Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2016, Article ID 4134735, 8 pages
http://dx.doi.org/10.1155/2016/4134735

Research Article

Efficient Sleep Scheduling Algorithm for Target Tracking in Double-Storage Energy Harvesting Sensor Networks

Hongbin Chen,[1] Qian Zeng,[2] and Feng Zhao[1]

[1]*Key Laboratory of Cognitive Radio and Information Processing, Guilin University of Electronic Technology, Guilin 541004, China*
[2]*School of Computer & Communication Engineering, University of Science and Technology Beijing, Beijing 100083, China*

Correspondence should be addressed to Hongbin Chen; chbscut@guet.edu.cn

Received 28 March 2016; Accepted 24 April 2016

Academic Editor: Zhipeng Cai

Target tracking is a typical application in wireless sensor networks. Both energy efficiency and tracking performance are important issues that need to be considered. They are a pair of contradictions most of the time. Saving energy often sacrifices tracking performance, while enhancing tracking performance needs to consume more energy. In this paper, an efficient sleep scheduling algorithm is put forward to tackle the above problem in energy harvesting sensor networks. At first, we modify the probability-based prediction and sleep scheduling (PPSS) algorithm to track the target and further use another sleep scheduling algorithm we proposed to wake tracking nodes when the target is likely to be missed (i.e., it is unsuccessful to wake next-moment tracking nodes). Secondly, a double-storage energy harvesting architecture is employed to increase residual energy of sensor nodes and to extend network lifetime. Simulation results reveal that the proposed sleep scheduling algorithm can improve tracking performance and prolong network lifetime compared with the PPSS algorithm and the proposed algorithm without energy harvesting.

1. Introduction

Earlier generations of wireless sensor networks used non-rechargeable batteries to supply electrical power, which has limited energy storage. When the power of the battery in a sensor node is run out, that sensor node cannot work anymore. Therefore, researchers have made great effort to reduce energy consumption and to prolong network lifetime in various application fields [1–7]. Recently, energy harvesting technologies have been proposed [8] and various problems in energy harvesting sensor networks have been studied [9–22].

Energy harvesting technologies can be applied in sensor networks to enhance performance or extend network lifetime. Many excellent works have been done toward this goal. For example, in [9] several clustering algorithms have been presented while energy harvesting nodes serve as dedicated relay nodes to prolong network lifetime. In [10], a fundamental guideline was provided to design efficient medium access control protocols to extend network lifetime. In [11], performance-sensitive processing and communication were combined with efficient energy management techniques to prolong network lifetime. The premise of it is to ensure energy neutral operation. To achieve this goal, it is required to balance energy consumption and harvesting capability of each node [12]. Following this principle, in [13] an energy flow control problem was formulated to keep the balance between energy supplies and demands. When the nodes need to harvest energy, they should turn into the sleep state, because the nodes in the sleep state consume the least energy. As the ambient energy is not sufficient for all the operation process, sleep scheduling is one of the effective methods for energy management [14]. However, sleep scheduling will cause working time delay, while designing an adaptive algorithm can vary the subsequent sleep interval to maximize the asymptotic event detection [15]. Low latency sleep and wake-up scheduling is challenging due to dynamic duty cycling. In [16], two dynamic duty-cycle scheduling schemes were proposed to reduce sleep latency. The bit-reversal permutation sequence can also be used to achieve this aim [17]. In [18], a new framework was developed to model the adaptive duty cycling in energy harvesting sensor networks as a Markov Decision Process. To achieve optimal performance, an optimal scheduling algorithm was proposed in [19] which considered the residual energy to achieve close-to-optimal utility. In [20], the sampling rate was adaptively determined to maximize the overall network performance. In [21], a new energy harvesting model was proposed for a network simulator that implements supercapacitor energy storage which is easily extensible. Both rechargeable battery and supercapacitor have their merits and disadvantages. NiMH battery and Li-based battery have high energy density and low self-discharge rate, but they have limited recharge cycle. On the contrary, supercapacitor has low energy density and high self-discharge rate, but it has theoretically infinite recharge cycle. Integrating these two devices into a battery will take the advantages of both and avoid their shortcomings [22].

Target tracking is a typical application in wireless sensor networks. Many target tracking algorithms have been proposed, such as Kalman filter [23], Bayesian tracking [24], Bernoulli filter [25], probability-based prediction, and sleep scheduling (PPSS) [26]. There are many requirements in the tracking procedure that need to be considered and energy efficiency is one of the key aspects [27]. Different methods were utilized to enhance energy efficiency of target tracking. In [28], the nodes in a spatial region surrounding a target were employed to achieve better tracking accuracy and energy efficiency. In the tracking area, one needs to consider the tradeoff between energy consuming and tracking accuracy, such as the expensive in-node computations and the imprecision tolerance when selecting subsequent tracking principals [29]. In [30], an algorithm was proposed where sensors were dynamically selected and allocated to track a target in order to conserve network energy and provide tracking coverage guarantee. As mentioned previously, sleep scheduling is an efficient way to save energy. In the tracking procedure, sleep scheduling means that only the sensors which are near to the target keep awake [31], while other sensors which are far away from the target stay in the sleep mode. In [32], a sleep strategy was proposed that each node autonomously determines the sleeping time. It also can make one subset of nodes active while others enter the sleep state to conserve energy and ensure that the target is tracked [33]. Besides, in [26] a PPSS algorithm was proposed to proactively wake tracking nodes.

Nowadays, few sleep scheduling algorithms for target tracking in sensor networks considered energy harvesting. In energy harvesting sensor networks, scheduling the node's sleep and wake-up time should consider both the target tracking procedure and the energy harvesting status. It is necessary to make the sleep scheduling algorithm adaptive to the energy harvesting situation. In this paper, first we combine the PPSS algorithm with another sleep scheduling algorithm we proposed [34] and make some improvement to it. Secondly, we utilize a double-storage energy harvesting architecture [22] to simplify the sleep scheduling process. For each sensor node, when it turns into sleep and the solar energy is available, it immediately harvests solar energy. If the target is about to be missed (i.e., it is unsuccessful to wake next-moment tracking nodes), the network will wake some nodes which are nearer to the target and have more residual energy to keep tracking the target [34]. Compared to the PPSS algorithm and the proposed algorithm without energy harvesting, the proposed sleep scheduling algorithm can improve the target tracking efficiency and increase the nodes' residual energy.

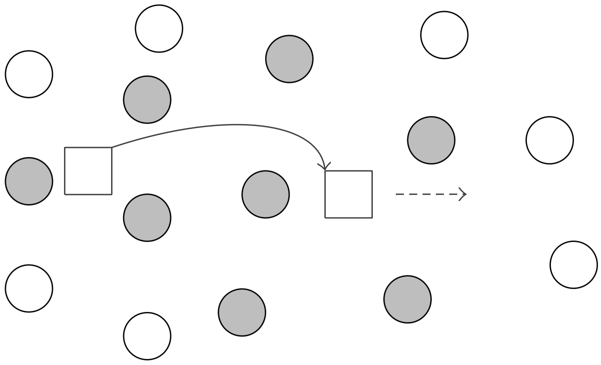


Figure 1: Target tracking in an energy harvesting sensor network.

The rest of this paper is organized as follows. The double-storage energy harvesting sensor network model and the target tracking model are introduced in Section 2. In Section 3, the PPSS algorithm and the proposed sleep scheduling algorithm are described in detail. Simulation results are presented in Section 4. Finally, some concluding remarks are made in Section 5.

2. System Model

We consider a wireless sensor network which is composed of static, uniformly distributed sensor nodes in a planar area. The nodes are equipped with double-storage energy harvesting components. The node's working time is divided into toggling periods (TP), and each TP is composed of sleep state and awaking state. The awaking state occupying the percentage of each TP is called a duty cycle (DC). The nodes in the sleep state have the lowest power consumption. They can harvest energy in the sleep state. The awaking nodes can detect the target and transmit and receive messages. Figure 1 illustrates a part of the energy harvesting sensor network for target tracking. White circles represent the nodes in the default toggling period, and grey circles represent the tracking nodes. The box denotes a target and the curve denotes its trajectory. The awaking nodes must be surrounding the target to keep tracking. There will be only one target passing through the area. When no target appears, all nodes keep in default duty cycle. If a target appears, the awaking nodes will detect the target and begin to track it. All nodes have sensing radius r and communication radius R. They are assumed to know their locations through GPS or other techniques. They are also locally time synchronized and have the ability to detect target's position (as in [24]). There is no sink in the sensor network.

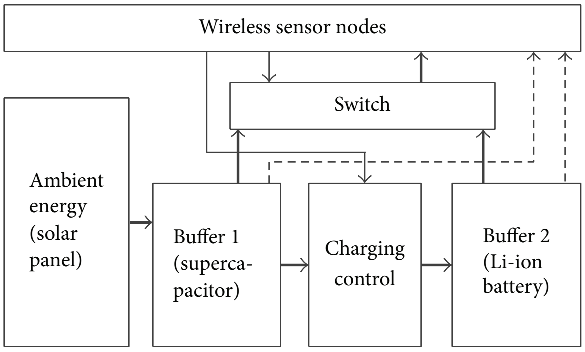

Wireless sensor nodes
Switch
Ambient energy (solar panel)
Buffer 1 (superca- pacitor)
Charging control
Buffer 2 (Li-ion battery)

Figure 2: Double-storage energy harvesting architecture.

The nodes can harvest energy only in the sleep state when the solar energy is available. The output of a solar panel E_n is expressed as [35]

$$E_n(t_0, T) = \int_{t_0}^{T+t_0} \eta_n f(t)\,dt = \eta_n \int_{t_0}^{T+t_0} f(t)\,dt, \quad (1)$$

where t_0 is the beginning of the energy harvesting time, T is the time duration, $f(t)$ is the statistical distribution of the solar energy, and η_n is the conversion efficiency of the solar panel.

According to the energy storage formula in [36] and (1), the stored energy $e_n(t)$ can be calculated as

$$e_n(t) = e_n(t - T) + kE_n(t_0, T) - e_{n\cos t}(t). \quad (2)$$

If there is no available energy, k equals 0; otherwise k equals 1. Here $e_{n\cos t}(t)$ denotes the consumed energy at the time t and $e_n(t - T)$ denotes the stored energy at the time $t - T$.

Nowadays, energy harvesting process is usually assumed to obey the Poisson distribution [37], and a single node's energy harvesting mechanism is comparatively mature. However, there is little research on energy harvesting mechanism for a large number of nodes. For convenience, we assume that energy arriving to each node in the sensor network follows an unknown random distribution. Besides, only the amount of energy is used in the algorithm.

The double-storage energy harvesting architecture [22] is shown in Figure 2. In general, the software program controls the energy usage and the recharging process of Buffer 1 and Buffer 2. Since the Li-ion battery has the limitation of deep charge (the battery is recharged after a complete drain-out) times, it is best to let these batteries undergo more shallow charge when the battery is partially discharged. Hence, having the supercapacitor as storage, we let the batteries experience more shallow charge when it is not fully discharged. The double-storage capability can take the advantages of both types of storage and avoid their disadvantages. It is an easy way to tackle the tradeoff between the target tracking performance and sleep scheduling.

3. Sleep Scheduling Algorithm

In this section, a sleep scheduling algorithm in the energy harvesting sensor network is derived. It makes some improvement to the PPSS algorithm and can increase target tracking efficiency.

3.1. Implementation of the PPSS Algorithm [26].

The PPSS algorithm selects the tracking nodes based on predicted target movement tendency to reduce the nodes' number and their active time. The tracking time is divided into time segments t_n. At first, the algorithm figures out the target state vector state(n) on the basis of state($n-1$) and the coordinates (x_n, y_n). Here state(n) $= (t_n, x_n, y_n, \overrightarrow{v_n}, \overrightarrow{a_n})$, $\overrightarrow{v_n} = (v_n, \theta_n)$, v_n is the target moving speed, θ_n denotes the target moving direction, and a_n is the acceleration. Afterwards, based on kinematics rules, the PPSS algorithm predicts $\overrightarrow{v_{n+1}}'$ and the target displacement $\overrightarrow{S_{n+1}}'$, where $\overrightarrow{S_{n+1}}' = (S_{n+1}, \Delta_{n+1})$. They can be expressed as

$$\begin{aligned} \overrightarrow{a_{n+1}}' &= \overrightarrow{a_n}, \\ \overrightarrow{v_{n+1}}' &= \overrightarrow{v_n} + \overrightarrow{a_{n+1}}' \cdot \mathrm{TP}, \\ \overrightarrow{S_{n+1}}' &= \overrightarrow{v_{n+1}}' \cdot \mathrm{TP} + \frac{1}{2}\overrightarrow{a_{n+1}}' \cdot \mathrm{TP}^2. \end{aligned} \quad (3)$$

Finally, it sets up Gaussian distribution models and a linear model for S_{n+1} and Δ_{n+1}, respectively, to predict the target location at the time instant t_{n+1}. However, when using the Gaussian distribution to predict S_{n+1}, that is, $S_{n+1} \sim N(\mu s_{n+1}, \delta^2 s_{n+1})$, according to the characteristics of Gaussian distribution, the accuracy will be approximately 68%. Using the linear model to predict Δ_{n+1} also has a certain amount of error.

At the beginning of each toggling period, the nodes which are called alarm nodes will be responsible for noticing the tracking nodes in the next round. They will broadcast state(n), $\overrightarrow{S_{n+1}}'$, μs_{n+1}, δs_{n+1}, and $\delta\Delta_{n+1}$ to awake neighbor nodes. When those nodes receive the messages, each node will, respectively, choose to wake or sleep according to the current situation. In each toggling period, some nodes are awaked. With the predicted target movement trajectory, they have the angle being less than $\sqrt{6}\delta\Delta_{n+1}$ and the distance d being in the range from $\mu s_{n+1} - \delta s_{n+1}$ to R. The nodes closer to the target have greater chance to wake up. The nodes' wake-up and sleep time are calculated below:

$$\begin{aligned} t_{\text{start}} &= t_{\text{alarmed}} + \frac{\max\{\mu s_{n+1} - \delta s_{n+1}, 0\}}{TS}, \\ t_{\text{end}} &= t_{\text{alarmed}} + \frac{d + r - (\mu s_{n+1} - \delta s_{n+1})}{TS}, \\ TS &= \frac{\mu s_{n+1}}{\mathrm{TP}}. \end{aligned} \quad (4)$$

3.2. Proposed Sleep Scheduling Algorithm

3.2.1. Shortcoming of the PPSS Algorithm. Although the PPSS algorithm improved energy efficiency, it increases the target loss rate. The reasons include the following: (1) If the target does not appear in the predicted direction and there are no awaking nodes around the target at that time, the target will be missed. (2) The target's movement is according to the predicted direction, but the speed is not in the prediction range, so that the target will be still in the last tracking region or will jump out of the current awake region. Both reasons will lead to target loss.

3.2.2. Improvement to the PPSS Algorithm. With the use of energy harvesting nodes, energy usage is no longer deficit. So the nodes can consume more energy in every tracking procedure. But due to the randomness of available energy harvesting time and the limit of solar panel conversion efficiency, the tracking procedure still needs to adapt to the energy harvesting situation. In the target tracking process, the first concern is to ensure the target is tracked, and the second concern is to optimize the energy utilization. We will modify the PPSS algorithm to solve the first problem. We combine the PPSS algorithm with another sleep scheduling algorithm we proposed [34], to predict the target loss situation and try to avoid it, and use double-storage energy harvesting sensor nodes to prolong network lifetime. In this way, it can improve the tracking performance and solve the energy limitation problem.

(i) Enhancing Target Tracking Performance. We used SS algorithm to represent the sleep scheduling algorithm in [34]. We first use PPSS algorithm to track the target, and we also add some communication processes of alarm nodes with their neighbor nodes. After alarm nodes send wake-up messages to neighbor nodes, alarm nodes immediately turn into the receiving status to wait for neighbor nodes' replying messages. Neighbor nodes will send reply messages about whether they decide to wake up at the next moment. If alarm nodes receive all sleeping messages, they will think it is unsuccessful to wake next-moment tracking nodes. The target is likely to be missed. By that time, alarm nodes will start using the SS algorithm to wake next tracking nodes immediately. After that, alarm nodes send another message to notice neighbor nodes while their distances are shorter than D to reply their location information and energy storage information. After receiving this information, the alarm nodes choose and wake n nodes which have more residual energy and are nearer to the target to track the target. Moreover, alarm nodes send target movement information to tracking nodes at the same time. At this moment, we assume that the target moves in uniform motion. So the choice of D is expressed as follows:

$$
\begin{aligned}
D &= \overline{S_n}, \\
\overline{S_n} &= \overline{v_n}\,\mathrm{TP}, \\
\overline{v_n} &= \frac{\left(\overline{v_n} + \overline{v_n} + \cdots + \overline{v_n}\right)}{n},
\end{aligned} \tag{5}
$$

where $\overline{S_n}$ denotes the target's average displacement and $\overline{v_n}$ denotes the target's average speed.

Alarm nodes will, respectively, choose $n/2$ nodes which have more residual energy (N_{ei}, $i = 1, \ldots, n/2$) and are nearer to the alarm nodes (N_{ni}, $i = 1, \ldots, n/2$) to keep awake in a whole default toggling period. The selecting methods are as follows: firstly, using selective sorting method to sort n nodes according to the stored energy from the largest to the smallest and sorting n nodes according to the distance to the target from the nearest to the farthest. According to the sorting results, choosing $n/2$ nodes with odd number to wake and track the target can make larger tracking coverage to find the target.

(ii) Using Double-Storage Energy Harvesting. The double-storage energy harvesting nodes are described below. If the supercapacitor charge is above a high threshold, it is used to power the node. If the supercapacitor charge is above a high threshold while the Li-ion battery level is below a high threshold, the battery is charged from the supercapacitor. If the solar energy is available and the supercapacitor charge is less than a low threshold, it begins to recharge. When the recharge opportunity is not available and the supercapacitor charge is less than a low threshold, the nodes are powered from the Li-ion battery until the Li-ion battery falls below a low threshold or until the solar energy is available.

(iii) Algorithm Description. For the actual implementation, we list flowchart of the algorithm.

Sleep scheduling algorithm is described as follows:

```
(1) Alarm nodes calculate target motion state and send
    messages to their neighbor nodes
(2) Alarm nodes wait reply messages
(3) if (all neighbor nodes choose to sleep)
(4) Alarm nodes send another message
(5) Neighbor nodes reply their $M_{di}$ (message of nodes
    nearer to the alarm nodes) and $M_{ei}$ (message of nodes
    have more residual energy)
(6) Alarm nodes find out $n/2$ $N_{ei}$ and $N_{ni}$ nodes
(7) Alarm nodes send wake up messages and target
    movement messages to the specific $n$ nodes
(8) end if
```

Energy harvesting battery control procedure is as follows:

```
(1) initialize the energy thresholds $V1$, $V2$, $V3$, $V4$, $V5$,
    $V6$; Let $V_{\text{super-capacitor}}$ denote the energy level of the
    super-capacitor and $V_{\text{Li-ion battery}}$ denote the energy
    level of the Li ion battery
(2) if (solar energy is available & $V_{\text{super-capacitor}} \leq V1$)
    Harvest energy and store it in the super-capacitor
(3) end if
(4) if ($V_{\text{super-capacitor}} \geq V2$)
(5) Super-capacitor powers the node
```

Table 1: Energy consumption parameters.

Status	Energy consumption
Active	9.6 (mJ/s)
Transmit	720 (nJ/bit)
Receive	110 (nJ/bit)
Sleep	0.33 (mJ/s)

```
end if
while $V_{\text{super-capacitor}} \geq V3$
    if ($V_{\text{Li-ion battery}} \leq V4$ & $V_{\text{super-capacitor}} \geq V5$)
        Super-capacitor charges the battery
    end if
end while
if (solar energy is not available & $V_{\text{super-capacitor}} \leq V6$)
    Battery powers the node
end if
```

4. Simulation Results

According to the system model, we conduct a simulation by using MATLAB to evaluate the performance of the proposed algorithm. The performance metrics are target loss ratio and remaining energy ratio. Target loss ratio is defined as the ratio of the times of target missed detection and the times of target detection. Remaining energy ratio is defined as the residual energy of the network after the target moves out and the residual energy of the network before the target enters.

4.1. Parameter Settings. The tracking region is a 150 × 150 m^2 square and nodes are randomly deployed in it. The density of nodes is changed from 1.5 nodes/100 m^2 to 5.5 nodes/100 m^2. The target speed is changed from 10 m/s to 30 m/s. $R/r = 3$, TP = 1 s, and DC = 20%. The target does curvilinear motion to pass the area after 3600 s. The nodes will not harvest energy before the target goes into the area. The experiment is repeated 100 times and the results are obtained by averaging. The energy consumption parameters are borrowed from [26], as shown in Table 1. Every communication delivers 250 bytes. The time required for communication is ignored.

The solar panels' area is 5 cm × 5 cm. The harvested energy is 0.1 W/cm^2 [38], and energy conversion efficiency is 15%. We consider the randomness of energy harvesting and assume that the probability of energy arrival is 50%. The initial energy of each battery is 2 J. Each battery has 200 deep charge cycles and has infinite shallow charge cycle. The initial energy of the supercapacitor is 0 J. Its max capacity is 1 J, and it has infinite recharge cycles.

From the energy harvesting and charging process and in order to simplify calculation, we use the amount of contained energy instead of voltage thresholds, as shown in Table 2.

Table 2: Contained energy thresholds.

Contained energy	Energy thresholds
$E1(V1)$	0.8 J
$E2(V2)$	0.3 J
$E3(V3)$	0.35 J
$E4(V4)$	0.4 J
$E5(V5)$	0.8 J
$E6(V6)$	0.1 J

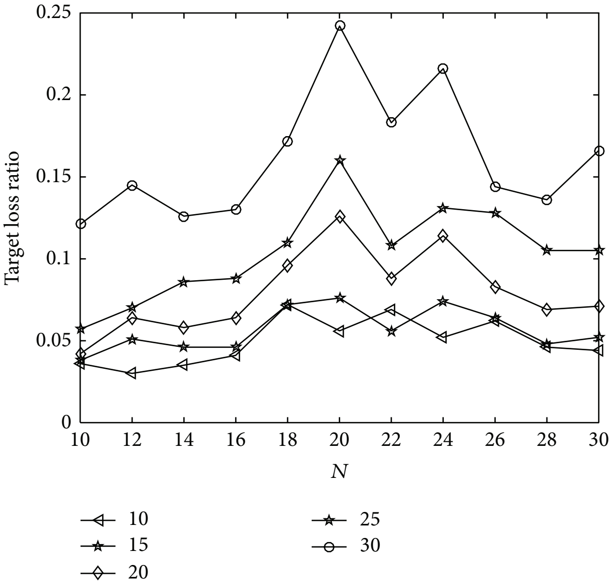


Figure 3: Target loss ratio versus number of nodes.

4.2. Discussions of Simulation Results

4.2.1. Target Loss Ratio versus Number of Nodes. The influence of number of nodes on the target loss ratio is shown in Figure 3, where the nodes' density is 2 nodes/100 m^2. The 5 curves represent different target speeds (m/s). As we can see, larger speed makes higher target loss ratio. But there is no linear relation or correlation between target speed and number of nodes. When the awaking nodes' number increases, it just increases the number of nodes in the same range but does not increase the target tracking area. As a result, the target loss ratio does not increase. Hence, in order to accelerate the program run, we choose N to be 10 in the following, if the number of nodes is not specified additionally.

4.2.2. Performance Comparison. Figures 4 and 5 illustrate the remaining energy ratio versus the target speed and node density, respectively. In Figure 4, the node density is 2 nodes/100 m^2. In Figure 5, the target speed is fixed on 15 m/s. The curve A1 represents the PPSS algorithm, curve A2 represents the proposed algorithm with no energy harvesting, and curve A3 represents the proposed algorithm with energy harvesting. Before the target goes into the area, the energy consumption is almost the same. As we can

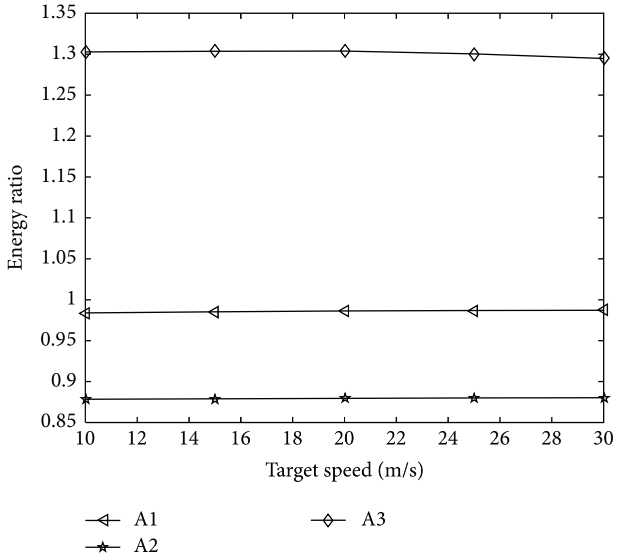


Figure 4: Remaining energy ratio versus target speed.

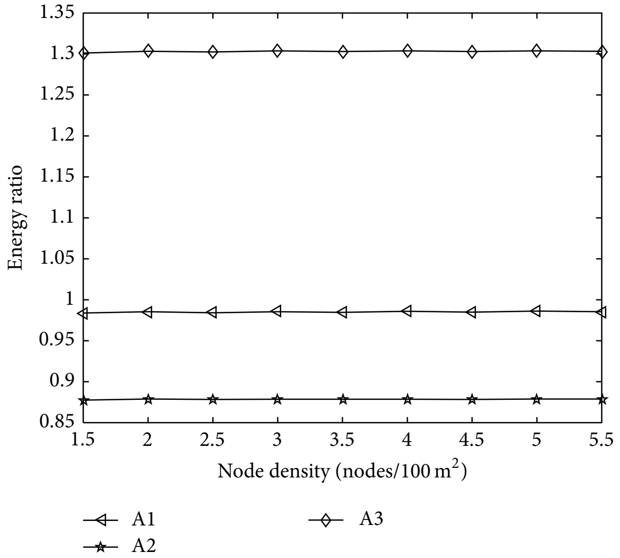


Figure 5: Remaining energy ratio versus node density.

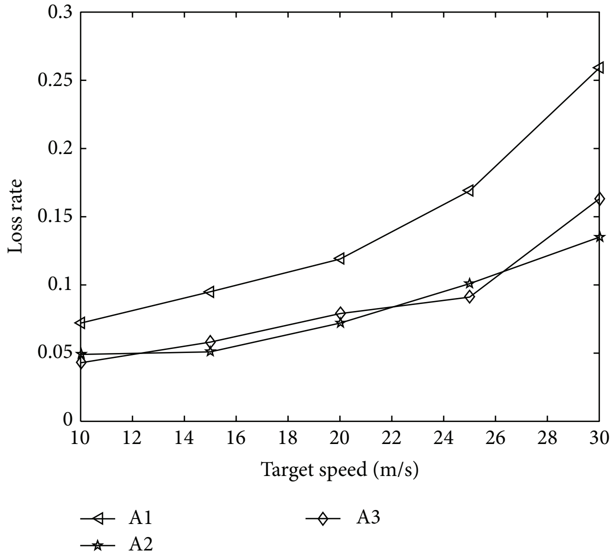


Figure 6: Target loss ratio versus target speed.

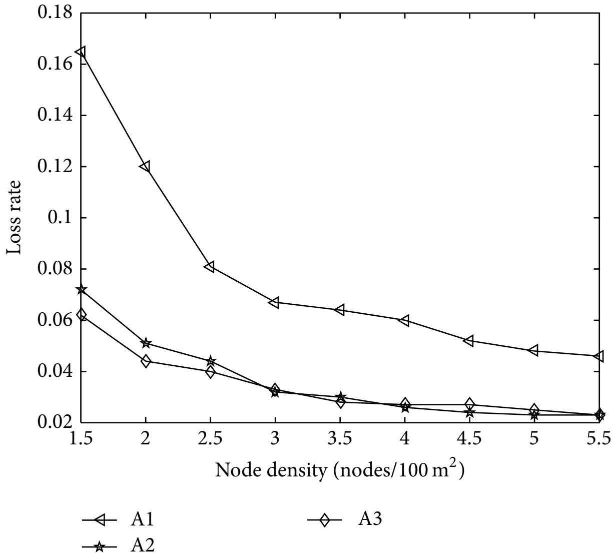


Figure 7: Target loss ratio versus nodes density.

see, both figures show that, without energy harvesting, the proposed algorithm has the least remaining energy ratio, which means that it consumes more energy than the PPSS algorithm. But with energy harvesting, it has the highest remaining energy ratio; namely, the energy harvesting nodes can increase the residual energy. In these two figures, lines are nearly parallel to the x-axis. It means that, with the increase of node density and target speed, the percentage of energy consumption is similar. The increased energy consumption has impact on target tracking loss ratio reduction.

Figures 6 and 7 illustrate the target loss ratio versus the target speed and the node density. Both figures illustrate that the proposed algorithm reduces the target loss ratio. In Figure 6, the node density is 2 nodes/100 m^2. As we can see, the proposed algorithm has lower target loss ratio than the PPSS algorithm, and the increase of target speed can lead to increased target loss ratio. The target loss ratio does not change regardless of energy harvesting or no energy harvesting. In Figure 7, the target speed is fixed on 15 m/s. The PPSS algorithm still has the highest target loss ratio, and the two curves' trends for the proposed algorithm are similar. As the increase of node density makes the increase of tracking nodes' number, the target loss ratio obviously drops. Both Figures 6 and 7 illustrate that, with the increase of node density, the tracking nodes' number increases and the target loss ratio decreases.

5. Conclusions

In this paper, we improve the PPSS algorithm and combine it with another sleep scheduling algorithm we proposed to decrease the target loss rate in energy harvesting sensor networks. We also utilize double-storage energy harvesting sensor nodes to prolong the network lifetime. In the tracking process, when the target is likely to be missed, we immediately use another sleep scheduling algorithm we proposed to ensure that the target is tracked. Simulation results show that the proposed sleep scheduling algorithm can enhance the target tracking performance, while the energy harvesting nodes make the sensor network work longer, compared with the PPSS algorithm and the proposed algorithm without energy harvesting.

Competing Interests

The authors declare that they have no competing interests.

Acknowledgments

This research was supported by the National Natural Science Foundation of China (61162008, 61172055, and 61471135), the Guangxi Natural Science Foundation (2013GXNSFGA019004, 2015GXNSFBB139007), the Fund of Key Laboratory of Cognitive Radio and Information Processing, Guilin University of Electronic Technology, China (2013ZR02), and the Fund of Guangxi Key Laboratory of Wireless Wideband Communication and Signal Processing (CRKL150104).

References

[1] S. He, J. Chen, X. Li, X. S. Shen, and Y. Sun, "Mobility and intruder prior information improving the barrier coverage of sparse sensor networks," *IEEE Transactions on Mobile Computing*, vol. 13, no. 6, pp. 1268–1282, 2014.

[2] Q. Yang, S. He, J. Li, J. Chen, and Y. Sun, "Energy-efficient probabilistic area coverage in wireless sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 1, pp. 367–377, 2015.

[3] S. He, D.-H. Shin, J. Zhang, J. Chen, and Y. Sun, "Full-view area coverage in camera sensor networks: dimension reduction and near-optimal solutions," *IEEE Transactions on Vehicular Technology*, 2015.

[4] J. Gao, J. Li, Z. Cai, and H. Gao, "Composite event coverage in wireless sensor networks with heterogeneous sensors," in *Proceedings of the IEEE Conference on Computer Communications (INFOCOM '15)*, pp. 217–225, IEEE, Hong Kong, April 2015.

[5] Z. He, Z. Cai, S. Cheng, and X. Wang, "Approximate aggregation for tracking quantiles and range countings in wireless sensor networks," *Theoretical Computer Science*, vol. 607, no. 3, pp. 381–390, 2015.

[6] S. Cheng, Z. Cai, J. Li, and X. Fang, "Drawing dominant dataset from big sensory data in wireless sensor networks," in *Proceedings of the IEEE Conference on Computer Communications (INFOCOM '15)*, pp. 531–539, Hong Kong, April 2015.

[7] T. Shi, S. Cheng, Z. Cai, and J. Li, "Adaptive connected dominating set discovering algorithm in energy-harvest sensor networks," in *Proceedings of the 35rd Annual IEEE International Conference on Computer Communications (INFOCOM '16)*, April 2016.

[8] R. V. Prasad, S. Devasenapathy, V. S. Rao, and J. Vazifehdan, "Reincarnation in the ambiance: devices and networks with energy harvesting," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, pp. 195–213, 2014.

[9] P. Zhang, G. Xiao, and H.-P. Tan, "Clustering algorithms for maximizing the lifetime of wireless sensor networks with energy-harvesting sensors," *Computer Networks*, vol. 57, no. 14, pp. 2689–2704, 2013.

[10] N. Tadayon, S. Khoshroo, E. Askari, H. Wang, and H. Michel, "Power management in SMAC-based energy-harvesting wireless sensor networks using queuing analysis," *Journal of Network and Computer Applications*, vol. 36, no. 3, pp. 1008–1017, 2013.

[11] B. Zhang, R. Simon, and H. Aydin, "Harvesting-aware energy management for time-critical wireless sensor networks with joint voltage and modulation scaling," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 514–526, 2013.

[12] H. Besbes, G. Smart, D. Buranapanichkit, C. Kloukinas, and Y. Andreopoulos, "Analytic conditions for energy neutrality in uniformly-formed wireless sensor networks," *IEEE Transactions on Wireless Communications*, vol. 12, no. 10, pp. 4916–4931, 2013.

[13] X. Shen, C. Bo, J. Zhang, S. Tang, X. Mao, and G. Dai, "EFCon: energy flow control for sustainable wireless sensor networks," *Ad Hoc Networks*, vol. 11, no. 4, pp. 1421–1431, 2013.

[14] V. Pryyma, D. Turgut, and L. Bölöni, "Active time scheduling for rechargeable sensor networks," *Computer Networks*, vol. 54, no. 4, pp. 631–640, 2010.

[15] N. Jaggi, S. Madakasira, S. R. Mereddy, and R. Pendse, "Adaptive algorithms for sensor activation in renewable energy based sensor systems," *Ad Hoc Networks*, vol. 11, no. 4, pp. 1405–1420, 2013.

[16] H. Yoo, M. Shim, and D. Kim, "Dynamic duty-cycle scheduling schemes for energy-harvesting wireless sensor networks," *IEEE Communications Letters*, vol. 16, no. 2, pp. 202–204, 2012.

[17] A. C. Valera, W.-S. Soh, and H.-P. Tan, "Energy-neutral scheduling and forwarding in environmentally-powered wireless sensor networks," *Ad Hoc Networks*, vol. 11, no. 3, pp. 1202–1220, 2013.

[18] W. H. Chan, P. Zhang, I. Nevat et al., "Adaptive duty cycling in sensor networks with energy harvesting using continuous-time Markov chain and fluid models," *IEEE Journal on Selected Areas in Communications*, vol. 33, no. 12, pp. 2687–2700, 2015.

[19] L. Huang and M. J. Neely, "Utility optimal scheduling in energy-harvesting networks," *IEEE/ACM Transactions on Networking*, vol. 21, no. 4, pp. 1117–1130, 2013.

[20] Y. Zhang, S. He, J. Chen, Y. Sun, and X. S. Shen, "Distributed sampling rate control for rechargeable sensor nodes with limited battery capacity," *IEEE Transactions on Wireless Communications*, vol. 12, no. 6, pp. 3096–3106, 2013.

[21] S. Climent, A. Sanchez, S. Blanc, J. V. Capella, and R. Ors, "Wireless sensor network with energy harvesting: modeling and simulation based on a practical architecture using real radiation levels," *Concurrency and Computation: Practice and Experience*, vol. 28, no. 6, pp. 1812–1830, 2016.

[22] X. Jiang, J. Polastre, and D. Culler, "Perpetual environmentally powered sensor networks," in *Proceedings of the 4th International Symposium on Information Processing in Sensor Networks (IPSN '05)*, pp. 463–468, IEEE, Los Angeles, Calif, USA, April 2005.

[23] E. Masazade, M. Fardad, and P. K. Varshney, "Sparsity-promoting extended kalman filtering for target tracking in wireless sensor networks," *IEEE Signal Processing Letters*, vol. 19, no. 12, pp. 845–848, 2012.

[24] P. Braca, P. Willett, K. Lepage, S. Marano, and V. Matta, "Bayesian tracking in underwater wireless sensor networks with port-starboard ambiguity," *IEEE Transactions on Signal Processing*, vol. 62, no. 7, pp. 1864–1878, 2014.

[25] M. B. Guldogan, "Consensus bernoulli filter for distributed detection and tracking using multi-static Doppler shifts," *IEEE Signal Processing Letters*, vol. 21, no. 6, pp. 672–676, 2014.

[26] B. Jiang, B. Ravindran, and H. Cho, "Probability-based prediction and sleep scheduling for energy-efficient target tracking in sensor networks," *IEEE Transactions on Mobile Computing*, vol. 12, no. 4, pp. 735–747, 2013.

[27] O. Demigha, W.-K. Hidouci, and T. Ahmed, "On Energy efficiency in collaborative target tracking in wireless sensor network: a review," *IEEE Communications Surveys and Tutorials*, vol. 15, no. 3, pp. 1210–1222, 2013.

[28] G. Wang, M. Z. A. Bhuiyan, J. Cao, and J. Wu, "Detecting movements of a target using face tracking in wireless sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 4, pp. 939–949, 2014.

[29] F. Zhou, G. Trajcevski, O. Ghica, R. Tamassia, P. Scheuermann, and A. Khokhar, "Deflection-aware tracking-principal selection in active wireless sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 61, no. 7, pp. 3240–3254, 2012.

[30] Y. Fu, Q. Ling, and Z. Tian, "Distributed sensor allocation for multi-target tracking in wireless sensor networks," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 48, no. 4, pp. 3538–3553, 2012.

[31] R. Sarkar and J. Gao, "Differential forms for target tracking and aggregate queries in distributed networks," *IEEE/ACM Transactions on Networking*, vol. 21, no. 4, pp. 1159–1172, 2013.

[32] J. Feng, B. Lian, and H. Zhao, "An energy-efficient sleep strategy for target tracking sensor networks," *Sensors and Transducers*, vol. 164-165, no. 2, pp. 92–99, 2014.

[33] C. Yang and K.-W. Chin, "Novel algorithms for complete targets coverage in energy harvesting wireless sensor networks," *IEEE Communications Letters*, vol. 18, no. 1, pp. 118–121, 2014.

[34] Q. Zeng, H. Chen, F. Zhao, and X. Li, "A sleep scheduling algorithm for target tracking in energy harvesting sensor networks," in *Proceedings of the International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC '14)*, pp. 323–330, Shanghai, China, October 2014.

[35] U. Alvarado, A. Juanicorena, I. Adin, B. Sedano, I. Gutiérrez, and J. de Nó, "Energy harvesting technologies for low-power electronics," *Transactions on Emerging Telecommunications Technologies*, vol. 23, no. 8, pp. 728–741, 2012.

[36] M. Gorlatova, A. Wallwater, and G. Zussman, "Networking low-power energy harvesting devices: measurements and algorithms," *IEEE Transactions on Mobile Computing*, vol. 12, no. 9, pp. 1853–1865, 2013.

[37] S. Zhang, A. Seyedi, and B. Sikdar, "An analytical approach to the design of energy harvesting wireless sensor nodes," *IEEE Transactions on Wireless Communications*, vol. 12, no. 8, pp. 4010–4024, 2013.

[38] S. Sudevalayam and P. Kulkarni, "Energy harvesting sensor nodes: survey and implications," *IEEE Communications Surveys and Tutorials*, vol. 13, no. 3, pp. 443–461, 2011.

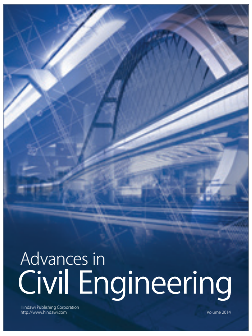
Journal of
Engineering

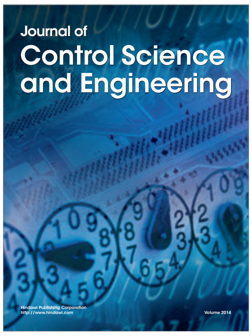
The Scientific World Journal

International Journal of
Rotating Machinery

Journal of
Sensors

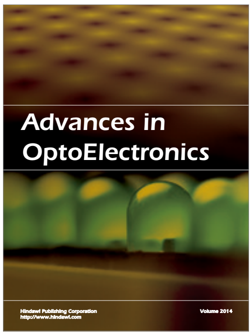
International Journal of
Distributed Sensor Networks


Advances in
Civil Engineering


Journal of
Control Science and Engineering

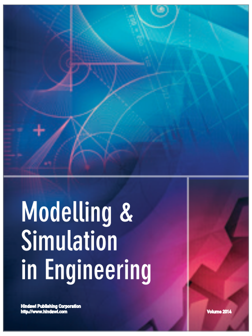
Journal of
Robotics

Journal of
Electrical and Computer Engineering

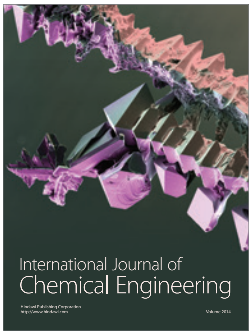

Advances in
OptoElectronics

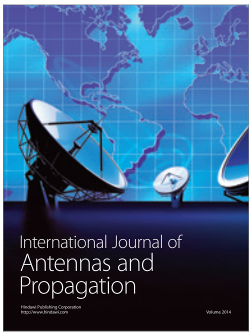
VLSI Design

International Journal of
Navigation and Observation

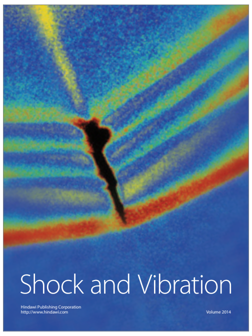

Modelling &
Simulation
in Engineering

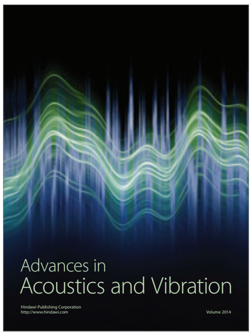
International Journal of
Aerospace Engineering


International Journal of
Chemical Engineering


International Journal of
Antennas and Propagation

Active and Passive
Electronic Components


Shock and Vibration


Advances in
Acoustics and Vibration