

An Energy-Efficient Algorithm Integrated with Target Tracking and Mobile Sensor Navigation

Amrutha Kakkoth¹, Rijin I K²

¹ PG Scholar, Department of Computer Science & Engineering, Malabar Institute of Technology, Anjarakandy, Kannur (Dt), Kerala, India

² Faculty, Department of Computer Science & Engineering, Malabar Institute of Technology, Anjarakandy, Kannur (Dt), Kerala, India

Abstract: *The main function of wireless sensors is to capture the location of targets, may be for message passing or for locating enemies as in military, by monitoring the physical surrounding. If the target's maneuver is not known a priori, the task becomes difficult. In this paper, this problem is tackled by the measuring mobile target signal's time of arrival(TOA).The network contains a mobile sensor, moving target, number of anchor nodes and a sensor controller that acts as a repository for storing location information. A min-max approximation scheme is used to estimate tracking location along with semi-definite programming (SDP) relaxation. A cubic strategy is applied to the mobile sensor navigation. Furthermore, an energy-efficient localization algorithm, called as probabilistic trap coverage protocol is included within each node configurations so as to improve the performance gain, throughput and hence minimize the energy consumption. Thereby, it enhances navigation control by minimizing the number of active sensors and attempts to achieve energy balance. Here performance analysis is done theoretically.*

Keywords: Wireless sensor networks (WSNs), target tracking, sensor navigation, TOA, mobile sensor.

1. Introduction

A Wireless Sensor Network monitors physical environment conditions such as temperature, sound, pressure, etc. and pass data cooperatively to a main location (base station) through the network. The most networks used today are bi-directional which enables the control of sensor activities. The wireless sensor networks' development was motivated by several military applications as in surveillance, many industrial and consumer applications, to monitor and control various industrial operations, monitor machine status, etc.

The WSN is built of nodes – from a few to several hundreds or even thousands, where each node is connected to one or more sensors. Each of them has typically several parts: a radio transceiver with an internal antenna or connection to another antenna, a microcontroller, an electronic circuit for interfacing with the sensors and an energy source, usually a battery or an embedded form of energy harvesting. A sensor node might vary in size from that of a shoebox down to the size of a grain of dust, although functioning "motes" of genuine microscopic dimensions have yet to be created. The cost of sensor nodes is similarly variable, ranging from a few to hundreds of dollars, depending on the complexity of the individual sensor nodes. Size and cost constraints on sensor nodes result in corresponding constraints on resources such as energy, memory, computational speed and communications bandwidth. The topology of the WSNs can vary from a simple star network to an advanced multi-hop wireless mesh network. The propagation technique between the hops of the network can be routing or flooding.

Typically target tracking follows two steps. In the first step, it needs to find target positions from noisy sensor data measurements, either through estimation or prediction. Next, it needs to control and monitor mobile sensor tracker to follow the moving target. A general TOA measurement model that accounts for the measurement noise due to

multipath propagation and sensing error is used here. Based on the model, a min-max approximation approach is proposed to estimate the location for tracking that can be efficiently and effectively solved by means of semi-definite programming (SDP) relaxation. Also apply the cubic function for navigating the movements of mobile sensors. In addition, the random localization of the mobile sensor and the target is also estimated. The efficient exploitation of measurement information paved the way to implement a weighted tracking algorithm. The use of the TOA measurement model has various advantages like easy to acquire because sensors only needs to identify a special signal feature such as a known signal preamble to record its arrival time, TOA as a practical model because the transmission start time of the signal need not be known in prior. Hence TOA model enables direct estimation of the source location by processing the TOA measurement data.

The localized algorithm PTCP guarantees energy efficiency operating in two phases i.e., an initial phase followed by an action phase. In the initial phase, they communicate with the neighbors and decide to stay active or switch to sleep mode. Sensors make decision locally and asynchronously. They contend to sleep to save energy with the priority being imparted along with their ID.

2. Related Works

The challenge of target tracking and mobile sensor navigation arises when a mobile target does not follow a known predictable path. Target tracking is typically an ordered location estimation problem. The target is probably a signal emitter whose transmissions are captured by a number of distributed sensors for location estimation. There are various target localization method-related various measurement models such as received signal strength (RSS), time of arrival (TOA), time difference of arrival (TDOA), signal angle of arrival (AOA), and their combinations.

Volume 4 Issue 4, April 2015

www.ijsr.net

Location estimation and tracking for the mobile devices have attracted a significant amount of attention for a long time. The different schemes based on location estimation have been widely adopted based on the radio signals transmitted between the mobile device and the base stations. The location finders associated with the Kalman filtering techniques are utilized to both acquire location estimation and trajectory tracking for mobile nodes. But most of the existing techniques have become unreliable for location tracking due to the deficiency of signal sources. In 2009 Tseng et.al [1] proposed two predictive location tracking algorithms to alleviate this problem. The Predictive Location Tracking (PLT) scheme utilizes the predictive information obtained from the Kalman filter in order to provide the additional signal inputs for the location calculator. Moreover, the GPLT (Geometric-assisted) scheme incorporates the Geometric Dilution of Precision (GDOP) information into the algorithm design. The GPLT scheme enables accuracy in location estimation, especially with inadequate signal sources. The experimental results show that the GPLT algorithm can achieve better precision while comparing with other network-based location tracking schemes.

In 2006, Huang et.al [2] presents a source localization algorithm based on the source signal's time-of-arrival (TOA) at sensors that are not synchronized with one another or others. This algorithm calculates source locations using a window of TOA measurements which, in effect, formulates a sensor array which is virtual. The Gaussian noise model makes use of maximum likelihood estimates (MLE) for the source position and displacement are obtained. The Cramer-Rao lower bound evaluation is used to address performance issues and considers the virtual sensor array's geometric properties. This localization algorithm is combined with the extended Kalman filter (EKF) and the unscented Kalman filter to track the source trajectory resulting in good tracking performance.

Mobility management is a major challenge in mobile ad hoc networks (MANETs) due in part to the dynamically changing network structures. For mobile sensor networks used for route or set of points with minimum length surveillance applications, it is essential to use a mobile scheme that can empower nodes to make better decisions regarding their positions such that strategic tasks such as target tracking can benefit from node movement. Zou et.al [3] in his paper describes a distributed mobility management scheme for mobile sensor networks. This scheme considers node mobility as part of a distributed optimization problem which integrates mobility-enhanced improvement in the quality of target tracking data.

In 2004, Rao et.al [4], considers a mobile ad hoc sensor network. The cost of communication and mobility are the two factors used along with consideration of the possible scanning tasks of the nodes for sensor design. In this approach, for any single mobile node, its local energy cost information is available. A distributed simulated annealing framework is used to govern the motion of the nodes and shows that a global objective function comprising mobility and communication energy costs will be minimized. This paper concludes with a simulation study focusing on mobile

sensors with dual roles of scanning and relaying higher priority tracking traffic from tracking nodes.

3. Problem Identification

We have multiple sensor nodes that are kept at different distance so as to track the target and make possible the data transmission at a faster rate. When these sensors perform sensing a large amount of energy is required each time it performs the work. So the mere tracking and localization cannot make sense to a wireless sensor network to perform efficiently in a network, if we alone consider TOA measurements. Hence it is essential to minimize the battery power and total throughput through an efficient energy perspective. This problem can be tackled with an energy-efficient algorithm which is the goal of this work.

4. Proposed Model

In the existing system the problem of mobile sensor navigation and mobile target tracking based on Time Of Arrival (TOA) measurement model is considered. It emphasizes on TOA measurement model that accounts for the measurement noise due to multi-path propagation and sensing error. Here a min-max approximation scheme is proposed to estimate the location for tracking that can be efficiently and effectively solved by means of semi-definite programming (SDP) relaxation. We apply the cubic function for navigating the movements of mobile sensors. In addition, we also investigate the simultaneous localization of the mobile sensor and the target to improve the tracking accuracy. We present a weighted tracking algorithm in order to exploit the measurement information more efficiently. The numerical result shows that the proposed tracking approach works well.

In order to enhance the performance, extend the concept of trap coverage into a realistic model and analyze the detection probability of mobile targets with various moving speeds traveling along an arbitrary path in a RoI theoretically, based on which probabilistic trap coverage is defined. We formulate and study the problem of scheduling the activation of sensors energy-efficiently while providing desired probabilistic trap coverage in large-scale WSNs. We design an efficient localized protocol to solve the problem. The lower bound of lifetime acquired by the protocol is proven to be nearly half the optimum lifetime. Extensive simulations are conducted to validate the efficiency of the protocol.

4.1 Mobile Sensor Navigation Strategy

A navigator in this case aims to control the mobile sensor to get close to the moving target from any initial position. Since the target maneuvers are not known a priori to the controller, solving the problem requires a real-time strategy. At time instant T_j , the mobile sensor is with a velocity v_j and angle α_j to the positive horizontal axis, and the target locates at $y_j = [y_{j1} \ y_{j2}]^T$. The radial line that connects the mobile sensor and the target is denoted by r_j , with angle ϕ_j to the positive horizontal axis. In polar coordinates, the mobile sensor and target move according to the following kinematics:

$$z_{j1} = v_j \cos \alpha_j, z_{j2} = v_j \sin \alpha_j \quad (4)$$

$$y_{j1} = \mu_j \cos \beta_j, y_{j2} = \mu_j \sin \beta_j \quad (5)$$

respectively.

A cubic navigation strategy has been proposed, where

$$\alpha_j = \phi_j + K \phi_j^3 \quad (6)$$

Assuming $v_j > \mu_j$, it has been proven that under this cubic law, the corresponding $r_j < 0$, and the mobile sensor will reach the target successfully. Because of the simplicity of this navigation law, we will apply this strategy in our work. Alternatively, we may be interested in keeping the mobile sensor at a given distance away from the target for surveillance purpose without being discovered. In such applications, we need to set $r_j = 0$. We have

$$\mu_j \cos(\beta_j - \phi_j) = v_j \cos(K \phi_j^3) \quad (7)$$

this gives the mobile sensor speed as

$$v_j = \mu_j \cos(\beta_j - \phi_j) / \cos(K \phi_j^3) \quad (8)$$

4.2 Tracking Algorithm

The first step of tracking is to estimate positions of both the target and mobile sensor. Since the measurement in the form of TOA information collected at the data fusion center is the same for both the target and the mobile sensor, we, therefore, focus our discussion on how to estimate the location vector y_j of the target at a given time instant T_j . We can modify the TOA model by rewriting (1) into

$$t_{ji} - t_{j0} = 1/c \|x_i - y_j\| + 1/c \|x_i - y_j\| n_{ji} + \delta_j \quad (9)$$

Squaring both sides, we get

$$(t_{ji} - t_{j0})^2 - 1/c^2 \|x_i - y_j\|^2 = (1/c \|x_i - y_j\| n_{ji} + \delta_j)(1/c \|x_i - y_j\| (2 + n_{ji}) + \delta_j) \quad (10)$$

The right-hand side of (10) is a noise term that is independent for different indices i . If n_{ij} and δ_j are zero, then the right-hand side of (10) would be zero. Therefore, one way to estimate the optimum y_j without assuming any particular characteristics on ω_{ji} is to minimize the l_a norm of ω_{ji} . This approach makes no assumption on the noise distribution or on the noise dependency. It simply tries to minimize the peak error. Therefore, its performance is expected to be less sensitive to the noise distribution or correlation. Thus, we propose to adopt the min-max criterion for location estimation via

$$\hat{y}_j = \arg \min \max |(t_{ji} - t_{j0})^2 - 1/c^2 \|x_i - y_j\|^2| \quad (11)$$

The min-max formulation (11) is non-convex, but is quite amenable to semi-definite relaxations. (11) can be rewritten as

$$\hat{y}_j = \arg \min \max |\varphi(t_{js}, t_{ji}, t_{j0}, y_{js}, x_i, y_j)| \quad (12)$$

In order to make the whole formulation convex, we relax the two equalities to inequalities $Y_{js} \geq y_j^T y_j$ and $t_{js} = t_{j0} t_{j0}$. These inequalities can also be expressed in linear matrix inequalities.

In addition, based on the location estimate at time instant T_{j-1} , we can obtain an approximate location vector for the target at time instant T_j . Let $\Delta T_j = T_j - T_{j-1}$ and μ_{j-1} be the estimated velocity vector of the target at time instant T_{j-1} . This can be used as additional constraints for the target location estimation at time instant T_j . Considering in 2D, the location change vector Δy_j is restricted to a box, then the corresponding y_j will also be constrained to a box, i.e., $y_{j1} \leq y_{j1} \leq y_{j2}$, $y_{j2} \leq y_{j2} \leq y_{j3}$. (13)

We can apply the Reformulation-Linearization-Technique (RLT) in order to obtain some extra constraints. In fact, based on RLT, inequalities can be relaxed as following matrix form:

$$\begin{bmatrix} \|a_j\|^2 & -2a_j^T & 1 \\ \|b_j\|^2 & -2b_j^T & 1 \\ -a_j^T b_j & a_j^T + b_j^T & -1 \end{bmatrix} \begin{bmatrix} 1 \\ y_j \\ y_{js} \end{bmatrix} \geq 0.$$

Here " ≥ 0 " denotes that each element in the vector is non-negative. Combining the above constraints, we obtain the following SDP optimization formulation:

$$\begin{aligned} & \min_{y_j, y_{js}, t_{j0}, t_{js}} \theta_j \\ & \text{s.t. } -\theta_j < \psi(t_{js}, t_{ji}, t_{j0}, y_{js}, x_i, y_j) < \theta_j, \\ & \begin{bmatrix} \mathbf{I} & y_j \\ y_j^T & y_{js} \end{bmatrix} \geq 0, \quad \begin{bmatrix} 1 & t_{j0} \\ t_{j0} & t_{js} \end{bmatrix} \geq 0, \\ & \begin{bmatrix} \|a_j\|^2 & -2a_j^T & 1 \\ \|b_j\|^2 & -2b_j^T & 1 \\ -a_j^T b_j & a_j^T + b_j^T & -1 \end{bmatrix} \begin{bmatrix} 1 \\ y_j \\ y_{js} \end{bmatrix} \geq 0. \end{aligned}$$

4.3 Mobile Sensor Localization

Similar to estimating the location of the target, we can reformulate the mobile sensor localization problem into an SDP relaxation problem. More specifically, we can estimate the mobile sensor location z_j via the similar formulation based on the TOA measurements at the anchor nodes from the signal received from the mobile sensor. Define $z_{js} = z_j T_j z_j$ and $T_{js} = T_{j0} T_{j0}$. Similarly, based on the input velocity vector v_{j-1} of the mobile sensor from the controller at time instant T_{j-1} , we can approximate the location change of the mobile sensor as $\Delta z_j = z_j - z_{j-1} = \Delta T_j v_{j-1}$. By applying the similar relaxations, we obtain the following SDP formulation:

$$\begin{aligned} & \min_{z_j, z_{js}, \tau_{j0}, \tau_{js}} \theta_j \\ & \text{s.t. } -\theta_j < \psi(\tau_{js}, \tau_{ji}, \tau_{j0}, z_{js}, x_i, z_j) < \theta_j, \\ & \begin{bmatrix} \mathbf{I} & z_j \\ z_j^T & z_{js} \end{bmatrix} \geq 0, \quad \begin{bmatrix} 1 & \tau_{j0} \\ \tau_{j0} & \tau_{js} \end{bmatrix} \geq 0, \\ & \begin{bmatrix} \|d_j\|^2 & -2e_j^T & 1 \\ \|e_j\|^2 & -2e_j^T & 1 \\ -d_j^T e_j & d_j^T + e_j^T & -1 \end{bmatrix} \begin{bmatrix} 1 \\ z_j \\ z_{js} \end{bmatrix} \geq 0. \end{aligned}$$

4.3 Energy-efficient Protocol-PTCP

Here, a localized algorithm called the Probabilistic Trap Coverage Protocol (PTCP) is proposed to guarantee (D, ϵ) -trap coverage and maintain energy efficiency in the RoI. The operation time is divided into time slots. Each time slot is divided into two parts, i.e., an initial phase followed by an action phase. Every sensor wakes up at the beginning of each time slot. In the initial phase, they communicate with neighboring sensors and decide whether to stay in active mode or switch to sleep mode. Sensors decide on its mode

locally and asynchronously. After the initial phase, during the action phase, if sensors choose to be active, they perform sensing, communication, and other tasks; otherwise, they switch to sleep mode to save energy.

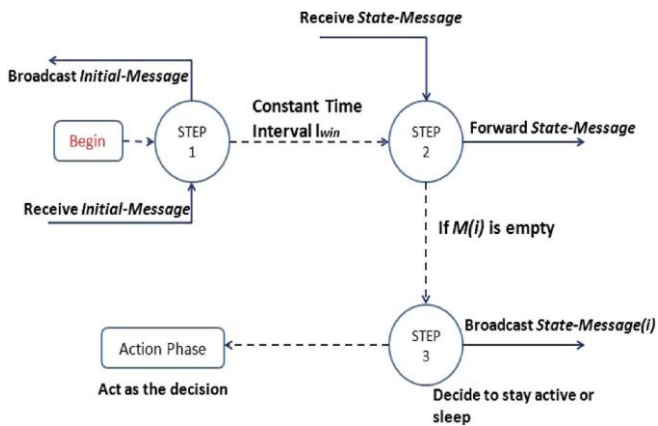


Figure 1: PTCP working

The PTCP runs during the initial phase of each slot. Sensors contend to sleep to save energy. If too many sensors choose to sleep, the requirement of probabilistic trap coverage will not be met. On the other hand, if too many sensors are active, it will be a waste of resources. Thus, a mechanism to coordinate sensors' decisions is desired. We, therefore, introduce *priority*. Every sensor has a unique ID. At the beginning of each time slot, each sensor is assigned a priority based on its residual energy and ID. Let pri denote the priority of sensor i . We define $pri = \{E_i, ID_i\}$, where E_i is the residual energy of sensor i , and ID_i is its ID. $pri > pri_j$ if 1) $E_i < E_j$ or 2) $E_i = E_j$, and $ID_i < ID_j$. If a sensor has a lower priority than its neighbors, it has to make a decision after the sensors with higher priority. Sensor i start to broadcast its priority and location information to neighbors, the sensors within its transmission range. The information is packed as Initial-Message(i). At the same time, i will receive the Initial-Messages from its neighboring nodes too. Multihop communication is entertained in PTCP so i may receive messages whose sender is out of its transmission range. Sensor i should check the distance between the sender and itself. If the distance is greater than twice its diameter, the information is abandoned since they are impossible to be in the same circular graph; otherwise, sensor i should record the received information and forward it to neighbors to perform multihop communication. A time window is set for sensor i to wait for all Initial-Messages. The length of time window l_{win} is determined by sensor deployment density and the range D . It needs to guarantee that all sensors within the range of $2D$ are recorded during the time window. Since information broadcast is usually very fast, the time window should not occupy much time. When the time window ends, sensor i start to determine whether to sleep. It will broadcast its decision packed as State-Message(i). There are two kinds of State-Message: 1) State-Message_{sleep}(i) and 2) State-Message_{active}(i), which denote the decision of sensor i , respectively. Here, we assume that C_i contains the recorded sensors from received Initial-Messages and that M_i contains sensors whose priority is higher than that of sensor i . If M_i is empty, i occupy the chance to make a decision since it is the sensor with the highest priority among the sensors within a

distance of $2D$. It will construct the circular graph and divide faces based on the information recorded in C_i . Note that sensor i itself is not contained in either C_i or M_i . Then, it employs Algorithm to determine whether the region within a distance of D is (D, ϵ) -trap covered. For the connectivity issue, sensor i also needs to guarantee that active sensors in a circular graph are connected if it chooses to sleep. Given the transmission range and the location information on all sensors in C_i , i can check whether all sensors in C_i are connected without i . If the region is covered and sensors in C_i are connected without i , sensor i will broadcast a State-Message_{sleep}(i) since it does not need to be active; otherwise, it broadcasts a State-Message_{active}(i) and chooses to stay active. If the region is still not (D, ϵ) -trap covered after sensor i chooses to stay active, it indicates that there are not enough sensors to provide (D, ϵ) -trap coverage and the lifetime of network terminates. If M_i is not empty, i have to wait for the State-Messages from other sensors in M_i . If i receive a new State-Message_{sleep} whose sender is in set C_i , it will record the information, forward the message to its neighbors, and remove the sender from set C_i and, M_i if in it; otherwise, it will abandon the message since the message is useless. Sensor i can only make a decision when M_i is empty. Then, sensor i construct the circular graph based on C_i . All sensors in C_i are viewed as active when sensor i make a decision. Similarly, sensor i chooses to stay active in the PTCP if probabilistic trap coverage is not guaranteed or sensors in C_i will be disconnected without i . After decision making, sensor i will act as its choice, either in active mode or in sleep mode. In summary, the PTCP puts sensors into sleep mode in the order of priority/residual energy. In this way, sensors with less residual energy have the higher priority to switch to sleep mode and leave sensors with more residual energy to perform sensing tasks for energy balance, which can prolong the network's lifetime.

Algorithm PTCP

1. Define $pri = \{E_i, ID_i\}$ as the priority of sensor i . $pri > pri_j$ if $E_i < E_j$ or $(E_i == E_j \text{ and } ID_i < ID_j)$;
2. Set timeout threshold τ .
3. At the beginning of each time slot, i turn into active mode;
4. Broadcast Initial-Message (i) to neighbors;
5. While Time window not end do
6. if Receive new Initial-Message(j) and $\text{distance}(i, j) < 2D$ then
7. Record Initial-Message(j);
8. Broadcast Initial-Message(j) to neighbors;
9. end if
10. end while
11. Define set C_i contains the recorded sensors from received Initial-Message;
12. Define set M_i contains sensors whose priority is greater than pri ;
13. while $C_i \neq \emptyset$
14. % Update C_i and M_i when receiving State-Message from sensors in C_i
15. if Receive new State-Message(j) then
16. if $\text{distance}(i, j) < 2D$
17. Record State-Message(j);
18. Broadcast State-Message(j) to neighbors;
19. if j decides to sleep then
20. Remove j from C_i ;

```
21. end if
22. if  $j \in M_i$  then
23. Remove  $j$  from  $M_i$ ;
24. end if
25. end if
26. end if
27. % Clear  $M_i$  if timeout
28. if  $M_i = \emptyset$  and exceed timeout threshold  $t_r$  then
29.  $C_i = C_i - M_i$ ;
30. Let  $M_i = \emptyset$ ;
31. end if
32. % Start to decide if  $M_i$  is empty
33. if  $M_i == \emptyset$  then
34. Assume set  $F_i$  as the faces who are covered by sensor  $i$ ;
```

Networks,” IEEE Trans. Mobile Computing, vol. 3, no. 3, pp. 225-231, Mar. 2004.

[5] Enyang Xu, Zhi Ding and Soura Dasgupta, “Target Tracking and Mobile Sensor Navigation in Wireless Sensor Networks”, IEEE Trans. Mobile Computing, vol. 12, no. 01, Jan.2013

5. Advantages of Proposed method

- The proposed method efficiently tracks the target using TOA even in higher noise rates.
- TCP minimizes amount of active sensors and hence energy balance is attained.
- Performance gain, throughput, and better navigation control is ensured through the theoretical study when compared with the state-of-art solutions.
- The number of stable nodes can be found with the energy constraint.

6. Conclusion

With unknown target and mobile sensor locations, we need to estimate the locations of the target and the mobile sensors first. Based on a more general TOA measurement model, convex optimization algorithms through SDP relaxation are developed for localization. Here is provided a sequential algorithm and a joint weighted localization algorithm before controlling the mobile sensor movement to follow the target. For the navigation of mobile sensors, the cubic law is applied. With these it's supposed to provide an efficient approach towards tracking. Along with this an energy efficient algorithm called PTCP is being used. The lower bound of lifetime acquired by the protocol is proven to be nearly half the optimum lifetime. This makes the environment much efficient towards tracking approach.

References

- [1] P.H. Tseng, K.T. Feng, Y.C. Lin, and C.L. Chen, “Wireless Location Tracking Algorithms for Environments with Insufficient Signal Sources,” IEEE Trans. Mobile Computing, vol. 8, no. 12, pp. 1676-1689, Dec. 2009.
- [2] T. Li, A. Ekpenyong, and Y.F. Huang, “Source Localization and Tracking Using Distributed Asynchronous Sensors,” IEEE Trans. Signal Processing, vol. 54, no. 10, pp. 3991-4003, Oct. 2006.
- [3] Y. Zou and K. Chakrabarty, “Distributed Mobility Management for Target Tracking in Mobile Sensor Networks,” IEEE Trans. Mobile Computing, vol. 6, no. 8, pp. 872-887, Aug. 2007.
- [4] R. Rao and G. Kesidis, “Purposeful Mobility for Relaying and Surveillance in Mobile Ad Hoc Sensor