Defence Science Journal, Vol. 67, No. 3, May 2017, pp. 316-324, DOI : 10.14429/dsj.67.10995

DTTA - Distributed, Time-division Multiple Access based Task Allocation Framework for Swarm Robots

Meetha V. Shenoy* and K.R. Anupama

Department of Electrical and Electronics Engineering, Birla Institute of Technology & Science, Pilani, KK Birla Goa Campus - 403 726, India

**E-mail: meetha@goa.bits-pilani.ac.in*

ABSTRACT

Swarm robotic systems, unlike traditional multi-robotic systems, deploy number of cost effective robots which can co-operate, aggregate to form patterns/formations and accomplish missions beyond the capabilities of individual robot. In the event of fire, mine collapse or disasters like earthquake, swarm of robots can enter the area, conduct rescue operations, collect images and convey locations of interest to the rescue team and enable them to plan their approach in advance. Task allocation among members of the swarm is a critical and challenging problem to be addressed. DTTA- a distributed, Time-division multiple access (TDMA) based task allocation framework is proposed for swarm of robots which can be utilised to solve any of the 8 different types of task allocation problem identified by Gerkey and Mataric´. DTTA is reactive and supports task migration via extended task assignments to complete the mission in case of failure of the assigned robot to complete the task. DTTA can be utilised for any kind of robot in land or for co-operative systems comprising of land robots and air-borne drones. Dependencies with other layers of the protocol stack were identified and a quantitative analysis of communication and computational complexity is provided. To our knowledge this is the first work to be reported on task allocation for clustered scalable networks suitable for handling all 8 types of multi-robot task allocation problem. Effectiveness and feasibility of deploying DTTA in real world scenarios is demonstrated by testing the framework for two diverse application scenarios.

Keywords: Swarm robotics; Task allocation; Search and rescue; Clustering

1. INTRODUCTION

Modern military forces operate as a network, with networked human-inhabited platforms in land or air for operations such as surveillance, search and rescue or even in warfare. Benefits of deploying human-uninhibited intelligent platforms in hostile situations like land mine detection, disaster recovery, etc. is recognised by defence forces all over the world, however the widespread deployment of robots as human substitutes in risk prone, unmapped or dynamic terrains is challenging and a dream yet to be true. The major challenges include, but are not limited to developing autonomy in robots to ensure required level reactiveness to dynamic environments, developing fault resilient robotic network, cost effective development of robotic systems, performance limitations of robot or multiple robots in completing the mission within the deadline.

Swarm robotics is a nascent branch of research which exploits the potential advantages of distributed computing like fault tolerance and scalability[1]. Swarm robotic systems when compared to the traditional multi-robotic system utilize decentralised, co-operative, self-organised, similar, simpler robots which utilize local interactions among each other to produce complex and emergent behaviours and thus can demonstrate potentials beyond the capabilities of individual robot. Several applications are proposed for utilising swarms in military operations, search and rescue, precision agriculture, inventory management in warehouse, etc.[2-5] thus mitigating human effort, provide a safe working environment or even save human lives.

Defence forces can operate swarm, with better coordination, intelligence and speed and most importantly reducing the risk to human personnel. With the recent advances in developing swarm robotic structures, the networking among the robots and task allocation among robots remains as the major factors holding back wide spread swarming.

'Task allocation' is to decide 'which robot executes which task, at what time'. This is described as multi-robot task allocation (MRTA) problem in multi-robot systems[6]. MRTA involves distributing and scheduling a set of tasks among a group of robots to achieve certain system goals taking into account the operational constraints. The complexity of task allocation increases with the size of the swarm and heterogeneity among the members of the swarm. Task allocation is an open problem of research in multi-robotic systems, but very less focus has been given to the inter-robot communication aspects of the task allocation problem. Originally swarm robots were nature inspired and hence their communication mechanism was limited to stigmergy or visual light patterns. To optimise

Received : 06 December 2016, Revised : 06 April 2017
Accepted : 12 April 2017, Online published : 24 April 2017

the time taken to accomplish any mission, robust methods of inter-robot communication is essential. Wireless sensor network is a mature branch of study and the same can be used for communication among members of the swarm[7].

2. LITERATURE REVIEW AND PROBLEM FORMULATION

The first reported work on formal classification of MRTA problems was by Gerkey and Mataric´[6]. They proposed a taxonomy for the classification of MRTA problem along three axes as follows,

(i) Single-task robots (ST) versus Multi-task robots (MT)
(ii) Single-robot tasks (SR) versus Multi-robot tasks (MR)
(iii) Instantaneous assignment (IA) versus Time extended assignment (TA)

ST-SR-IA is the simplest task allocation problem which assigns single robot to a single task and each task is attended by only one robot. The instantaneous assignment schemes perform task allocation without taking into account, current and future requirements of the system. MT-MR-TA is the most complex among all MRTA problems in which each robot is assigned multiple tasks considering the current and future requirements and each task requires multiple co-operating robots for its completion.

A new taxonomy, iTax was proposed, which is an extension to the taxonomy proposed by Gerkey and Mataric´, taking into account the interrelated utilities and constraints among tasks[8]. Task allocation in robotic systems seeks to determine a feasible assignment of tasks to robots that optimises the utility of the task. Based on the degree of interdependency four possible classifications were proposed as follows :

(i) No dependencies: the utility of robot-task does not depend on task of any other robot or its own schedule
(ii) In-schedule dependencies: the utility of robot–task depends on its own schedule
(iii) Cross-schedule dependencies: the utility of robot–task depends on its own schedule and the schedules of other robots, and
(iv) Complex dependencies: the utility of robot–task depends not only the schedules of other robots but also on their decomposition.

The solutions suggested for MRTA problem in literature can be classified into two major categories- Optimisation based approaches and Market based approaches. Optimisation based approaches attempt to find a solution such that, for the given union of state of the system, it is impossible to find a better solution than the obtained one. When treated as an optimisation problem, out of the 8 classifications proposed by Gerkey and Mataric´[6] only the ST-SR-IA problem can be solved in polynomial time, while remaining problems are strongly NP (Nondeterministic Polynomial time) hard. Similarly, only problems in 'no dependencies' category can be solved in polynomial time, while the other problems are NP-hard if treated as an optimisation problem. Several works[9] are reported on solving MRTA problem as a formulation of TSP (travelling salesman problem) or mTSP (multiple travelling salesman problem) which solves one specific type of MRTA problem. For example TSP can be formulated as SR-MT-IA with in-schedule dependencies. Population based approaches like genetic algorithm or heuristic approaches like simulated annealing[9] are proposed to find optimal allocation of robots to tasks such that the cost function is minimised and the utility is maximised. The most common cost functions are the distance travelled, time taken for travel, etc. Although in static environments and ideal scenarios, the distance to be travelled and the time required to cover a distance by the robot may be calculated, the assessment of the same is not a realistic in case of dynamic environments or unmapped terrains. The optimisation based approaches are difficult to implement for practical scenarios due to it computational and communication complexity.

Market based approaches for task allocation is emerging as a popular choice within robotics research community in recent years. In market based approaches instead of finding the optimal solution, focus is on finding a feasible solution as fast as possible. In market-based approaches the task allocation is based on a bidding-auctioning procedure between a central agent (auctioneer) and the robots (bidders). The auctioneer announces the task along with the information pertaining the task such as, the number of robots required to perform the task, energy budget for the task, capabilities of the robot to service the task, deadline of the task, etc. The robots which listen to the auction announcement will bid for the messages. Based on how the winner of the bidding is declared the market based approaches can be centralised or distributed[10,11]. Most commonly, the central auctioneer will evaluate the bids and assign the task to the robot with the best bid. The communication overhead increases drastically in these scenarios when the size of the network increases. Task allocation problem is a dynamic decision problem, that needs to be iteratively reconsidered over time rather than as a static assignment problem. Moreover, most of the algorithms available in literature caters only to one or few specific types of MRTA problem proposed in the classification of Gerkey and Mataric´.

We propose a task allocation framework for handling all types of MRTA problems as proposed by Gerkey and Mataric´[6]. To this account, we redefine the MRTA problem as follows. Given a set of available robots- $R = \{R_1, R_2, ...R_n\}$ and available tasks- $T = \{T_1, T_2, ...T_n\}$. Each task may require multiple robots to complete the task and each robot may be assigned to multiple tasks, the framework should provide a feasible task allocation so that all the tasks are completed as per the required performance goals by minimizing the number of communication messages and computations required for task allocation.

Authors proposed a task allocation framework with the following features.

(i) A distributed framework for task allocation is designed which can be used to solve any instance of MRTA problem
(ii) In-addition to assigning tasks based on the present environmental conditions, the scheme also allows extended task assignments and task migration when a robot may not be able to complete its assigned task within deadline. In such scenarios, the robots should be capable of modifying their schedule pro-actively, post the initial

task assignment to handle challenges like obstacles, loss of nodes due to the risk involved in the mission, complex terrain, etc.

(iii) The scheme is independent of path planning or navigation of robots, so that the scheme can be used for task allocation of land/air-borne robots or for co-operating networks in land and air.

(iv) The scheme clearly identifies the interdependencies between the other layers of the protocol stack so that the same can be incorporated into any existing robotic system.

(v) The scheme is scalable at the same time computationally efficient with lesser communication overhead.

3. DTTA FRAMEWORK

DTTA is a distributed task allocation framework which can be utilised to solve 8 different types of MRTA problem identified by Gerkey and Mataric´ i.e. SR-ST-IA, SR-MT-IA, MR-ST- IA, MR-MT-IA and their time-extended assignment (TA) counterparts. Another key significance of DTTA framework is that the task allocation scheme is reactive and thus can accommodate any of the schedule dependencies as suggested by iTax taxonomy through extended task assignments. Unlike most of the solutions suggested for MRTA problems[9,10,12], DTTA framework isolates path planning and navigation from task allocation problem and hence can be utilised for any kind of robot in land or for co-operative systems comprising of land robots and air-borne robots/drones. The framework clearly identifies the dependencies with other layers of the protocol stack and provides a quantitative analysis of the computational complexity of the framework. DTTA can be effectively utilised in clustered and non-clustered scenarios. To our knowledge, this is the first work to be reported on MRTA for clustered scalable networks. We demonstrate the effectiveness of DTTA framework by modelling two diverse application scenarios.

The DTTA framework assumes that the robots are equipped with wireless communication devices and the members of the swarm can communicate via the wireless network. The type of wireless network can vary according to the deployment scenario, but the same can be as simple network as in a wireless sensor network (WSN)[7]. It is assumed that the robots can maintain notion of time and the nodes are coarsely time synchronised so that the synchronisation error among members of the swarm is not more than couple of milli-seconds which is easily achievable[13] in a robotic network which is usually deployed for short duration of time and may not require continuous operation for over 24 h. Each robot should also have the capability to localize itself in area of interest.

The task allocation strategy employed in DTTA is inspired from market based task allocation strategies which are more reactive when compared to optimisation based approaches[14,15]. In traditional market/auction based methods, the robots competitively bid for task or tasks in response to the task announcement by the auctioneer. The major challenge in implementing competitive bidding is the containment of bidding messages in the network, as the size of the network increases. DTTA replaces competitive bidding with co-operative self-task assignment. Individual members of swarm assign tasks to itself rather than the same being assigned by the auctioneer.

Based on how the winner of bidding is decided, the auction based methods can be centralised or distributed. Centralised task allocation is not preferred for autonomous swarm robots. In decentralised market based task allocation schemes like M+[15], each robot broadcasts its utility (bid), and on receipt of bids from others, robots perform a greedy task selection, finds the tasks for which it has the highest utility among robots and selects the task with highest utility. The computational and communicational complexity per iteration is $O(mn)$ where m and n are the number of robots and tasks respectively. DTTA propose batch processing of tasks, in which a smaller number of tasks are considered at a time for task assignment, thus reducing the number of computations required for task selection. DTTA framework is distributed, hence each robot will receive the information as indicated in Table 1 from the auctioneer via task announcement message. Each robot which receives the task announcement message may assign task/tasks for itself. The robot then announces the count of tasks and the ID's of tasks it will service through task acceptance message.

The steps involved in the task allocation can be represented as a state flow diagram as in Fig. 1. Task announcement messages by the central auctioneer marks the beginning of hyperperiod. A hyperperiod is the duration of time for which the task assignment/allocation is performed by robots. The Auction messages are of the format - *SourceID : Type : TTL :CC :CIi : RC : RIi : TIn : MTR* (as shown in Table 1 and Fig. 2). After the broadcast of task announcement message by auctioneer, the hyperperiod starts and the robots participate in role selection process.

The robots assume role of either relay node, worker node or cluster head node. The preference order of the roles is as cluster head (highest), relay node and worker node (lowest) respectively. If a role in higher preference is already filled by other robots, then the robots will assume the next available role. In case of small scale network with lesser number of robots, clustering is not required and hence the available roles are as relay or worker node. Each robot utilizes its TDMA slot to assign role and task for itself and to announce the same via task acceptance message. The TDMA slots for a particular hyperperiod are self-assigned by robots based on the priority of robots. The initial priority assignment is based on the ID number of the robot. The robot with the highest priority is assigned the first TDMA slot. Each robot maintains a 'Task queue' to store the task acceptance messages from the other robots. To assign the task to itself, the robots inspect its task queue for available task acceptance messages from the robots of higher priority for the given hyperperiod.

In DTTA, the auctioneer announces the robot resources including the energy budget required for completing a particular task in the TI_n field. Only available robots with the requested features and energy budget will assign the task to itself during the task assignment process. The robots obey the following rules during assignment process.

(1) The robot of a certain priority will attempt to service the task with lowest ID first.

(2) If Task with lowest ID (e.g.Task1) is already accepted by higher priority robot and the 'count' of robots required to service the task is met then the robot attempts to accept

Table 1. List of abbreviations

Symbol	Item	Comments
SourceID	ID of the robot originating the message	
Type	Type of message being transmitted	Different type of messages includes- (a) task announcement message by auctioneer, (b) extended task announcement messages, (c) task acceptance messages, (d) extended task acceptance messages.
TTL	Time to live	Dead line of tasks. Requires multiple fields if TTL is different for each task
CC	Cluster control information	Indicates if clustering is required ($C = 1$), Clusters are of equal size ($E = 1$) and number of clusters (*NC*). Refer Fig. 2 for details.
CI_i	Cluster information	Includes required features of cluster members, number of robots/cluster (*RC*). Number of *CI* fields *i*; $i = 0,..NC$
W	No of tasks announced/ task announcement message	
RC	Relay node control information	Indicates if relay nodes are required, and the relay node count (*RN*)
RI_i	Relay information	Relay node location and characteristics; $i = 0,...RN$
TI_n	Task information	Information about task (being announced/accepted) such as Task ID, number of robots required/task (*CH*), Location of the task, characteristics of the robot, Energy budget for task (*EB*), Extended Task assignment allowed (*TA* = 1), number of *TI* fields depends on n; n = 1...W
MTR	Maximum number of tasks a robot can service/task announcement message	
Lx, *Ly*	Current position of the robot	
Count	Count of tasks which a robot will service/task announcement message	

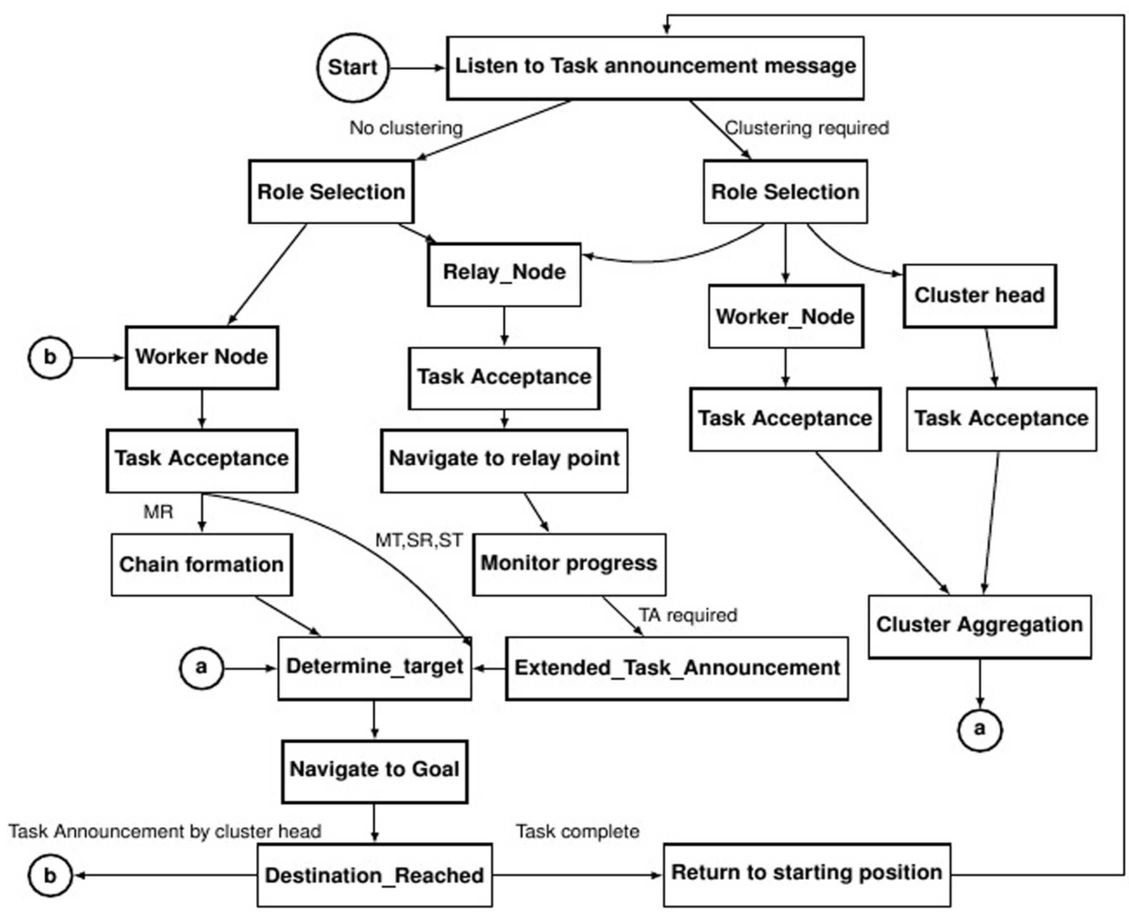

Figure 1. DTTA- state flow diagram.

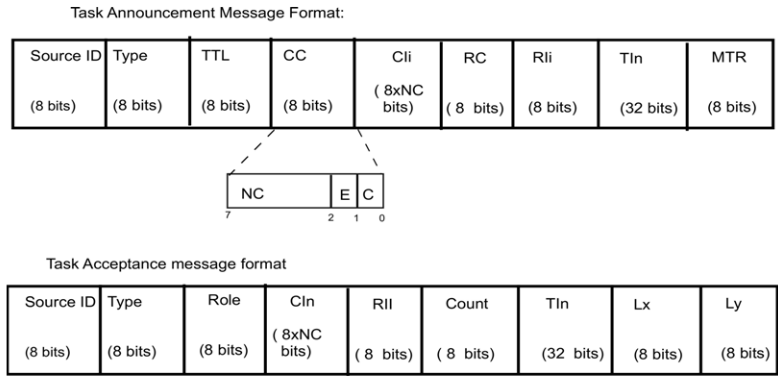

Figure 2. Task announcement and task acceptance message formats.

the next task (e.g. Task2).

(3) If the count of robots required to service a task is 'greater than unity' then the robot services the task along with the other robot servicing the task.

(4) If a robot accepts a single robot (SR) task, then the robot can accept to service 'MTR' (Table 1) number of single robot tasks if the targets are close to each other.

(5) A robot assigned for a multi-robot (MR) task will not commit to service other tasks. (Robot assigned to a MR task is allowed to respond to extended task announcement messages)

(6) After the completion of task hyperperiod, or certain hyperperiods based on the application, the priority of robots is rotated in order to ensure a fair distribution of load to all the robots, thus extending the life time of the system.

It has to be noted that message transmissions are not required for TDMA slot assignment or rotation of priority as the same is based on the unique ID of robots. Each robot broadcasts the count and ID of the tasks which it will service via task acceptance messages. The task acceptance messages (Table 1, Fig. 2) are broadcast messages of the format - *SourceID : Type: Role :* CI_i *:* RI_i *: Count :* TI_n *:* L_x *:* L_y. Once the hyperperiod is complete, the robots start servicing the tasks. The robots which are designated to serve the same will form a chain and move towards the destination. Chaining reduces the communication and computational overheads in path planning. The number of messages required for task assignment in DTTA is equal to the number of robots required to service the tasks in contrast to the traditional auction based schemes where the number of messages transmitted for task allocation is equal to the number of robots bidding for the task and the additional messages required to announce task assignment.

DTTA can be extended to cluster based large scale networks as follows. After the task announcement by the auctioneer, the robots perform role selection. The cluster heads are selected first, which in-turn will announce their selected tasks via task acceptance messages. Based on the task announcement, relay nodes will be selected. The worker nodes in their task acceptance messages will indicate the cluster which they would join in the CI_n field. Once task assignment is complete, the nodes in a cluster aggregate together and move to the target location assigned to the cluster. On reaching the target the cluster head broadcasts task announcement messages to delegate the work among the cluster members.

Once the mission is accomplished, the members of cluster will again aggregate together before returning to their starting point.

Relay nodes ensure that there is a communication relay established between the cluster heads and base-station throughout the mission irrespective of the target locations. Once the robots start moving to their destination they monitor their own progress towards the mission. If the robots are unable to progress further due any unforeseen situation (e.g.: a hurdle on the path difficult to cross), the same will be conveyed to the peers via extended task announcement messages. If there are robots in SR-ST mission nearby, then the robots can assist the trapped robot to progress further by crossing hurdles through chaining. The relay node will ensure that the extended task announcement messages are attended by either the robots already on mission or by idle robots so that the mission can be continue without interruption. Extended task announcement messages are of the same format as that of task announcement messages with unique 'type' field. Similarly, task acceptance and extended task acceptance messages are of the same format and can be distinguished by the robots based on their 'type' field. The task acceptance queue is cleared by worker nodes once the announced tasks during hyper period are accepted by the robots. Although robots on 'MR' mission are not assigned multiple tasks (MR-MT missions) in response to initial task announcement messages to avoid the inter-schedule dependencies, and to maximize the speed of service of tasks, the robots in MR mission are allowed to respond to extended task announcement messages.

Table 2 provides comparison of the computational and communicational complexity of DTTA with some of the popular market based approaches for SR-ST scenario. The analysis is provided for a scenario with *m* robots and *n* tasks. It has to be noted that DTTA supports all variants of MRTA problems reported by Gerkey and Mataric´.

Table 2. Complexity analysis

Name	Computational complexity	Communication complexity	MRTA Type
Alliance	mn	m	SR-ST
M+	mn	mn	SR-ST
Murdoch	1-bidder, n-auctioneer	n	SR-ST
BLE	mn	mn	SR-ST
DTTA	No. of tasks to be serviced /task announcement	1 (if robot accepts task)	SR-ST

4. SIMULATOR SETUP

DTTA framework is tested on Argos, a multi-robot simulator[16]. The Argos simulator supports model for three types of robots designed as a part of EU funded Swarmanoid project. The three types robots are -

1. Foot-bots (land based robots),
2. Hand-bots (robot which can climb walls) and
3. Eye-bots (robots which can fly).

Any algorithm developed on Argos can be loaded onto the actual robot. On-board clock on robots serves as the time reference. Foot-bots navigates using wheels and tracks which allows it to move on complex terrains. The foot-bots and eye-bots supports range and bearing modules and Wi-Fi for wireless communication. The range and bearing module, upon receipt of a message, can calculate the relative position (distance and angle) of the sender. Unlike other multi-robot simulators like Gazebo or VREP, Argos provide interface to network simulators like ns-2 and ns-3 enabling simulation of networked robots[17]. Foot-bot is equipped with two cameras- an omnidirectional camera, and a camera that can be mounted looking upwards or frontally, which can be used for creation of maps or for path planning. Eye-bot has pan and tilt camera with actuators to control the attitude of camera. The foot-bot also support IR proximity sensors, around the robot for near obstacle detection. Another notable feature of foot-bots is the gripper, using which it can hold onto other robots or other objects for cooperative task completion. By gripping on to other foot-bots, the robots can help each other to cross obstacles or drag/push an object from one location to another, thus performing tasks beyond capabilities of a single robot. We implemented obstacle sensing and avoidance using cameras and IR sensors. Localisation of robots is implemented using position sensors. Argos can simulate well over 200 robots.

5. APPLICATION SCENARIOS

To validate DTTA framework, we have simulated two application scenarios on Argos simulator as described in this section.

5.1 Application Scenario 1 - Small Scale Swarm Task Allocation Scenario

The application scenario considered is a transportation task in which the robots must transport objects from one location to another. The robots can be used to transport goods during a defence mission or during search and rescue missions where the task identified through surveillance can be announced by a central auctioneer. The central auctioneer can be a land robot or a drone based on the mission.

The simulation scenario is modelled as follows. We have assumed that there are 8 foot-bots in the system, 6 worker nodes, one auctioneer and one relay node. Three objects (tasks) are to be retrieved from a fixed location and taken to another location at regular intervals of time by the robots (Fig. 3). Maximum workload or number of tasks that can be announced by the auctioneer (W) is '3' and maximum number of tasks that can be serviced by a robot (MTR) is '2'. The number of robots required to pick up an object is variable. For example, the number of robots required to serve 'Task1' may be '1' or '2' . Thus 6 different types of MRTA problems are covered in the simulation scenario as indicated in Table 3. The combinations which is not tested are MR-MT-IA and MR-MT-TA. Although the DTTA framework allows the allocation of these problems as well, the robot is not designed to collaboratively grip and hold two or more objects. 8 rounds of experiments were conducted to validate DTTA framework. One round includes 8 task announcement sequences as shown in Table 3. Each task announcement sequence will include announcement of 3 tasks. The priority of robots is rotated only after each round.

Since the tasks involves only small number of robots, clustering is not implemented whereas one relay node (Fig. 3) is utilised to monitor the progress of the worker nodes. In case the worker nodes could not progress as expected, the worker

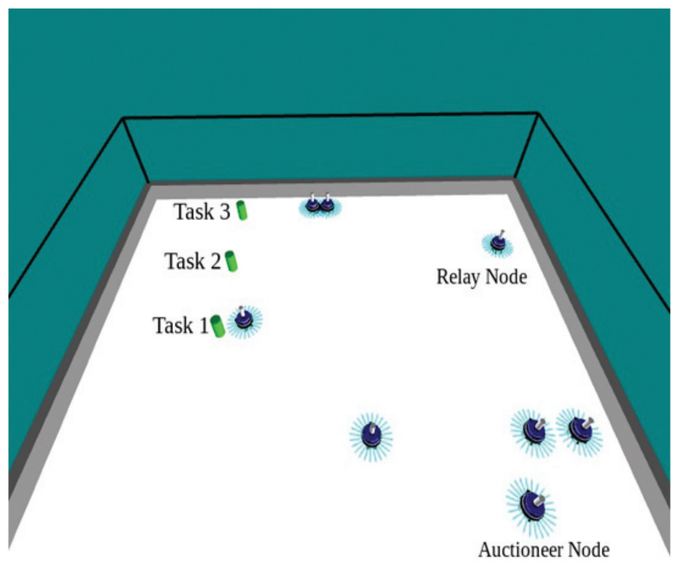

Figure 3. Application scenario 1- Foot-bots performing object pick-up task.

Table 3. Task announcement sequences/round for application scenario 1.

Task 1 : No. of robots required	Task 2: No. of robots required	Task 3: No. of robots required	MRTA type	No. of robots assigned	No. of task acceptance messages
1	1	1	SR-MT, SR-ST	2	2
2	2	2	MR-ST	6	6
2	2	1	MR-ST, SR-ST	5	5
2	1	2	MR-ST, SR-ST	5	5
1	2	2	MR-ST, SR-ST	5	5
1	2	1	MR-ST, SR-MT	3	3
1	1	2	SR-MT, MR-ST	3	3
2	1	1	MR-ST, SR-MT	3	3

node broadcasts extended task announcement messages. The relay nodes may further announce extended task announcement messages to ensure successful completeness of the work by other nodes which are free. This guarantees fault tolerance in the system. In MR scenario, when two robots are required to perform a task, chaining of robots is implemented so that, the multiple robots involved in the task proceed to the target location as a chain (Fig. 3). Figure 3, depicts the task scenario where the task announcement is as follows. Number of robots required for Task 1 = 1, Number of robots required for Task 2=1 and Number of robots required for Task 3 = 2. According to DTTA, Task 1 and 2 will be serviced by one robot whereas Task 3 will be serviced by 2 robots which move towards the target as a chain. The computational and communication overhead can be reduced if the robots proceed as chain towards the destination rather than as individual robot as only the leading robot will have to perform path planning and obstacle detection. The test case in Fig. 3 also represent the scenario where robot serving Task 1 and Task 2 exhibit in-schedule dependency and robots serving Task 3 exhibit cross-schedule dependency.

Table 3 also provide the number of robots assigned and number of task acceptance messages transmitted per task announcement message. Figure 4 indicates the number of tasks served by robots during each round of experiment. Figure 5 indicates the total number of tasks served by each robot for 8 rounds. It can be observed from Fig. 5 that the protocol is a fair protocol which ensures balanced distribution of tasks among the set of robots.

The worst case computational complexity for generic MRTA problem is $O(3*W*M)$ where M is the number of robots required to perform the announced set of tasks and W is the maximum number of tasks announced/task announcement message. The multiplication factor of '3' in the complexity equation is owing to the fact that for any situation other that SR-ST, the count of tasks to be serviced and count of robots required/task should also be taken into consideration before assigning the task. When the application scenario involves only SRST problem, the computational complexity is equal to the number of robots required to service the tasks in the task announcement message unlike other protocols which is dependent on the total number of robots in the system (Table 2). Also, the number of task acceptance/assignment messages (communication complexity) is equal to 1, if the robot is accepting task/tasks belonging to the task announcement message unlike the other protocols where communication cost is involved even if a robot is not assigned a task.

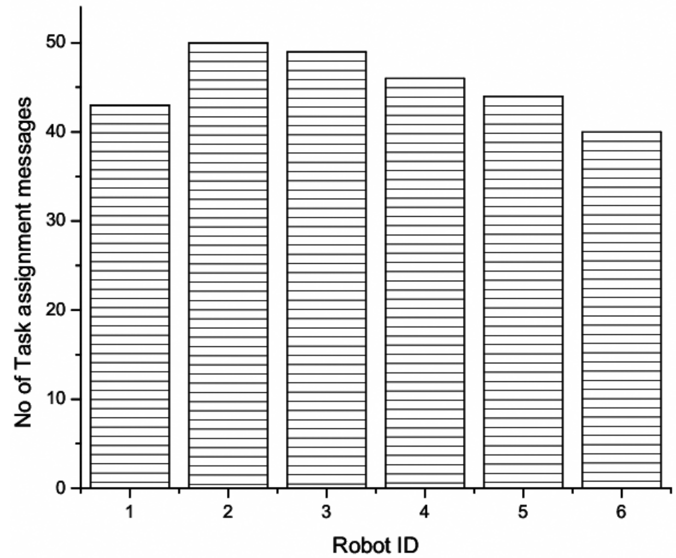

Figure 5. Total number of task assignment messages/robot and tasks serviced/robot for 8 rounds - application scenario 1.

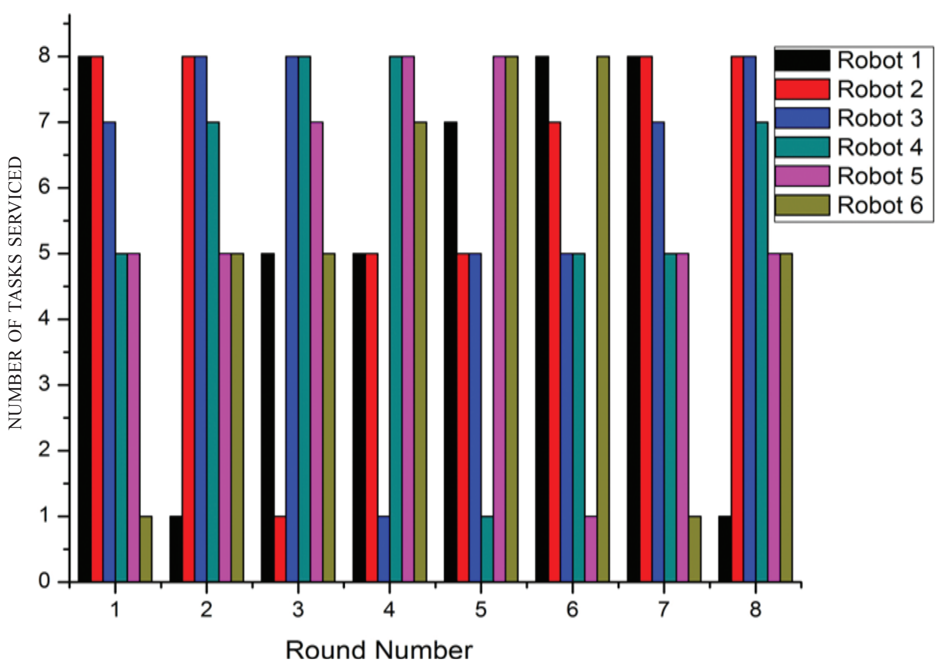

Figure 4. Number of tasks serviced by robots/round for application scenario 1.

5.2 Application Scenario 2 - Medium Scale Swarm Task Allocation Scenario

To validate the suitability of DTTA for a medium scale swarm network and to demonstrate the suitability of DTTA for diverse robotic structures, we have modelled a simulation scenario with 30 worker robots, one relay robot and one task auctioneer robot. The robots are assigned the mission to cover an area of 10 m x 30 m searching for a particular object. An analogy to the real-world scenario is the problem of detection of land mines in an area of interest. For this application, we selected eye-bot, the air-borne robot as the auctioneer and relay node. The worker nodes are foot-bots. The auctioneer surveys the area to be covered and performs task announcement. The task announcement message will include announcement for 3 tasks i.e. formation of 3 clusters with 5, 6, and 19 foot-bots respectively. The announcement message will also announce for a relay node through *RC* and *RI* fields of task announcement message. The target position of the cluster head and relay node is conveyed to robots through the *TI'* and *RI* field of task announcement message. Once the robots receive the task announcement messages, they self-assign tasks and broadcast task acceptance messages. Once hyperperiod is complete the robots move towards the target location. The robots move in such a way that at any point in time the distance between its neighbouring nodes in a cluster is equal. The distance between the nodes is maintained using range and bearing sensor. After the clusters reach the desired area of interest, cluster head performs the required task announcements for its cluster members to ensure coverage of area under each cluster. Cluster members accept the new tasks via task acceptance messages, following which the members of swarm disperse to complete the mission as in Application scenario 1. Fig. 6 depicts the scenario where three clusters have reached the target successfully using DTTA framework.

The rays between the foot-bots and the eye-bot indicate that the corresponding robots are in communication range. It can be observed that auctioneer is in communication range of cluster 3, whereas auctioneer can communicate with cluster 1 and cluster 2 only via the relay node. Thus, the suitability of DTTA for medium scale network is also validated.

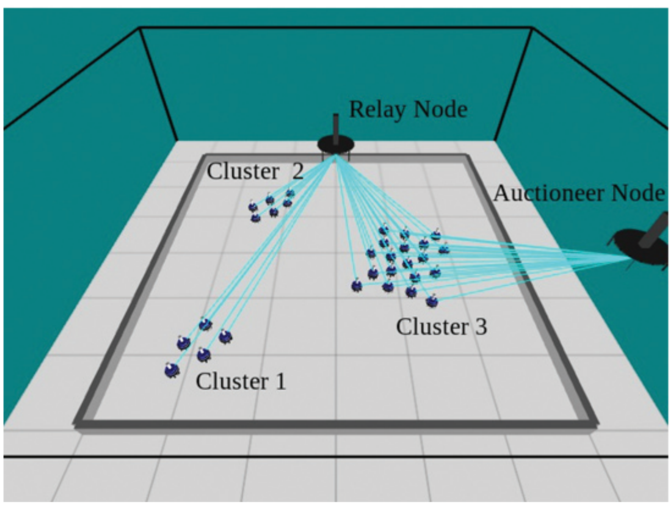

Figure 6. Foot-bot clusters at their target location-application scenario 2.

6. CONCLUSION

Authors proposed DTTA- a distributed, TDMA based Task Allocation framework for swarm of robots. The framework is reactive and can perform task migration via extended task assignments to complete the mission in case of failure of the assigned robot. DTTA framework isolates path planning and navigation from task allocation problem and hence can be utilised for any kind of robot in land or for co-operative systems comprising of land robots and air-borne robots/ drones. This is the first work to be reported which is suitable for solving all 8 types of MRTA problem identified by Gerkey and Mataric´ and is suitable for clustered scalable networks. The major advantage of the protocol is that the computational complexity of the protocol is proportional to the number of robots required to service a given task rather than the total number of robots in the system. Similarly, the communication overhead is involved for a robot only if it will service a task/ tasks. This framework can be utilised in any robotic structure which maintains the notion of time and has the capability to determine its relative position in a given environment. Data collected over a distributed network has no relevance if it does not carry a timestamp and location stamp. Any practical implementation of robotic network will include implementation of time synchronisation and localisation and hence DTTA can be easily incorporated into the system.

REFERENCES

1. Tan, Y. & Zheng, Z. Research Advance in Swarm Robotics. *Defence Technology*, 2013, **9**(1), 18–39. doi: 10.1016/j.dt.2013.03.001
2. Mullen, R. J; Barman, S; Remagnino, P. & , D. Towards autonomous robot swarms for multi-target localisation and monitoring with applications to counter IED operations. *Int. J. Intelligent Def. Supp. Syst.,* 2011, **4**(1). doi: 10.1504/IJIDSS.2011.037809
3. O'Grady, R; Pinciroli, C; Groß, R.; Christensen, A.L; Mondada, F.; Bonani, M. & Dorigo, M. Swarm-bots to the rescue. *In* Proceedings of the 10th European Conference on Artificial Life, Springer, Budapest, Hungary, Lecture Notes in Computer Science, 2009 , **5777**, pp. 165–172. doi: 10.1007/978-3-642-21283-3_21
4. Anil, H.; Nikhil, K.S; Chaitra, V. & Gurusharan, B.S. Revolutionizing farming using swarm robotics. *In* 6th International Conference on Intelligent Systems, Modelling and Simulation, 2015, pp. 141–147. doi: 10.1109/ISMS.2015.26
5. Wurman, P.R; D'Andrea, R. & Mountz, M. Coordinating hundreds of cooperative, autonomous vehicles in warehouses. *AI Magazine*, 2008, **29**(1), 1752-1759. doi: 10.1609/aimag.v29i1.2082
6. Gerkey, Brian P. & Mataric´, M. J. A formal analysis and taxonomy of task allocation in multi-robot systems. *Int. J. Robotics Res.*, 2014, **23**(9), 939–954. doi: 10.1177/0278364904045564
7. Li, W. & Shen, W. Swarm behavior control of mobile multi-robots with wireless sensor networks. *J. Network Comput. Appl.*, 2011, **34**(4), 1398–1407. doi: 10.1016/j.jnca.2011.03.023
8. Korsah, G. A; Stentz, A. & Dias, M. B. A comprehensive taxonomy for multi-robot task allocation. *Int. J. Robotics Res.*, 2013, **32**(12), 1495–1512. doi: 10.1177/0278364913496484
9. Badreldin, M; Hussein, A. & Khamis, A. A comparative study between optimization and market-based approaches to multirobot task allocation. *Adv. Artif. Intell.,* 2013, pp. 1–11. doi: 10.1155/2013/256524
10. Trigui, S; Koubaa, A; Cheikhrouhou, O; Youssef, H; Bennaceur, H; Sriti, M.F. & Javed, Y.A distributed marketbased algorithm for the multi-robot assignment problem. *Procedia Comput. Sci.,* 2014, **32**, 1108–1114. doi: 10.1016/j.procs.2014.05.540
11. Gerkey, B.P. & Mataric´, M.J. Multi-robot task

allocation: analyzing the complexity and optimality of key architectures. *In* IEEE International Conference on Robotics and Automation, 2003, pp. 3862–3868.
doi: 10.1109/ROBOT.2003.1242189
12. Woosley, B. & Dasgupta, P. Multirobot task allocation with real-time path planning. *In* Proceedings of the Twenty-Sixth International Florida Artificial Intelligence Research Society Conference (FLAIRS-26 Conference), FL, USA, 2013.
13. Huang, G; Zomaya, A. Y; Delicato, F.C. & Pires, P.F. Long term and large scale time synchronization in wireless sensor networks. *Computer Communications*, 2013, **37**, 77-91.
doi: 10.1016/j.comcom.2013.10.003
14. Parker, L.E. ALLIANCE: An architecture for fault tolerant multirobot cooperation. *IEEE Trans. Robotics Automation*, 1998, **14**(2), 220–240.
doi: 10.1109/70.681242
15. Botelho, S.C. & Alami, R. M+: A scheme for multi-robot cooperation through negotiated task allocation and achievement. Proceedings of International Conference on Robotics and Automation, 1999, pp. 1234–1239.
doi: 10.1109/ROBOT.1999.772530
16. Brutschy, A; Brambilla, M; Caro, G.D; Ducatelle, F; Dorigo, M; Ferrante, E; Gutierrez, A.; Gambardella, L.M.; Mathews, N.; O'Grady, R; Pinciroli, C.; Pini, G.; Stirling, T. & Trianni, V. ARGoS: A pluggable, multi-physics engine simulator for heterogeneous swarm robotics. IRIDIA – Technical Report Series. 2011, pp. 1–22.
doi: 10.1109/IROS.2011.6048500
17. Kudelski, M; Gambardella, L.M. & Di Caro, G. A. RoboNetSim: An integrated framework for multi-robot and network simulation. *Robotics Autonomous Sys.,* 2013, **61**(5), 483–49.
doi: 10.1016/j.robot.2013.01.003

CONTRIBUTORS

Ms Meetha V. Shenoy is working as Lecturer in the Department of Electrical and Electronics Engineering, BITS Pilani, KK Birla Goa Campus. Her research interest includes : Design and development of suitable hardware, software and protocol stack for efficient deployment of swarm robots and mobile wireless sensor networks in practical scenarios like search and rescue, intelligent monitoring, surveillance, etc. Currently she is working on aspects of synchronisation, localisation and schedulability aspects of dynamic large scale networks.
Contribution in the current study includes the design of DTTA framework, implementation and simulation of framework for different application scenarios described in the paper.

Dr K.R. Anupama is currently working as Professor in the Department of Electrical and Electronics Engineering, BITS Pilani K K Birla Goa campus. She is currently supervising 7 PhD scholar in the field of mobile robotics, wireless sensor networks and underwater sensor networks.
The author has provided valuable insights during the requirements analysis phase of DTTA framework. She also reviewed the DTTA framework and validated the implementation results.