J Intell Robot Syst (2011) 64:33–55
DOI 10.1007/s10846-010-9536-2

Complex Task Allocation in Mobile Surveillance Systems

Alaa M. Khamis · Ahmed M. Elmogy · Fakhri O. Karray

Received: 21 March 2010 / Accepted: 27 December 2010 / Published online: 21 January 2011

Abstract In mobile surveillance systems, complex task allocation addresses how to optimally assign a set of surveillance tasks to a set of mobile sensing agents to maximize overall expected performance, taking into account the priorities of the tasks and the skill ratings of the mobile sensors. This paper presents a market-based approach to complex task allocation. Complex tasks are the tasks that can be decomposed into subtasks. Both centralized and hierarchical allocations are investigated as winner determination strategies for different levels of allocation and for static and dynamic search tree structures. The objective comparison results show that hierarchical dynamic tree task allocation outperforms all the other techniques especially in complex surveillance operations where large number of robots is used to scan large number of areas.

Keywords Surveillance systems · Task allocation · Market-based techniques

1 Introduction

Effective monitoring of persistent and transient objects and events is a key to the effective protection of any area of interest (AOI). Surveillance is the systematic observation of AOI by visual, audio or other means. This systematic observation includes the timely detection, localization, recognition and identification of objects and events, their relationships, activities, and plans, in a given AOI in order to

A. M. Khamis · A. M. Elmogy (✉) · F. O. Karray
University of Waterloo, Waterloo, ON, Canada
e-mail: aelmogy@pami.uwaterloo.ca

A. M. Khamis
e-mail: akhamis@pami.uwaterloo.ca

F. O. Karray
e-mail: karray@pami.uwaterloo.ca

determine whether they are behaving normally or there is any deviation from their expected behavior. To achieve this complete situation awareness, the system starts by collecting the relevant data in order to identify situation entities and their relationships. Then the system performs a relational analysis of objects-events followed by intent estimation and consequence prediction.

Mobile surveillance systems incorporate self-organized networks of mobile sensing nodes, data and information fusion nodes, acting nodes and control nodes. These self-organized nodes can sense collaboratively and continuously AOIs and physically manipulate and interact with it. The sensing nodes represent a set of spatially distributed mobile sensing agents of different modalities that can sense collaboratively and continuously an AOI. These nodes include but are not limited to vision system, sonar/infrared/laser range finders, microphone array or RFID mounted on mobile bases in order to overcome the limitations of the static sensors. Mobile sensing nodes are capable of sensing, processing, mobilization and communication with other nodes. They can sample the environment at different locations, exchange the information with other nodes, and collaboratively achieve the required mission. Fusion nodes combine information from the sensing nodes about the targets and events in order to determine whether they are behaving normally or if there is any deviation from their expected behavior. Actuation may be a direct physical action upon the process, such as moving a camera to keep track an agile target; or a physical making of an electrical circuit, which in turn has a direct effect upon the process. An example would be an actuator (relay) that activates an alarm, a fire extinguisher or hard-kill/soft-kill weapons. Control nodes manages sensing, fusion and acting nodes to provide timely detection, localization, recognition and identification of targets and events, their relationships, activities, and plans, in a given AOI. In recent years, mobile surveillance systems research has received an increasing amount of attention from researchers in academia, government laboratories and industry [12, 20, 24, 27, 33, 34, 41]. This research activity has yielded some fruit in tackling some of the challenging problems of mobile surveillance that are still open. These research problems include, but are not limited to, task allocation, mobile sensing agent deployment, multisensor management, cooperative object detection and tracking, decentralized data fusion and interoperability and accessibility of system nodes. The problem of task allocation in mobile surveillance systems is a twofold problem. First it addresses how to assign a set of tasks to a set of mobile sensing agents. Second it considers how to coordinate the behavior of the sensor team in order to do the cooperative tasks efficiently. This problem is usually studied as Multi-robot Task Allocation (MRTA) problem [4]. In spite of the great number of MRTA algorithms reported in the literature, important aspects have, to date been given little attention. These aspects include but are not restricted to allocation of complex tasks, dynamic task allocation, and constrained task allocation. In this paper, we are trying to address these aspects by giving an organizational framework to study this problem in a formal manner. The paper discuses centralized and hierarchical dynamic and fixed tree task allocation approaches to solve the MRTA problem.

The remainder of the paper is organized as follows. Section 2 introduces the task allocation problem definition and formulation followed by reviewing the related work in Section 3. Section 4 discusses the different components of the proposed market-based approach. Experimental results are presented and discussed in Section 5 and finally, the conclusion and future work are summarized in Section 6.

2 Complex Task Allocation

This section provides problem definition for both simple and complex task allocations and formulation for only complex task allocation.

2.1 Problem Definition

Definition 1 (Simple Task Allocation) Given a set of mobile sensing agents S each looking for one task, and a set of tasks T each requires one mobile sensing agent. The simple task allocation can be defined by a function $A : T \rightarrow S$, mapping each task to a mobile sensing agent in order to be executed. Similarly, S_T is the set of all allocations of tasks T to the team of sensors S.

Definition 2 (Complex Task Allocation) Given a set of mobile sensing agents S, and a set of tasks T. Let $G \subset T$ is a group or a bundle of tasks that is decomposable into other tasks $M \in G$. The complex task allocation can be defined by a function $B : M \rightarrow S$, mapping each subtask to a mobile sensing agent to be responsible of completing it. Equivalently, S_M is the set of all allocations of subtasks M to the team of sensors S.

For both simple and complex task allocation, the goal is to assign sensors to tasks so as to maximize overall expected performance, taking into account the priorities of the tasks and the skill ratings of the sensors. Appropriate functions are needed to map possible task outcomes into revenue values and to map possible schemes for performing the task into cost values. The goal is to assign tasks to sensorss such that the overall profit (the excess of revenue over cost) is maximized. Generally, mobile sensing agents receive revenue and incur costs for accomplishing a specific team-task. A mobile sensing agent can also receive revenue from another agent in exchange for goods or services. The price dictates the payment amount for the good or service. A common approach is to bid for a good or service in order to arrive at a mutually acceptable price [9].

2.2 Problem Formulation

The problem of task allocation can be formulated in many ways. Given our surveillance application domain, it can be formulated as follows:

1. AOI: two dimensional, bounded area of interest.
2. S: a team of mobile sensing agents s_i, $i = 1, 2, \ldots n$. It is assumed that each mobile sensing agent carries sensors (such as cameras, sonar and laser range finders)
3. T: a set of tasks t_j, $j = 1, 2, \ldots m$.
4. U: a set of sensors utilities, u_{ij} is the utility of sensor i to execute task j.

For single sensor task, the problem is to find the optimal allocation of sensors to tasks, which will be a set of sensors and tasks pair [19]:

$$(s_1, t_1), (s_2, t_2), ...(s_k, t_k) \ for\ 1 \leq k \geq m \tag{1}$$

For the general case, the problem is to find the optimal allocation of a set of tasks to a subset of sensors, which will be responsible for accomplishing it [40]:

$$A : T \rightarrow S \tag{2}$$

Each mobile sensing agent $s \in S$ can express its ability to execute a task $t \in T$, or a bundle of tasks $G \subseteq T$ through bids $b_s(t)$ or $b_s(G)$. The cost of a bundle of tasks can be simply computed as the sum of costs of the individual tasks:

$$b_s(G) = \sum_{k=1}^{f} b_s(t_k) \{t_k \in G\} \tag{3}$$

where f is the number of tasks of the bundle G. The group's assignment determines the bundle $G \subseteq T$ of tasks that each mobile sensing agent $s \in S$ receives. These bundles can be characterized as follows:

$$\beta = \left\{ (G_1, G_2, ...G_w) | G_{k_1} \cap G_{k_2} = \varphi, \bigcup G_w = T \right\} \tag{4}$$

The global objective function can vary depending on the requirements of the system or the preferences of the designer. The most common global objective is to minimize the sum of the team member costs, which can be described mathematically as follows:

$$C(A) = \sum_{s=1}^{n} b_s(G_s) \tag{5}$$

where $C(A)$ is the total required cost for executing the allocation A, and G_s is the bundle of tasks that is won by sensor s.

Though the mobile sensing agent team members may have well-defined cost or utility functions, these functions still rely on having accurate models of the world state and may require computationally expensive operations. When there are multiple goal locations like in surveillance application, determining the cost to perform even one task can require solving multiple path planning problems. Thus an instance of the traveling salesman problem (TSP) [32, 37] might be used. In the theory of computational complexity, the decision version of TSP belongs to the class of NP-complete problems. Thus, it is assumed that there is no optimal algorithm for solving traveling salesman problems. In this work, we are using a shortest sequence planning algorithm (SSP) [36] in order to find the minimum cost path for each mobile sensing agent given the tasks locations. In this algorithm, an agent is tasked with visiting a set of points and the goal is to find in which order it should visit these points with minimum traveling distance without going back to its original place. Thus accomplishing the required tasks with near optimal system performance.

3 Related Work

MRTA approaches can be classified based on problem description, task allocation category, the planning meth-od, the used organizational paradigm, and problem solving techniques. A framework for studying multi-robot task allocation problem is introduced in [15]. According to this framework, the multi-robot task allocation problem can be seen as an instance of optimal assignment problem (OAP) [13]. This is can be defined in the following way: given n robots, m single robot tasks, assign robots to tasks so as to achieve maximum overall profit. Because the problem of task allocation is a dynamic decision problem that varies in time with phenomena including environmental changes, this static assignment problem should be solved iteratively over time [15]. Thus, dynamic task allocation is a class of task allocation in which the assignment of robots to sub-tasks is a dynamic process and may need to be continuously adjusted in response to changes in the task environment or group performance [21].

A formal analysis and taxonomy of multi-robot task allocation is also introduced in [16]. The authors in this survey paper tried to provide a particular taxonomy for studying MRTA, based on organizational theory from several fields, including operations research, economics, scheduling, network flows, and combinatorial optimization. Also, a complete analysis and description for single-task (ST), multi-task (MT) robots, single robot (SR), multi-robot (MR) tasks, instantaneous assignment (IA) and time-extended assignment (TA) are provided. ST means that each robot is capable of executing as most one task at a time, while MT means that some robots can execute multiple tasks simultaneously. Very similarly, SR means that each task requires exactly one robot to achieve it, while MR means that some tasks can require multiple robots. In IA approaches [4, 14, 26] the available information concerning the robots, the tasks, and the environment permits only an instantaneous allocation of tasks to robots (i.e. tasks independence is a strong assumption). These approaches are sometimes used in order to avoid the need for highly computationally scheduling algorithms. At the other extreme is continuous task allocation or time extended assignment (TA) approaches [9, 31, 40] where more information is available, such as the set of all tasks that will need to be assigned. Because robots have to reason about the dependencies between tasks, TA is more demanding from a planning perspective. The work presented here can be categorized as single robot task-single task robot-instantaneous assignment task allocation (ST-SR-IA).

Existing task allocation techniques can be categorized to: (1) Allocation of simple tasks, (2) Allocation of complex tasks. Simple tasks are tasks that can be accomplished by a straightforward manner [4, 9, 11, 14, 26, 31] while complex tasks are the tasks that have several possible ways to be implemented [22, 38, 40]. When dealing with complex tasks, the structure and semantics of the tasks can be exploited to produce more efficient team plans by giving individual robots the ability to come up with new ways to perform a task, or by allowing multiple robots ("mobile sensing agents" henceforth) to cooperate by sharing the subcomponents of a task, or both [40]. Motivated by the little attention given to formal modeling, and analysis of complex task allocation, complex tasks that can be decomposed into different subtasks is the current scope of this paper.

From the perspective of planning, there are two common approaches to task allocation problem; decompose-then allocate and allocate-then decompose. In the first technique, the complex mission is decomposed to simple subtasks and then these subtasks are allocated to the team member based on their capability and availability to complete the subtasks as required [1, 25]. In this type of techniques, the cost of the final plan cannot be fully considered because the task decomposition is done without knowing to whom tasks will be allocated. Another disadvantage of this type is inflexibility to changes in the designed plan. So, the plan designed by the central agent cannot be rectified even if it is found costly. On the other side, in allocate-then decompose approach [4], the complex tasks are allocated to mobile sensing agents, and then each mobile sensing agent decomposes the awarded tasks locally. The main disadvantage of this approach is the allocation of all tasks to only one mobile sensing agent and thus, the preferred task decomposition is purely dependent on the plan of that mobile sensing agent which increases the possibility of having suboptimal solution. It may be more beneficial to allocate tasks to more than one mobile sensing agent in order to consider different plans for the required task. While the decompose-then-allocate and the allocate-then decompose methods may be capable of finding feasible plans, there are drawbacks to both approaches. Motivated by these drawbacks, Zlot and Stentz proposed in [40] a market based task allocation approach to allocate complex tasks among a robot team. They proposed a solution concept that unifies the decompose-then allocate and allocate-then decompose stages by not decoupling the solution into separate allocation and decomposition phases.

Another line of comparison between task allocation approaches is the classification according to team organization: centralized and hierarchical approaches. In centralized approaches, a single agent is employed to coordinate the team. Theoretically, this agent gathers all relevant information from the team members, does planning for the entire team, and broadcasts commands in order to allocate tasks to robots. Practically, fully centralized approaches can be computationally intracta-ble, brittle, and unresponsive to change. Thus, for applications where teams are small and the environment is static or global state information is easily available, centralized approaches are the best-suited solution. Not surprisingly, many MRTA architectures implement some form of this approach [5, 6, 8, 26, 29]. On the other hand, in hierarchical task allocation approaches, mobile sensing agents rely solely on local knowledge. Such approaches have many advantages over centralized approaches such as; flexibility, robustness, and low communication demands. However, because a good local solution may not sum to a good global solution, hierarchical approaches can produce highly suboptimal solutions. Fully hierarchical schemes are best suited in applications where large teams carry out relatively simple tasks without efficiency restriction. In order to take the advantages of both schemes, many market-based approaches have been proposed [3, 10, 30]. Thus having centralized and hierarchical elements can help in accomplishing task allocation mission.

This paper presents a market-based approach to complex task allocation. Both centralized and hierarchical allocations are investigated as winner determination strategies for different levels of allocation and for static and dynamic search tree structures. A task tree structure is used as an implementation of tasks before auctioning. The task tree is constructed as a generic representation of tasks by performing a hierarchal decomposition on the abstract task. Details of the proposed approach are presented in the next section.

4 Proposed Market-based Approach

Market-based approaches have received significant attention and are growing very fast in the last few decades especially in multi-agent domains [3, 10, 23, 30, 40].

These approaches are considered as hybrid approaches that combine the centralized and distributed strategies (i.e. market-based approaches have elements that are centralized and distributed). Motivated by this great attention, a market-based approach for dynamic task allocation for multisensor surveillance systems is presented in this paper.

4.1 Single-shot and Combinatorial Auctioning

So far, researchers have studied single-item auctions at which items are auctioned off one at a time [39]. However, if there are strong synergies between the items of the bidders, highly suboptimal team solutions can result from single-item auctions [2]. Two items are said to exhibit positive or negative synergy for a bidder if the combined bid of this bidder on these two items is larger or smaller than the sum of its individual bids on each item separately.

An example of that is shown in Fig. 1. There is positive synergy between AOI-1 and AOI-2 for the mobile sensing agent S_1 because they are close to each other. The mobile sensing agent S_1 can reach AOI-2 with a short distance (5 m) after it has reached AOI-1 (5 m). So, the sum of the single bids of S_1 on AOI-1 and AOI-2 ($12 = 5 + 7$) is more than the combined bid of S_1 on both areas ($10 = 5 + 5$). On the other hand, there is a negative synergy between AOI-1 and AOI-3 for S_1 because they are on opposite sides of the S_1, and hence the mobile sensing agent S_1 can therefore reach either one of the areas only with a long travel distance after it has reached the other one.

Generally speaking, combinatorial auctions attempt to overcome the disadvantages of single-item auctions by allowing bidders to bid on bundles of items [2, 40]. If a bidder wins a bundle, they win all the items in that bundle.

4.2 Auction Design

The task allocation approach proposed in this paper imitates the auction process of buying and selling services through bidding. Sellers or auctioneers are responsible of

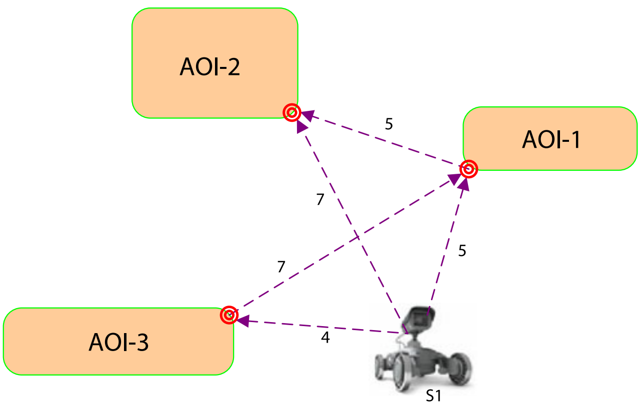

Fig. 1 Single-shot and combinatorial auctioning

processing the bids sent by buyers or bidders and determining the winning bidder. In this subsection, a brief idea about how to design the auction process will be based on maximizing a utility function.

4.2.1 Utility Function

Utility is the quality or state of being useful. For task allocation problem, utility is a satisfaction (value of profit) derived by a mobile sensing agent s_i from accomplishing a task t.

Given a mobile sensing agent s and a task t, if s is capable of executing t, utility can be defined [40] on some standardized scale as:

$$u = p(t) - d(t) \tag{6}$$

Where $p(t)$ is the total payment it receives after executing the task t, and $d(t)$ is the total distance it travels to reach the task. The priorities of tasks to be executed should be taken into account while designing the task allocation framework. Our objective is to find the optimal assignment of tasks T to sensors S in order to minimize cost and thus maximize the overall utility. Consequently, system performance is ideally optimized. Thus, the goal is to assign sensors to tasks so as to minimize the cost as we assume that there is no payement received after executing the task.

4.2.2 Search Tree

Most of task allocation approaches treated tasks as atomic units [4, 9, 14, 31]. Thus allowing only static description for each task and so the only degree of freedom is determining to which sensor the task will be assigned. While this description is fine in case of simple tasks, it is not with complex tasks. Search over all possible allocations can be used as a winner determination strategy to find to optimal allocation that maximize the revenue of the whole team. In this case, a search tree can be used as a better description for the tasks. In this tree, mobile sensing agents are permitted to bid on nodes representing varying levels of task abstraction, thereby enabling hierarchical planning, task allocation, and optimization among the team members.

In our work, the complex tasks to be allocated are structured as an ordered tree. In a set theory, a tree is defined to be a set E and a relation F where $F \subseteq E$ such that:

- F is a partial-ordering of E.
- For any $e \in E$,$\{ v \in E \mid vFe\}$ is well-ordering. A set X is well-ordering if every non-empty subset of X has a least an element under the ordering.

The nodes (elements of the tree) that are immediately greater than a node are called its children, while the node that is immediately less is its parent (if it exists). Any node less is an ancestor and any node greater is a descendant. A node with no ancestors is a root. The partial ordering represents distance from the root, and the well-ordering requirement prohibits any loops or splits below a node (that is, each node has at most one parent, and therefore at most one grand-parent, and so on). In other words, if xFz then there is exactly one yFz such that xFy and there is nothing between x and y. Perhaps the best way to illustrate the mechanics of the task tree is through a simple example. Figure 2 shows a surveillance scenario, which represents a shopping mall in the city of Waterloo, Ontario, Canada. The mission is to monitor a set of

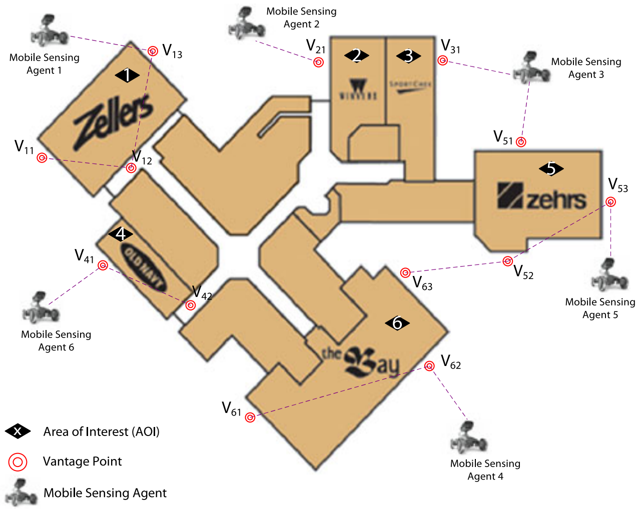

Fig. 2 Surveillance scenario

areas of interest (AOIs) such Zellers, Old Navy, the Bay, Zehrs, Winners and Sports Check. For the small areas like (Old Navy, Winners and Sports Check), only one or two vantage points are enough to achieve the monitoring task, while three points are required to visit for the large areas like (Zellers, the Bay, and Zehrs).

The *AND/OR* task tree is constructed by decomposing the surveillance mission as complex task into two subtasks (scan large areas and scan small areas) as illustrated in Fig. 3. Accomplishing the requested mission requires achieving both scan large and small areas. In other words, these two subtasks are related to each other by the logical operator AND, which means that both tasks are required to be executed. The subtask (Scan large areas) is in turn decomposed to other simpler tasks such as Scan AOI-1, AOI-5 and AOI-6. The simple tasks can be executed by one of two plans, which contain the most primitive tasks. For example, to scan AOI-1, Plan-1 or Plan-2 can be chosen. These two alternative covering plans are computed based on the minimum traveling distance and the second minimum traveling distance. Plan-1 contains a list of primitive tasks (Goto V_{13}, Goto V_{12}, Goto V_{11}) that must be executed sequentially. Similarly, Plan-2 contains same primitive tasks but with different order (Goto V_{11}, Goto V_{12}, Goto V_{13}) as shown in Figs. 2 and 3.

This decomposition is done initially by an operator or by the selected initial auctioneer. Once the task tree is constructed and the decomposition is complete, the auctioneer holds a task tree auction, distributing tasks among the team and allowing other robots to use their own plans when appropriate. The auctions then proceed in rounds in which each mobile sensing agent holds a task tree auction (if it has any tasks) in a round-robin fashion.

Other logical operators like *XOR*, and *NAND* can also be used. *XOR* operator can be used in order to implement the surveillance of sensitive areas (sensitive area must be surveyed exclusively by the assigned sensor), while *NAND* can be used to prevent redundancy (i.e. each area of interest is surveyed by only one sensor).

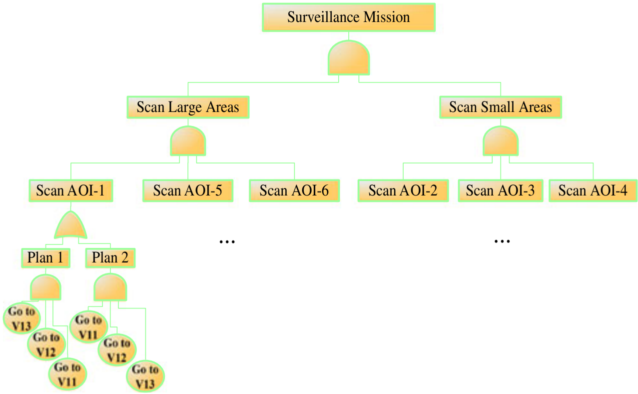

Fig. 3 *AND/OR* task tree

4.3 Allocation Levels

From the perspective of planning, there are different allocation levels for complex tasks. In this subsection, these allocation levels are discussed in detail.

4.3.1 Point-level Allocation

The complex mission is decomposed to simple subtasks and then these subtasks are allocated to the team member based on their capability and availability to complete the subtasks as required [1, 25]. In this type of techniques, the cost of the final plan cannot be fully considered because the task decomposition is done without knowing to whom tasks will be allocated. Another disadvantage is inflexibility to changes in the designed plan. So, the plan designed by the central agent cannot be rectified even if it is found costly. This is also called decompose then allocate approach [22].

As shown in Fig. 4, the mission is initially decomposed by the auctioneer into a set of surveillance points. All auctions are only for tasks in this set of goal points.

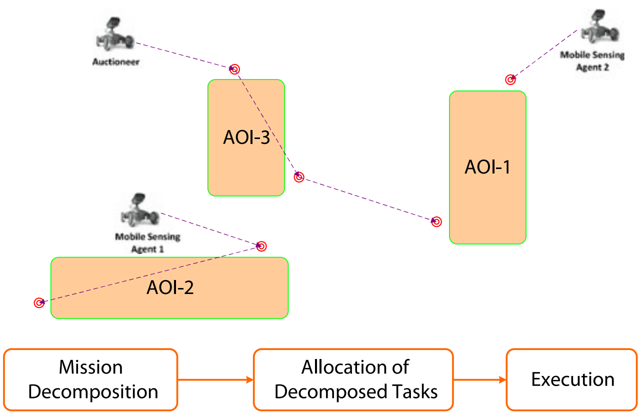

Fig. 4 Decompose-then-allocate approach

In other words, there is no notion of abstract tasks; no further decomposition occurs. Each winner put the awarded subtask into its schedule to be considered for execution.

4.3.2 Area-level Allocation

In this type of allocation techniques, the situation is different. The complex tasks are allocated to one mobile sensing agent, which in turn decomposes the awarded tasks locally. This is also called allocate-then-decompose approach [19]. One disadvantage of this approach is that it may be beneficial to allocate subcomponents of these tasks to more than one sensor. As shown in Fig. 5, the mission is allocated to one of the sensors (auctioneer), which in turn will be responsible of decomposing it to a set of surveillance areas. All auctions are only for tasks in this set of goal areas. In other words, there is no notion of single tasks; no further decomposition occurs. Each winner put the awarded subtask area into its schedule to be considered for execution. The auctioneer finds itself the best to execute area 3, while it awards area 1 to mobile sensing agent 2, and area 2 to mobile sensing agent 1.

4.3.3 Mission-level Allocation

In this type of allocation, the auction is for the whole mission and so it is considered as single-shot type of auctioning. Thus the entire mission is awarded to only one sensor, which can decompose the mission to subtasks. The decomposition is considered only for execution not for reallocating the subtasks. This is can be called higher-level allocate-then decompose [40]. The main disadvantage of this approach is the allocation of all tasks to only one sensor and thus, the preferred task decomposition is purely dependent on the plan of that sensor which increases the possibility of having suboptimal solution. It may be more beneficial to allocate tasks to more than one sensor in order to consider different plans for the required task.

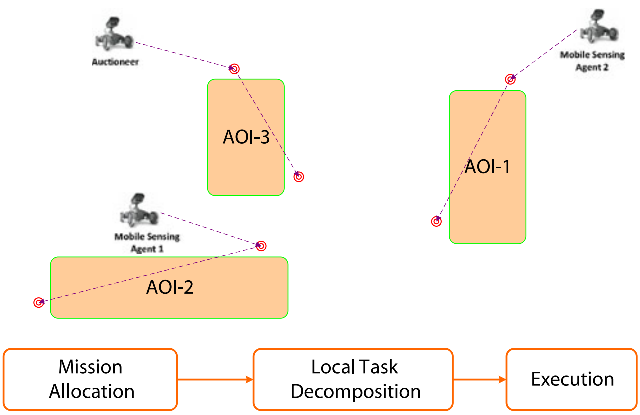

Fig. 5 Allocate-then-decompose approach

4.4 Winner Determination Strategies

The winner determination strategy addresses how to optimally find the set of bids that maximize the bidder's revenue. In combinatorial auction, winner determination is NP-hard problem [28] as searching for all possible allocations of items to mobile sensing agents is computationally intractable and no approach will work in polynomial time.

Winner determination strategy is highly affected by the type of description of tasks to be allocated. As mentioned previously, the complex tasks to be allocated are represented as an ordered tree. A breadth-first search algorithm is used to find the task allocation solution from this task tree structure. Two organizational paradigms, namely, centralized and hierarchical allocation are used during the allocation process. These paradigms determine the roles, the relationships, and the structures, which govern the auction process.

4.4.1 Centralized Allocation

In this type of auctioning, an auctioneer holds a series of auctions to allocate the surveillance tasks to the mobile sening agents in order to maximize the system utility. An example for this is shown in Fig. 6. The auctioneer holds auctions in rounds to allocate the tasks it has to the mobile sensing agents S_1, S_2, S_3, and S_4 if it finds that the system utility will increase.

4.4.2 Hierarchical Allocation

As shown in Fig. 7, the tasks are allocated initially to the mobile sensing agents S_1, S_2, S_3, and S_4 via a central auctioneer. Each mobile sensing agent can hold auctions in rounds for the tasks it wins in the initial auction.

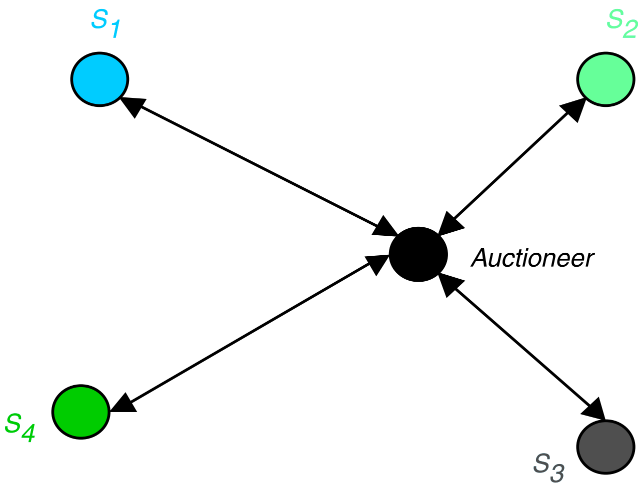

Fig. 6 Centralized auctioning

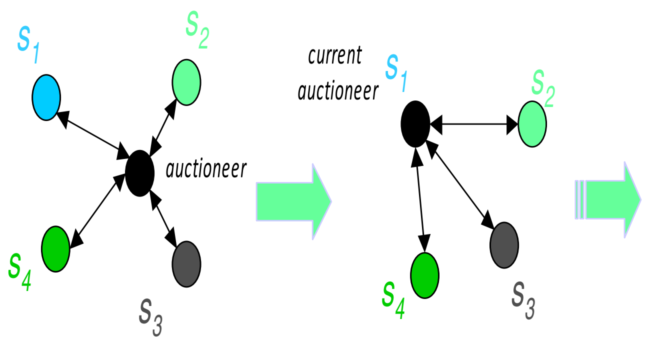

Fig. 7 Hierarchical auctioning

4.5 Fixed Tree Task Allocation

The key to effective task allocation for multi-sensor systems is to iterate the assignment, in order to deal with changing conditions [10] such as changes in the environment, including malfunctioning agents. Any tasks undertaken by a mobile sensing agent that malfunctions can be re-bid to other agents, and the entire task can be accomplished. For that, an iterated market-based complex task allocation approach is developed to allocate tasks to the mobile sensing agent team members through contract negotiation. A manager agent can offer tasks to other agents, which may submit bids based on their ability to perform the tasks. Centralized and hierarchical auction mechanisms are developed for complex task allocation. Using fixed and dynamic tree are explained in the following subsections.

Consider a team of mobile sensing agents assembled to perform a particular task. Consider further, that each mobile sensing agent is capable of executing one task at once, and each task can be accomplished by one sensor. The task information is continuously available to the mobile sensing agents team. Thus, the proposed approach in this case can be framed as iterated instances of ST-SR-IA (Single-Task Single Robot Instantaneous Assignment). The goal of the team is to perform the task efficiently while minimizing costs. This is can be done by modeling each mobile sensing agent as self-interested agent which aims to minimize the team cost and so maximize the whole team performance by following a greedy algorithm. Each mobile sensing agent is cooperating with other members of the team to achieve an outcome greater than that possible by each member alone, thus eliminating waste and inefficiency. A system such as this can be highly seen in the economy and so many desirable characteristics from the market mechanisms might be used.

In the market-based task allocation approaches [9, 15, 40], the mission task to be executed is awarded to one of the agents, which is called the operator or the auctioneer. This auctioneer is responsible for providing a plan for executing this mission task to the other team members. The proposed plan by the auctioneer is implemented in form of task tree as mentioned above. Each mobile sensing agent maintains several lists of trees it has agreed to handle. The constructed task tree will have abstract nodes (have children) and primitive nodes (no children). The proposed algorithm allows the auctioneer to sell any primitive node through sequential

Algorithm 1: CenBFFixTree2P($T, Aucid, Ns, Na$) Centralized Fixed Tree *(AND/OR)*

Data: Na: = number of areas;
Ns: number of mobile sensing agents;
$Aucid$: the ID of the auctioneer
Result: $A : S \longleftarrow T$; the allocation of the task tree T to the set of mobile sensing agents S;
$Totalcost$:the cost of executing the tree T

1 **begin**
2 Compute the best task tree T_1 using SSP (shortest sequence planning) algorithm;
3 Initialization: $Aucid \longleftarrow T_1$; mark the whole tree as won by $Aucid$;
4 **for** $i = 1$ *to* N_a **do**
5 Compute $cost_1(T_1)$; the traveling cost of the tree T_1 before auctioning ;
6 **for** $j = 1$ *to* Ns **do**
7 **while** $j \neq Aucid$ **do**
8 Compute $b_j(i) = \sum b_j(h) \{h \in i\}$; the bid of mobile sensing agent j on node i ;
9 Find $bestsensor_1$; the ID of sensor which has the best bid (i.e. lowest cost) on node i. This is done using SSP algorithm ;
10 $T_{bestsensor1} \longleftarrow T_{bestsensor1} + i$;
11 $T_{Aucid} \longleftarrow T_{Aucid} - i$;
12 Compute $cost_2(T_1)$; using SSP algorithm ;
13 the traveling cost of the tree T_1 after auctioning ;
14 **if** $cost_2(T_1) > cost_1(T_1)$ **then**
15 $T_{bestsensor1} \longleftarrow T_{bestsensor1} - i$;
16 $T_{Aucid} \longleftarrow T_{Aucid} + i$;
17 **if** $\exists$ *set* $t \subset T_{Aucid}$ **then**
18 **for** $c = 1$ *to* t **do**
19 Compute $cost_3(T_1)$; the traveling cost of the tree T before auctioning the single nodes;
20 **for** $o = 1$ *to* Ns **do**
21 **while** $o \neq Aucid$ **do**
22 Compute $b_o(c)$; the bid of mobile sensing agent o on node c ;
23 Find $bestsensor_2$; the ID of mobile sensing agent which has the best bid (i.e. lowest cost) on node c. This is done using SSP algorithm ;
24 $T_{bestsensor_2} \longleftarrow t_{bestsensor2} + c$;
25 $T_{Aucid} \longleftarrow T_{Aucid} - c$;
26 Compute $cost_4(T_1)$; the traveling cost of the tree T after auctioning ;
27 **if** $cost_4(T_1) > cost_3(T_1)$ **then**
28 $T_{bestsensor2} \longleftarrow T_{bestsensor2} - c$;
29 $T_{Aucid} \longleftarrow T_{Aucid} + c$;
30 $Totalcost_1 \longleftarrow cost(T)$;
31 Compute the second best tree T_2 ((i.e. second lowest cost tree)using SSP (shortest sequence planning) algorithm;
32 Initialization: $Aucid \longleftarrow T_2$; mark the whole tree as won by $Aucid$;
33 Repeat steps from 4 to 28 replacing T_1 with T_2 $Totalcost_2 \longleftarrow cost(T)$;
34 **if** $Totalcost_2 > Totalcost_1$) **then**
35 $T \longleftarrow T_1$;
36 **else**
37 $T \longleftarrow T_2$;
38 $Totalcost \longleftarrow cost(T)$;
end

Algorithm 2: HBFFixTree2P($T, Aucid, Ns, Na$) Hierarchical Fixed Tree *(AND/OR)*

Data: Na: = number of areas;
Ns: number of mobile sensing agents;
$Aucid$: the ID of the auctioneer
Result: $A : S \longleftarrow T$; the allocation of the task tree T to the set of mobile sensing agent S;
$Totalcost$: the cost of executing the tree T

begin
  Compute the best task tree T_1 using SSP algorithm;
  Initialization: $Aucid \longleftarrow T_1$; mark the whole tree as won by $Aucid$;
  CenBFFixTree2P($T, Aucid, Ns, Number$);
  for $i = 1$ *to* Ns **do**
    CenBFFixedTree2P($T, i, Ns, Number$);
  $Totalcost \longleftarrow cost(T)$;
end

single-shot auctions. Also, combinatorial auctions are adopted by selling some bundles of tasks or abstracted nodes. Our algorithm begins by auctioning the abstracted nodes in the constructed tree from the top to the bottom of the tree. Selling the most top abstracted or apex node in the tree means that one sensor will be responsible of executing the whole mission task. This is of course if its execution cost is less than the execution cost of the auctioneer. If after auctioning all the abstracted nodes, there are still nodes in the auctioneer tree. It tries to sell them by running auctions on the level of the primitive nodes if it is profitable for him to do. The winner of any auction will insert the node it wins in its task tree and thus be responsible of completing it either by executing it itself, or by selling the whole task or part of it to other teammates.

In the context of fixed task tree allocation, a set of constrains dictates that the whole auction mechanism is based only on one task tree, which is proposed by the operator or the auctioneer. The proposed algorithm allows using only one auctioneer from the start to the end of auctioning, and so considered a centralized task allocation. It also allows changing the auctioneer during auctioning while considering only the plan of the original operator. In this case, our proposed mechanism can be seen as a hierarchal task allocation mechanism. Another constraint dictates that at most one node can be sold to each bidder per auction. This is because upon awarding one node to a bidder the bid prices on other nodes become invalid due to the fact that bid prices are conditioned on the current commitments of each participant. The details of our fixed task tree allocation algorithms (centralized and hierarchical) for *AND/OR* bidding language are shown in Algorithms 1 and 2 respectively. The *AND/OR* bidding language means that the current auctioneer will have some abstract areas that must to be surveyed (*AND*) with two alternative plans for each area (*OR*), and only one plan will be sold.

4.6 Dynamic Tree Task Allocation

The proposed fixed task tree allocation described in the previous section could be seen as an instance of decompose-then-allocate approach. The main drawback of this approach is that the cost of the final plan cannot be fully considered because the

Algorithm 3: CenBFDynTree2P($T, Aucid, Ns, Na$) Centralized Dynamic Tree *(AND/OR)*

Data: Na: = number of areas;
Ns: number of mobile sensing agents;
$Aucid$: the ID of the auctioneer
Result: $A : S \longleftarrow T$; the allocation of the task tree T to the set of mobile sensing agent S;
$Totalcost$:the cost of executing the tree T

1 **begin**
2 Compute the best task tree T_1 using SSP (shortest sequence planning) algorithm;
3 Initialization: $Aucid \longleftarrow T_1$; mark the whole tree as won by $Aucid$;
4 **for** $i = 1$ *to* N_a **do**
5 $\forall x$ (abstract node), find the area index $AOI(x)$;
6 Compute $cost_1(T_1)$; the traveling cost of the tree T_1 before auctioning ;
7 **for** $j = 1$ *to* Ns **do**
8 **while** $j \neq Aucid$ **do**
9 Compute $P_j(AOI(i))$; the plan of mobile sensing agent j for area $AOI(i)$;
10 Find $bestsensor_1$; the ID of mobile sensing agent which has the best plan on node i. This is done using SSP algorithm ;
11 $T_{bestsensor1} \longleftarrow T_{bestsensor1} + P_{bestsensor1}(AOI(i))$;
12 $T_{Aucid} \longleftarrow T_{Aucid} - i$;
13 Compute $cost_2(T_1)$; the traveling cost of the tree T after auctioning ;
14 **if** $cost_2(T_1) > cost_1(T_1)$ **then**
15 $T_{bestsensor1} \longleftarrow T_{bestsensor1} - P_{bestsensor1}(AOI(i))$;
16 $T_{Aucid} \longleftarrow T_{Aucid} + i$;
17 **if** $\exists$ *set* $t \subset T_{Aucid}$ **then**
18 **for** $c = 1$ *to* t **do**
19 Compute $cost_3(T_1)$; the traveling cost of the tree T_1 before auctioning the single nodes;
20 **for** $o = 1$ *to* Nr **do**
21 **while** $o \neq Aucid$ **do**
22 Compute $b_o(c)$; the bid of sensor o on node c ;
23 Find $bestsensor_2$; the ID of sensor which has the best bid (i.e. lowest cost) on node c. This is done using SSP algorithm ;
24 $T_{bestsensor_2} \longleftarrow t_{bestsensor2} + c$;
25 $T_{Aucid} \longleftarrow T_{Aucid} - c$;
26 Compute $cost_4(T_1)$; the traveling cost of the tree T after auctioning ;
27 **if** $cost_4(T_1) > cost_3(T_1)$ **then**
28 $T_{bestsensor2} \longleftarrow T_{bestsensor2} - c$;
29 $T_{Aucid} \longleftarrow T_{Aucid} + c$;
30 $Totalcost_1 \longleftarrow cost(T)$;
31 Compute the second best tree T_2 (i.e. second lowest cost tree) using SSP (shortest sequence planning) algorithm;
32 Initialization: $Aucid \longleftarrow T_2$; mark the whole tree as won by $Aucid$;
33 Repeat steps from 4 to 30 replacing T_1 with T_2;
34 $Totalcost_2 \longleftarrow cost(T)$;
35 **if** $Totalcost_2 > Totalcost_1)$ **then**
36 $T \longleftarrow T_1$;
37 **else**
38 $T \longleftarrow T_2$;
39 $Totalcost \longleftarrow cost(T)$;
end

Algorithm 4: HBFDynTree2P($T, Aucid, Ns, Na$) Hierarchical Dynamic Tree *(AND/OR)*

Data: Na: = number of areas;
Ns: number of mobile sensing agents;
$Aucid$: the ID of the auctioneer
Result: $A : S \longleftarrow T$; the allocation of the task tree T to the set of mobile sensing agent S;
$Totalcost$: the cost of executing the tree T

begin
  Compute the best task tree T_1 using SSP algorithm;
  Initialization: $Aucid \longleftarrow T_1$; mark the whole tree as won by $Aucid$;
  CenBFDynTree2P($T, Aucid, Ns, Number$);
  for $i = 1$ *to* Ns **do**
    CenBFDynamicTree2P($T, i, Ns, Number$);
  $Totalcost \longleftarrow cost(T)$;
end

complex task is decomposed by the auctioneer without knowledge of the eventual task allocation. Also, backtracking is not allowed in this approach, and so any costly mistakes in the auctioneer decompositions cannot be rectified. Generally, the allocate-then-decompose method tries to avoid the drawbacks of the decompose-then-allocate method. However, there are still some disadvantages. Motivated by the drawbacks of both methods, we are proposing dynamic tree allocation to allow backtracking in order to recover the bad plans made by the auctioneers. The algorithm allows auctioning on all levels of abstraction of the mission task implemented by the task tree from the top to the bottom. Each mobile sensing agent evaluates its ability to execute the required task based on its plan not on the plan of auctioneer. Our proposed dynamic algorithm is either executed by allowing only one auctioneer (centralized allocation) or allowing different auctioneers (hierarchical allocation). The details of these algorithms for *AND/OR* bidding language are shown in Algorithms 3 and 4 respectively.

5 Results and Discussion

In order to evaluate the proposed approach, we consider an area surveillance application where the goal is to monitor some areas in a physical space like in malls and airports with a team of mobile nodes, each equipped with a vision system, and laser ranger sensor. It is assumed that the physical space is priori known with some obstacles. To tackle this application, it is assumed that for each area, a set of surveillance points (vantage points) is selected from which the mobile sensing agents can view the interior of the area as illustrated in Fig. 2. The architecture under study achieves the surveillance task while keeping in mind the minimization of the total traveling distance of the whole team. The mobile sensing agents team frequently uses the SSP algorithm when bidding, and when reordering schedules after trades or task completion.

Our initial experiments are performed within a 2D simulation environment using Player/Stage simulator [17, 35]. The above scenario was run with six different cases

Table 1 Test cases

Case	Number of areas	Number of mobile sensing agents
1	1	2
2	4	2
3	5	3
4	7	5
5	6	6
6	4	7

as shown in Table 1 with 50 runs for each case. These areas are represented within the Player/Stage environment as a graph. As shown in Figs. 8 and 9, the average cost of the task tree algorithms (fixed and dynamic) is compared with that of the allocate-then decompose, decompose-then allocate, and higher-level allocate-then decompose algorithms explained above for each type of allocation mechanisms (centralized and hierarchical).

The average cost is computed by calculating the cost of executing the mission task using 50 runs and then taking the average. In terms of this average cost, the results in Figs. 8 and 9 show that both fixed and dynamic tree allocations (centralized and hierarchical) consistently outperform the other algorithms especially for the complex cases like in cases 4, 5, and 6. It is also seen that the dynamic tree allocation outperforms fixed tree allocation, which was expected as the replanning ability is added to the sensors in the dynamic tree allocation.

On average, the hierarchical task tree algorithm is better than the centralized task tree algorithm besides its good feature of relying on different auctioneers compared to one auctioneer in the centralized algorithm. This is because the hierarchical auctioning allows more auction rounds to happen and so the mobile sensing agents may find themselves in better positions to win more tasks than their old positions if

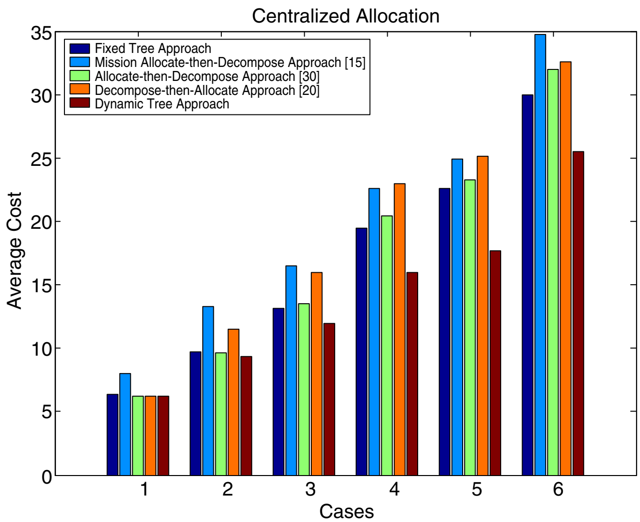

Fig. 8 Comparison of the average cost for different allocation algorithms

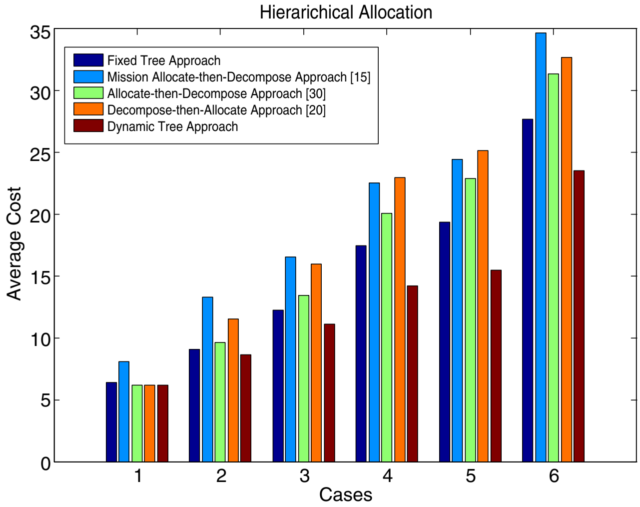

Fig. 9 Comparison of the average cost for different allocation algorithms

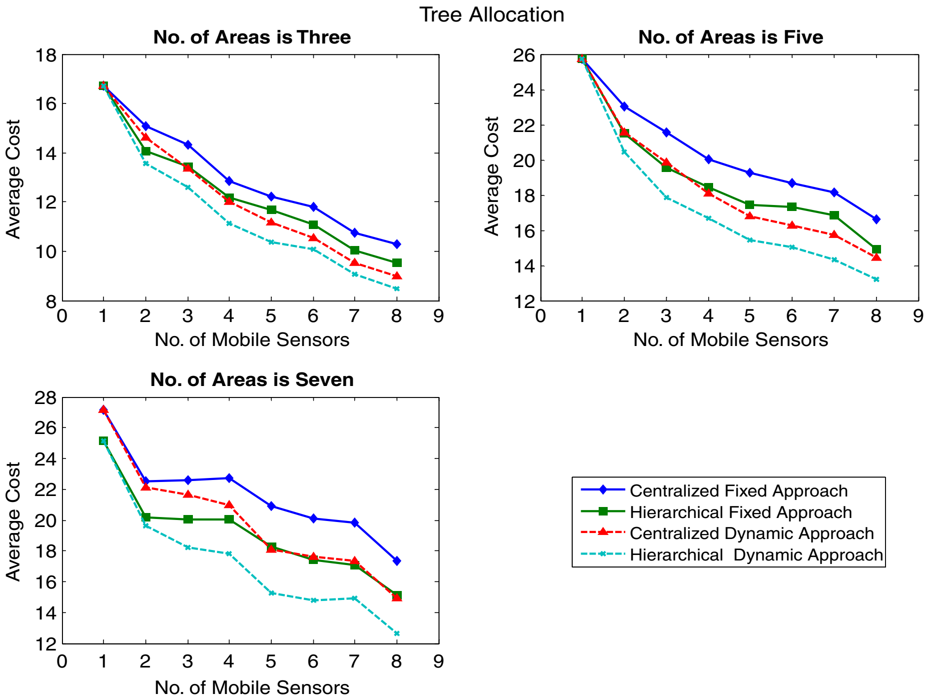

Fig. 10 A comparison of the solution quality of task tree allocation varying the number of mobile sensing agents

it is found beneficial to the whole system. Thus the hierarchical auctioning increases the possibility of improving the system performance over the centralized auctioning.

In order to determine how the proposed tree allocation algorithms are affected by problem complexity, two sets of experiments are conducted. In the first set, the number of mobile sensing agents is varying while the number of areas is kept fixed, and in the second set, the number of areas is varying while the number of mobile sensing agents is kept fixed. The results for both sets (shown in Figs. 10 and 11) are deduced using the average of 50 trials. The quality of the proposed algorithms is again evaluated using the average traveling cost like the experiments shown in Figs. 8 and 9. For each type of allocation mechanism (centralized and hierarchical), the average traveling cost decreases as the number of mobile senosrs increases as shown in Fig. 10. It is also noticed from Fig. 10 that the average traveling cost increases as more areas are incorporated in the environment. For the second set of the experiments shown in Fig. 11, as the number of areas increases, the average traveling cost increases. Also, the average traveling cost is decreased as more mobile sensing agents are used to achieve the surveillance mission. A general trend in the results shown in Figs. 10 and 11 is that the hierarchical tree allocation outperforms the centralized one.

The conducted experiments give objective comparison results between hierarchal and centralized dynamic and fixed tree task allocation. This comparison shows

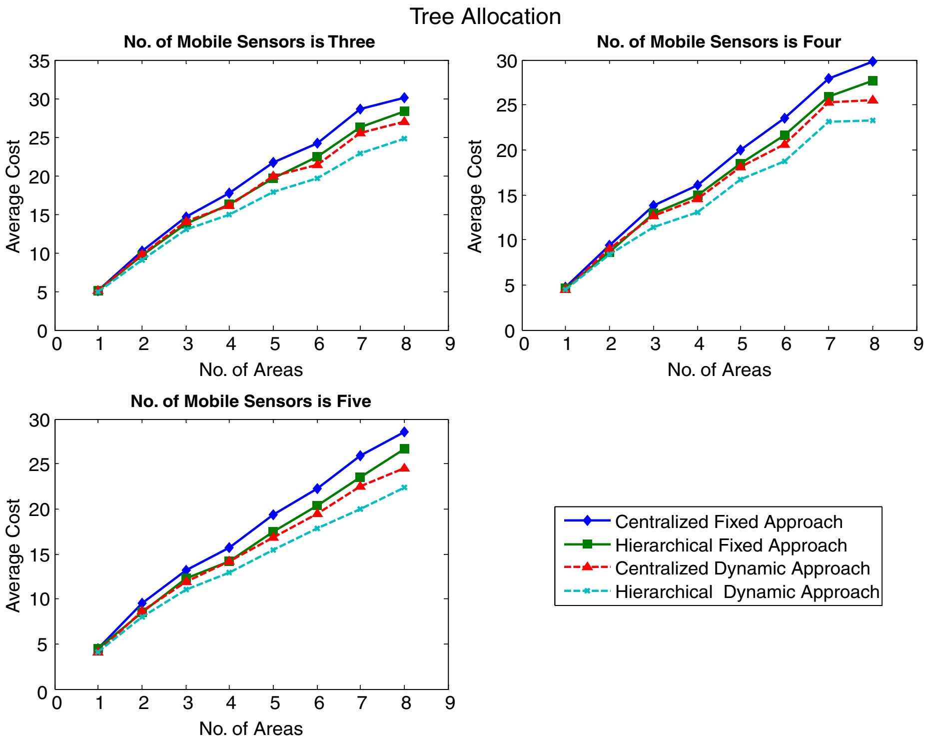

Fig. 11 A comparison of the solution quality of task tree allocation varying the number of areas

that hierarchal dynamic tree task allocation outperforms all the other techniques especially in complex surveillance operations where large number of mobile sensing agents is used to scan large number of areas.

6 Conclusion

In this paper, a market-based approach has been presented to solve the complex task allocation problem in mobile surveillance systems. Centralized and hierarchical algorithms with fixed and dynamic trees have been examined focusing on complex tasks, i.e., the tasks that can be decomposed into subtasks. The results of the conducted experiments showed that hierarchical dynamic tree task allocation outperforms all other techniques especially in complex surveillance operations wh-ere large number of mobile sensing agents is used to scan large number of areas. In the future, we consider extending the proposed algorithms so that constrained and tight tasks can be handled. An example for constrained is two tasks that cannot be done independently as the same sensor would obviously have to do both of them. Tight tasks cannot be decomposed into further single sensor tasks. In this case, a subgroup of mobile sensing agents could determine their joint costs and submit joint bids for such type of tasks. Also to ensure cost independence between the sub-teams, the proposed framework should be extended to include the constraint that the sub-teams being awarded tight tasks are disjoint.

Acknowledgements This work has been supported by Ontario Research Fund-Research Excellence (ORE-RE) program through the project "MUltimodal- SurvEillance System for SECurity-REl-aTed applications (MUSES SECRET)" funded by the Government of Ontario, Canada.

References

1. Aylett, R., Barnes, D.: A multi-robot architecture for planetary rovers. In: Proc. 5th ESA Workshop on Advanced Space Technologies for Robot. Autom. Noordwick, The Netherland (1998)
2. Berhault, M., Huang, H., Keskinocak, P., Koenig, S., Elmaghraby, W., Griffin, P., Kleywegt, A.: Robot exploration with combinatorial auctions. In: Proceedings of the International Conference on Intelligent Robots and Systems, pp. 1957–1962 (2003)
3. Bernardine Dias, M., Stentz, A.: A market approach to multi-robot coordination. Tech. Rep., Carnegie Mellon University (2001)
4. Botelho, S.C., Alami, R.: M+: a scheme for multi-robot cooperation through negotiated task allocation and achievement. In: Proc. IEEE Int. Conf. Robot. Autom. (ICRA), pp. 1234–1239 (1999)
5. Brumitt, B.L., Stentz, A.: GRAMMPS: a generalized mission planner for multiple mobile robots in unstructured environments. In: Proc. IEEE Int. Conf. on Robot. Autom., pp. 1564–1571. Leuven, Belgium (1998)
6. Caloud, P., Choi, W., Latombe, J.C., Pape, C.L., Yim, M.: Indoor automation with many mobile robots. In: Proc. IEEE Int. Workshop on Intelligent Robots and Syst. (IROS), pp. 67–72 (1990)
7. Carley, K.M., Gasser, L.: Computational organization theory. In: Lawrence Earlbaum Associates. Hillsdale, NJ (1994)
8. Castelpietra, C., Iocchi, L., Nardi, D., Piaggio, M., Scalzo, A., Sgorbissa, A.: Communication and coordination among heterogeneous mid-size players: ART99. Lect. Notes Comput. Sci. **2019**, 86–95 (2001)
9. Dias, M.B.: TraderBots: a new paradigm for robust and efficient multi-robot coordination in dynamic environments. PhD Thesis, Robotics Institute, Carnegie Mellon University (2004)

10. Dias, M.B., Stentz, A.: A free market architecture for distributed control of a multi-robot system. In: Proc. 6th Int. Conf. on Intelligent Autonomous Syst. (IAS-6), pp. 115–122 (2000)
11. Dias, M.B., Zinck, M., Zlot, R., Stentz, A.: Robust multi-robot coordination in dynamic environments. In: Proc. of IEEE Int. Conf. Robot. Autom. (ICRA), pp. 3435–3442 (2004)
12. Elmogy, A.M., Karray, F.O., Khamis, A.M.: Auction-based consensus mechanism for cooperative tracking in multisensor surveillance systems. Journal of Advanced Computational Intelligence and Intelligent Informatics (JACIII), vol. 14, no. 1 (2010)
13. Gale, D.: The Theory of Linear Economic Models. McGraw-Hill Book Company, Inc., New York (1960)
14. Gerkey, B.P., Mataric, M.J.: Sold!: auction methods for multi-robot control. IEEE Trans. Robot. Autom. (Special Issue on Multi-Robot Systems) **18**(5), 758–768 (2002)
15. Gerkey, B.P., Mataric, M.J.: A framework for studying multi-robot task allocation. In: Schultz, A.C., and others (eds.) Multi-Robot Syst.: From Swarms to Intelligent Automata, vol. 2, pp. 15–26. Kluwer Academic Publishers, The Netherlands (2003)
16. Gerkey, G.P., Mataric, M.J.: A formal analysis and taxonomy of task allocation in multi-robot systems. Int. J. Rob. Res. **23**(9), 939–954 (2004)
17. Gerkey, B., Vaughan, R., Howard, A.: The player/stage project: tools for multi-robot and distributed sensor systems. In: Proc. 11th Int. Conf. on Advanced Robotics, pp. 317–323 (2003)
18. Horling, B., Lesser, V.: A survey of multi-agent organizational paradigms. Knowl. Eng. Rev. **19**(4), 281–316 (2004)
19. Kalra, N., Ferguson, D., Stentz, A.: Hoplites: a market-based framework for complex tight coordination in multi-robot teams. In Proc. IEEE Int. Conf. Robot. Autom. (ICRA), pp. 1170–1177 (2005)
20. Kumar, V., Rus, D., Singh, S.: Robot and sensor networks for first responders. IEEE Pervasive Computing **3**(4), 24–33 (2004)
21. Lerman, K., Jones, C., Galstyan, A., Mataric, J.: Analysis of dynamic task allocation in multi robot systems. Int. J. Rob. Res. **25**(3), 225–241 (2006)
22. Manisterski, E., David, E., Kraus, E., Jennings, N.: Forming efficient agent groups for completing complex tasks. In: Proc. 5th Int. Conf. on Autonomous Agents and Multi-Agent Syst., pp. 834–841. Hakodate, Japan (2006)
23. Martinez-Jaramillo, S., Tsang, E.P.K.: An heterogeneous, endogenous and coevolutionary GP-based financial market. IEEE Trans. Evol. Comput. **13**(1), 33–55 (2009)
24. Miao, Y.Q., Khamis, A., Kamel, M.: Coordinated motion control of mobile sensors in surveillance systems. International Conference on Signals, Circuits and Systems (SCS'09). Djerba, Tunisia (2009)
25. Ortiz, C.J., Vincent, R., Morisset, B.: Task inference and distributed task management in the Centibots robotic system. In: Proc. 4th Int. Joint Conf. on Autonomous Agents and Multi-Agent Syst., pp. 860–867. Utrecht University, The Netherlands (2005)
26. Parker, L.: Alliance: an architecture for fault-tolerant multi-robot cooperation. IEEE Trans. Robot. Autom. **14**(2), 220–240 (1998)
27. Quinonez, Y., de Lope, J., Maravall, D.: Cooperative and competitive behaviors in a multi-robot system for surveillance tasks. EUROCAST 2009, LNCS 5717, pp. 437–444. Springer-Verlag, Berlin (2009)
28. Rothkopf, M.H., Pekec, A., Harstad, R.M.: Computationally manageable combinational auctions. Manage. Sci. **44**(8), 1131–1147 (1998)
29. Simmons, R., Apfelbaum, D., Fox, D., Goldman, R., Haigh, K., Musliner, D., Pelican, M., Thrun, S.: Coordinated deployment of multiple heterogeneous robots. In: Proc. Intelligent Robots and Syst. (IROS), pp. 2254–2260. Takamatsu, Japan (2000)
30. Tang, F., Parker, L.: A complete methodology for generating multi-robot task solutions using ASyMTRe-D and market-based task allocation. In: Proc. IEEE Int. Conf. On Robot. Autom. Roma, Italy (2007)
31. Tang, F., Saha, S.: An anytime winner determination algorithm for time-extended multi-robot task allocation. In: Proc. Int. Conf. on Automation, Robotics, and Control Syst., pp. 123–130 (2008)
32. Tsai, H.K., Yang, J.M., Tsai, Y.F., Kao, C.Y.: An evolutionary algorithm for large traveling salesman problems. IEEE Trans. Syst. Man, Cybern. Part B, Cybern. **34**(4), 1718–1729 (2004)
33. Tseng, Y., Wang, Y., Cheng, K., Hsieh, Y.: iMouse: an integrated mobile surveillance and wireless sensor system. IEEE Comp. Soc. **40**(6), 60–66 (2007)

34. Valera, M., Velastin, S.A.: Intelligent distributed surveillance systems: a review. IEE Proc. Vis. Image Signal Process. **152**(2), 192–204 (2005)
35. Vaughan, R.: Massively multi-robot simulation in stage. J. Swarm Intell. **2**(2–4), 189–208 (2008)
36. Wurll, C., Henrich, D., Worn, H.: Multi-goal path planning for industrial robots. In: International Conference on Robotics and Application (RA99). Santa Barbara, USA (1999)
37. Xie, X.F., Liu, J.: Multiagent optimization system for solving the traveling salesman problem (TSP). IEEE Trans. Syst. Man Cybern. Part B, Cybern. **39**(2), 489–502 (2009)
38. Zaki, B., Gammoudi, M.: Decentralized method for complex task allocation in massive MAS. In: Proc. Intelligent Syst. and Automation: 1st Mediterranean Conf. on Intelligent Systems and Automation (CISA 08), vol. 1019, pp. 287–293 (2008)
39. Zheng, X., Koenig, S., Tovey, C.: Improving sequential single-item auctions. In: Proceedings of the IEEE International Conference on Intelligent Robots and Systems (IROS), pp. 2238–2244 (2006)
40. Zlot, R., Stentz, A.: Market based multirobot coordination for complex tasks. Int. J. Rob. Res. **25**(1), 73–101 (2006)
41. Zou, Y., Chakrabarty, K.: Distributed mobility management for target tracking in networks. IEEE Trans. Mob. Comput. **6**(8), 872–887 (2007)