Adaptive Energy Optimization for Object Tracking in Wireless Sensor Network

Juan Feng, Baowang Lian and Hongwei Zhao
School of Electronic Information Northwestern Polytechnical University
Xi'an, 710072, P.R China
[e-mail: fengjuankh@hotmail.com]
*Corresponding author: Juan Feng

Received May 8, 2014; revised January 27, 2014; accepted February 27, 2015;
published April 30, 2015

Abstract

Energy efficiency is critical for Wireless Sensor Networks (WSNs) since sensor nodes usually have very limited energy supply from battery. Sleep scheduling and nodes cooperation are two of the most efficient methods to achieve energy conservation in WSNs. In this paper, we propose an adaptive energy optimization approach for target tracking applications, called Energy-Efficient Node Coordination (EENC), which is based on the grid structure. EENC provides an unambiguous calculation and analysis for optimal the nodes cooperation theoretically. In EENC, the sleep schedule of sensor nodes is locally synchronized and globally unsynchronized. Locally in each grid, the sleep schedule of all nodes is synchronized by the grid head, while globally the sleep schedule of each grid is independent and is determined by the proposed scheme. For dynamic sleep scheduling in tracking state we propose a multi-level coordination algorithm to find an optimal nodes cooperation of the network to maximize the energy conservation while preserving the tracking performance. Experimental results show that EENC can achieve energy saving of at least 38.2% compared to state-of-the-art approaches.

Keywords: Target tracking, energy saving, sleep schedule, WSNs

This work was supported by National Natural Science Foundation of China (61301094) and NPU Foundation for Fundamental Research (NPU－FFR－ JCY20130135).

http://dx.doi.org/10.3837/tiis.2015.04.005 **ISSN : 1976-7277**

1. Introduction

A typical Wireless Sensor Network (WSN) consists of a large amount of small, low-cost, and wirelessly connected sensor nodes deployed in an unattended natural environment. The sensor nodes are usually battery-powered and therefore have a limited lifetime. It is infeasible to replenish energy via replacing their battery after deployment. Thus, reducing energy consumption to prolong the network lifetime is the most critical issue in the design of WSNs.

Sleep/wake scheduling of sensor nodes is one common approach to conserving energy after the network is deployed. The main idea of sleep scheduling is to keep a minimal number of sensor nodes active, in order to provide a necessary coverage of the sensing field, and to put the other nodes in sleep mode. When a sensor node is put into sleep mode, it is completely shut down except for one timer, which must keep on in order to wake up the node after the designated sleep time and consumes extremely low power. A sleep scheduling scheme makes the decision of: (1) which nodes can sleep, and (2) how long they can sleep.

Since WSNs are highly application-specific, an efficient sleep scheduling scheme should take the special features of an application into account. In this paper, we target the target tracking application, which is one of the most important WSN applications. In a target-tracking WSN, the events of interest occur usually randomly and therefore the time of their occurrences is hardly predictable. After entering the sensing field, the roaming path of the target is also random. Hence, for a sensor node it is difficult to know when a target will enter its sensing area. If it always keeps active, the long time in idle mode is the major source of energy waste. On the other hand, if the sensor nodes are blindly turned off to save energy, it is very likely to miss the target. Therefore, it is highly desired to find a sleep scheduling scheme that adapts to the dynamic features of target tracking applications [1, 2, 3].

Some previous approaches assign each sensor node a fixed sleep/wake schedule [4, 5, 6]. This simple sleeping scheme cannot adapt to the dynamic features of target tracking applications. Instead, some coordinate approaches suit better by dynamically adjusting the sleep/wake schedule of each node using information from neighboring nodes [7, 8, 9]. However, by using this scheme two other problems arise: (1) the frequent communications among neighboring nodes introduce a large overhead on traffic load, and (2) unsynchronized sleeping of the nodes increases the end-to-end delay of multi-hop data transmission. Hence, none of the previous approaches has achieved all goals of adaptability to moving targets, high energy conservation and high tracking performance. Moreover, in [22] we propose a hierarchically power management approach, which divides the network into hierarchical layers. Just one layer is in tracking state in one time instant and the others are in sleep state to save energy. However, the central sink controls the type of hierarchical layer and sleep strategy in each hierarchical layer, in which the information of moving target needs to be transmitted to the sink in time in order to make a decision. In [15], we propose a distributed sleep strategy and delay reduction approach in which each GH decides its sleep interval by the information from its neighbours instead of the sink. Nevertheless, it does not consider the relation of network size and nodes cooperation. This paper improves previous two approaches by optimal the nodes cooperation according to network size for maximum energy efficiency.

In this paper, we propose a novel energy optimization scheme, called Energy-Efficient Node Coordination (EENC), for target tracking WSNs to achieve the aforementioned goals.

EENC is based on a grid-based network structure, where the entire sensing field is divided into virtual grids. The grid structure is well studied in the literature [1, 10, 11, 12, 13, 14].

In EENC, the sleep scheduling is locally synchronized and globally unsynchronized. Locally in each grid, all nodes have the same sleep time and synchronized by the head. Globally, all grids can have independent sleep schedules. We define two network states: surveillance state when no event of interest occurs and tracking state when a target enters the sensing field. In surveillance state we propose to assign a static sleep time to each grid according to the shortest distance to the border, which in tracking state we propose a multi-level coordination scheme for finding optimal nodes cooperation around the target to maximize the energy conservation. *Summarizing:* the contribution of this paper is to propose the EENC scheme. The advantages of EENC are summarized as follows:

- High energy conservation: novel sleep scheduling schemes are proposed for both surveillance state and tracking state to maximize the sleep time of nodes. In our coordinated scheme the collaboration is established among grids. The amount of control packets sent among the grid heads is much less that among the nodes, leading to less traffic overhead.
- Good adaptability to moving targets: the sleep scheduling scheme in tracking state can find an optimal node cooperation around the target. Although the target moves dynamically and randomly, the nodes around it can be woken up in time to avoid target missing.
- High tracking performance: In each grid the grid head is always active, and therefore, data transmission paths with good connectivity can be established among the grid heads. Good connectivity and low target miss rate lead to very high tracking performance.

The rest of this paper is organized as follows: Section 2 gives an overview of related work. Then, Section 3 presents the proposed approach in detail. The experimental results are shown in Section 4. Finally, we conclude the paper in Section 5.

2. Related Work

A simple scheme for sleep scheduling of sensor nodes called Fully Synchronized Pattern is proposed in [4]. In this scheme, all nodes periodically wake up for a fixed time. However, it requires global synchronization of the nodes, which is very difficult to implement. In [5], a self-scheduling mechanism is presented, which allows nodes to independently decide whether to be active or to sleep. Each node keeps a timer to record how long no event has been detected and goes to sleep after this timer times out. An asynchronous scheme is proposed in [6]. It allows each node to independently wake up by guaranteeing that the active periods of neighbors are always overlapped within a specified cycle. Although this scheme does not need a tight synchronization among nodes, it usually leads to long data transmission delay. Moreover, in this scheme the length of sleep and active periods is fixed and cannot adapt to variations in target tracking applications.

An on-demand scheme, as introduced in [16], is ideal for target tracking WSNs. In this scheme, a node can be woken up anytime when it needs to execute the task. The implementation of such a scheme typically requires two different channels: a data channel for normal data communication and a wakeup channel for waking up the sleeping nodes [16]. However, the wakeup channel works only for a short distance between nodes, typically a few meters. This strongly limits the applicability of this scheme. Moreover, the wakeup radio is costly and is not shipped with commonly used sensor platforms.

Zamora et al. propose an adaptive coordinated sleep scheduling scheme [8, 9], where each node periodically broadcasts a message to its neighbors to indicate whether there is a target being detected. In this way, nodes are aware of the status of their neighbours. A node can enter sleep mode if its neighbors do not detect any target. However, due to the high density of sensor nodes in WSNs, the frequent broadcasts will dramatically increase the traffic load and lead to congestions and high energy consumption. A similar scheme is also proposed in [7].

In [17] a prediction algorithm is developed to predict the target movement. Thus, only the nodes around the target's next position need to wake up. However, such prediction algorithms have high accuracy for regularly moving objects but not for highly randomly moving objects such as wild animals. Inaccurate predication will result in low tracking performance.

3. Energy-Efficient Node Coordination Approach

In EENC, the network operations of a target tracking WSN have two states: surveillance state and tracking state. In surveillance state, since there is no target to track, intuitively, the sensor nodes are allowed to work in the low-power mode and have longer sleep time. Nevertheless, they need to keep a certain level of vigilance in order to detect the intrusion of a target in time. In tracking state, the network is responsible for tracking the path of the moving target and reporting the location information to the sink. In this state, the nodes around the target must wake up to respond to the target's movement, while the other nodes can still sleep to conserve energy. However, they must turn into a higher level of vigilance, because they must assure that they are awake when the target enters their sensing areas.

3.1 Network Model

We consider a static WSN, which consists of one sink and n randomly and evenly distributed sensor nodes in a two dimensional $M\times M$ sensing field. The sink is assumed to have an infinite power supply, and it gathers the sensing data from the sensor nodes. We assume each node is aware of its location after deployment (e.g., by running a light-weight localization algorithm such as the one proposed in [18]). The whole sensing field is divided into $K\times K$ grids of the same size. **Fig. 1** illustrates the grid structure. The size of each grid is $\alpha\times\alpha$, and therefore, $K = M/\alpha$. In each grid, one node with the most residual energy is selected as the grid head (GH). The role of GH rotates when the energy of the current GH falls under some level. In the definition of virtual grid, any pair of nodes in two adjacent grids can directly communicate with each other [10, 11]. Thus, the following condition must be fulfilled:

$$(2\alpha)^2+(2\alpha)^2\le R_{trans}{}^2 \quad \text{or} \quad \alpha\le\frac{R_{trans}}{2\sqrt{2}} \tag{1}$$

where R_{trans} is the transmission range of a sensor node.

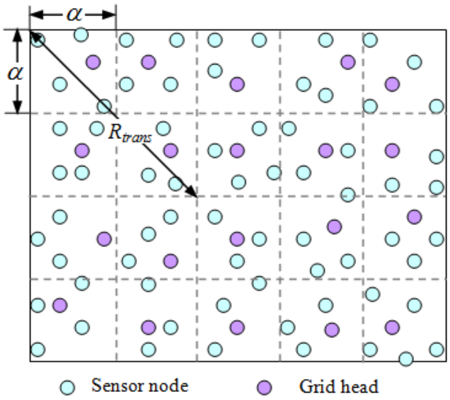


Fig. 1. Illustration a grid structured sensor network

All sensor nodes have active and sleep modes. Active mode is further divided into working and idle states. In sleep mode, a sensor node cannot be woken up externally and has to set an internal timer to determine the time of waking up.

3.2 Sleep Optimization in Surveillance Stage

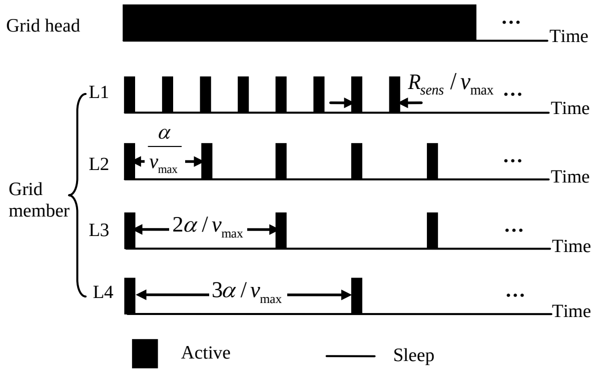
In target tracking sensor network, the users are only interested in the occurrence of a certain event. These interesting events don't happen frequently. When no target occurs, it would be a significant waste of energy if all the nodes always keep active. Sleep optimization is dynamically getting a part of nodes sleep as long as possible to reduce energy consumption and guarantee tracking performance. In this paper, the sensor nodes have active and sleep states. In detail, the active state includes work and idle states. In work state, nodes sense a target, send or receive data, while in idle state, nodes do not detect any target and just listening from others. In sleep mode, the node could not execute any function, and it could not be woken up externally. Although sleep state has less power consumption, it incurs a longer latency and a higher energy to awaken. Therefore, it is clear that the node should get into a sleep state only when its idle period will be long enough. Thereby, estimating the idle time of nodes is crucial in sleep optimization.

Initially, the idle time of a node is estimated by the distance between the node and the network border, because the target has to pass the network border before it gets into the sensing area. As illustrated in **Fig. 2**, the grids in a square-formed field are divided into layers. From the border to the center the layer number increases and the estimated idle time of the node also increases. Thus, the estimated idle time of the nodes in the border is $t_{L1} = R_{sens} / v_{\max}$, where $v_{\max}$ is the maximum speed of the target and R_{sens} is the sensing range of the node.The idle time of the nodes in the other layers is decided by the shortest distance from their grid border to the border of the sensing field. For example, for a grid in L2 the idle time is estimated as $\alpha / v_{\max}$. In the same way, the estimated idle time of the grids in layer L_i is $(i-1)\times\alpha / v_{\max}$ in surveillance state.

Since the idle time of the nodes have uncertainty and are closely related to the events occurred in the sensing area, the estimated idle time is not always accuracy. Considering the situation at the most of the time, the movement trajectory of the target is continuous in the local area. As a result, the two contiguous idle periods of the node are correlated and continuity. We can utilize this characteristic to make the estimated idle time more precise. To deal with the emergencies and correct a deviation of the estimated idle time, the estimated idle time of the node n_j can be expressed by the follow equation.

$$t_j(p+1)=e^{-\theta\bar{t}_j(p)/t_j(p)}\bar{t}_j(p)+(1-e^{-\theta\bar{t}_j(p)/t_j(p)})t_j(p) \tag{2}$$

Where p ={0,1,2,…}. $\bar{t}_j(p)$ and $t_j(p)$ are the actual and estimated idle time in the current time, respectively. $t_j(p+1)$ is the estimated idle time for the next time instant. θ $(0.1\le\theta\le10)$ is a constant used to adjust the exponential convergence rate. $e^{-\theta\bar{t}_j(p)/t_j(p)}$ denotes the correct coefficient of the estimated deviation. When the actual idle time is equal to the estimated idle time in the current time i.e. $\bar{t}_j(p)=t_j(p)$, the next estimated value is $t_j(p+1)=\bar{t}_j(p)=t_j(p)$. When $\bar{t}_j(p)>>t_j(p)$, then $e^{-\theta\bar{t}_j(p)/t_j(p)}\approx0$ and $t_j(p+1)=t_j(p)$. That is, the next estimated idle time is equal to the current estimated idle time when the actual idle time is larger than the estimated one at the current time. This way avoids overestimating the next idle time when the current idle time is much longer than the normal situation. When $\bar{t}_j(p)<<t_j(p)$, then $e^{-\theta\bar{t}_j(p)/t_j(p)}\approx1$ and $t_j(p+1)=\bar{t}_j(p)$. Here, the next idle time is estimated as the current actual idle time value because the two contiguous idle periods of a node are correlated and continuity. Moreover, it also avoids overestimating the next idle time in the case of $\bar{t}_j(p)<<t_j(p)$. After acquiring the corrected idle time value, each node adjusts its sleep/active state in terms of the corrected idle time value.



Sleep schedule in the surveillance stage

Fig. 2. The sensor nodes near to the network border keep more alert and have less sleep time. By contraries, the nodes near to the network center have more sleep time. The sleep time of nodes is calculated by how far from the grid to network border.

3.3 Dynamic Sleep Optimization in Tracking State

When a target is detected in a grid, the node that catches the target first sends the message "*target_detected*" to its GH, The Grid Members (GMs) in the same grid will overhear this message and then immediately turn into tracking state to collaboratively measure the target's location. The GH then forwards the message hop-by-hop to the sink. After receiving the message, the sink immediately broadcasts it to all GHs in the sensing field to inform them to turn all grids into tracking state. When the target moves out of the sensing field, the sink will broadcast another message "*target_ left*" to all GHs to switch all grids back to surveillance state. In the last subsection, we have discussed the sleep scheduling of each node in surveillance state. In this subsection, we deal with the dynamic sleep scheduling of each node in tracking state.

In our dynamic sleep optimization algorithm, the grids are divided into three parts: 1) **tracking grid**, which includes the target. All nodes in a tracking grid should remain active to perform the tracking operation. The target's information collected by the grid members is aggregated at the GH, which then transmits the aggregated data through a multi-hop path to the sink. 2) **neighboring coordinated grid**, whose GH can receive the detected information from the GH of tracking grid; 3) **border grid**, which is at the network border and should keep alert to prepare for detecting another targets.

The detected information broadcasted by the GH who detects a target or gets the detected information form its GMs.The range of neighboring coordinated grid decided the sleep time of the interior GMs. One hop neighbors coordination means when a GH makes PM decisions, it just considers the detected information from its immediate neighbor GHs (shown as **Fig. 3** (a)). If a GH does not receive the detected information, it can obtain the distance from the target to itself is larger than the side length of one grid α so that the sleep interval of the GMs in these grids is the same as the GMs in the layer L2 (shown as **Fig. 3** (b)). If the detected information is transmitted to the *H-th* hop GHs, the GHs who do not receive the detected information, it can obtain the distance from the target to itself is larger than $H\alpha$. Therefore, the GMs in these grids has the longer sleep interval (shown as **Fig. 3** (c,d)).Therefore, the more hops the detected information is transmitted, the more sleep time GMs can get, however, the more transmission energy costs. Thus, it is a tradeoff between the sleep time and the transmission energy cost. A GH calculates the sleep time for its GMs in three different cases.

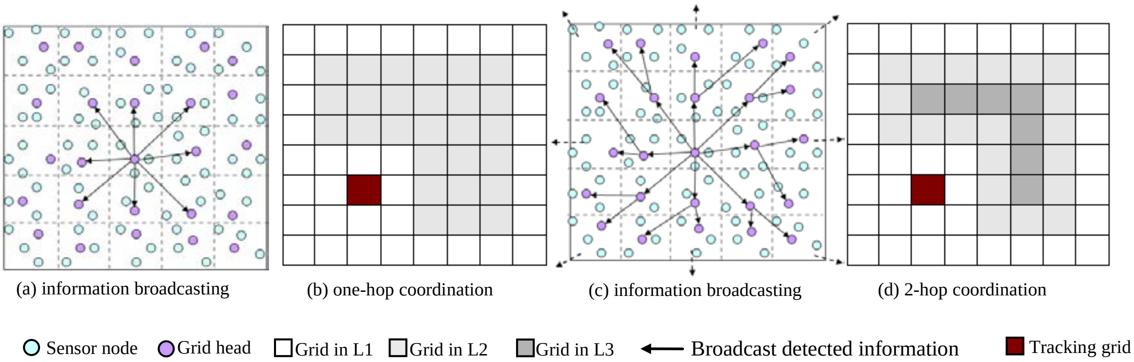

(a) information broadcasting (b) one-hop coordination (c) information broadcasting (d) 2-hop coordination

Fig. 3. Dynamic sleep optimization in different level neighbors coordination

The first case is that the GH neither detects any target nor receives any target information from others during a time period. The maximum sleep time of GMs is decided by the following equation,

$$t_{trac} = \frac{H\alpha}{v_{\max}} \tag{3}$$

where H is the number of the coordinated hops.

The second case is that the target is detected by the GH or its GMs. The GMs keeps active until the target moves out of sensing area of its grid, and it decides the sleep time of its GMs

$t_{trac} = 0$. Then the GH broadcasts a detected information message to its neighbour GHs so that they can prepare to track the target at the next moment.

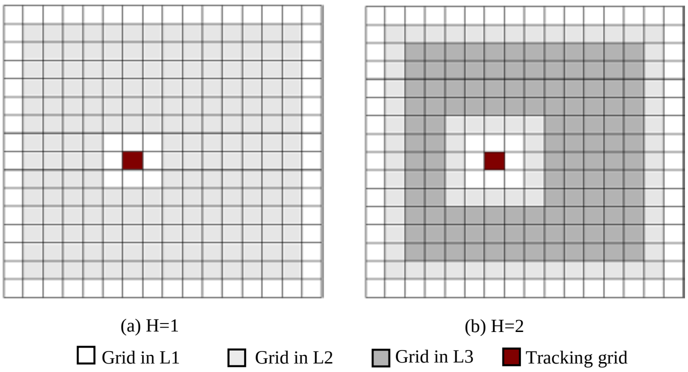
The third case is that the GH receives the target information from its neighbors. Thus the GH can calculate the maximum sleep time for its GMs by following equation,

$$t_{trac} = \frac{(h-1)\alpha}{v_{\max}} \tag{4}$$

where h is the h-*th* hop the detected information transmitted from the GH of the tracking grid. Therefore, the sleep time of the GMs in tracking stage can be set as min $\{ t_{trac}, t_{surv} \}$, in which t_{surv} is the sleep time of the GMs in the surveillance stage. Each GH decides if the sleep time of its GMs needs to be changed based on the information it received. When there is a change on the sleep time value of GMs, the GH informs its GMs about the updated sleep time in the next active instant. The GMs receive a new sleep time from its GH and adaptively adjust the current sleep mode.

In this way, sensor nodes will get the target movement information from the GHs in neighboring area and accurately estimate the sleep state and time interval. In addition, transmitting the target information only among GHs instead of that among all the sensor nodes can reduce the communication cost and information redundancy.

When we increase the neighboring coordinated grids located multi-hop away from the tracking grid, the communication costs are also increased. This requires a balance between the transmitting energy consumption and the energy saving of GMs who can sleep more. For example, in **Fig. 3** *K*=8, when the neighboring coordinated grids are only immediate neighbor of the current grid, the worst case is that 36 grids are in layer L1 and 27 grids are in layer L2. When the coordinated multi-hop number *H*=2, the better case is that 36 grids are in layer L1, 20 grids are in layer L2 and 7 grids are in layer L3 which have more sleep time than that in layer L1 and L2, however, when the target moves to the network center, the worst case is the same as the situation when *H*=1. Moreover, the communication cost when *H*=2 is larger than that when *H*=1. Therefore, when *K*=8, the increase of the coordination level from one to two hops does not make any grid sleep longer but consumes more energy for broadcasting messages. Whereas, when the network is enlarged to 16×16, as shown in **Fig. 4**, the increase of the coordination level from one to two allows many grids to sleep for one level longer. (There are 119 grids have longer sleep time when *H*=2 than that when *H*=1.) Thus, increasing *H* results in optimal energy saving when *K*=16. In what follows, we conduct an analysis.


(a) H=1

(b) H=2

Fig. 4. Dynamic sleep optimization in H-hop neighbors coordination

On the one hand, when *H* increases, the energy saving is brought by more GMs can have longer sleep time. Since the sleep time of GMs is set as min $\{t_{trac}, t_{surv}\}$, if the value of *H* should be satisfied Inequation (5), the number of grids whose members can have longer sleep time in the deeper layer is increased and the multi-hop coordination will make sense.

$$(K-2H)^2-(2H+1)^2>0 \tag{5}$$

therefore, $H<(K-1)/4$. For a period of time *T*, the energy saving by the longer GMs sleep from *H-1* hops to *H* hops coordination is,

$$E_{save}=\left(\frac{T}{(H-1)\alpha/v_{\max}+t_{active}}-\frac{T}{H\alpha/v_{\max}+t_{active}}\right)t_{active}P_{active}\times\left[(K-2H)^2-(2H+1)^2\right]N_C \tag{6}$$

where N_C is the average number of the GMs in each grid, and t_{active} is the active intervals of GMs.

On the other hand, when *H* increases, communication cost is also increased by GHs broadcasting detected information. For example, when *H* is increased from 1 to 2, the communication cost is increased by 8 GHs broadcasting detected information. For a period of time *T*, the communication cost is increased by GHs broadcasting from *H-1* hops to *H* hops coordination is,

$$E_{\cos t}=\frac{Tv_{\max}}{\alpha}\times 2[E_{br}+(8(H-1)-8)E_{br}] \tag{7}$$

where E_{br} is the energy consumption of detected information broadcasting, and $\alpha/v_{\max}$ is the frequency of detected information broadcast. When the saving energy and the cost are satisfied Inequation (8), the multi-hop coordination PM will make sense.

$$E_{save}-E_{\cos t}>0 \tag{8}$$

We should select appropriate coordinated multi hop number *H* to obtain the optimal energy saving according to the network parameters.

3.4 Adaptive Transmission for Sensed Data

In the tracking grid, the GMs need to report the sensed data to their GH. Then, the GH sends the aggregated data toward the sink. According to the energy consumption model, to reduce the energy consumption of data transmitting we should shorten the distance between nodes or decrease the packet size. In this paper, the tracking chain is built for transmitting the sensed data in the tracking grid to shorten the tramsmitting distance between two nodes. As **Fig. 5** shows, each GM just transmits the sensed data to its nearest neighbor node instead of its GH. Consequently, the data packets can be aggregated step by step when they are transmitted through the tracking chain to achieve energy saving.

Algorithm 1 The tracking chain construction

```
input: GH and $q$ GMs
calculate the distance between GH and each GM
sort the distance in ascending order {|GH, $GM_1$|,…,|GH, $GM_q$|}
$C_{left}$ =$GM_1$, $C_{right}$ =$GM_2$, $CHAIN \leftarrow$ { $GM_1$-GH-$GM_2$}
$p$=3
while($p<q$)
    for each $(GM_j \notin CHAIN, j \in [3,q])$
        calculate the distance $d_{1j}$=| $C_{left}$, $GM_j$|, $d_{1j}$=| $C_{right}$, $GM_j$|
        do $Min\_d_1 = \min\{|C_{left}, GM_j|\}, Min\_d_2 = \min\{|C_{right}, GM_j|\} j \in [3,q]$
            if ( $Min\_d_1 < Min\_d_2$ )
            add ($CHAIN$, $C_{left} \leftarrow GM_j$) && $C_{left}$ =$GM_j$
            else ($CHAIN$, $C_{right} \leftarrow GM_j$) && $C_{right}$ =$GM_j$
    end for
    $p$=$p$+1
end while
output $CHAIN$
```

Algorithm 1 builds a tracking chain in the tracking grid. First, the GH chooses its two nearest GMs as its neighbors to initiate the process of chain constructing. Then, these two GMs respectively choose one nearest neirghbor to add the chain each time and the process is repeated until all the nodes are in the chain. Finally, the tracking chain which takes the GH as the center of data collection is built.

After the tracking chain forming, the GMs send the data to their nearest neighbour though the chain. Each GM aggregates its own data with the received data from its upstream GM to generate a single packet of the same length. At last, GH sends the aggregated data to the sink.

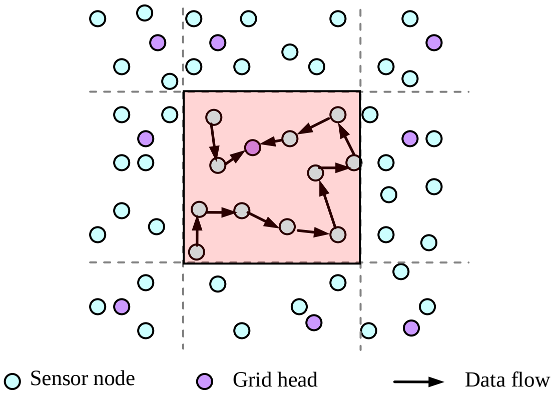


Fig. 5. Sensed data transmitting in the tracking grid

4. Theoretical Analysis

In order to evaluate the performance of the proposed approach, we analytically compare the energy consumption of EENC with two existing approaches, which are the network with fixed sleep schedule [6] (Fixed SS: all the nodes sleep periodically for a fixed time) and the network with coordinated sleep schedule [7, 8, 9] (Coordinated SS: sleep schedule is dynamically adjusted according to the target information from neighbors).

1) In surveillance state, there is no target. The sleep interval of a node in the different sleep levels in EENC is calculated as follows,

$$\begin{cases} t_{L1}(EENC) = R_{sens} / v_{\max} \\ t_{L2}(EENC) = \alpha / v_{\max} \\ \cdots \\ t_{Li}(EENC) = (i-1)\alpha / v_{\max} \end{cases} \tag{9}$$

From Equation (9), we can see that the nodes have longer sleep intervals when they are located nearer to the network center. Nevertheless, the sleep intervals of all the nodes in Fixed SS and Coordinated SS approaches are the same as $R_{sens} / v_{\max}$. Consequently, the interior nodes in EENS have more time to stay in sleep state. This way can not only save energy but also balance the energy consumption on border and interior nodes because the interior nodes take more transmitting tasks.

2) In tracking state, there are targets in the sensing area. The energy consumption of the network mainly includes two parts as Equation (10) shows.

$$E_c(tracking) = \sum_{N_i \in tracking} [P_{active} t_{active}(N_i) + E_{trans}(N_i)] + \sum_{N_j \notin tracking} P_{active} t_{active}(N_j) \tag{10}$$

Where $\sum_{N_i \in tracking} [P_{active} t_{active}(N_i) + E_{trans}(N_i)]$ denotes the energy consumption of the nodes, called tracking nodes, whoes sensing range reach the target and which execute the tracking mission. P_{active} is the node power consumption in active state and $t_{active}(N_i)$ is the time periods of node N_i in active state. $E_{trans}(N_i)$ is the sensed data transmitting energy consumption of the node N_i. Since the tracking nodes are in active state for tracking in all the three approaches, the energy consumption of $\sum_{N_i \in tracking} P_{active} t_{active}(N_i)$ has the same value. However, in the case of EENC, the tracking nodes just transmit the sensed data to their nearest neighbors in stead of transmitting the data to their GHs in the other two approaches. The transmitting distance in EENC is shortend so that $\sum_{N_i \in tracking} E_{trans}(N_i)$ in EENC is the smallest among the three approaches. In addition, $\sum_{N_j \notin tracking} P_{active} t_{active}(N_j)$ denotes the energy consumption of the nodes whoes sensing range do not reach the target. In Fixed SS, the value of $\sum_{N_j \notin tracking} P_{active} t_{active}(N_j)$ is the same as that in the surveillance state. In Coordinated SS, just the nodes neighbored the tracking nodes can receive the target information to adjust their sleep interval, whereas in EENC the adaptive multi hop coordination can be utilized to obtain the optimal energy saving. In a word, EENC has the optimal energy saving in both the surveillance and tracking state.

5. Experimental Results and Analysis

In the experiments, we evaluated the performance of the proposed EENC scheme for different network conditions and compared it to the state-of-the-art approaches by means of simulations using Castalia [19] based on OMNet++4.1 [20]. We chose Castalia [19] as our network simulator because it provides realistic channel models, including radio models like CC2420 and CC1000 and MAC models like IEEE 802.15.4 [21]. The approaches we compared with in the experiment include (1) target tracking without sleep scheduling at all (No SS: nodes are always on), (2) Fixed SS, and (3) Coordinated SS.

4.1 Experiment Environment

Table 1 lists the initial simulation parameters. We adopted the MAC model of IEEE 802.15.4 and chose Texas Instruments CC1000 as the radio. The power consumption of CC1000 in transmission, reception, idle and sleep mode are 44.4 mW, 22.2 mW, 22.2 mW and 0.0006 mW, respectively. The power consumption of CPU and memory is assumed to be constant and 6 mW in total. For each simulation setup, we run at least 5 times with different random node distributions, while the sink is always in the middle of the sensing field. Each result is averaged over these runs.

Table 1. Simulation Parameters

Parameters	**values**
Field size	400 m ×400 m
Number of nodes	1600
Grid size	50 m ×50 m
Number of grids	64
Sensing range	15 m
Transmission range	150 m
Data reporting frequency	every 0.5 s
Data packet size	128 bytes
Control message size	8 bytes
Velocity of the target	10 m/s
Location of sink	(200, 200)
R_{sense}^{FG}	25 m
Active intervals of GMs	0.2 s
Simulation time	600 s

4.2 Simulation Results

1) Performance evaluation in surveillance state: **Fig. 6** shows the average energy consumption of the sensor nodes in surveillance state versus simulation time. It can be seen that EENC saves energy by 43.9%, 65.9% and 90.8% compared to Coordinated SS, Fixed SS and No SS, respectively, after 1000s simulation time. In the experiment, we have calculated the average sleep time and time awake of all sensor nodes. The percentage of sleep time and time awake is illustrated in **Fig. 7**. As shown, using EENC the sleep time of a node is accounted for 91.3% on average, while Coordinated SS and Fixed SS allow a node to sleep 84.7% and 73.1% on average, respectively. Since no sleep scheduling is applied in No SS, all nodes are active all the time. This explains why EENC is most energy efficient compared to the other three schemes.

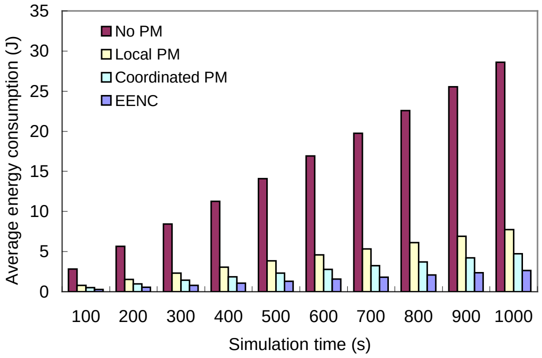


Fig. 6. Average energy consumption in surveillance stage

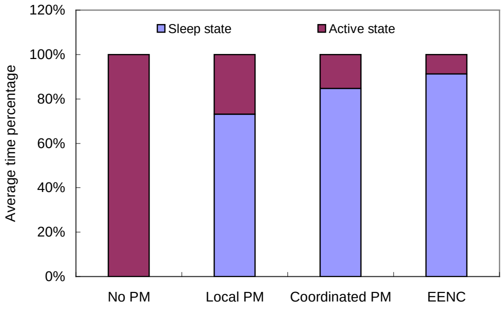


Fig. 7. Sleep/awake time distribution in surveillance state

2) Performance evaluation in tracking state: Next, we evaluated the performance of EENC in tracking state. We assume the target enters the field at a random location of the border and moves continuously and randomly in the sensing field with the maximum speed of 10m/s. **Fig. 8** shows the average energy consumption of the sensor nodes in tracking state versus simulation time. If we compare **Fig. 8** and **Fig. 6** we will find that the energy consumption of No SS and Fixed SS in tracking state has only a slight increase compared with that in surveillance state. This is because the two approaches have the same static sleep scheduling in both states. The slight increase in energy consumption is caused by data transmissions. Whereas, EENC and Coordinated SS have a significant increase in energy consumption in tracking state, because they apply dynamic sleep scheduling schemes which adjust the sleep schedule of each node according to the changing position of the target. From surveillance state to tracking state, using EENC the average energy consumption of one node increases from 2.64J to 5.13J after 1000s simulation time. Compared to Coordinated SS, Fixed SS and No SS, EENC still achieves energy saving of 38.2%, 47.0% and 82.7%, respectively.

Similar as for surveillance state, we calculated the average sleep time and time awake. Here, we further refine the sleep state to two states: *Sleep and missing target* and *Sleep without missing target*, and also refine the state awake to *Active and detecting target* and *Active and idle listening*, in order to evaluate both energy efficiency and tracking performance. By *Active and detecting target* the node is active when the target enters its sensing area, while by *Active and idle listening* there is no event of interest during the awake time of the node. By *Sleep without missing target* the node sleeps safely, while *Active and missing target* is not desirable. A good sleep scheduling scheme should have short time in *Active and idle listening* and long time in *Sleep without missing target*. With respect to these states, a comparison of the selected sleep scheduling schemes is illustrated in **Fig. 9**. Using EENC the time of missing target accounts for only 0.7%. Still, EENC has much better tracking performance than Fixed SS (4.3% of missing target) and Coordinated SS (2.1% of missing target). Since using No SS all nodes are active there is no target missing. As shown, EENC has the shortest time in *Active and idle listening* and the longest time in *Sleep without missing target*.

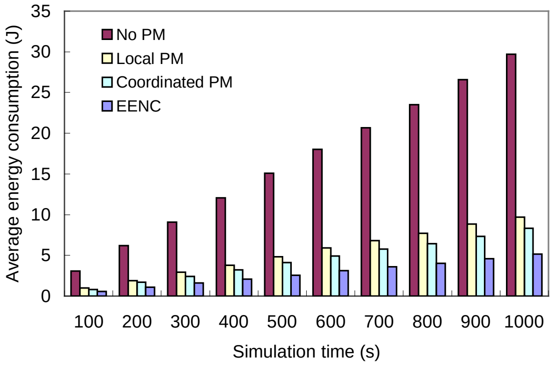


Fig. 8. Average energy consumption in tracking stage

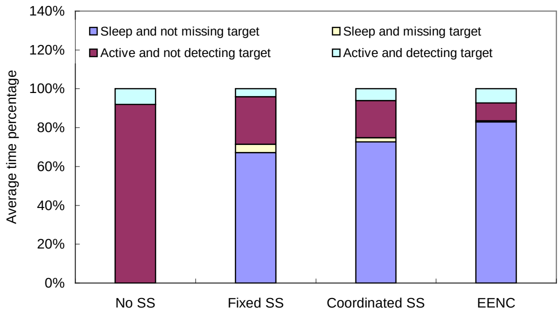


Fig. 9. Sleep/awake time distribution in tracking state

In addition to energy efficiency, tracking performance is another critical criterion for target tracking WSNs. A comparison of tracking successful ratio is illustrated in **Fig. 10**. According to many existing localization algorithms such as [18], it is reasonable to suppose that the sink can localize the target successfully if it receives a certain number of sensed data from the tracking nodes. Hence, the tracking successful ratio in this paper is defined as the ratio of time when more than five pieces of data about the current target's location are successfully received. The tracking successful ratio in No SS is highest because all the nodes are always active. EENC and Coordinated SS have the higher tracking successful ratio than Fixed SS because the tracking nodes can be woken in prior. The results validate that EENC achieves the highest energy efficiency but without sacrificing tracking performance.

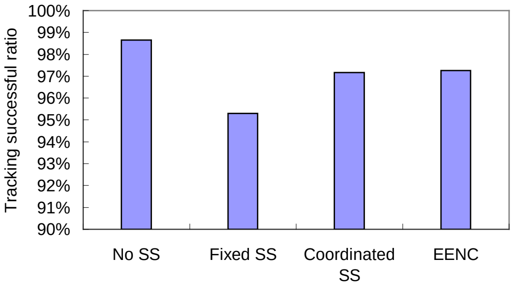


Fig. 10. Tracking successful ratio

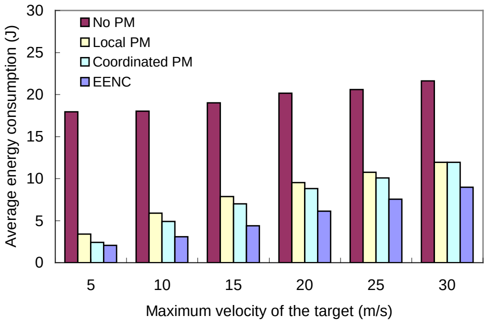


Fig. 11. Average energy consumption vs the velocity of the target

4) Impact of target's velocity: Finally, we studied the impact of a target's velocity on the energy efficiency and the tracking performance of EENC, by varying the target's velocity from 5m/s to 30m/s. **Fig. 11** and **Fig. 13** illustrate the impact of the target's velocity on the energy consumption and the data transmission delay, respectively. As shown in **Fig. 11**, with the increase of the target's velocity, the energy consumption of No SS increases very slightly because the increase of the target's velocity only leads to more data reporting but does not affect the sleep schedule. Whereas, in EENC, Fixed SS and Coordinated SS, the sleep time of a node is affected by the target's velocity and higher velocity leads to shorter sleep time. Therefore, in the figure we can see a significant increase of energy consumption of the three approaches with the increase of the target's velocity. Note that in Fixed SS the sleep schedule is fixed for a specific scenario with the known v_{max}. However, when the scenario changes and

v_{max} changes, the sleep time will also change, given by $t = r_{sense}/ v_{max}$.

Fig. 12 shows the tracking successful ratios changing with the target's velocity. Increasing the target's velocity from 5 m/s to 30 m/s, the tracking successful ratio in No SS is almost not affected, while the successful ratio in Fixed SS is reduced because some nodes may miss the target. Although the successful ratio in coordinated SS and EENC also decreases with the increasing target's velocity, it is always above 96%, showing enough tracking performance.

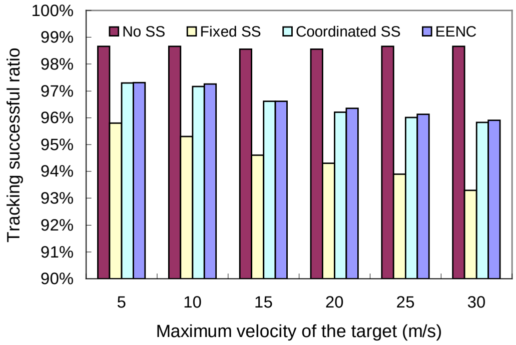


Fig. 12. Tracking successful ratio vs. the target's velocity

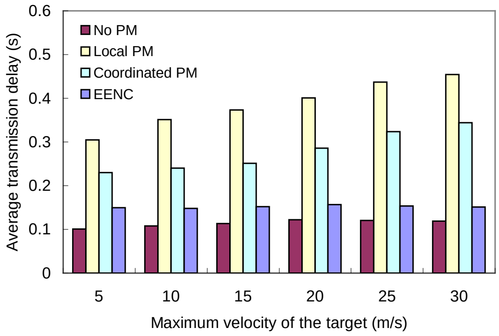


Fig. 13. Average transmission delay vs the velocity of the target

Using EENC, the data is transmitted hop by hop through grid heads, which are always active. Therefore, EENC have no connectivity problem. As shown in **Fig. 13** the average data transmission delay of EENC is almost constant and is immune to the target's velocity. No SS has surely no connectivity problem either. Whereas, using Fixed SS and Coordinated SS the data transmission is often delayed by the sleeping relay nodes. Therefore, they have much larger delays and the delay increases with respect to the target's velocity.

5. Conclusion

This paper proposed a novel Energy-Efficient Node Coordination (EENC) for tracking target in WSNs. It can reduce the energy consumption and increase the network lifetime. EENC is aimed to find an optimal nodes cooperation to conserve the energy without degrading the tracking performance. This is realized by assigning each grid a static sleep schedule according to its shortest distance to the border in surveillance state and by assigning each grid a dynamic sleep schedule in tracking state by means of the proposed multi-level coordination algorithm. Our experiments proved that EENC outperformed the state-of-the-art approaches and saves energy by at least 38.2%.

References

[1] Y. Xu, J. Heidemann, and D. Estrin, "Geography informed energy conservation for ad hoc routing," in *Proc. of the ACM 7th annual international conference on Mobile computing and networking(MobiCom'01)*, pp. 70–84, 2001.

[2] M. Cardei and J. Wu, "Handbook of Sensor Networks – Chapter: Coverage in Wireless Sensor Networks," CRC Press.

[3] L. Wang and Y. Xiao, "A survey of energy-efficient scheduling mechanisms in sensor networks," *Mob. Netw. Appl*., vol. 11, no. 5, pp. 723–740, 2006. Article (CrossRef Link)

[4] C. Chiasserini and R. Rao, "Improving energy saving in wireless systems by using dynamic power management," *IEEE Transactions on Wireless Communications*, vol. 2, pp. 1090–1100, 2003.

Article (CrossRef Link)

[5] T. V. Dam and K. Langendoen, "An adaptive energy-efficient mac protocol for wireless sensor networks," in *Proc. of the ACM Conference on Embedded Networked Sensor Systems (Sensys'03)*, pp. 171–180, 2003.

[6] Y. F. Wong, L. H. Ngoh, W. C. Wong, and W. K. G. Seah, "A combinatorics-based wakeup scheme for target tracking in wireless sensor networks," *in Proc. of the IEEE Wireless Communications and Networking Conference (WCNC'07)*, pp. 3569–3574, 2007.

[7] C. Gui and P. Mohapatra, "Power conservation and quality of surveillance in target tracking sensor networks," in *Proc. of the 10th annual international conference on Mobile computing and networking, ser. MobiCom '04*, pp. 129–143, 2004.

[8] N. Zamora, J. Kao, and R. Marculescu, "Distributed power management techniques for wireless network video systems," in *Proc. of the Design Automation and Test in Europe (DATE'07)*, pp. 1–6, 2007.

[9] N. H. Zamora and R. Marculescu, "Coordinated distributed power management with video sensor networks: Analysis, simulation, and prototyping," in *Proc. of the ACM/IEEE International Conference on Distributed Smart Cameras (ICDSC'07)*, pp. 4–11, 2007.

[10] G. Takahara, K. Xu, and H. Hassanein, "Efficient coverage planning for grid-based wireless sensor networks," in *Proc. of the IEEE International Conference on Communications (ICC'07)*, pp. 3522–3526, 2007.

[11] M. A.-T. Fadi, S. H. Hossam, and A. I. Mohamed, "Quantifying connectivity of grid-based wireless sensor networks under practical errors," in *Proc. of the IEEE 35th Conference on Local Computer Networks (LCN'10)*, pp. 220–223, 2010.

[12] N. Meghanathan, "Grid block energy based data gathering algorithm for lower energy, delay and longer lifetime in wireless sensor networks," in *Proc. of the ISCA International Conference on Sensor Networks and Applications (ICSNA'09)*, pp. 1–6, 2009.

[13] C. Luo, F. Wu, J. Sun, and C. W. Chen, "Compressive data gathering for large-scale wireless sensor networks," in *Proc. of the ACM 15th annual international conference on Mobile computing and networking(MobiCom'09)*, pp. 145–156, 2009.

[14] H. Jeon, K. Park, D.-J. Hwang, and H. Choo, "Sink-oriented dynamic location service protocol for mobile sinks with an energy efficient gridbased approach," *Sensors*, vol. 9, pp. 1433–1453, 2009. Article (CrossRef Link)

[15] F. Juan, B. Lian and Z. Hongwei. "Smart Power Management and Delay Reduction for Target Tracking in Wireless Sensor Networks." *Journal of Electrical and Computer Engineering Volume (3)*, 2014. Article (CrossRef Link)

[16] J. Ansari, D. Pankin, and P. M¨ah¨onen, "Radio-triggered wake-ups with addressing capabilities for extremely low power sensor networkapplications," in *Proc. of the 5th European conference on Wireless Sensor Networks (EWSN'08)*, pp. 1–5, 2008.

[17] S. S., A.-H. M., and B. A., "A predictive energy-efficient technique to support object tracking sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 60, pp. 656–663, 2011. Article (CrossRef Link)

[18] N. Bulusu, J. Heidemann, and D. Estrin, "Gps-less low-cost outdoor localization for very small devices," *Personal Communications, IEEE*, vol. 7, no. 5, pp. 28 –34, 2000. Article (CrossRef Link)

[19] "National ICT australia - castalia," http://castalia.npc.nicta.com.au/.

[20] "OMNeT++ Network simulator," http://www.omnetpp.org/.

[21] "IEEE 802.15.4," Article (CrossRef Link).

[22] F. Juan, B. Lian and Z. Hongwei. "Hierarchically Coordinated Power Management for Target Tracking in Wireless Sensor Networks." *Int J Adv Robot Syst*, 2013. Article (CrossRef Link)

Juan Feng received the B.S. degree in communication engineering from Northwestern Polytechnical University, Xi'an, China, in 2005 and the M.S. degree in information system engineering from Northwestern Polytechnical University, in 2009. She is currently pursuing the Ph.D. degree in communication and information system at Northwestern Polytechnical University. Her research interests are wireless communications, adaptive signal processing and wireless sensor network.

E-mail:fengjuankh@hotmail.com

Baowang Lian was born in 1962. He received the B.S., M.S. and the Ph.D. degree in communication engineering from Northwestern Polytechnical University, Xi'an, China, in 1983, 1986 and 2007, respectively. Since 1999, he has been a Professor with the School of Electronic Information in Northwestern Polytechnical University. He has more than 50 research articles, and more 2 inventions. His research interests include navigation positioning systems, wireless communications, and embedded systems.

E-mail: bwlian@nwpu.edu.cn

Hongwei Zhao was born in He Nan province, China, in 1980. He received the B.S. degree in communication engineering from Zheng Zhou University, Zheng Zhou, China, in 2001. Then he received the M.S., and Ph.D. degrees in communication engineering from Northwestern Polytechnical University, Xi'an, China in 2006 and 2012, respectively, and he is currently a lecturer and post-doctor in Northwestern Polytechnical University.

His research interests include satellite navigation and positioning, adaptive array processing, and embedded systems.

E-mail: hongvi_zhao@126.com.