ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/272829555

Multi-Robot Coordination in Complex Environment with Task and Communication Constraints

Article *in* International Journal of Advanced Robotic Systems · May 2013

DOI: 10.5772/54379

CITATIONS
21

READS
376

5 authors, including:

Yabo Liu
Zhejiang University
6 PUBLICATIONS **51** CITATIONS
SEE PROFILE

Yao Zheng
Zhejiang University
436 PUBLICATIONS **3,147** CITATIONS
SEE PROFILE

Z. Wu
533 PUBLICATIONS **11,628** CITATIONS
SEE PROFILE

Min Yao
Zhejiang University
84 PUBLICATIONS **604** CITATIONS
SEE PROFILE

All content following this page was uploaded by Yao Zheng on 27 February 2015.

The user has requested enhancement of the downloaded file.

INTECH
open science | open minds

OPEN ACCESS
ARTICLE

International Journal of Advanced Robotic Systems

Multi-Robot Coordination in Complex Environment with Task and Communication Constraints

Regular Paper

Yabo Liu[1], Jianhua Yang[2,*], Yao Zheng[1], Zhaohui Wu[2,3] and Min Yao[3]

1 School of Aeronautics and Astronautics, Zhejiang University, P. R. China
2 The Sci-Tech Academy, Zhejiang University, P. R. China
3 College of Computer Science and Technology, Zhejiang University, P. R. China
* Corresponding author E-mail: jhyang@zju.edu.cn

Received 22 Jul 2012; Accepted 15 Oct 2012

DOI: 10.5772/54379

© 2013 Liu et al.; licensee InTech. This is an open access article distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/3.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Abstract The tasks would fail to be assigned to any robots in the task allocation phase as a consequence of the inherent communication constraints in multi-robot systems (MRS). This negative effect becomes even more serious in tasks with temporal constraints. We therefore propose the constraint-based approach (CoBA), a market-based task allocation approach to enable multi-robot coordination in domains with temporal constraints between subtasks of a complex task and network constraints between robots. We handle network constraints by having each robot maintain a dynamic acquaintance network of robots that it knows about, and allowing a robot to submit a bid on behalf of another robot during a task auction ("indirect bidding"). In order to model the complex task, we introduce the AND/OR task tree with temporal constraints. An auction-clearing routine, which supports the AND/OR task tree with temporal constraints and direct/indirect task auction, is proposed to enable effective multi-robot task allocation in spite of various constraints. The solution was validated in both simulation and physical environments by a series of experiments in disaster response domains. Specifically, we study the system performance by separately varying the number of robots, the expected rate of task issuance, the communication reliability factor, the compositions of MRS, as well as the acquaintance relationship parameter, in simulation experiments. The results suggest that our solution outperforms others, that is, robots were able to complete the tasks more promptly and effectively.

Keywords Multi-Robot System, Coordination and Cooperation, Task Allocation

1. Introduction

It seems that robots, which are being charged with increasingly varied and difficult tasks, will become a ubiquitous part of our daily lives in the future [1]. Researchers generally agree that multi-robot systems (MRS) with inherently distributed character may behave more robustly and effectively, and accomplish coordinated tasks that are not possible for single robot systems [2-5]. Gustafson [6] has suggested that blindly increasing robots, targets and sensor range will be harmful without efficient cooperation models. Thus, suitable strategies are required to organize and control these systems. The problem of assigning tasks to a group of robots is a fundamental issue for MRS and several techniques have been studied [7, 8]. A multi-robot robot system can be considered as a multi-agent system (MAS); the techniques for coordination and cooperation in MAS are often not well suited for dealing with the typical uncertainty and modelling incompleteness of robotics [9, 10]. The problem becomes more difficult when a cooperative situation must accommodate various constraints on the goals, tasks, resources and robots themselves. Hence, this topic has received significant and increasing interest in the research community.

In many domains, tasks are temporally constrained with respect to one another [11, 12]. In the case of partially ordered tasks, a marked-based central allocator can be used to auction only when those tasks' predecessors have been completed [13, 14]. Robots can alternatively incorporate the cost of meeting constraints into their bids during assignment [15]. Frequently, robots must also coordinate during execution to reschedule and accommodate changes that occur after the initial allocation [16]. Petri-net-based cooperative frameworks [17-19] are also sometimes used owing to the Petri net's inherent capabilities of handling concurrency. In addition, a complex task can be introduced using a task tree representation, and subtasks are related with logic operators, precedence constraints and joint actions [20].

Recent works have focus on the time-extended allocation in domains where the constraints introduce coupling between robots' schedules. A time-extended multi-robot coordination method was proposed by Jones *et al.* for domains with intra-path constraints [21]. They used tiered auctions and two heuristic techniques (i.e., clustering and opportunistic path planning) to perform a bounded search of possible time-extended schedules and allocations. The problem raised by temporal constraints between tasks in the COMETS project was addressed by dynamically specifying temporary hierarchies among the tasks [22]. These researchers chose to put the constraint "execution around date d" on the tasks, and constraint-link two tasks (i.e., mother/child tasks). Koes *et al.* described a Constraint Optimization Coordination Architecture (COCoA) which mapped the team planning problem to a mixed integer linear programming problem and then combined domain specific heuristics and standard integer programming algorithms to efficiently generate optimal or near optimal plans [23, 24]. Kalra *et al.* employed a market-based framework named Hoplites to provide tight coordination and planned coordination in security sweep domain [25].

A vast majority of coordination approaches strictly depend on team-wide (i.e., global) connectivity, while robots also face communication challenges. The cooperative back-off adaptive scheme (COBOS) was proposed for task allocation amongst a team of heterogeneous robots in regions with limited communication ranges, i.e., disjoint sub-networks may exist at different locations [26]. Therefore, in the absence of team-wide communications robots are still able to respond appropriately. Frequently, few domains are so benign that the initial plans can be successfully executed, and the plans must be revised during execution while communication failures hamper this replanning. Hence, care must be taken to minimize disruption to the schedule of other robots when the fractured subteam carries out replanning [27, 28]. On the one hand, it is sometimes worth trying to maintain connectivity while performing the assigned task [29-31] or using techniques with no direct inter-robot communication [32].

Some work has also been carried out on solving the constraints in coalition formation [33-35], path/trajectory schedule [36, 37], cooperative transport [38], pursuit-evasion [39], etc. However, until now there are very few studies simultaneously addressing cases where tasks have temporal constraints and where robots have communication constraints. We consider a scenario where complex tasks are perceived by robots when they observe new information about their surroundings. These tasks are interdependent and temporally constrained and are distributed over the agents. Moreover, the broadcasting of each robot is limited and finite, since the dynamic environment is divided into several disjoint sub-networks (or sections). Robots in a sub-network are able to communicate with each other, directly or indirectly, but not with the rest of the team outside the sub-network.

Most existing works implicitly assume that there are always some robots to which current tasks can be assigned (sooner or later); however, no work has been done so far to explicitly discuss a situation where sufficient robots cannot be found to perform the current task. We therefore introduce the terminology *candidate absent* to refer to the situation where robots able to implement the task in question alone or with other robots are unable to receive the message of task announcement due to the limitation of network range. Consequently, tasks that could be successfully completed can fail to be assigned to any robots in the task allocation phase; this challenge is common but unfortunately disregarded. This detrimental effect becomes more serious in tasks with temporal constraints, because the delay of a basic task may result in the failure of the whole complex task.

Our market-based solution to multi-robot coordination, which we call CoBA (Constraint-based Approach for MRS), enables tasks to be traded among a team of heterogeneous robots in a complex environment with task and communication constraints. More specifically, complex tasks are modelled with an AND/OR task tree with temporal constraints and traded at variable levels of abstraction. Given a task node at a certain level, the robot supports both direct evaluation of whether it can accomplish the node by itself and indirect evaluation of whether one of its acquaintances is able to do so. As a result, the mechanism of the acquaintance network indirectly involves robots in other sub-networks to enter the phase of task allocation. Then, an auction clearing routine is provided to find the optimal allocation of the tasks to the robots when all evaluations have been received or a preset auction deadline elapses. We also implement a preliminary version of an accountability system, in which the manager of every task takes responsibility for dynamic constraint and satisfaction of the corresponding tasks. The CoBA was validated in both simulation and physical environments by using a series of experiments in disaster response domains.

The rest of this paper is structured as follows. Section 2 briefly illustrates the problem with task and communication constraints. Section 3 describes a model of a dynamic acquaintance network. Section 4 presents detailed discussion on the task model and evaluation. An auction clearing routine which is executed to find the optimal allocation of the offered tasks to the bidders is also introduced in section 4. Simulated and experimental evaluations of the approach are given in section 5, and we conclude the paper with a summary and a description of the outlook for the future in section 6.

2. Problem Statements

Market-based techniques [14, 40] have been used by many researchers [5, 20, 25, 29, 41-44] to solve multi-robot coordination problems since Smith [45] first introduced the Contract Net Protocol, addressing the problem of how agents can negotiate to collectively solve a set of tasks. In a market economy, robots receive revenue and incur costs when trying to accomplish tasks, and act as greedily as possible to maximize their individual rewards and minimize their individual costs. The goal for robots is to trade tasks through auctions/negotiations so that self-interests lead to globally efficient solutions. A technical goal of the market-based approach is to opportunistically optimize resource utilization.

The traded tasks in the market can be simple tasks or complex tasks. In this paper, we focus on a complex task, which is further decomposed into task trees. A task tree is defined as a rooted set of task nodes connected by directed edges that specify parent-child and constraint relationships between the tasks. Contracts can be sold for executing tasks at variable levels of abstraction, and every robot can perform its own decomposition of the abstract task when the new decomposition is better. Given the mentioned aspects, we can pose the cooperation scenario as the following optimization problem.

Definition 1 (*Multi-robot coordination*) Given a complex task T, let $\mathcal{T}$ be a set of task tree representation of T, $NODES(\mathcal{T}^i)$ represent the nodes of tree $\mathcal{T}^i$ ($i = 1, \dots, |\mathcal{T}|$), and $\mathcal{C}_{\mathcal{T}^i}$ represent the temporal constraints in tree $\mathcal{T}^i$; given a set of robots R, let $\mathcal{R} = 2^R$ the set of all possible robot sub-teams. For each $\mathcal{T}^i \in \mathcal{T}$, a set of robots $\mathcal{R}_j$ ($j = 1, \dots, |\mathcal{R}|$) bid and negotiate to carry out the sub-trees in $\mathcal{T}^i$, a cost function $\mathcal{U}_{\mathcal{R}_j}$: $2^{NODES(\mathcal{T}^i)} \longrightarrow \mathbb{R}^+ \cup \{\infty^+\}$ for each subset of robots $\mathcal{R}_j$ specifying the utility of performing each sub-tree of tasks, which measures the quality and cost factors of the resulting actions. Hence, if we define $\mathcal{A}_{\mathcal{R}}^{NODES(\mathcal{T}^i)}$ as the set of all possible allocations of the task tree $\mathcal{T}^{\mathrm{i}}$ to the team of robots R, the goal is to find an allocation $A \in \bigcup_{i=1,\dots,|\mathcal{T}|} \mathcal{A}_{\mathcal{R}}^{NODES(\mathcal{T}^i)}$ which assigns all tasks to the robots to maximize the global objective function $\mathcal{U}\colon \bigcup_{i=1,\dots,|\mathcal{T}|} \mathcal{A}_{\mathcal{R}}^{NODES(\mathcal{T}^i)} \longrightarrow \mathbb{R}^+ \cup \{\infty^+\}$ subject to the constraints $\mathcal{C}_{\mathcal{T}^i}$.

A team of robots will perform more efficiently if team members can communicate with each other [46]. However, communication failures occur in a variety of domains, especially in outdoor environments, and range from occasional loss of messages to complete loss of touch. Most recent approaches for MRS suppose present team-wide communication; even if the packet loss is considered, a simple acknowledgment protocol or reallocation of tasks is often applied, which may allocate one task to many robot teams in duplicate. In this paper, we deal with a situation where the robot team is partitioned into several disjointed sections.

In order to deal with the candidate absent problem, we further extend the market-based approach [20] to support task tree auction with *direct* and/or *indirect* bids. Once a new task is found by a robot, it decomposes the task into a local task tree, refining an abstract task into a set of hierarchical and temporal constrained tasks. As soon as the decomposition is complete, the robot holds a task tree auction, expecting to distribute the tasks among the team and potentially to allow other robots to perform their own re-decomposition of the tasks appropriately.

On hearing the auction announcement, each candidate robot performs direct and/or indirect evaluation of tasks. The direct evaluation is performed for the robot itself while the indirect evaluation is performed on behalf of other robots. Specifically, upon direct evaluation, a robot uses the individual utility function specified for each robot to quantify the robot's preferences for its individual resource usage and contribution to the team, given its current state. Given a task node T at an arbitrary level of abstraction, the direct bid submitted by robot r is given by:

$$b_r(T) = \min\{c_r^{val}(T), c_r^{dec}(T)\} \qquad (1)$$

where $c_r^{val}(T)$ is the cost of completing T by robot r based on the auctioneer's original decomposition, while $c_r^{dec}(T)$ is based on the bidder's new decomposition. Note that the task node T may be still a complex task with several temporal constraints, except with leaf nodes.

An indirect evaluation is then performed, but only when the direct evaluation is infinite, that is, the robot cannot complete the task on its own. We suppose that some robots inside its dynamic acquaintance network, who may be not able to hear the auction announcement, are competent for the task. The indirect bid of task T submitted by robot r is given by:

$$b'_r(T) = \min_{s \in AC_r}\left\{\frac{c_s(T)+c_r(s)}{(a_{rs})^2}\right\} \qquad (2)$$

where a_{rs} is the degree of acquaintanceship between robot r and robot s, $c_s(T)$ is the estimated cost of complete task T by robot s, $c_r(s)$ is the additional cost for robot r to subcontract the task to robot s, and AC_r is the dynamic acquaintance network maintained by robot r. Actually, locally each robot maintains a dynamic acquaintance network, which is a dynamically updated knowledge base that describes the other robots' sensing and behavioural capabilities, as well as action plans. The degree of acquaintanceship reflects the level of dependability of knowledge. More details on the acquaintance network will be discussed in the next section.

The indirect evaluation is only a potential suggestion, and considered inferior to direct evaluation. In fact, some partially malfunctioning robots which still can move may devote themselves to some particularly tough tasks while passing other tasks to potential robots in other disjoint sub-networks, acting as relay robots.

An auction clearing routine is executed to find the optimal allocation of the offered tasks to the bidders when all bids have been received or a preset auction deadline elapses. Hence, robots are able to response appropriately in the absence of team-wide communication, as illustrated in Figure 1. In this simple example, robot A initially finds a complex task to be executed in location L_5, and then publishes it to all the robots in the network section I, i.e., robots B and C. However, the three robots in the current section cannot complete this task without other robots' help, and all the other robots are outside of this network section. Consequently, robot B resorts to its acquaintance network and invites robot D to their mission. Eventually, these robots successfully carry out the complex task through the cooperation.

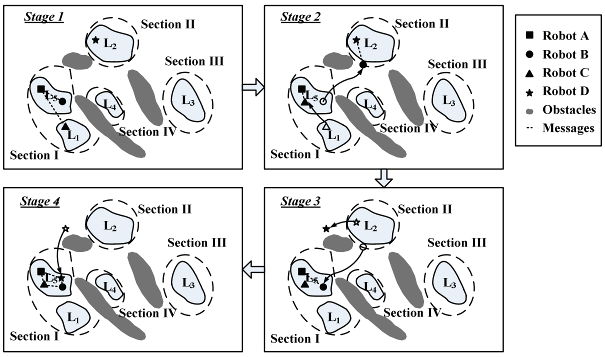


Figure 1. A series of stages of multi-robot cooperation in an environment with task and communication constraints. Different shapes represent the heterogeneity of the robots. The arrow that connects open and filled shapes indicates the movement of robots.

3. Dynamic Acquaintance Network

The environment in which robots operate is modelled as several disjoint sub-networks (or sections). Robots in a sub-network are able to communicate with each other, directly or indirectly, but not with the rest of the team outside the sub-network.

Definition 2 (*Disjoint Sub-networks*) Let $N = \{N_j | N_j \in W, \text{for } j = 1, \cdots, n_{sn}\}$ be the disjoint sub-networks of the workspace W, and n_{sn} be the number of sub-networks.

Definition 3 (*Candidate Absent*) Given a complex task T, and $L_T \in N_j$, where L_T is the location of task T. An allocation of a set of tasks T to R is a function, $F: T \rightarrow 2^R$ mapping each task to a subset of robots responsible for completing it. We define the set of all possible allocations of the tasks T to the team of robots R as $\bigcup_{i=1,\ldots,|\mathcal{T}|} \mathcal{A}_{\mathcal{R}}^{NODES(\mathcal{T}^i)}$, where $\mathcal{R} = 2^R$. Equivalently, $\bigcup_{i=1,\ldots,|\mathcal{T}|} \mathcal{A}_{\mathcal{S}}^{NODES(\mathcal{T}^i)}$ is the set of all allocations of the tasks T to the team of robots S_j^t in sub-network N_j at time t, where $\mathcal{S} = 2^{S_j^t}$. It is called candidate absent for task T at time t when $\bigcup_{i=1,\ldots,|\mathcal{T}|} \mathcal{A}_{\mathcal{S}}^{NODES(\mathcal{T}^i)} = \emptyset$ and $\bigcup_{i=1,\ldots,|\mathcal{T}|} \mathcal{A}_{\mathcal{R}}^{NODES(\mathcal{T}^i)} \neq \emptyset$ hold.

Therefore, we put forward a dynamic acquaintance network to facilitate the maintenance of inter-robot relationships, and the degree of acquaintanceship can be used in the indirect task bid. As mentioned in the previous section, each robot locally maintains a dynamic acquaintance network, a dynamically updated knowledge base that describes the other robots' sensing and behavioural capabilities and action plans. As a result, the mechanism of acquaintance network indirectly involves robots in other sub-networks to enter the task allocation phase.

Definition 4 (*Dynamic Acquaintance Network*) Let dynamic acquaintance network $G = (V, E, A)$ be a weighted digraph (or directed graph) of order n with the set of nodes $V = R = \{r_1, \cdots, r_n\}$, set of edges $E \subseteq V \times V$, and a weighted adjacency matrix $A = [a_{ij}]$ with non-negative adjacency elements a_{ij}. An edge of G is denoted by $e_{ij} = (r_i, r_j)$, representing the acquaintanceship between robots.

The adjacency elements associated with the edges of the graph are positive, representing the degree of acquaintanceship, and $a_{ij} \in [0,1]$. For notational simplicity, let $e_{ii} \in E$ and $a_{ii} = 1$ for $i = 1, \cdots, n$, that is, a robot is its own acquaintance. Hence, every robot r_i locally maintains the matrix $A[i,\cdot]$ and the set of acquaintances of robot r_i is denoted by $G_i = \{r_j | r_j \in R \wedge a_{ij} > 0\}$.

The acquaintance network simply defines the awareness links that exist between robots; an arc $\mathrm{r_i} \rightarrow \mathrm{r_j}$ indicates that robot $\mathrm{r_i}$ is aware of robot $\mathrm{r_j}$ with the degree $\mathrm{a_{ij}}$, but not necessarily that robot $\mathrm{r_j}$ is aware of robot $\mathrm{r_i}$. Hence,$\mathrm{e_{ij}} \in E \nRightarrow \mathrm{e_{ji}} \in E$ and $\mathrm{e_{ij}} \in E \wedge \mathrm{e_{ji}} \in E \nRightarrow \mathrm{a_{ij}} = \mathrm{a_{ji}}$, because acquaintanceships are not always symmetrical. For example, the degree of acquaintanceship between r_8 and r_{14} in Figure 2 (top right corner) are 0 and 0.91, respectively.

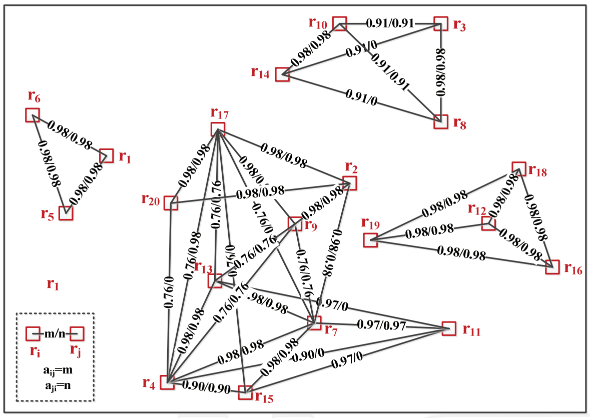


Figure 2. An illustration of a dynamic acquaintance network with 20 robots. The squares represent robots, and lines between robots represent acquaintanceships. The degrees of acquaintanceships are marked on corresponding lines.

Robots use Algorithm 1 for the exchange and management of dynamic acquaintance network among robots to enlarge the acquaintance scope. The degree of acquaintanceship declines over time according to a sigmoid function, which is defined as follows:

$$z(t) = \frac{1}{1+e^{\left(-2\frac{-t+\mu_1}{\delta_1}\right)}} \tag{3}$$

where (μ_1, σ_1) are adjustable parameters and the argument t is the elapsed time since the direct establishment of acquaintanceship between two robots. When the degree of acquaintanceship drops below a threshold τ, the robot will give up the acquaintanceship with the corresponding robot. According to equation (3) and Algorithm 1, the *expiration interval* of the acquaintance relationship is then calculated by:

$$\Delta T = \mu_1 + \frac{1}{2} \cdot \sigma_1 \cdot \ln\left(\frac{1}{\tau} - 1\right) \tag{4}$$

```
Robot $r_i$ broadcasts $G_i$ when it is arriving
  at or departing from sub-network $N_j$;
for all $r_k \in S_j^t$ do
    Robot $r_k$ and $r_i$ mark each other as acquaintances
    and update the information;
    $a_{ki} = a_{ik} = 1$;
    for all $r_m \in G_i$ do
        if $a_{km} < a_{im}$ then
            Robot $r_k$ marks $r_m$ as its acquaintance
            and updates the information;
            $a_{km} = a_{im}$;
        end if
    end for
end for
When $a_{xy}$ drops below the specified threshold $\tau$
  (e.g., 0.2), the robot $r_x$ will give up the acquaintance
  relationship with the corresponding robot $r_y$.
```

Algorithm 1. Management of dynamic acquaintance network

4. Coordination with Complex Task

4.1 Task Model and Evaluation

Complex tasks are modelled as AND/OR task trees with temporal constraints, which is inspired by the work of Zlot and Stentz [20]. The tree is defined as a rooted set of task nodes connected by directed edges that specify parent-child and constraint relationships between the tasks, as illustrated in Figure 3. To execute an AND task, all of its subtasks must be executed; to execute an OR task, any one of its subtasks may be executed. Each successive level of the tree represents a further refinement of a complex task: the root is the most abstract description, and the lowest level contains basis tasks (or actions) that can be executed directly in the system. As mentioned in the previous section, every robot may have its own decomposition of the abstract task. For example, robot *A* may decompose the task t_{32} into e_1 and e_2 while robot *B* may decompose the same task into f_1 and f_2, as shown in Figure 3.

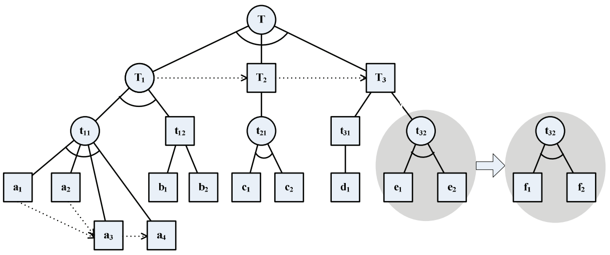


Figure 3. An example of an AND/OR task tree with temporal constraints. At the AND nodes, shown as circles, every subtask must be dealt with, as indicated by the arc linking the subtasks. Square nodes are OR nodes where any one of the subtasks may be executed. The dotted lines with arrow ends represent various temporal constraints, and the shaded zones show different decompositions of the task node.

To stop the problem becoming excessively complicated, the temporal relationships are restricted in siblings under the AND node in the task tree, that is, if task A and B are siblings under OR nodes or from different parents, then there are no direct temporal constraints upon them. The temporal constraints used are qualitative interval constraints and qualitative interval-point constraints [47].

The combined costs of a node's children are calculated according to the node's type. If we let S be the set of children of a task node N_p, $op(x)$ be the node type of x, $P(x)$ be the lowest cost bid, and $P'(x)$ be the lowest indirect cost bid, then

$$P(S) = \begin{cases} \sum_{C\in S} P(C) & \text{if } op(N_p) = \text{AND}, \\ \min_{C\in S} P(C) & \text{if } op(N_p) = \text{OR}. \end{cases} \tag{5}$$

$$P'(S) = \begin{cases} \sum_{C\in S} \min\,(P(C), P'(C)) & \text{if } op(N_p) = \text{AND}, \\ \min_{C\in S} P'(C) & \text{if } op(N_p) = \text{OR}. \end{cases} \tag{6}$$

In order to determine whether to accept the combination of the node's children or the node itself at each level, we can compare the costs between them. When any of the following equations (7)(8)(9) holds true, the node itself can be considered to have an advantage over the combination of its children, and $P(N_p)$ and $P'(N_p)$ remain unchanged.

$$P(S) \neq \infty^+ \wedge P(N_p) < P(S) \text{ [condition 1]} \tag{7}$$

$$P(S) = \infty^+ \wedge P(N_p) \neq \infty^+ \text{ [condition 2]} \tag{8}$$

$$P(S) = \infty^+ \wedge P(N_p) = \infty^+ \wedge P'(N_p) < P'(S) \text{ [condition 3]} \tag{9}$$

Otherwise, the combination of the children nodes should be chosen over the parent node; meanwhile, $P(N_p)$ and $P'(N_p)$ are reassigned with $P(S)$ and $P'(S)$, respectively. $P(X) \neq \infty^+$ indicates that at least one feasible solution exists for X, because the bid (i.e., cost) for X is initialized to infinity and the value remains fixed when no robots bids for it.

Algorithm 2 illustrates a task tree traversal algorithm, where *in-degree* $R^+(t)$ is the number of constraints that end at t and *out-degree* $R^-(t)$ is the number of constraints that start at t. In the traversal procedure, we firstly calculate the priority level $\phi(N)$ of task node N to evaluate the constraint conditions, and then determine the lowest cost of task nodes and the corresponding robots to carry out the task nodes.

```
Firstly breadth-first traversal of the tree, and set
$\Phi(N) := 10 \cdot R^+(parent(N)) + R^+(N)$ for every task node $N$;
while $T \neq \emptyset$ do
    Let $L$ be the set of tasks on leaves;
    $N_{ms} := \{N \mid \Phi(N) = \max_{M\in L} \Phi(M), N \in L\}$;
    for all $N_{min} \in N_{ms}$ do
        Let $N_p$ be the parent of $N_{min}$, and $S$
        be the set of children of $N_p$;
        Determine whether to accept $S$ or $N_p$;
        $T := T - S$;
    end for
end while
```

Algorithm 2. Traversal of task tree *T*

Although the issue of job constraints is addressed by the scheduling literature, the addition of such constraints generally increases the difficulty of the problem, and tractable algorithms exist for only the simplest kinds of constraints [7]. A possible way to approach this problem is using techniques for dynamic constraint satisfaction. In this paper, we implement a preliminary version of an accountability system. For example, as shown in Figure 3, suppose that the manager of the task *T* is robot *R*₀, and robot *R*₁ locally form a subgroup together with other robots. They win the sub-tree *T*₁ (including node *T*₁ and its descendant nodes), *R*₂ wins *T*₂, and *R*₃ wins *T*₃, respectively. Then, *R*₁ is responsible for supervising the execution or reallocation of the subtasks of *T*₁ when necessary. When all sub-tasks of *T*₁ are successfully finished, *R*₁ will report the completion of *T*₁ to *R*₀ because *R*₀ is the manager of task *T*, which is the parent node of task *T*₁. *R*₀ is aware of the fact that *R*₂, which is the manager of task *T*₂, is waiting for *T*₁ to be finished; hence, *R*₀ will notify *R*₂ of the completion of *T*₁. Consequentially, *R*₂ receives the message and realizes that all constraints of *T*₂ are satisfied; it then begins to start *T*₂. In other words, the manager of every task should take responsibility for dynamic constraint satisfaction of the corresponding tasks.

4.2 Task Auction and Allocation

An auction clearing routine is executed to find the allocation of the offered tasks to the bidders when all bids have been received or a preset auction deadline elapses. Unfortunately, clearing a general task tree, even in a relatively simplified case, is NP-hard. Therefore, we have designed a polynomial-time heuristic auction clearing algorithm. First of all, it may be possible that a robot *r* submits an indirect bid on behalf of one of its acquaintances named robot *s*, while robot *s* is also in the same sub-network and has submitted direct bids on its own. Thus, the invalid indirect bids should be firstly cleaned up before proceeding. Afterwards, the solution is initialized by marking all the leaf nodes of the task tree as won by the auctioneer, and it is determined whether to accept the combination of the children of a node or the node itself at each level through a traversal according to Algorithm 2.

Finally, there are dependencies between the prices of any combination of awardable bids; the algorithm should ensure that no bidder wins multiple nodes. Algorithm 3 handles examination and elimination of multiple awards according to priorities. For easy notation, we let T_R be a set of nodes in a task tree with root R, B be a set of direct bids on task nodes, $b_r(N)$ be the direct bid of robot r on node N, B' be a set of indirect bids on task nodes, and $b'_{r,s}(N)$ be an indirect bid of robot r on node N on behalf of robot s. In order to clear the entire tree, multiple auction rounds must usually be held. Award messages are sent to the auction winners after the winners are determined.

```
for all $r \in \{1, \cdots, |B|\}$ do
  Find set $W$ contains nodes which are won by bidder $r$;
  if $W \neq \emptyset$ then
    $W_{tmp} \coloneqq \{C | C \in W, \Phi(C) = \max_{w \in W} \Phi(w)\}$; //a set of nodes with maximal priority;
    $\mathrm{F_r} \coloneqq \arg\max_{w \in W_{tmp}} \left( \left( \min_{t \in \{1,\cdots,|B|\}, t \neq r} b_t(w) \right) - P(w) \right)$; //largest difference between best & second-best;
    mark the node corresponding to $\mathrm{F_r}$ in $\mathrm{T_R}$;
  end if
end for
for all $s \in \{1, \cdots, |B'|\}$ do
  Find set $W'$ contains nodes which are indirectly won by bidder $s$ via other robots;
  if $\mathrm{W'} \neq \emptyset$ then
    $W'_{tmp} \coloneqq \{C | C \in W', \Phi(C) = \max_{w \in W'} \Phi(w)\}$; //a set of nodes with maximal priority;
    $W'_{tmp2} \coloneqq \emptyset$ ;  //initialize set $W'_{tmp2}$;
    for all $w' \in W'_{tmp}$ do  //check whether the robot corresponding to $w'$ is free;
      if $\mathrm{F}_{w' \to r} = \emptyset$ then
        $W'_{tmp2} \coloneqq W'_{tmp2} \cup \{w'\}$;
      end if
    end for
    if $W'_{tmp2} = \emptyset$ then  //all the robots corresponding to nodes in $W'_{tmp}$ are not free;
      for all $w' \in W'_{tmp}$ do
        if $\Phi(\mathrm{w'}) > \Phi(\mathrm{F}_{w' \to r})$ then   //compare the priority;
          $W'_{tmp2} \coloneqq W'_{tmp2} \cup \{w'\}$;
        end if
      end for
    end if
    $\mathrm{F'_s} \coloneqq \arg\max_{w' \in W'_{tmp2}} \left( \left( \min_{t \in \{1,\cdots,|B|\}, t \neq s} b'_{r,t}(w') \right) - P'(w') \right)$;
    mark the node corresponding to $\mathrm{F'_s}$ in $T_R$;
    unmark the node corresponding to $\mathrm{F}_{\mathrm{F'_s} \to r}$ in $T_R$;
  end if
end for
```

Algorithm 3. Examination and elimination of multiple awards

When award messages arrive at bidders, robots can choose to accept or reject the awarded tasks depending on whether their own states or the states of others and the environment have significantly changed since the time of the bid. When a directly awarded task is accepted, the robot simply inserts the task into its commitments list, and re-optimizes its schedule according to the task's constraint. However, the robot is responsible for passing the indirectly awarded task to an appropriate robot in some disjoint sub-networks when this type of task is accepted. The responsible robot can easily reach the appropriate robot because it maintains a dynamic acquaintance network and knows the appropriate approximate state. Of course, the robot should assume the risks that the appropriate robot may reject the task due to significant changes in states (e.g., malfunction, unknown environment). Consequently, the responsible robot needs to seek out another appropriate candidate from its dynamic acquaintance network, or re-decompose and subcontract the task to other robots; otherwise, it will be punished. Note that a new incoming task is not the prerequisite to hold an auction; robots can subcontract or re-decompose on the task tree during execution at any moment, if necessary.

5. Experimental Results

5.1 Experiments in Simulated Environment

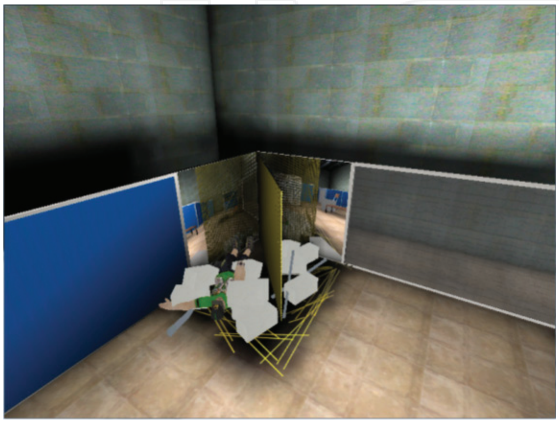
Our approach has been validated in both simulation environments and physical robots by a series of experiments in disaster response domains. In the simulation setup, a number of sites are distributed in an unknown environment. A team of heterogeneous robots is required to accomplish the tasks in a site, which may involve finding the victim, cleaning scattered garbage and obstacles, transporting the victim to safety, etc. A site is considered satisfied when all the tasks in this site are completed timely, and this requires three types of robots (TYPE1, TYPE2, and TYPE3) working together.

Performing experiments with a large team of robots, to achieve statistical significance we evaluated our approach in the realistic simulation experiments on USARSim [48]. A set of typical sites in the simulated environment is shown in Figure 4. In order to make it more suitable for our scenario, we have extended the simulation with two additional functions. Firstly, we developed a lossy communication model that generated Gaussian packet loss probability distributions for each distance. Secondly, we employed a Poisson process to represent the stochastic task arrival. The parameter λ represents the expected rate or intensity of task issuance; the number of tasks in time interval $(t, t+\tau]$ follows a Poisson probability distribution with associated parameter λτ:

$$P\left[\left(N(t+\tau) - N(t)\right) = k\right] = \frac{e^{-\lambda\tau} \cdot (\lambda\tau)^k}{k!} \qquad k = 0, 1, \cdots \quad (10)$$

where $N(t+\tau) - N(t)$ describes the number of events in time interval $(t, t+\tau]$. Here, a task arrival means the task is injected into the system and waiting to be discovered by some robots.

(a)



(b)

Figure 4. A set of typical sites in the simulated environment.

The experimental setup consisted of a 400×400 arena with three types of mobile robots, which simulate realistic Talon, P2AT, and Soryu robots, respectively. The Talon is a lightweight tracked vehicle built by Foster-Miller for missions ranging from reconnaissance and weapons delivery to rescue. The P2AT is a four-wheel drive all-terrain pioneer robot from ActivMedia Robotics. The IRS Soryu is a serpentine robot used for search and rescue, developed by the Tokyo Institute of Technology and the International Rescue System Institute, which has the ability to easily intrude narrow spaces as well as traverse rough terrain.

For a quantitative evaluation, the aforementioned scenario was run separately by varying some particular parameter settings while keeping other parameters fixed, with each persisting for 100,000 steps. The fixed parameter settings used during experiments are shown in Table 1. The first row of this table indicates that the 20 robots comprise eight TYPE1 robots, eight TYPE2 robots, and four TYPE3 robots. Each case was run 20 times to determine the mean and standard deviation of the total number of completed tasks (NCT) as well as the total number of failed tasks (NFT) in experiments. The sum of NCT and NFT is not a constant due to the stochastic task arrival; hence, they are counted separately.

Parameter	Description	Default value
n_r	number of robots	20 (8/8/4)
λ	expected rate of task issuance	8
(μ_1, σ_1)	acquaintance relationship parameter	(300,20)
(μ_2, σ_2^2)	communication reliability factor	(120,10)

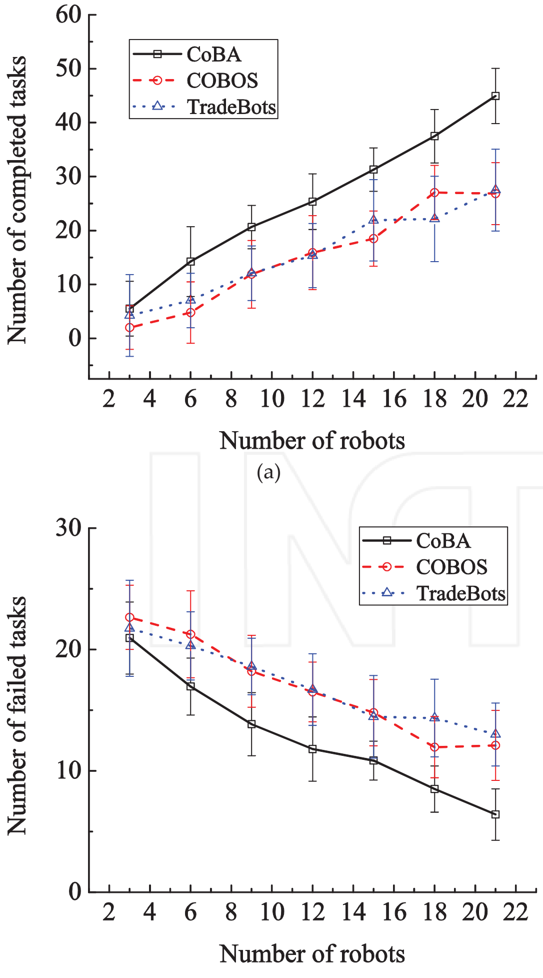
Table 1. Parameter settings used during experiments

The benefits of using our constraint-based approach for MRS were investigated in a series of experiments. This is aimed to assess the robustness of the approach in terms of different parameters and the improvement of the team's performance to other approaches. We compared this approach with other two typical approaches: TraderBots [43] and COBOS (the cooperative back-off adaptive scheme) [26]. TraderBots is a market-based coordination approach. It is inherently distributed, but also opportunistically forms centralized sub-groups to improve efficiency. The COBOS operates in regions with limited communication ranges, and is robust against robot malfunctions and uncertain task specifications, with each task potentially requiring multiple robots. Both TraderBots and COBOS are implemented in a distributed manner, and are extended to support the allocation of tasks with temporal constraints.

In Figure 5, we present the results obtained by varying the number of robots n_r involved in the tasks. The number of each type of robot is n_r/3. The results suggest that the differences between approaches can be considered negligible when the robot team is small. Not surprisingly, in Figure 5 we can see an upward trend of the percentage of successful tasks proportional to the number of robots. It shows that all three approaches are able to scale with the team size, to ensure an acceptable task assignment.

However, CoBA demonstrates significantly higher percentage of successful tasks than the other two approaches, regardless of whether the team size is small or large. The performances of COBOS and TraderBots are very similar with respect to the number of completed and failed tasks. Nevertheless, the standard deviation of COBOS is lower than that of TraderBots. This indicates that COBOS is more stable.

We measured the system performance imposed by the rate of task issuance λ, which is illustrated in Figure 6. When the tasks are sporadic, a number of robots are idle. Consequently, these idle robots can take part in the indirect auction to improve the overall solution quality. On the other hand, while the number of tasks increases to the extent that almost every robot is busy executing tasks, it is not worth carrying out the indirect auction.



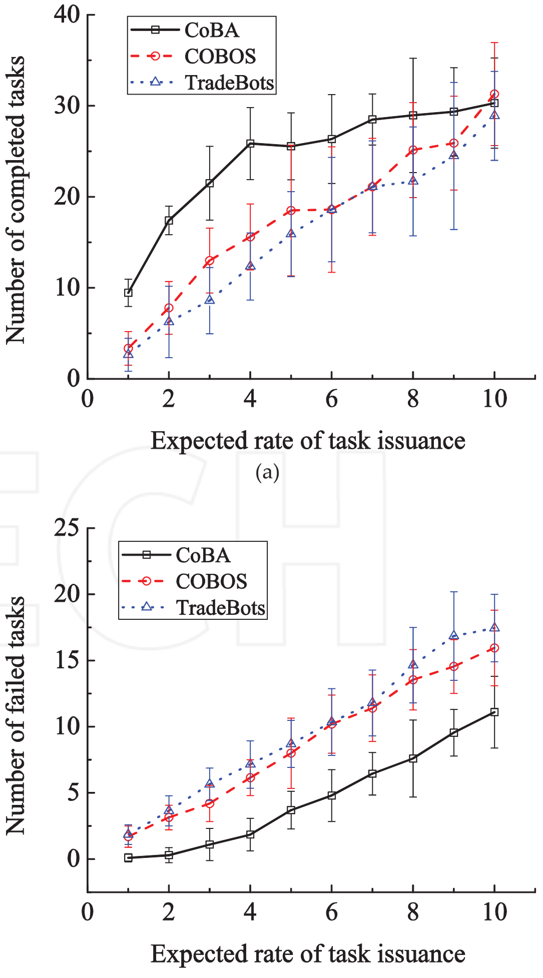
(a)

(b)

Figure 5. Comparison of solution qualities among the three approaches with changes in the number of robots n_r.

Figure 6 shows an interesting behaviour with respect to the number of failed tasks: the number of failed tasks can be reduced to a relatively low level with our approach; however, they could never be minimized to zero. Task failure occurred for two main reasons: 1) candidate absent problem; 2) tasks were not found by robots. This oversight is inevitable in a large environment because tasks are injected into the system anywhere, waiting to be discovered. This shows that, to a certain extent, our approach is able to ensure an efficient task allocation from a low up to a high expected rate of task issuance.

As shown in Figure 6, CoBA achieves and maintains the number of completed tasks at about 30 from λ equals 4 to 10. In other words, 30 is the maximum number of tasks that robots can complete under the current parameter settings. This indicates that our approach can quickly get robots busy on tasks instead of standing idle, while the other two approaches cannot fully utilize available robots until tasks are extremely saturated. This confirms the intuition that adding more tasks does not heavily influence performance in terms of number of completed tasks.



(a)

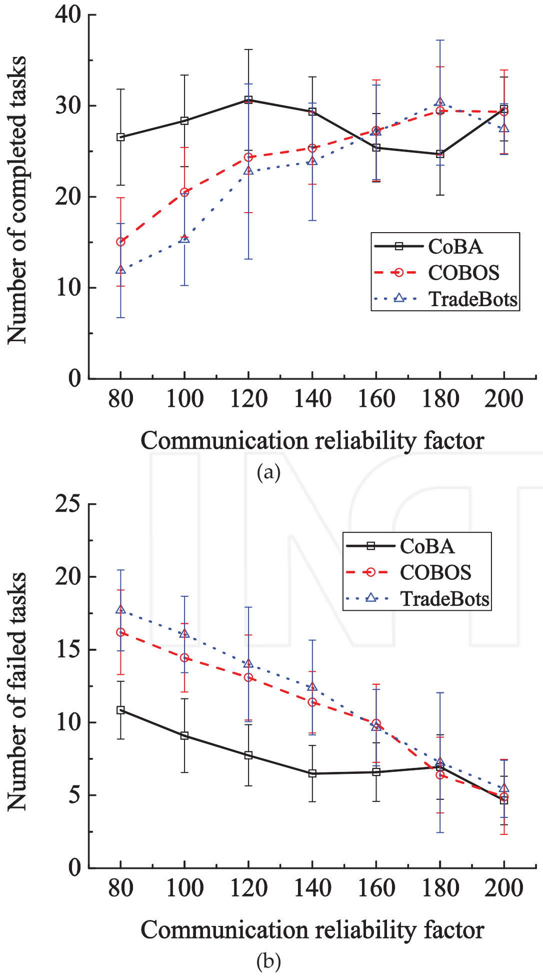
(b)

Figure 6. Comparison of solution qualities among the three approaches with changes in the expected rate of task issuance λ.

Figure 7 shows that the system performance is imposed by the parameter μ_2 of the lossy communication model. A higher communication reliability factor implies a better communication condition. We used communication reliability factors from 80 to 200 for a meaningful comparison because all three approaches are based on an intentional cooperation model. With extremely low communication reliability, robots cannot communicate, let alone cooperate. The solution quality of all approaches other than our approach is heavily influenced by the communication condition, according to the experimental results. The advantage of our approach is evident in the absence of team-wide communication.

One interesting observation is that CoBA completes fewer tasks than the other two approaches when the communication reliability factor is at 180. In fact, indirect auctions may be costly and unworthy when the communication is nearly team-wide. The candidate absent problem can be transient because some candidates may enter the current section by themselves in the near future. Hence, we can discard some costly indirect auctions unless the task is emergent to improve the overall system performance.

INTECH



(a)

(b)

Figure 7. Comparison of solution qualities among the three approaches with changes in the communication reliability factor μ_2..

It has to be emphasized that each robot may have a different work capacity. The degree of heterogeneity within an MRS can be measured in an information-theoretical way using Balch's social entropy [49]:

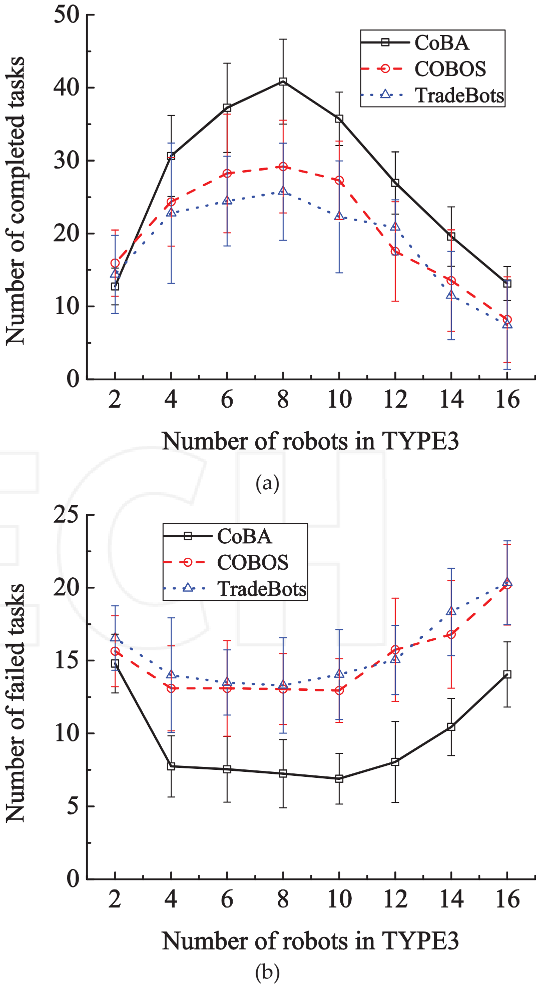
$$\mathrm{p_i} = \frac{|C_i|}{\sum_{j=1}^{c}|C_j|} \quad (11)$$

$$\mathrm{Het}(R) = -\sum_{i=1}^{c}(p_i \cdot \log_2(p_i)) \quad (12)$$

where c is the number of classes of robots and p_i is the proportion of robots in the ith class. In our experiments, these differences are the work capacity of each robot; therefore, two robots belong to the same group if they have the same work capacity.

Figure 8 reports the results obtained by varying the compositions of MRS with the total robot number n_r fixed at 20. In order to reduce the complexity, experiments were performed on eight selected compositions (the total number of compositions is 33). The eight selected compositions of MRS used are given in Table 2, and the numbers of the three types are subject to:

$$n_{\mathrm{TYPE1}} = n_{\mathrm{TYPE2}} = \frac{n_r - n_{\mathrm{TYPE3}}}{2} \quad (13)$$



(a)

(b)

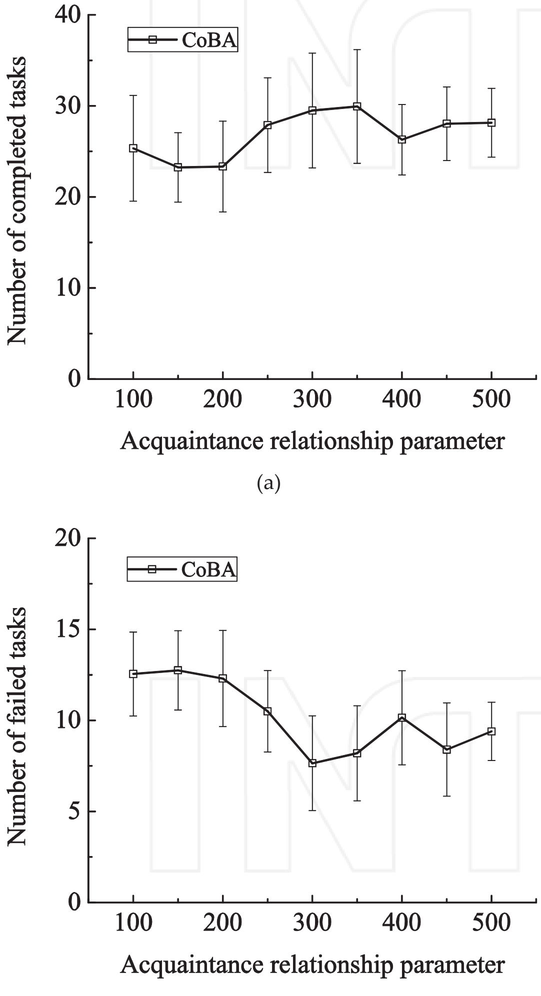
Figure 8. Comparison of solution qualities among the three approaches with changes in the compositions of MRS.

Composition of MRS (n_{TYPE1}/ n_{TYPE2}/n_{TYPE3})	9/9/2	8/8/4	7/7/6	6/6/8	5/5/10	4/4/12	3/3/14	2/2/16
Social entropy	1.37	1.52	1.58	1.57	1.50	1.37	1.18	0.92

Table 2. Difference compositions of MRS

The main goal of this experiment is to study how the percentages of the successful tasks change when the compositions of MRS vary. Figure 8 indicates that the higher the degree of heterogeneity, the better the three approaches' performance. Especially CoBA demonstrates significantly higher percentages of successful tasks than other two approaches, usually by a factor of 1.5-2, when the social entropy of MRS is above 1.5. This is explained by the fact that our approach utilizes dynamic acquaintance network to address the candidate absent problem, and the dynamic acquaintance network is more profitable when the MRS is more heterogeneous. It is interesting that, in contrast to the other two approaches, our approach performs better on the composition 2/2/16 than on the composition 9/9/2. One possible explanation for this different behaviour is that the former composition is more likely to form a useful acquaintance network than the latter one.

Finally, Figure 9 suggests that our approach is slightly affected by the acquaintance relationship parameter μ1. It appears that the parameter μ_1 should not be set too small in case the acquaintance relationship becomes invalid too early, and that means robots cannot use this relationship sufficiently. However, an unreasonably long acquaintance relationship implies the information is more likely to be inconsistent. The robots can resort to subcontracting or re-decomposition to compensate for the inconsistency. We note that the parameters controlling the acquaintance relationship under the CoBA architecture can be adapted over time based upon learning. Although some progress has been made in methods for learning and improving these parameters, further work is required.



(a)

(b)

Figure 9. Comparison of solution qualities of CoBA with changes in the acquaintance relationship parameter μ1.

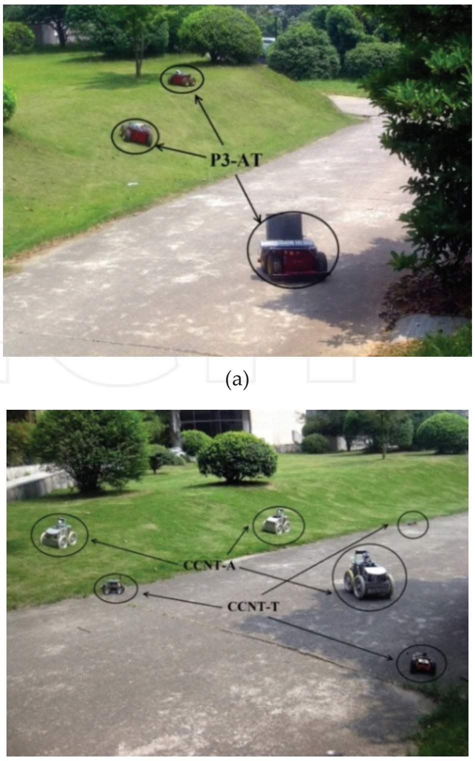
5.2 Implementation on Physical Robots

In order to further evaluate the advantage of our approach, the aforementioned scenario has also been implemented and tested on a team of physical robots with different capabilities. The set of robots forming our application scenario is composed of three P3-AT robots as illustrated in Figure 10(a) and the two types of our self-made mobile robots (three CCNT-A robots and three CCNT-T robots) as illustrated in Figure 10(b). The P3-AT and CCNT-A robots are medium sized, while the CCNT-T robot is small, which enables it to easily intrude narrow spaces. Table 3 briefly shows the onboard sensors of these three types of robots, respectively. The software architecture on our robot platforms is essentially the same as that used in simulation experiments.

Sensors	Robots		
	P3-AT	CCNT-A	CCNT-T
Camera	√	√	×
Bumper	√	√	√
Sonar	√	√	×
Infrared	×	√	√
Odometry	√	√	√
Sound	×	√	√
Network	√	√	√

Table 3. Different capabilities of the physical robots.

In the experiment, robots are charged with a disaster response mission, in which six sites are specified in an outdoor environment. Three sites are introduced at the beginning, and another three sites are introduced 600 seconds later. For the sake of convenience and simplicity, obstacles and victims are represented with different coloured boxes. Due to the lack of gripper, robots choose to push the objects. Initially, all the robots spread out from starting positions to find the sites. Once a site is observed, the robot subsequently holds an auction to allocate tasks to others.



(a)

(b)

Figure 10. Robots in the start positions: (a) P3-AT robots; (b) CCNT-A and CCNT-T robots.

In order to successfully cooperate in this scenario, a mica mote is mounted on every robot acting as a communication module. The mote utilizes a powerful ATMega128L micro-controller and a frequency-tunable radio with extended range. In this way, it is possible to obtain a different communication condition via tuning transmittance power.

Figure 11 shows the results obtained by varying the communication distance of the mobile robot. The results clearly coincide with those observed in the simulated environment. The results suggest that CoBA demonstrates relatively stable performance under different communication settings. In contrast, the performances of the other two approaches are heavily influenced by communication settings. However, the numbers of completed tasks of the three approaches are similar when global communication is possible (i.e., communication distance is 60 m).

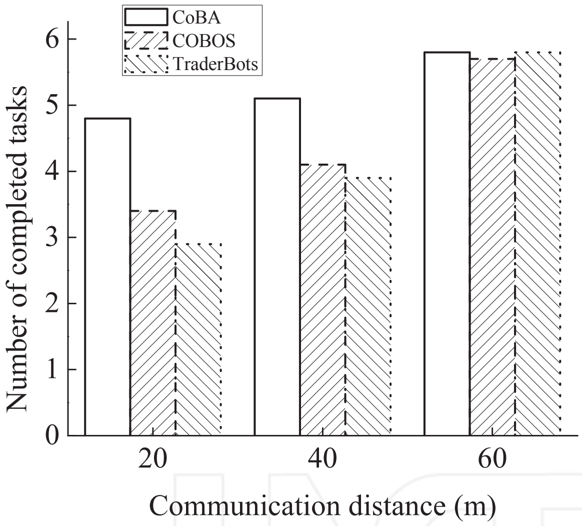


Figure 11. Comparison of solution qualities among the three approaches with changes of communication distance.

We also compared the performance of our approach under constrained communication with that under perfect communication. Figure 12 shows the variation in the number of tasks that remain in the environment over time for two different communication conditions running separately. The completion times of tasks may be longer under constrained communication than that under perfect communication. These additional time costs are due to acquaintance recruitment across different sub-networks. Nevertheless, our approach enables robots to ultimately complete all the tasks successfully. As expected, the experimental results show that our approach is able to gracefully degrade its performance and maximize the efficiency in the absence of team-wide communication.

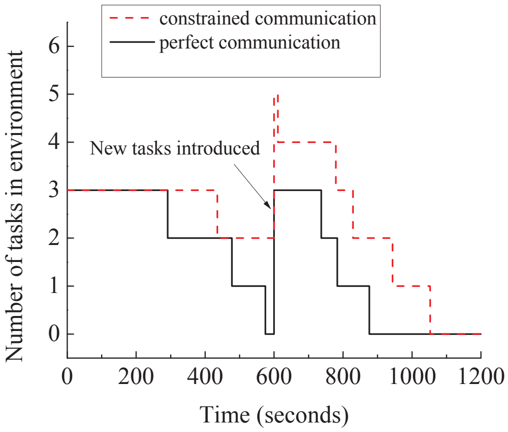


Figure 12. Comparison of solution qualities of CoBA between constrained communication and perfect communication.

6. Conclusions

Large-scale coordination tasks in hazardous, uncertain, and time-stressed environments are becoming increasingly important for fire, rescue, and military operations. Substituting robots for people in the most dangerous activities could greatly reduce the risk to human life. We present a novel, robust, and effective multi-robot coordination approach, CoBA, to meet this demand. The approach is validated in both a simulation environment and on physical robots by a series of experiments in disaster-response domains.

Although the work presented in this paper is preliminary, it proves that it is feasible to employ an improved market-based approach, which supports an AND/OR task tree with temporal constraints and direct/indirect task auction to enable effective multi-robot coordination in spite of various constraints. As future work, several interesting extensions could be considered in order to realize a more robust system: enhancing the dynamic temporal constraint satisfaction process, replanning teamwork with selective disruption minimization to others, etc. We also plan to carry out experiments with a large heterogeneous mobile robot team.

7. Acknowledgements

This work was supported by National Natural Science Foundation of China (61173177), the Defence Industrial Technology Development Program of China (B1420110149), and the China Postdoctoral Science Foundation (2012M511369).

8. References

[1] B. Gates (2007) A robot in every home. Sci. Am. 296(1): 58-65.
[2] T. Arai, E. Pagello, and L. E. Parker (2002) Editorial: advances in multirobot systems. IEEE T. Robotic. Autom. 18(5): 665-661.

[3] L. E. Parker (2008) Multiple mobile robot systems. Springer Handbook of Robotics. Berlin Heidelberg, Springer-Verlag. pp. 921–941.

[4] A. Farinelli, L. Iocchi, and D. Nardi (2004) Multirobot systems: a classification focused on coordination. IEEE Trans. Syst. Man Cybern. Part B-Cybern. 34(5): 2015-2028.

[5] A. Howard, L. E. Parker, and G. S. Sukhatme (2006) Experiments with a Large Heterogeneous Mobile Robot Team: Exploration, Mapping, Deployment and Detection. Int. J. Robot. Res. 25(5-6): 431-447.

[6] S. Gustafson and D. A. Gustafson (2006) Issues in the scaling of multi-robot systems for general problem solving. Auton. Robot. 20(2): 125-136.

[7] B. P. Gerkey and M. J. Mataric (2004) A formal analysis and taxonomy of task allocation in multi-robot systems. Int. J. Robot. Res. 23(9): 939-954.

[8] G. A. Korsah, M. B. Dias, and A. Stentz (2012) A Comprehensive Taxonomy for Multi-Robot Task Allocation. Technical report, Carnegie Mellon University.

[9] M. M. Veloso and D. Nardi (2006) Special issue on multirobot systems. P. IEEE. 94(7): 1253-1256.

[10] L. Parker (2008) Distributed intelligence: Overview of the field and its application in multi-robot systems. J. Phys. Agents. 2(1): 5-14.

[11] Y. Zhang and L. E. Parker (2012) Considering inter-task resource constraints in task allocation. Autonomous Agents and Multi-Agent Systems. online first: 1-31.

[12] A. Tsalatsanis, A. Yalcin, and K. P. Valavanis (2012) Dynamic task allocation in cooperative robot teams. Robotica. online first: 1-10.

[13] S. C. Botelho and R. Alami (1999) M+: a scheme for multi-robot cooperation through negotiated task allocation and achievement. Proc. IEEE International Conference on Robotics and Automation. pp. 1234-1239.

[14] M. B. Dias, R. Zlot, N. Kalra, *et al.* (2006) Market-based multirobot coordination: a survey and analysis. P. IEEE. 94(7): 1257-1270.

[15] D. C. MacKenzie (2003) Collaborative tasking of tightly constrained multi-robot missions. Multi-Robot Systems: From Swarms to Intelligent Automata (Proceedings of 2nd International Workshop on Multi-Robot Systems), Washington D.C. pp. 39-50.

[16] D. Goldberg, V. Cicirello, M. B. Dias, *et al.* (2003) Market-based multi-robot planning in a distributed layered architecture. Multi-Robot Systems: From Swarms to Intelligent Automata, Proceedings of 2nd International Workshop on Multi-Robot Systems, Washington D.C. pp. 27-38.

[17] V. A. Ziparo, L. Iocchi, D. Nardi, *et al.* (2008) Petri net plans: a formal model for representation and execution of multi-robot plans. Proceedings of the 7th international joint conference on autonomous agents and multiagent systems (AAMAS), Estoril, Portugal. pp. 79-86.

[18] Y. T. Kotb, S. S. Beauchemin, and J. L. Barron (2007) Petri Net-Based Cooperation In Multi-Agent Systems. Proceedings of the Fourth Canadian Conference on Computer and Robot Vision, IEEE Computer Society. pp. 123-130.

[19] W. Sheng and Q. Yang (2005) Peer-to-peer multi-robot coordination algorithms: petri net based analysis and design. Proceedings of IEEE/ASME International Conference on Advanced Intelligent Mechatronics. pp. 1407-1412.

[20] R. Zlot and A. Stentz (2006) Market-based multirobot coordination for complex tasks. Int. J. Robot. Res. 25(1): 73-101.

[21] E. Jones, M. Dias, and A. Stentz (2011) Time-extended multi-robot coordination for domains with intra-path constraints. Auton. Robot. 30(1): 41-56.

[22] T. Lemaire, R. Alami, S. Lacroix, *et al.* (2004) A distributed tasks allocation scheme in multi-UAV context. Proceedings of IEEE International Conference on Robotics and Automation (ICRA 04). pp. 3622-3627.

[23] M. Koes, I. Nourbakhsh, and K. Sycara (2005) Heterogeneous multirobot coordination with spatial and temporal constraints. Proc. the Twentieth National Conference on Artificial Intelligence and the Seventeenth Innovative Applications of Artificial Intelligence Conference (AAAI). pp. 1292-1297.

[24] M. Koes, I. Nourbakhsh, and K. Sycara (2006) Constraint optimization coordination architecture for search and rescue robotics. Proc. IEEE International Conference on Robotics and Automation (ICRA). pp. 3977-3982.

[25] N. Kalra, D. Ferguson, and A. Stentz (2005) Hoplites: A Market-Based Framework for Planned Tight Coordination in Multirobot Teams. Proc. IEEE International Conference on Robotics and Automation. pp. 1170-1177.

[26] C. Fua and S. S. Ge (2005) COBOS: Cooperative backoff adaptive scheme for multirobot task allocation. IEEE T. Robot. 21(6): 1168-178.

[27] M. Koes, K. Sycara, and I. Nourbakhsh (2006) The best laid plans of robots and men. Proc. AAAI Spring Symposium on Distributed Plan and Schedule Management, Menlo Park, CA, USA. pp. 49-56.

[28] B. Thomas and D. Edmund (2003) Limiting disruption in multiagent replanning. Proceedings of the second international joint conference on Autonomous agents and multiagent systems, Melbourne, Australia, ACM. pp. 49 - 56.

[29] N. Kalra, D. Ferguson, and A. Stentz (2006) Constrained exploration for studies in multirobot coordination. Proc. IEEE International Conference on Robotics and Automation (ICRA). pp. 4300-4302.

[30] N. Kalra, D. Ferguson, and A. Stentz (2007) A Generalized Framework for Solving Tightly-coupled Multirobot Planning Problems. International Conference on Robotics and Automation (ICRA), Roma. pp. 3359-3364.

[31] A. H. Mong-ying, P. Srivastava, V. Kumar, *et al.* (2004) Composable communication constraint-based control. Mobile Robots XVII, Proceedings of SPIE Vol. 5609, Philadelphia, PA. pp. 192-200.

[32] T. S. Dahl, M. Mataric, and G. S. Sukhatme (2009) Multi-robot task allocation through vacancy chain scheduling. Robot. Auton. Syst. 57(6-7): 674-687.

[33] L. E. Parker and F. Tang (2006) Building multirobot coalitions through automated task solution synthesis. P. IEEE. 94(7): 1289-1305.

[34] L. Vig and J. A. Adams (2006) Multi-robot coalition formation. IEEE T. Robot. 22(4): 637-649.

[35] S. D. Ramchurn, M. Polukarov, A. Farinelli, *et al.* (2010) Coalition formation with spatial and temporal constraints. Proc. of 9th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2010). pp. 1181-1188.

[36] S. Carpin and E. Pagello (2009) An experimental study of distributed robot coordination. Robot. Auton. Syst. 57(2): 129-133.

[37] F. L. Lian and R. Murray (2003) Cooperative task planning of multi-robot systems with temporal constraints. Proceedings of IEEE International Conference on Robotics and Automation (ICRA), IEEE. pp. 2504-2509.

[38] F. Yang, S.-r. Liu, and F. Liu (2010) Cooperative transport strategy for formation control of multiple mobile robots. J. Zhejiang Univ.-SCI. C. 11(12): 931-938.

[39] B. Gerkey, S. Thrun, and G. Gordon (2006) Visibility-based pursuit-evasion with limited field of view. Int. J. Robot. Res. 25(4): 299-315.

[40] S. Koenig, P. Keskinocak, and C. Tovey (2010) Progress on agent coordination with cooperative auctions. Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence (AAAI-10). pp. 1713-1717.

[41] B. P. Gerkey and M. J. Mataric (2002) Sold!: Auction methods for multirobot coordination. IEEE T. Robotic. Autom. 18(5): 758-768.

[42] M. B. Dias, M. Zinck, R. Zlot, *et al.* (2004) Robust multirobot coordination in dynamic environments. Proc. IEEE International Conference on Robotics and Automation. pp. 3435-3442.

[43] M. B. Dias (2004) TraderBots: A New Paradigm for Robust and Efficient Multirobot Coordination in Dynamic Environments. Ph.D. dissertation, Carnegie Mellon University.

[44] M. Nanjanath and M. Gini (2010) Repeated auctions for robust task execution by a robot team. Robot. Auton. Syst. 58(7): 900-909.

[45] R. G. Smith (1980) The Contract Net Protocol: High-Level Communication and Control in a Distributed Problem Solver. IEEE Trans. Comput. 29(12): 1104-1113.

[46] Y. Liu, J. Yang, and Z. Wu (2011) Ubiquitous and cooperative network robot system within a service framework. Int. J. Humanoid Robot. 8(1): 147-167.

[47] E. Schwalb and L. Vila (1998) Temporal constraints: A survey. Constraints. 3(2): 129-149.

[48] S. Carpin, M. Lewis, J. Wang, *et al.* (2007) USARSim: a robot simulator for research and education. IEEE International Conference on Robotics and Automation (ICRA), Roma, Italy, IEEE. pp. 1400-1405.

[49] T. Balch (2000) Hierarchic Social Entropy: An Information Theoretic Measure of Robot Group Diversity. Auton. Robot. 8(3): 209-238.

INTECH

View publication stats