Energy-Efficient Distributed Target Tracking using Wireless Relay Robots

Chia Ching Ooi and Christian Schindelhauer

Abstract This paper considers the problem of mobile target tracking, in which a pursuer robot has to capture a mobile target with the aid of multiple wireless relay robots. We assume that the relay robots are able to self-organize to form an ad hoc robotic network when a tracking event is triggered, and act as static multihop relays to disseminate the latest target trajectory they detect to the pursuer. The task of the pursuer is to plan its path to capture the target based on the information received from the relays. Only the first position of target is given to the pursuer initially. No global information and no centralized infrastructure are provided. The heterogeneous robot team has to solve the problem using a distributed approach. We address this challenge by introducing Distributed Relay-robot Algorithm (DRA), which consists of two parts: information dissemination of target trajectory by relay robots using *tAdv* message and prediction of target position based on the first part. We study DRA using a network simulator and compare the results with Centralized-Control Approach (CCA). The outcomes show that DRA is more reliable, and it is feasible to use multihop communications to aid target tracking in a distributed way.

1 Introduction

Recently, networked robotics has been gaining increasing attention among robotic and mobile networking research community. With the unprecedented growth of wireless communication technologies witnessed over the past two decades, many modern robots are already equipped with wireless communications capability [4, 2], using either radio frequency, for instance, bluetooth or IEEE 802.11-compliant [3] network interfaces, or infra-red communications.

Multi-robot systems (MRS) [10] take the advantage of collective and cooperative behavior of robots through inter-robot communications. Swarm robotics [5] is

Albert-Ludwigs-Universität Freiburg, Computer Networks and Telematics, Georges-Köhler-Allee 51, 79110 Freiburg, Germany. {ooi,schindel}@informatik.uni-freiburg.de

an example of MRS that deploys and coordinates large number of relatively simple physical robots to complete assigned tasks as a team. These swarm robots can form a heterogeneous robotic team with larger, more powerful robots to cooperate to achieve common goals. Among them, mobile target tracking problem can be solved by employing such a heterogeneous robotic team. Due to their low power, the lightweight robots can act as static wireless relay robots to disseminate information of detected target, while the high power robots work as pursuers to chase and capture mobile targets.

The co-location of the motion and task control systems with the communications mechanisms in a robot enables the co-exploitation of individual subsystems, providing a completely new kind of networked autonomy that is beneficial to many robotic applications. However, only little on-going researches make use of this advantage. Our paper explores this advantage to solve mobile target tracking problem using a team of heterogeneous robots communicating wirelessly.

The remainder of the paper is organized as follows. Section 2 reviews related work on target tracking. In Section 3, we present the system and the energy models used in our research and define the problem of mobile target tracking using multiple wireless relay robots. Based on these foundations, we propose and analyze a distributed strategy called Distributed Relay-robot Algorithm (DRA) in Section 4. The proposed scheme is evaluated through simulation described in Section 5. Lastly, we make concluding remarks and discuss future works.

2 Related Work

Target tracking problem has been a popular topic among the research community of Wireless Sensor and Actor Networks (WSANs) [6] since the past few years. Among them, binary-sensor network [7] introduces a binary sensor model, in which one-bit information is detected and broadcast to a base station. The one-bit information determines if an object is moving away or closer to the sensor, and therefore, target direction can be predicted. Using one-bit data is reasonable since sensor nodes have very limited energy and computational resources. However, it uses a centralized approach, with the base station keeping the bit information from every sensor.

In [17], the concept of convoy tree based collaboration is introduced to track moving target. The target trajectory determines the addition and removal of sensor nodes from the convoy tree. This paper contributes mainly to the energy efficiency issue. Both of the papers do not consider the path planning of a pursuer robot.

Pursuer-evader tracking is a specific tracking in which the evader always attempts to avoid pursuer by maximizing the distance between both nodes. The authors of [9] present a self-stabilizing approach that dynamically maintains a tracking tree of wireless sensor nodes rooted at evader. Only when the pursuer has found the tracking tree that it can follow the tree towards the evader. This paper restricts the speed of pursuer to be larger than the target speed. Moreover, the energy consumption for pursuer mobility is proportional to the distance traveled by target. In our proposed algorithm, we use target position prediction based on previous target trajectory information obtained from relay robots, leading to a more energy-efficient path of pursuer.

On the other hand, [13] develops a hierarchical control system involving a multi-layer data fusion in wireless sensor network. The control system has been demonstrated on a real sensor network deployment. However, a centralized controller is required to coordinate the pursuers to capture the evaders.

Generally, a lightweight robot used in swarm robotics has higher power than a wireless sensor node. The wireless capability of robot enables it to act as a communications relay. To our best knowledge, no previous work addresses mobile target tracking problem using low power robots as wireless multihop relays, which assist a higher power pursuer robot in a distributed way. When a pursuer can obtain target information via the relay robots, a more efficient path of pursuer can be computed, leading to highly efficient energy and time consumption of the whole robotic team.

3 Preliminaries

We consider an autonomous MRS that is employed to perform multiple tasks to achieve its assigned goals. The robot team is heterogeneous, and consists of both high and low power robots. Each robot is equipped with sensors to detect and identify moving objects and wireless data communications capability. Some tasks to be performed are event-driven. Among them, the robots have to form a mobile wireless network to track mobile target(s) in, for instance, an intruder tracking system.

Due to the heterogeneity of the MRS and the different energy requirements for movement and radio communications, only the high power robots are assigned to act as pursuers, while the low power robots serve as wireless relays. In this scenario, we call the high power robot a pursuer and the low power robot a relay robot. The pursuer has to detect, chase and capture the moving targets with the aid of the relay robots.

The energy required for robot movements [14] is generally much bigger than that for communications, if the communications do not involve high amount of data, e.g. multimedia data. According to [16], a wheeled vehicle with rubber tires at one kilogram moving on concrete must expend 0.1J/m to overcome 0.1N force of dynamic friction.On the other hand, depending on the transceiver sensitivity, the energy required by the power amplifier of transceiver to transmit one bit data over the distance of one meter ranges from some pico- to nano-Joule per bit per meter$^{\alpha}$ [1].

We assume that every robot is aware of its own location by using localization [8] technique, and does time synchronization [15] prior to or during the tracking event. In addition, each relay robot is assumed to be able to detect and measure the target trajectory, including speed, direction, and the distance between the moving target

[1] α is the path loss exponent of the transmission medium ranging from 2 to 6

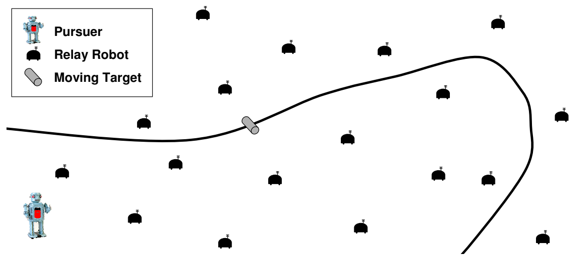

Fig. 1 Mobile Target Tracking using Wireless Relay Robots

and the relay robot using the motion detector, whenever the target is within a robot's sensing range.

In the next sub-sections, we introduce the system and the energy models, and define the problem of tracking a mobile target by a team of heterogeneous robots.

3.1 System Model

The MRS consists of a single pursuer P and multiple relay robots. The tracking area of the MRS is modeled on a two dimensional Euclidean space as illustrated in Fig. 1. When a tracking event is triggered, the robots self-organize themselves to form a wireless ad hoc network [11]. Relay robots are static while the pursuer is mobile. Let N_s be the number of robots over the tracking region $\mathscr{R} \subset \mathbb{R}^2$, $s_0 \in \mathscr{R}$ be the initial location of P, and $s_i \in \mathscr{R}$ be the position of i-th relay where $i \neq 0$, the robot set is defined as $S = s_i : 0 \leq i < N_s$. Let R_c and R_s be the communication and sensing range of the robots respectively. $G = (S, E)$ is a communication graph of the wireless robot network such that $(s_i, s_j) \in E$ if and only if $||s_i, s_j||_2 \leq R_c$, where $||u, v||_2 = \sqrt{(u_x - v_x)^2 + (u_y - v_y)^2}$ denotes the Euclidean distance between two robots.

Let $t_s = 0$ be the time the target first enters the sensing area of the MRS, and t_e denote the time the pursuer captures the target. We indicate the pursuer and target position as continuous function of time $p : [0, t_e] \mapsto \mathscr{R}$ such that $p_p(t)$ and $p_q(t)$ denote the position of the pursuer robot and the mobile target respectively at time t in Cartesian Coordinates $p(t) = (p_x(t), p_y(t))$. When a target Q appears at $p_q \in \mathscr{R}$, robot i can detect Q if and only if $||s_i, p_q||_2 \leq R_s$.

Both the target and pursuer are moving at constant speed v_q and v_p respectively. When a relay robot detects a target, it transmits a message containing the target trajectory to the pursuer through multihop communications. Upon receiving the message, the pursuer predicts the target position based on the trajectory information retrieved from the message and the communication delay. Then, it computes a path towards the estimated target position. The proposed algorithm is described in Section 4.

3.2 Energy Model

The robot team consumes energy for both communications and movement. We adopt the energy models used in [14]. The movement energy depends on the mass of the pursuer, the surface friction (air or ground), gravity and acceleration, and the distance traveled. We use the energy model for wheeled robot defined as $E_m = m \cdot d_m$, where the movement parameter, m, measured in Joule/meter, is a constant based on the aforementioned factors, and d_m is the distance traversed by the pursuer in *meter*.

On the other hand, the energy consumed to transmit ℓ bits of data over distance d_c measured in *meter* is defined as $E_{tx}(\ell, d_c) = \ell \cdot (d_c^{\alpha} \cdot e_{tx} + e_{cct})$, where e_{tx} is the energy required by the power amplifier of transceiver to transmit one bit data over the distance of one meter, and e_{cct} is the energy consumed in the electronic circuits of the transceiver to transmit or receive one bit, measured in the unit of Joule/bit. The energy consumption for receiving ℓ bits of data $E_{rc}(\ell) = \ell \cdot e_{cct}$ is independent of the distance between communicating nodes.

3.3 Mobile Target Tracking Problem Statement

In this problem, a team of heterogeneous robots have to track and capture a mobile target with the aid of target information dissemination in a wireless ad hoc network.

Definition. *The mobile target tracking problem using multiple relay robots. Given a set of heterogeneous robots S that consists of one single pursuer robot P and N_s-1 relay robots. All robots have to form a wireless ad hoc network. The relay robots have to detect a mobile target Q moving at constant speed v_q and disseminate the sensor information of target trajectory to the pursuer P. Using the information provided by relay robots, the pursuer P, moving at constant speed v_p, has to compute its path to track and capture Q.*

4 Algorithms

Our tracking algorithm is composed of two parts: information dissemination of target trajectory and path planning based on future target position. The previous is performed by multiple relay robots while the latter is run by a pursuer robot *P*.

4.1 Information Dissemination of Target Trajectory

Algorithm 1 is invoked and run in all wireless relay robots during the tracking event until the target *Q* is captured by the pursuer *P*. Each relay repeatedly senses if *Q* appears in its sensing range R_s. Let the relay robot detecting the target at time *t* be

s_q. When Q is detected, s_q gathers the sensor information and produces three target trajectory information: target position, $p_q(t)$, target moving direction $dir_q(t)$, and target speed, $v_q(t)$.

Algorithm 1 Target Info Dissemination: Wireless Relay Robot

```
Input: motion sensor information
Output: tAdv message
initialize tAdvList
repeat
    sensing Q
    if Q or its trajectory change is detected then
        generate and flood tAdv to P with initTtl
        update tAdvList
    end if
    if receive tAdv not in tAdvList then
        if P is my neighbor then
            forward tAdv to P
            update tAdvList
        else
            if ttl < maxTtl then
                ttl ⇐ ttl + incTtl
                flood tAdv with ttl to neighbors
                update tAdvList
            else
                discard tAdv
            end if
        end if
    end if
until Q is captured by P
```

To disseminate these information to the pursuer P via multihop communications (through multiple relay robots), we introduce a target information advertisement message called *tAdv*. Other than the source address of s_q and the destination address of P, a *tAdv* message consists of $p_q(t)$, $dir_q(t)$, $v_q(t)$ and t (the time Q is detected by s_q). Note that s_q also detects if Q changes its trajectory, e.g. its direction. Upon any change detection, s_q initiates a new *tAdv* to notify P about the latest target information. This is important to work with the prediction algorithm described in Section 4.2. For example, in case Q changes its direction before reaching the predicted position, P has to be informed to re-predict target position based on the latest *tAdv* contents.

Moreover, instead of blind flooding, *tAdv* is disseminated based on the expanding ring search algorithm [11] that works as follows. The source node searches successively larger areas until the destination node is found, by expanding the *search ring* centered at the source node. In our algorithm, the relay robot increments the value of time-to-live (ttl) in every *tAdv* it receives for retransmission until P is found. Using IEEE 802.11 in static network, the ideal one-hop delay [12] for 11 Mbps data rate is around 2 ms, and about 7 times longer delay is expected for 1 Mbps. A *tAdv list* is maintained to avoid redundant rebroadcast of *tAdv*.

A *tAdv* is generated by s_q with *ttl=initTtl*. The relay robot receiving this *tAdv* increments the *ttl* by *incTtl* if *P* is not its neighbor and *ttl* has not reached *maxTtl*. Then, it forwards *tAdv* with the new *ttl* to all of its neighbors. This is repeated hop-by-hop by the relay robots in the tracking region until *P* receives *tAdv* or *maxTtl* is reached. The value of *maxTtl* can be determined based on N_s upon the establishment of wireless ad hoc robot network [11].

4.2 Prediction of Mobile Target Position

P runs Algorithm 2 to compute its path to reach *Q*. At the beginning, *P* moves at v_p towards the target position, $p_q(t_s)$, where it is first detected when it enters the tracking area at t_s. We assume that *P* obtains the first target information by other means, for instance, through a centralized controller or base station, which also notifies the robot team of the tracking event. Whenever a *tAdv* is received, *P* invokes Algorithm 3 to estimate the future position of *Q* at $p_{q'}$ and thus, plan its path to reach $p_{q'}$. This is repeated until *P* captures *Q* (or *Q* is within R_s of *P*).

Algorithm 2 *tAdv*-based Path Planning: Pursuer Robot, *P*

Input: First detected target position, $p_q(t_s)$
Output: Path plan for *P*
$p_{q'} \Leftarrow p_q(t_s)$
repeat
    move to $p_{q'}$ at v_p
    if receive *tAdv* **then**
        run Algorithm 3
    end if
until *Q* is captured

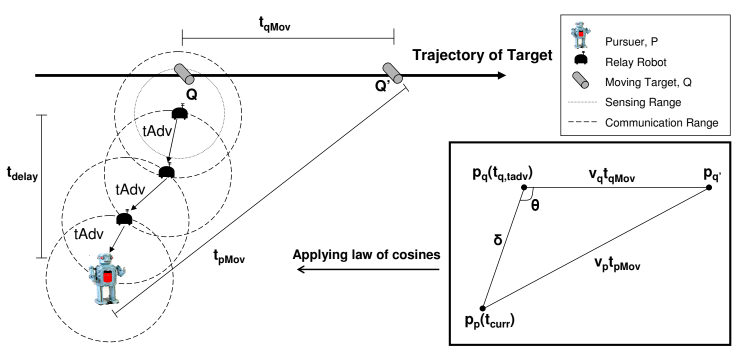

Fig. 2 Prediction of Mobile Target Location

Algorithm 3 is explained next with the aid of Figure 2. *P* first checks the target detection time, $t_{q,tadv}$ in *tAdv* it receives. Only the *tAdv* with $t_{q,tadv}$ larger than that

Algorithm 3 Prediction of Target Position, $p_{q'}$

Input: previous $p_{q'}$, latest Q detection time, $t_{q,latest}$
Output: predicted Q position, $p_{q'}$
retrieve $t_{q,tadv}$ in *tAdv*
if $t_{q,tadv} > t_{q,latest}$ **then**
  $t_{q,latest} \Leftarrow t_{q,tadv}$
  $t_{delay} \Leftarrow t_{curr} - t_{q,tadv}$
  get current position of P, $p_p(t_{curr})$
  retrieve $p_q(t_{q,tadv})$, v_q, and dir_q in *tAdv*
  compute $\|p_p(t_{curr}), p_q(t_{q,tadv})\|_2$
  compute t_{pMove} to find $p_q(t_{q,tadv} + t_{qMove})$ using *law of cosines*
  if $t_{pMove} > 0$ **then**
    $t_{qMove} \Leftarrow t_{pMove} + t_{delay}$
  else
    $t_{qMove} \Leftarrow t_{delay}$
  end if
  $p_q(t_{q,tadv} + t_{qMove}) = p_q(t_{q,tadv}) + v_q \cdot t_{qMove}$
  $p_{q'} = p_q(t_{q,tadv} + t_{qMove})$
else
  Discard *tAdv*
end if
return $p_{q'}$

of previously processed *tAdv* will be handled. Otherwise, the *tAdv* is discarded. Two time-related parameters are important to predict $p_{q'}$: the multihop transmission delay of *tAdv*, t_{delay} and the traveling duration of P from its current position to $p_{q'}$, t_{pMove}. t_{delay} is calculated based on $t_{q,tadv}$ and the time P receives *tAdv*, t_{curr}. On the other hand, P retrieves $p_q(t)$, $dir_q(t)$, $v_q(t)$ from *tAdv* to compute t_{pMove} by using the law of cosines as follows.

Let $\delta = \|p_p(t_{curr}), p_q(t_{q,tadv})\|_2$, $\beta = (2 \cdot t_{delay} \cdot \delta \cdot v_q \cdot \cos\theta) - (v_q \cdot t_{delay})^2 - \delta^2$, $\alpha = v_q^2 - v_p^2$ and $\gamma = (2 \cdot v_q^2 \cdot t_{delay}) - (\delta \cdot v_q \cdot \cos\theta)$, where θ is the angle between $p_p(t_{curr})$ and $p_q(t_{q,tadv})$ that is computed using arctan with these two positions. When $v_p = v_q$, $t_{pMove} = \beta/\gamma$. Otherwise, $t_{pMove} = -\gamma \pm \sqrt{\gamma^2 + 4 \cdot \alpha \cdot \beta}/(2 \cdot \alpha)$. Depending on the trajectory of Q, and the position of P and Q at t_s, $p_{q'}$ can not always be found based on t_{pMove}. For instance, when v_q is higher than v_p and P is behind Q moving with linear mobility at $t_{q,tadv}$. In this scenario, we base the computation of $p_{q'}$ only on t_{delay}. Planning a path of P towards this $p_{q'}$ enables P to receive more recent *tAdv* about Q's trajectory from the relay robots. Once $p_{q'}$ is computed, P continues with Algorithm 2 until Q is reached.

5 Performance Evaluation

As described in Section 4, the proposed tracking algorithm computes a path for P based on *tAdv* contents disseminated in wireless networks. Thus, we have to consider communications parameters involved, such as transmission delay of *tAdv*.

Parameter	Value(s)
Field size, (m^2)	800×600
Number of pursuer	1
Number of relay robot	90
Number of target	1
Max. sensing range, R_s (m)	100
Max. transmission range, R_c (m)	100
Path loss exponent, α	3
Speed of P, v_p (m/s)	2
Speed of Q, v_q (m/s)	1, 2, 3, 4

Parameter	Value(s)
Energy consumed by transceiver circuitry to transmit or receive a bit, e_{cct} (Joule)	10^{-7}
Energy consumed by transceiver amplifier to transmit one bit data over one meter, e_{tx} (Joule)	10^{-12}
Energy to receive a bit, e_{rc} (Joule)	10^{-7}
ttl: Initial, Incremental, Maximum	1, 1, 10
tAdv Data Size (bits)	48
Physical, MAC, Network Layer Header Size (bits)	192,272,160
Energy to move P over one meter, m (Joule)	3

Fig. 3 Simulation Parameters

We evaluate the performance of the algorithm using the network simulator OMNet++ [1] throughout our work. We compare the results with another approach, in which P always knows the current position of Q without the aid of other robots. In the rest of the paper, we call this approach *Centralized Control Approach (CCA)* and our proposed algorithm *Distributed Relay-robot Algorithm (DRA)*. Note that *CCA* assumes perfectly reliable and real time communications between P and its central controller. Two main criteria of our simulation study are the duration taken to capture Q and the total energy consumed by all robots (considering both mobility and communications costs).

5.1 Simulation Setup

We simulate 90 static relay robots cooperating with a pursuer robot on an area of 800m$\times$600m. Only one mobile target is simulated. All robots apply IEEE 802.11 as their MAC layer protocols and no special routing protocol is used. The flooding area of *tAdv* is controlled by expanding ring search as explained in Section 4.1. Figure 3 shows the simulation parameters. For simplification, we fix R_s and R_c at 100m, which is applicable to both IEEE 802.11 and some existing motion detectors. As mentioned in Section 3, m at 3 J/m is suitable for wheeled robots at 30-kg moving on flat surface. Depending on the application, a lower surface friction still allows a lighter weight robot to act as a pursuer.

At the beginning, all relay robots are distributed uniformly and randomly over the simulation area. Once deployed, a robot connects itself to its neighbors within its communication range. The mobility of Q is simulated using *Rectangle Mobility* to show the impact of its direction change to *tAdv* generation by relay robots(s). A simulation run is terminated whenever Q is detected within R_s of P.

5.2 Simulation Results

Using the simulation parameters in Figure 3, we perform 50 simulation runs for each v_q and compute total energy consumed by the robot team, and total time P

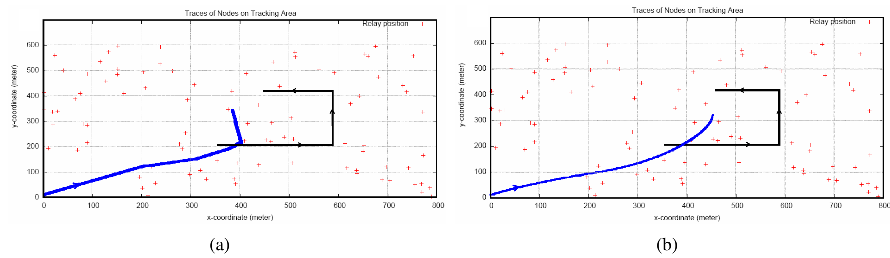

Fig. 4 Example Results using **(a)** *DRA* and **(b)** *CCA*: Pursuer (on blue track) and Target (on rectangular black track)

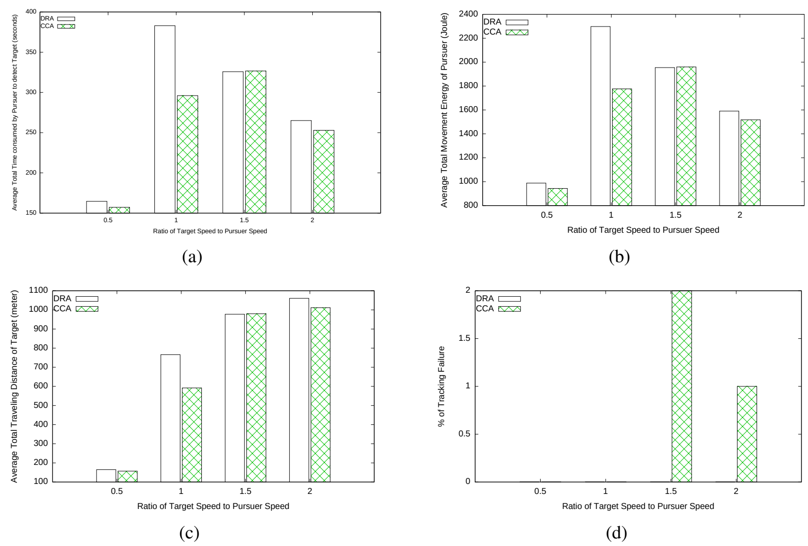

Fig. 5 Comparison of Tracking Strategies using Distributed DRA and Centralized CCA: **(a)** Average Total Time Consumption **(b)** Average Total Movement Cost of P **(c)** Average Total Traveling Distance of Q **(d)** % of Tracking Failure (pursuer failed to reach target within 1500s)

takes to capture Q using both *DRA* and *CCA*, and compare the two results. First, we show two example results of a simulation scenario in Figure 4, using *DRA* and *CCA* strategies in Figure 4(a) and 4(b) respectively. The initial robot-target distance is set at 406 m. Using *DRA*, total time and energy consumed are 291.40 s and 1782.54 J, in which communication cost constitutes only 1.27 J. Due to space limit, we will not display the resulting communication cost expended. In all simulations, wireless communication cost is negligible compared to movement cost.On the other hand, *CCA* consumes 286.28 s and 1717.802 J of movement energy. These results show that the proposed distributed *DRA* strategy needs only an additional 1.76% time and 3.63% energy compared to the centralized approach.Moreover, if *CCA* involves long-range communications (ignored in our simulation) with the centralized controller, e.g. a base station, additional communication cost and delay have to be considered.

Figure 5 illustrates the comparison results of the proposed distributed approach, *DRA* and the centralized *CCA* strategy. To simulate various types of target being tracked, we vary the target speed from 1 to 4 m/s, while pursuer speed is fixed at 2m/s. First, in Figure 5(a), we compute and compare the average amount of time consumed by each algorithm to capture the target, with both approaches using the same scenario setting. Initial position of *P* is the same for all simulations, and *Q* always starts moving away from the first position of *P*. Thus, it is reasonable to observe that the average tracking duration reduces over higher speed ratio since the target mobility is based on rectangle mobility that *Q* goes back to its origin towards completing one round. Overall, *CCA* performs 4.49% to 22.7% better than *DRA* considering that *DRA* does not provide instantaneous target information due to transmission delay of *tAdv*, while in *CCA*, *P* always knows the current position of *Q*, with an exception at ratio 1.5. We deduce that the performance gap between *DRA* and *CCA* decreases for scenarios with speed ratio ≥ 1. This is because the prediction algorithm of *DRA* described in Section 4.2 takes target speed as one of the factors for computing future target position. Figure 5(b) shows the average mobility cost of *P* to capture *Q*. Since the robot speed is constant and fixed over all simulation runs, based on the energy model described in Section 3.2, the result is identical to that in Figure 5(a).

Figure 5(c) shows the average total distance traveled by *Q*. This metric is important for tracking application that aims at minimizing the total distance traveled by a target, for instance, due to the larger loss an intruder may cause the further it manages to travel in the tracking area. The increasing total distance over higher target speed can be observed in both approaches, which show a maximum difference of only 48.64 meter, except in the case $v_p = v_q$. An appropriate approach can be chosen based on the application requirement.

As some tracking applications require the pursuer to capture the target within an allocated time, we set the maximum simulation duration at 1500 s. Figure 5(d) shows that *DRA* performs without any failure, while *CCA* has the failure rates of 2% and 1% at v_q=3 and 4 respectively. Note that all failed cases have been excluded from the previous statistics to fairly compare both approaches. *CCA* approach always directs the pursuer to current target position without considering the target trajectory. On the other hand, *DRA* predicts target position in advance based on the last detected target speed.

6 Conclusions and Future Works

In this paper, we study the feasibility of using wireless communications of a robotic team in a mobile target tracking application. We introduce a distributed approach, *DRA* by employing low power robots as wireless relays. By predicting target position based on the target trajectory information supplied by relay robots, the pursuer robot computes its path towards capturing the target. The results of the simulations performed using a network simulator show that it is feasible to use multihop communications to assist target tracking in a distributed means. Lastly, *DRA* is more energy-efficient because observing the much lower communication cost compared to mobility cost, it is beneficial to take the advantages of relay robots to provide useful target information for pursuer to compute an energy-efficient path.

As the communication cost is much lower than the movement cost, it is promising to develop an application-specific ad hoc routing protocol to maintain a multihop route between pursuer and relays closest to the target, which distance should decrease over time. We have also found several variations of mobile target tracking problem worthy of study using robots as relays. They include changing the number of pursuer and target, utilization of relay mobility, and various target trajectory impacted by using different mobility models, and lastly, the case in which a target acts as an evader.

Acknowledgements This research is partly supported by *DFG-Sonderforschungsbereich SPP 1183: Organic Computing. Smart Teams: Local, Distributed Strategies for Self-Organizing Robotic Exploration Teams.*

References

1. OMNet++, discrete event simulation environment. http://www.omnetpp.org
2. EPFL Education Robot. http://www.e-puck.org (2007)
3. IEEE 802.11 WIRELESS LOCAL AREA NETWORKS - The Working Group for WLAN Standards. http://www.ieee802.org/11/ (2007)
4. K-Team Corporation. http://www.k-team.com (2007)
5. Swarm-robotics.org. http://www.swarm-robotics.org/ (2007)
6. Akyildiz, I.F., Kasimoglu, I.H.: Wireless sensor and actor networks: Research challenges. Ad Hoc Networks (Elsevier) (2004)
7. Aslam, J., Butler, Z., Constantin, F., Crespi, V., Cybenko, G., Rus, D.: Tracking a moving object with a binary sensor network. In: Proceedings of ACM SenSys, pp. 150–161 (2003)
8. Bachrach, J., Taylor, C.: Localization in sensor networks (2005)
9. Demirbas, M., Arora, A., Gouda, M.: Pursuer-evader tracking in sensor networks. In: Sensor Network Operations, 5, pp. 587–608 (2007)
10. Farinelli, A., Iocchi, L., Nardi, D.: Multirobot systems: A classification focused on coordination. IEEE Trans. Syst., Man, Cybern. (2004)
11. Ilyas, M., Dorf, R.C. (eds.): The handbook of ad hoc wireless networks. CRC Press, Inc., Boca Raton, FL, USA (2003)
12. Li, F.Y., Hafslund, A., Hauge, M., Engelstad, P., Kure, O., Spilling, P.: Dilemma of using high datarate in ieee 802.11b based multihop ad hoc networks. In: International Conference on CIIT (2004)
13. Oh, S., Schenato, L., Chen, P., Sastry, S.: Tracking and coordination of multiple agents using sensor networks: System design, algorithms and experiments. Proceedings of the IEEE **95**(1), 234–254 (2007)
14. Ooi, C.C., Schindelhauer, C.: Minimal energy path planning for wireless robots. In: ROBOCOMM (2007)
15. Römer, K., Blum, P., Meier, L.: Time synchronization and calibration in wireless sensor networks. In: I. Stojmenovic (ed.) Handbook of Sensor Networks, pp. 199–237. John Wiley and Sons (2005)
16. Tipler, P.A.: Physics For Scientists and Engineers. Worth Publishers (1991)
17. Zhang, W., Cao, G.: Dctc: dynamic convoy tree-based collaboration for target tracking in sensor networks. IEEE Transactions on Wireless Communications **3**(5), 1689–1701 (2004)