International Journal of Electrical, Electronic and Communication Sciences
ISSN: 2517-9438
Vol:3, No:6, 2009

A Cooperative Multi-Robot Control Using Ad Hoc Wireless Network

Amira Elsonbaty, Rawya Rizk, Mohamed Elksas, and Mofreh Salem

***Abstract*—**In this paper, a Cooperative Multi-robot for Carrying Targets (CMCT) algorithm is proposed. The multi-robot team consists of three robots, one is a supervisor and the others are workers for carrying boxes in a store of 100×100 m^2. Each robot has a self recharging mechanism. The CMCT minimizes robot's worked time for carrying many boxes during day by working in parallel. That is, the supervisor detects the required variables in the same time another robots work with previous variables. It works with straightforward mechanical models by using simple cosine laws. It detects the robot's shortest path for reaching the target position avoiding obstacles by using a proposed CMCT path planning (CMCT-PP) algorithm. It prevents the collision between robots during moving. The robots interact in an ad hoc wireless network. Simulation results show that the proposed system that consists of CMCT algorithm and its accomplished CMCT-PP algorithm achieves a high improvement in time and distance while performing the required tasks over the already existed algorithms.

***Keywords*—**Ad hoc network, Computer vision based positioning, Dynamic collision avoidance, Multi-robot, Path planning algorithms, Self recharging.

I. INTRODUCTION

TODAY robot systems are becoming more and more significant in various aspects of human life, for example in industrial, commercial and scientific applications. A key driving forces in the development of robotic systems is their potential for reducing the need for human presence. As a result of scientific achievements and industrial development, the number of robots being used in industrial projects is increased. Autonomous robots performing a common task not achievable by a single robot must coordinate their actions in order to success [1].

While all fields of robots have progressively based on rapid advances in computing and information systems. The field of distributed and cooperative robots has been catalyzed in particular by new networks and communications technologies. In robotics context, the linkage of information systems through ad hoc network facilitates interactions that quickly scale from two robots shaking hands to a swarm of multi robots demonstrating cooperative behaviors. Mobile ad hoc networks are a collection of mobile nodes or terminals that cannot rely on centralized or organized connectivity. These networks have no base stations, no fixed routers and no centralized administration. All nodes may move randomly and are connecting dynamically to each other. Therefore all nodes are operating as routers and need to be capable of discovering and maintaining routes to every other node in the network and to propagate packets accordingly.

This paper presents team of three robots work together without collision. They have self recharging mechanism, where their batteries are recharged periodically according to the total worked time passed from the previous charge. The paper focuses on the interactions and the constraints among robots to detect the position of boxes. The robots communicate through an ad hoc network.

The rest of the paper is organized as follows. Section II presents the related work. Section III introduces the proposed cooperative multi-robot for carrying targets. Section IV presents the proposed path planning algorithm. Section V introduces the simulation results. Finally, Section VI presents the main conclusion and the future work.

II. RELATED WORK

Several multi-robot systems have been proposed. Most research efforts on multi-robots have focused on multi-robot control, path planning for robot navigation in a dynamic environment, or the communication among robots.

The Cooperative Multi-robot Observation of Multiple Moving Targets (CMOMMT) was proposed in [2]. It is a cooperative team of autonomous sensor-based robots for the observation of multiple moving targets. It focuses primarily on developing the on-line distributed control strategies that allow the robot team to attempt to minimize the total time. In [3], team of robots assists firefighters by searching and inspection of the building was used. A robust ad hoc communications between firefighters was used. Distributed In-Network Task Allocation (DINTA) was designed in [4] to solve the online multi-robot task allocation problem. DINTA combined the benefits of a sensor network with the mobility and functionality of robots. In [5], some problems that are expected to happen in dynamic modeling of a multiple-degree-of-freedom robot arm were studied. The model can be derived by calculating the kinetic energy and potential energy of the links and the joints. A dynamic model combining the dynamics of both robot arm and platform were developed.

Networked robot system called MiNT is designed to support high fidelity wireless network emulation [6]. The communications among these robots are through wireless LAN interfaces. MiNT has self-charging mechanism, and computer vision-based robot positioning system. In [7], the

---

Eng. Amira A. Elsonbaty is with Alzarka Higher Institute for Administration and Computer Sciences.

Dr. Rawya Y. Rizk* is with Electrical Engineering Dept., Suez Canal University (*Corresponding author; e-mail: r.rizk@ scuegypt.edu.eg).

Dr. Mohamed S. Elksas is with the Computers & Systems Engineering Dept., Mansoura University, Egypt.

Prof. Mofreh M. Salem is with the Computers & Systems Engineering Dept., Mansoura University, Egypt.

drawbacks of wireless networking were overcomed by developing a communication system based on optical fiber links, with high bandwidth and converters at both ends. A method to evaluate control and coordination strategies for group of wireless networked robots was proposed in [8]. An integrated robot and network simulation tool was developed to allow investigation of the behavior of robot controllers combined with detailed communication models embedded in a virtual environment.

Hierarchical Path Finding A* **(**HPA*) for reducing problem complexity in path finding on grid based maps was proposed in [9]. It abstracted a map into linked local clusters, at the local level. The optimal distances for crossing each cluster were computed and cached. The global level, clusters were traversed in a single big step. Small clusters are grouped together to form larger clusters. Computing crossing distances for a large cluster used distances computed for the smaller contained clusters. A dynamic path finding algorithm for a vehicle-based automated material handling system (AMHS) was discussed in [10]. Distance between nodes, node penalties, number of vehicles queued and historical data from the AMHS was used to calculate the total cost of a path. It discussed how to effectively utilize such data in critical situations to improve overall AMHS performance. The accuracy of position estimation for groups of mobile robots performing cooperative localization was studied in [11]. Analytical expressions for the upper bound on their expected positioning uncertainty were provided by using relative position measurement graph (RPMG).

III. THE PROPOSED COOPERATIVE MULTI-ROBOT FOR CARRYING TARGETS

Several multi-robot systems have many attractive features, but most of them have problems of which the crucial one and how to control effectively a group of robots. Number of multi-robot systems work with complex mechanical models. Various systems have complex equipments for building robot's body. Some robot groups have environment including obstacles, which move with low-quality motion planning algorithm. Some systems need the human role to recharge robot's battery or for another task. In a multi-robot system, robots interact with each other by a network that may be affected with wired cables and limited bandwidth.

The Cooperative Multi-robot for Carrying Targets (CMCT) is proposed as a combination between centralized and distributed control. It overcomes some drawbacks founded in the other approaches by supporting the following features:

- It works with straightforward mechanical models (model of moving robot, model of manipulator, and model of color camera) by using simple cosine lows.
- It detects the robot's shortest path for reaching the target position avoiding obstacles by using a proposed path planning algorithm.
- It minimizes robot's worked time for carrying many boxes during day by working in parallel. That is, the supervisor detects the required variables in the same time another robots work with previous variables.
- It prevents the collision between robots during moving.
- It provides a good interaction between the team of robot using ad hoc wireless network.

A. Working Space Environment

Working space is a store with area 100 × 100 m^2. Objects are boxes. Load threshold is measured according to the size of the motor of robot. Robot's worked time threshold is the threshold of battery used (6 hours) as in [12]. Traveled distance threshold is greater than twice the work width of the robot, or less than approximately ten times the work width of the robot, according to the velocity of the robot [13].

B. Components of the Proposed System

The proposed system includes three robots as shown in Fig. 1. First and second robots are Worker 1 and Worker 2. Each consists of arm (2-links as serial manipulator), hand, four conventional wheels (rolling robot), platform, timer to calculate the total worked time and counter to calculate the total traveled distance. The third robot (Supervisor) consists of rolling robot with four conventional wheels and two color cameras to detect around. The three robots share in two charging stations. Ad hoc wireless network is used to communicate between the three robots.

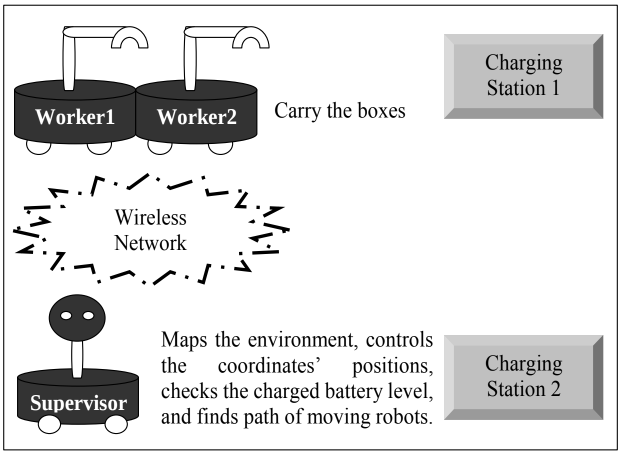

Fig. 1 The components of the proposed CMCT work space

C. Assistance Mechanisms

Assistance Mechanisms use in CMCT algorithm are; self recharging mechanism, static path planning algorithm for navigation, computer vision based positioning system, dynamic collision avoidance algorithm, and NS_2 network simulator. Each item is explained in the following:

- *Self Recharging Mechanism:*

Self recharging mechanism is designed to determine the recharge time for each robot. Typically charging a robot's battery is a manual process that requires the administrator to physically take the robot to a charging station [14]. The proposed CMCT supports automatic recharging of robot battery when it becomes low. Fig. 2 shows the charging station. It has an AC power plug and an infrared (IR) beacon.

The robot's IR sensors perform the beacon detection. When the capacity of the robot’s battery drops below a certain threshold, it starts looking for any beacon signal emitted by a docking station. It uses this signal to home into the docking station and recharge its battery. A timer is used to compute the total work time of each robot. If this time reaches a certain threshold, the robot seeks for a charging station to recharge its battery.

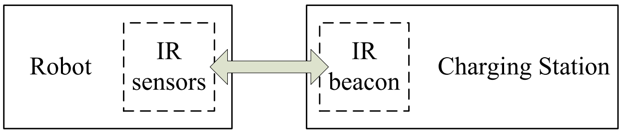

Fig. 2 Self charging station

For sharing a charging station, the robots need to share the station efficiently as shown in Fig. 3. If the robot reaches the station while there is another robot recharging its battery, it will take one of two decisions. First, it waits few seconds. Second, it seeks another charging station.

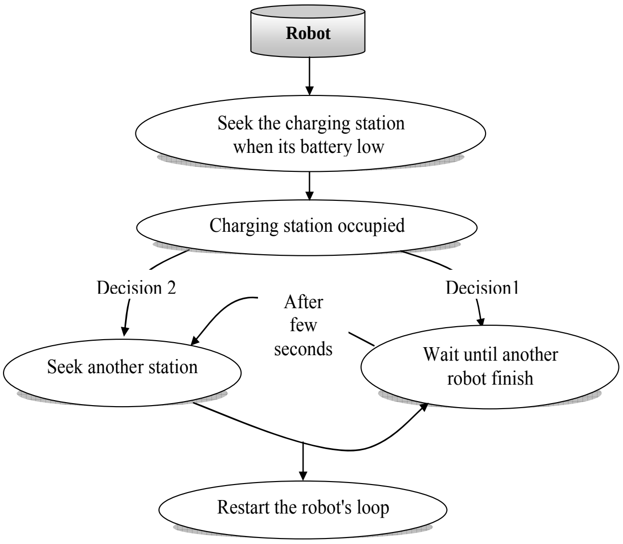

Fig. 3 Sharing a charging station

- *Static Path Planning Algorithm for Navigation:*

The aim of path planning is to decide which route to take [15]. The supervisor robot needs to compute the path for each robot with static environment. Then it sends the corresponding movement commands to the robots at the proper moment. The supervisor takes the current position and the target destination of a robot, then checks if there is a direct path between the worker robot and the box. If such paths do not exist, the algorithm identifies the obstacle closest to the source position and detects the best moving path for robots. There are different algorithms for solving the path planning problem based on an environment map for mobile robots [16] such as Roadmap algorithm, Cell Decomposition algorithm, and Potential Field algorithm. A* is a well known algorithm for finding path planning for mobile robots. In this paper, a comparison between A* algorithm and the proposed CMCT algorithm is carried out.

- *A Computer Vision Based Positioning System:*

A Computer vision based positioning system tracks the position of each robot for collision free robot movement [17]. The robot positioning system in CMCT is necessary to maintain the location and orientation information of each robot. One simple way to track the position of a mobile robot is to use its static path planning with respect to a fixed direction, send position coordinates to a mobile robot from supervisor, and process by mechanical model to get results.

- *Dynamic Collision Avoidance Algorithm:*

If any two robots are closer than a threshold distance, including when they already collide with each other, the supervisor stops both of them, computes a new path for each of them if necessary, and moves them on their new path.

- *NS_2 Network Simulator:*

NS_2 is a discrete event simulator written in C++. It supports TCP, routing algorithm, queuing algorithms, and multicast protocols over wired and wireless (local and satellite) networks. It is freely distributed and all source code is available [18]. Wireless NS_2 protocols support ad hoc routing and mobile IP and is used in wireless and satellite networking. The simulator (NS), the network animator (nam), visualizing ns output, and nam editor are the components of NS_2.

C. The Proposed CMCT Algorithm

The flow chart of the proposed CMCT algorithm is shown in Fig. 4. The following considerations are assumed: TDIS(i) is the robot's worked time, where i is the robot number (i= 1,2 or 3). TDS is the robot's worked time threshold. DIS is the distance between two robots. Ds is the nearest distance between two robots to avoid collision. B is a box. The Coordinates of the initial position of a box are Xs and Ys. The coordinates of the target position of a box are Xf and Yf. NoB is the number of boxes in the store. N is the variable for determining the number of boxes carried. W is the Box's weight. Ws is the load threshold. R1, R2 are worker robots, and R3 is the supervisor robot.

In the work space environment:

1) First the operator checks the supervisor's battery level. If the level is high (not reached battery threshold value), then it begins the work. Else (the supervisor's battery level is low), then it automatically seeks the nearest charging station and recharges its battery.
2) The supervisor measures the battery level for the two worker robots.
   - If the battery level for any robot is low (it is equal to the battery threshold value), then the robot with the low battery automatically seeks the nearest charging station by applying the self recharging mechanism.

- Else (Each of the batteries of the two robots is less than battery threshold value), the supervisor does the following steps:

3) It detects worker robots' positions, initial box's position, target box's position and obstacles' positions by making map of the environment.
4) It measures the weight of the box by a load sensor connected at the end effectors of the robot arm.
   - If the weight of the box is smaller than or equal to a load threshold, it will be carried by one worker and the supervisor detects another box's variables to the other worker to carry it.
   - Else (it's weight larger than load threshold), it will be carried by the two workers.
5) If the supervisor's battery level becomes low, it automatically seeks the nearest charging station, recharges its battery, and repeats the Step 1.

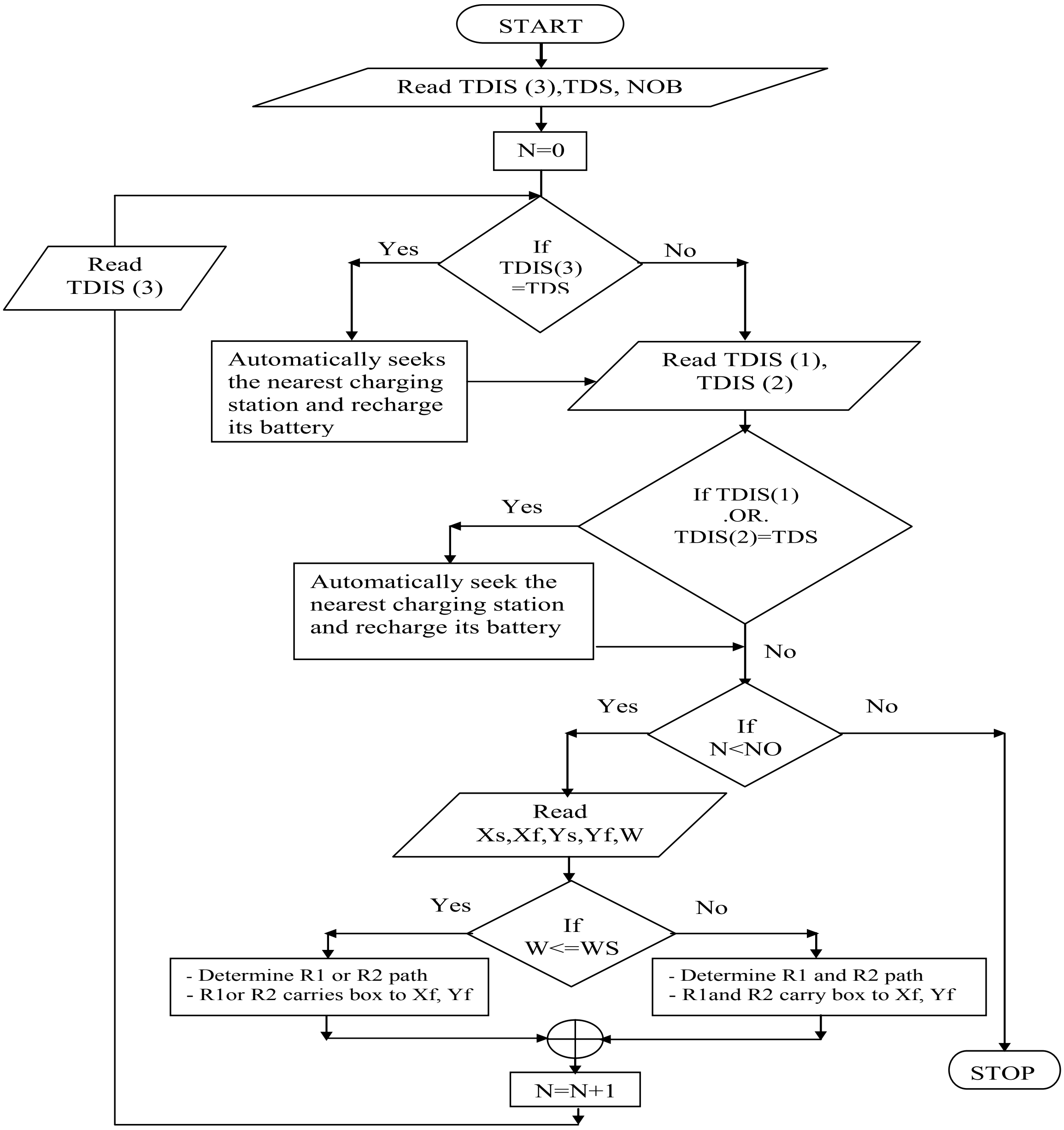

Fig. 4 The flow chart of the proposed CMCT algorithm

D. Kinematics Models Mechanical

This section presents the kinematics models mechanical for moving robot, manipulator, and color camera.

- *Moving robot: Aanalysis one wheel*

It is assumed that ø1: Angle for first start position with origin (wheel), ø2: Angle for first stop position with origin (wheel), (X1,Y1): Start coordination, (X2,Y2): Target coordination. Ex: Error signal for X, Ey: Error signal for Y, Cx: Moving Step for X, Cy: Moving Step for Y, X_m: in-between point from (X1, X2) and Y_m: in-between point from (Y1, Y2). The Position coordinate of one wheel for moving robot is shown in Fig. 5.

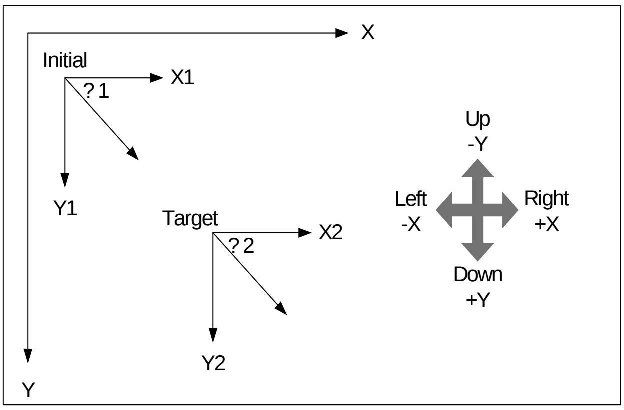

Fig. 5 Position coordinates of one wheel for a moving robot

i. *The angle for start and stop position*

Tan (ø1)= Y1/ X1, ø1= ArcTan (Y1/ X1) (1)

Tan (ø2)= Y2/ X2, ø2= ArcTan (Y2/ X2) (2)

ii. *The error signal for x and y*

Ex= Abs (X2-X1), Ey= Abs (Y2-Y1) (3)

$$E=\sqrt{Ex^{2}+Ey^{2}} \tag{4}$$

iii. *The moving step in direction x and y*

$$Cx=\frac{Abs\ \ (X2-X1)}{10} \tag{5}$$

$$Cy=\frac{Abs\ \ (Y2-Y1)}{10} \tag{6}$$

iv. *The value of Xm, Ym for different directions*

Right (+X): X_m = X1 + Cx , Y_m =Y1 (7)

Left (-X): X_m = X1- Cx, Y_m =Y1 (8)

Up (-Y): X_m = X1, Y_m =Y1-Cy (9)

Down (+Y): X_m =X1, Y_m =Y1+Cy (10)

- *Manipulator*

The 2-links that are serial manipulator is shown in Fig. 6 [19]. The following parameters are assumed in the analysis: θ_1 is the angle for joint_1of manipulator. θ_2 is the angle for joint_2 of manipulator. $\bar{\theta}_1$=180- θ_2. L_1 is the segment_1 length of manipulator. L_2 is the segment_2 length of manipulator. X and Y are coordinates of end effector point.

$$X^2+Y^2=L_1{}^2+L_2{}^2-2L_1L_2\quad Cos(180-\theta_2) \tag{11}$$

$$Cos(\theta_2)=\frac{X^2+Y^2-L_1{}^2-L_2{}^2}{2L_1L_2} \tag{12}$$

$$\theta_2=ArcCos\left[\frac{X^2+Y^2-L_1{}^2-L_2{}^2}{2L_1L_2}\right] \tag{13}$$

$$\frac{Sin(\theta_1)}{L_2}=\frac{Sin(180-\theta_2)}{\sqrt{X^2+Y^2}}=\frac{Sin(\theta_2)}{\sqrt{X^2+Y^2}} \tag{14}$$

$$\theta_1=ArcSin[\frac{L_2\quad Sin(\theta_2)}{\sqrt{X^2+Y^2}}]+ArcTan[2\frac{y}{x}] \tag{15}$$

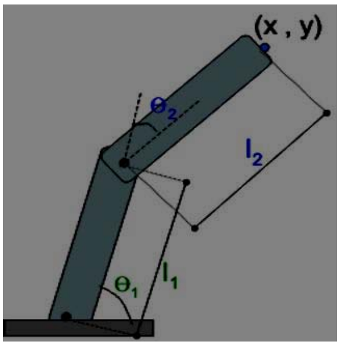

Fig. 6 End effector's coordinates of a manipulator

- *Model for two color cameras*

Two color cameras making map for around environment by using a high-level computer language (Visual Basic). Any video capture device is automatically detected and used to capture robot video making the robot vision system. The video from the color camera is sent to the computer from the mobile robot through the instruction: (VideoOCX1.start) which starts the video communication links.

IV. PATH PLANNING ALGORITHMS

This section presents the proposed CMCT path planning (CMCT-PP) algorithm and introduces a comparison between the proposed algorithm and the A* algorithm.

A. A Algorithm*

A* is a best-first, graph search algorithm that finds the least-cost path from a given initial node to one goal node (out of one or more possible goals) [9]. The algorithm traverses various paths from start to goal. For each node x traversed, it maintains 3 values:

g(x): the actual shortest distance traveled from the initial node to the current node.
h(x): the estimated (or "heuristic") distance from the current node to the goal node.
f(x): the sum of g(x) and h(x).

The Algorithm

Starting with the initial node;

1. The algorithm maintains a priority queue of nodes to be traversed, known as the open set.
2. The lower f(x) for a given node x, the higher its priority.
3. At each step of the algorithm, the node with the lowest f(x) value is removed from the queue, the f and h values of its neighbors are updated accordingly.
4. These neighbors are added to the queue.
5. The algorithm continues until the goal node has a lower f value than any node in the queue (or until the queue is empty).

B. The Proposed CMCT Path Planning Algorithm

In the proposed CMCT-PP algorithm, the work space area is assumed 100×100 m^2. To determine the shortest path for moving robot from source to target position, the work space is divided into cells each cell has area (10×10 m^2). The following parameters are assumed**:** Xs and Ys are the initial position coordinates for moving. Xf and Yf are the target position coordinates for moving. Xm1 is the new X position for path1, Ym1 is the new Y position for Path 1, Xm2 is the new X position for Path 2, Ym2 is the new Y position for Path 2, Xm3 is the new X position for Path 3 and Ym3 is the new Y position for Path 3. Cx is the X moving step and Cy is the Y moving step. D1 is the length of Path 1, D2 is the length of Path 2 and D3 is the length of Path 3. Dt is the total distance. The proposed CMCT-PP algorithm is shown in Fig. 7.

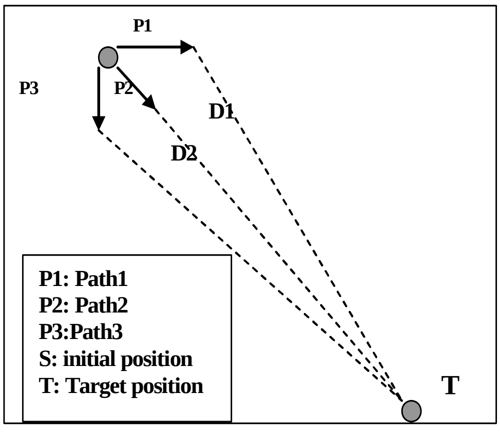

Fig. 7 The proposed CMCT-PP algorithm

The Algorithm

```
1. The Supervisor reads the coordinates, Xs, Ys, Xf, Yf, D1, D2.
2. $Cx = \frac{Abs\ (Xs-Xf)}{10}$, $Cy = \frac{Abs\ (Ys-Yf)}{10}$.
3. Dt =0.
4. Xm1= Xm2= Xm3= Xs,
   Ym1= Ym2= Ym3= Ys.
5. Check the available paths (Path 1, Path 2, Path 3) by the following steps
   – Path 1: Xm1= Xm1+Cx, Ym1=Ym1
       $D1=\sqrt{(Xf-Xm1-Cx)^2}+\sqrt{(Yf-Ym1)^2}$
   – Path 2: Xm2= Xm2+Cx, Ym2=Ym2+Cy.
       $D2=\sqrt{(Xf-Xm2-Cx)^2}+\sqrt{(Yf-Ym2-Cy)^2}$
   – Path 3 Xm3= Xm3, Ym3=Ym3+Cy.
       $D3=\sqrt{(Xf-Xm3)^2}+\sqrt{(Yf-Ym3-Cy)^2}$
   – If (D1>D2.and.D1>D3) Then draw line between
       (Xs,Ys) and (Xm1,Ym1), Xs=Xm1, Ys=Xm1,
       Dt=Dt+D1, Go to 6.
   – Else if (D2>D1.and.D2>D3) Then draw line between
       (Xs,Ys) and (Xm2,Ym2), Xs=Xm2, Ys=Xm2,
       Dt=Dt+D2, Go to 6.
   – Else draw line between (Xs,Ys) and (Xm3,Ym3),
       Xs=Xm3, Ys=Xm3, Dt=Dt+D3.
6. If ( (Xs=Xf) .And. (Ys=Yf)) Then print Dt, go to 7.
   Else Go to 4.
7. Stop.
```

V. SIMULATION RESULTS

A. The Proposed CMCT-PP Algorithm

The comparison between CMCT-PP and A* algorithms is centered around the following criteria: the total time required for carried out a certain task, the distance that expresses the path length to reach the required position, the accuracy of position and the processing needs. Figs. 8-11 present the comparison between CMCT-PP and A* algorithms.

Fig. 8 shows the required time versus the number of performed tasks in the case of the absence of obstacles. The time increases with the increase of the number of tasks. It is clear that, the proposed algorithm achieves the required tasks at lower time than the A* algorithm. The time saving in the CMCT-PP algorithm reaches about 100% (At 4 tasks, the time of the proposed algorithm is 7 s as opposed to 14 s in the A* algorithm). Fig. 9 presents the total distance of the two algorithms against the number of performed tasks without obstacles. It is shown that, the CMCT-PP algorithm find shorter path to reach the required positions than the A* algorithm. The distance of the CMCT-PP is about half the distance of the A* algorithm.

Figs. 10 and 11 present the time and the distance of the two algorithms in the case of the presence of obstacles. They confirm the above results. It is shown that, both the time and the distance of the two algorithms increases in the case of obstacles.

Another comparison metrics such as complexity and accuracy prove the efficiency of the proposed CMCT-PP algorithm. The CMCT-PP algorithm is simpler than A* since it doesn't need memory. A* algorithm needs memory terms to save open list and close list. The CMCT-PP algorithm can obtain the accurate total distance while A* algorithm does not guarantee the accurate total distance.

B. *The Proposed CMCT Algorithm*

The comparison between the proposed CMCT algorithm and the CMOMMT algorithm [2] is presented. The CMOMMT algorithm works with variable number of robots and speeds and without supervisor. The CMCT algorithm works with fixed number of robots (three robots), fixed speed for the two worker robots and the supervisor. In the simulation, the variable number of robots, speeds, and obstacles in both algorithms are ignored (i.e., the same environment is assumed for the two algorithms).

Fig. 12 shows time versus distance in the proposed CMCT and the CMOMMT algorithms using the CMCT-PP algorithm. It is clear that, CMCT saves more time than CMOMMT since that CMOMMT is a distributive control algorithm, while the robots in CMCT work in parallel. Thus, while the two worker robots move toward their targets, the supervisor robot detects the new variables for the new tasks. Fig. 13 presents the comparison between CMCT and CMOMMT in the case of using A* path planning algorithm. It is shown that, the time of the two algorithms increase than in the case of using CMCT-PP. The proposed CMCT always achieves better time than the CMOMMT algorithm.

VI. CONCLUSION

This paper studies the performance of cooperative multi-robot team, focusing on the mobile robot navigations. Ad Hoc wireless network is used for communication between a group of robots over the range of a store. In this paper, a new algorithm called CMCT is proposed. It is based on self recharging mechanism. An accomplished path planning algorithm (CMCT-PP) is also proposed for use with the CMCT algorithm. The comparison between CMCT-PP algorithm and the well known path planning algorithm (A*) proves the efficiency of the proposed algorithm. It is shown that, CMCT-PP achieves a 100% improvement of time and distance than A*. In addition it has less processing needs and provides more accurate total distance. Another comparison between CMCT and the already existed cooperative algorithm (CMOMMT) shows that, CMCT improves the performance of the cooperative multi-robot team.

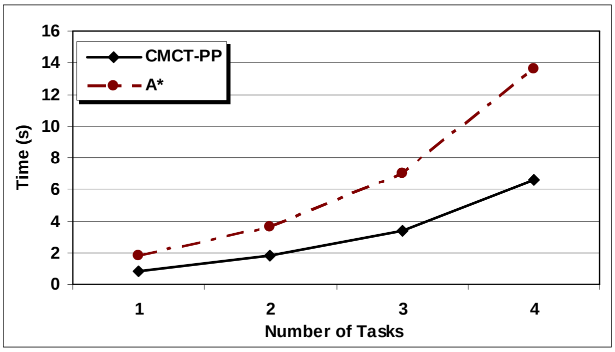

Fig. 8 Time versus the number of performed tasks in both A* and CMCT-PP algorithms without obstacles

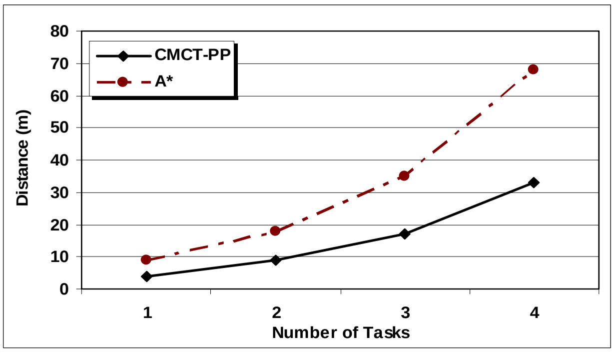

Fig. 9 Distance against the number of performed tasks in A* and CMCT-PP algorithms without obstacles

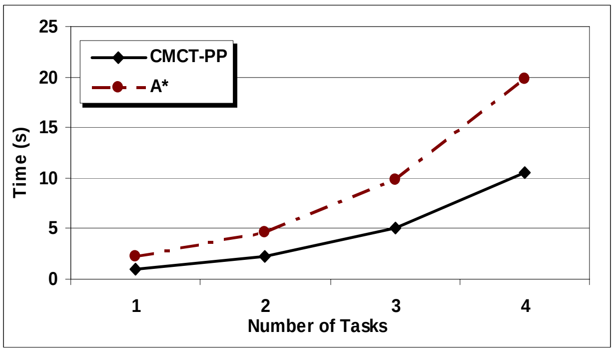

Fig. 10 Time versus the number of performed tasks in both A* and CMCT-PP algorithms with obstacles

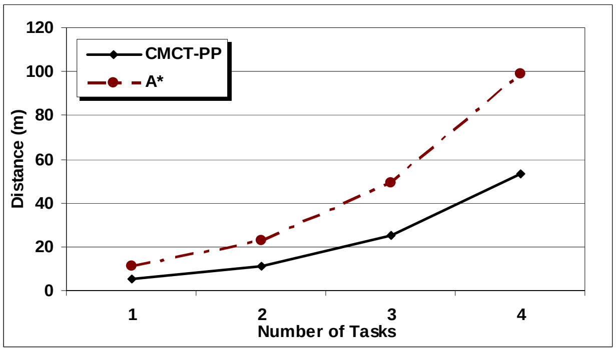

Fig. 11 Distance against the number of performed tasks in A* and CMCT-PP algorithms with obstacles

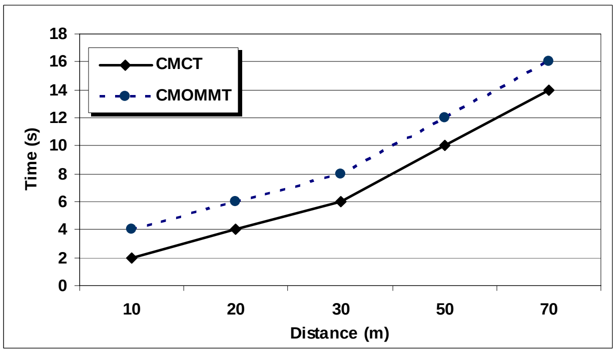

Fig. 12 Time versus distance in the CMCT and the CMOMMT algorithms using CMCT-PP algorithm

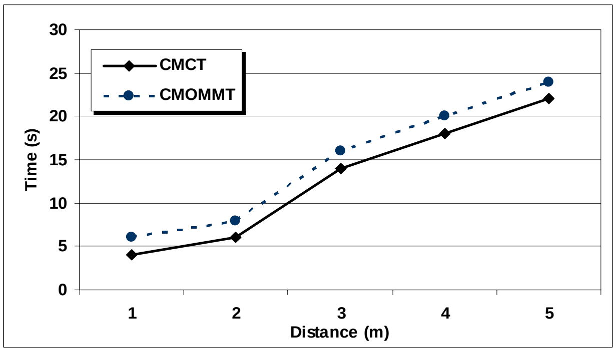

Fig. 13 Time versus distance in the CMCT and the CMOMMT algorithms using A* path planning algorithm

REFERENCES

[1] T. Zheng, R. Li, W. Guo, and L. Yang, “Multi-robot Cooperative Task Processing in Great Environment,” in Proc. of IEEE Conference on Robotics, Automation and Mechatronics, pp. 1113-1117, September 2008.

[2] L. Parker, "Distributed algorithms for multi-robot observation of multiple moving targets," Autonomous Robots Journal, vol.12, pp. 231–255, 2002.

[3] Larp Euron, "Ad-hoc network communication infrastructure for multi-robot systems in disaster scenarios," in Proc. of Robotics for Risky Intervention and Environmental Surveillance - Benicàssim, 2008.

[4] M. J. Johnson, K. Intlekofer, H. Jung, and J. M. Bradshaw, et. al, " Coordinated operations in mixed teams of humans and robots," International Journal of Robotics Research 2, pp. 431-447, no. 5,6, 2006.

[5] J. Wang "Investigation of a mobile manipulator over wired/wireless control System,” in Proc of Skima–International conference on software knowledge information management and applications, 2006.

[6] T. Chiueh, R. Krishnan, P. De, and J. Chiang, "A networked robot system for wireless network emulation", in Proc. of the 4th International Conference on Mobile Systems, Applications, and Services, Uppsala, Sweden, pp. 124-137, 2006.

[7] A. Birk, and C. Condea "Mobile robot communication without the drawbacks of wireless networking" Springer, vol. 4020, pp. 585 – 592, 2006.

[8] A. R. Wagner, and R.C. Arkin, “Internalized plans for communication sensitive robot team behaviors”, in Proc. of. IEEE Int. Conference on Intelligent Robotics and Systems, pp 2480-2487, 2003.

[9] A. Botea, M. Müller, and J. Schaeffer "Near optimal hierarchical path-finding", Journal of Game Development, vol. 1, issue 1, pp. 7-28, 2004.

[10] M. Fukunari, S. Rajanna, R. J. Gaskins, and M. E. Sparrow "Data-based node penalties in a path-finding algorithm in an automated material handling system,” in Proc. of the 34th conference on Winter simulation, California, pp1383-1386, 2002.

[11] A. I. Mourikis, and S. I. Roumeliotis, "Performance analysis of Multirobot cooperative localization," IEEE Transactions on Robotics, vol. 22, issue 4, pp. 666-681, 2006.

[12] S. Oh., A. Zelinsky, and K. Taylor, "Autonomous battery recharging for indoor mobile robots," in Proc. of Australian Conference on Robotics and Automation (ACRA 2000), pp. 1-6, 2000.

[13] J. L. Jones, and P. R. Mass "Method and system for multi mode coverage for an autonomous robot," Citation of provisional application, Jones, US Patent 7, Google Patents, 2007.

[14] K. kouzoubov, and D. Austin, "Autonomous recharging for mobile robotics," in Proc. of Australasian Conference on robotics and automation Auckland, pp. 27-29, 2002.

[15] J. Chen., and L. Ren, "Path planning protocol for collaborative multi-robot systems " in Proc. of IEEE International Symposium, Computational Intelligence in Robotics and Automation, pp.721–726, June 2005.

[16] B. Graf, J. Wandosell, and C. Schaeffer, "Flexible path planning for nonholonomic mobile robots," in Proc. of the Fourth European Workshop on Advanced Mobile Robots, 2001.

[17] S. Jantz, Keith, and L Doty, "Visual perception based behaviors for a small autonomous mobile robot," in Proc. of Florida Conference on Recent Advances in Robotics, Florida, 2000.

[18] http://www.isi.edu/nsnam/ns,10/2008.

[19] Y. Yong, and G. Kamal, "Puma 560 software client's report", Advanced Robotics , vol. 14, issue 22, no. 6, pp. 515-536, 2000