ISSN 1000-9825, CODEN RUXUEW
Journal of Software, Vol.20, No.1, January 2009, pp.164−176
doi: 10.3724/SP.J.1001.2009.00562

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel/Fax: +86-10-62562563

面向移动节点定位的传感器网络预唤醒策略*

刘玉恒[1+], 陈真勇[1], 吴 晶[1,2], 熊 璋[1]

[1](北京航空航天大学 计算机科学与工程学院,北京 100191)
[2](中国移动通信研究院 产业市场研究所,北京 100053)

Proactive Wakeup and Sleep Scheduling Scheme for Localizing Mobile Sensors

LIU Yu-Heng[1+], CHEN Zhen-Yong[1], WU Jing[1,2], XIONG Zhang[1]

[1](School of Computer Science and Engineering, BeiHang University, Beijing 100191, China)
[2](Department of Industry and Market Research, Research Institute of China Mobile, Beijing 100053, China)
+ Corresponding author: E-mail: hengly@cse.buaa.edu.cn

Liu YH, Chen ZY, Wu J, Xiong Z. Proactive wakeup and sleep scheduling scheme for localizing mobile sensors. *Journal of Software*, 2009,20(1):164−176. http://www.jos.org.cn/1000-9825/562.htm

Abstract: When localizing a mobile node in sensor networks, it is possible that a seed node is passed by the mobile node during its sleeping mode, which might lead to failed localization. This paper proposes a dynamic node scheduling scheme P-SWIM based on proactive wakeup and sleep. In P-SWIM, each seed is proactively notified if a mobile node is moving toward it. Hence, only these seeds remain active in full duty when the mobile node passes by them, while the other seeds still stay in sleeping mode. Simulation results indicate that localization algorithms based on P-SWIM can achieve better localization performance than those based on the other two static node scheduling schemes, RIS and GAF. Moreover, P-SWIM incurs least running overhead to the network overall power consumption among the three schemes. In addition, the paper evaluates the effect of node scheduling schemes on localization performance by tuning the parameters of each scheme, which presents guidelines for efficient network deployment for mobile sensor positioning applications.
Key words: mobile sensor networks; self localization; node scheduling scheme; proactive wakeup and sleep

摘 要: 当无线传感器网络对移动节点进行定位时,锚节点可能会因为处于休眠状态而没有响应移动节点的定位请求,从而导致定位失败.提出一种基于预唤醒机制的动态功耗控制策略 P-SWIM,该策略提前通知移动节点周边的锚节点进入全勤的工作方式,而网络内其他锚节点则仍然处于低功耗的工作方式.仿真实验结果表明,移动节点定位方法采用 P-SWIM 相比于采用静态功耗控制策略(RIS 和 GAF)能够显著地提高定位性能,且 P-SWIM 引入的功耗也是 3 种策略中最低的.此外,通过大量的仿真实验,评估了调节 3 种策略的各项参数对移动节点定位方法性能的影响,为在实际应用中高效的部署网络提供了参考方案.
关键词: 移动传感器网络;自身定位;节点休眠调度策略;预先唤醒与休眠

---

* Supported by the Aero Science and Technology Foundation of China under Grant No.05E551010 (中国航空支撑科技基金); the Key Subject Foundation of Beijing of China under Grant No.XK100060423 (北京市重点学科基金); the Graduate Practice and Innovation Foundation of BUAA of China under Grant No.2007.20 (北京航空航天大学研究生创新实践基金)

Received 2007-12-09; Accepted 2008-05-06

中图法分类号: TP393　　文献标识码: A

1 Introduction

Location awareness of an individual node is a critical requirement for the deployment and implementation of wireless sensor networks in a wide variety of applications[1,2]. Location dependent examples include forest fire detection[3], animal migrations tracking[4] and emergency navigation[5]. In recent years a number of self-localization algorithms have been proposed specifically for sensor networks[6]. These algorithms share one common mechanism: the unknown node collects the location information of neighboring seeds; when enough seeds are collected, the unknown node performs the approaches of multilateration or center of gravity to estimate its location. In mobile sensor networks, the localization of mobile unknown sensors can be achieved by rerunning the localization algorithms after some time interval[7].

Recent research results show that increasing the number of visible seeds can improve the possibility of successful localization and the localization accuracy. However, more seeds will increase the cost and complexity of deployment. Moreover, in most cases, it is not practical for all seeds to work in full duty and wait for the request from unknown nodes at any time since replenishment of power sources in the sensor networks might be impossible. In fact, seeds generally adopt some node scheduling schemes to conserve limited energy. For example, each seed switches between wakeup and sleep according to a predefined low duty cycle, where duty cycle is defined as the fraction of time that a seed is active. In case an active seed hears a request from an unknown node, it enters full duty mode and remains active in the whole localization process. After the unknown node leaves, the seed returns to the previous low power mode. Under such static wakeup and sleep scheduling schemes, current mobile localization algorithms iteratively using information from neighboring seeds will suffer from severe information delay caused by scheduled sleeping and wakeup switching. Therefore, it is possible that a seed is passed by a mobile node during its sleeping period, so that the mobile node can pass across the seed without being responded by the seed, which might result in corrupted localization quality. In this paper, we refrain from proposing a specific localization method. Instead, we study dynamic power conservation protocols for positioning mobile sensors. We are concerned with the problem of how to adjust sleep and wakeup modes of seeds dynamically so that a certain degree of connectivity can be achieved for successful localization of mobile sensor nodes.

Random Independent Sleeping (RIS)[8] is the most simple node scheduling scheme proposed to maintain connectivity of sensor nodes. At each node, the time is divided into time slots of equal length T_{slot}. The interval of each time slot is divided into two parts: the active period and the sleeping period. The duration of active period is $p\times T_{slot}$ where p is the duty cycle or active probability. The sleeping period takes the remaining part of a time slot, that is $(1-p)\times T_{slot}$. Since the sleeping probability is independent at each seed, in a macroscopic view, the expected number of active nodes within the area at any given time is $p\times n$, where n is the total number of seeds. Based on RIS, a number of promising power management protocols have been developed for connectivity maintenance. LEACH[9] divides the network into clusters in each of which sensors choose one sensor as the cluster head based on a randomized election algorithm. The cluster head takes the role of data collection and aggregation among clusters so that the cluster members can remain sleeping in most of time. In ASCENT[10], each sensor assesses its connectivity and schedules its sleeping time in the multi-hop network topology so that robust data routes can be guaranteed. GAF[11] is a power-effective sleeping planning scheme to keep network connectivity. GAF divides the target region into virtual grids. Sensors in the same grid contend to be the only active sensor in this grid, while the other sensors go into sleeping. As a given time expires, a new round of competition begins. Due to the static mechanism, however, the above schemes might fail to maintain network connectivity in presence of mobile sensor nodes. Moreover, they are designed to reduce the unnecessary power consumption during the packet delivery process and do not consider the spatiotemporal constraints in localization of mobile sensor networks.

In this paper, we present a novel node scheduling scheme, Proactive Sleep and Wakeup Scheduling for Localization in Mobile Sensor Networks (P-SWIM). As the mobile node approaches, P-SWIM dynamically predicts the seeds in the future neighborhood (local or multi-hop) of the mobile node and proactively informs them to stay in full duty mode during the presence of the mobile node. By waking up the right seeds at the right time, the mobile node always has a desired number of neighboring seeds as it moves. Since most seeds stay in low power mode before the mobile node arrives, the limited energy can be conserved. Furthermore, this paper provides detailed quantitative comparison of the impact of node scheduling schemes on localization quality and power efficiency under different localization algorithms. The extensive experiment results and analysis provide guidelines for system configuration upon different requirements in mobile sensor positioning applications.

2 Network Model and Problem Definition

We consider the following mobile sensor positioning applications: monitoring wildlife habitat in mountain areas and navigating firefighters in smoky buildings. In such sensor networks, sensors include the mobile node carried by the mobile object and the fixed seeds pre-deployed in the target region. Since localization process is periodical and lasts for a short period of time when the mobile object is present, the mobile node may adopt no power saving operations and always stay in active mode (Moreover, power source in the mobile node is refreshable). For seeds, however, replenishment of batteries might be impossible due to inaccessibility. Therefore, it is crucial for seeds to perform energy-efficient sleep and wakeup scheduling schemes to maximize the network life-time and minimize the cost of repeated deployments.

Based on the characteristics of the above application, we use the following network model. The sensor network is deployed in a square region ($l \times l$) in which the total number of fixed seeds and mobile nodes are n and m respectively ($n >> m$). The deployment of the n seeds follows the random uniform fashion, where each seed has equal likelihood of falling at any location in the area, independently of the other seeds. We assume that seeds run the basic RIS in which each time slot is T_{slot} and duty cycle is p. Each mobile node moves in a random manner and is unaware of its moving speed and direction, other than knowing its speed is less than $v_{\max}$. The localization algorithm running at each node is an h-hop type, which means it requires seeds of h hops in the neighborhood to participate the location estimate. The iterative localization process takes place after a time interval of t.

We assume a disc-based communication model where each active sensor has a communication radius of r. A mobile node is said to be k-connected if it is within the communication ranges of at least k active seeds. Therefore, the problems we address in this paper are listed as follows:

1. As a mobile node moves, how to adjust the sleep and wakeup schedule of seeds dynamically to keep a given level of connectivity which meets the requirements of the h-hop localization algorithm?
2. The dynamic node scheduling scheme incurs communication overhead, therefore, the trade-off between communication cost and connectivity is another problem to be addressed.

3 P-Swim Scheme

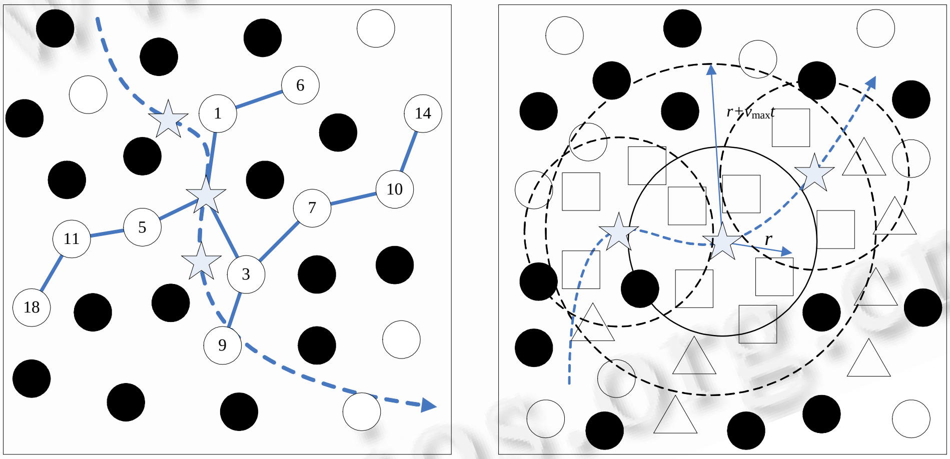
In this section we describe our novel node scheduling scheme, Proactive Sleeping and Wakeup for localization In Mobile sensor networks (P-SWIM). The general idea of P-SWIM is that the active seeds in the localization area of the mobile node are proactively informed to switch to full duty cycle and wait for the arrival of the mobile node;

when the mobile node leaves, the seeds switch back to low power mode to conserve energy. The basic pseudo code of P-SWIM is presented as shown in Fig.1.

```
ConstructInitialLocalizationTree($LT_0$); // construct the initial localization tree
for each $t_i$ with an interval of $t$ do {
    LocalizeMobileNode ($t_i$); // run a specific localization algorithm to estimate its position
    ProcactivelyWakeup($LT_i$);            // wakeup the corresponding seeds proactively
    ReconfigureLocalizationTree($LT_i$); // add or remove seeds from Localization Tree
}
```

Fig.1 Basic structure of P-SWIM

When the mobile node enters the monitoring area, an initial localization tree LT_0 is built. Then the iterative localization process begins by running a specific localization algorithm at a certain time interval t. As the mobile node moves, it predicts the seeds to be inside/outside of its localization area and proactively notifies them to go into wakeup/sleep mode. Then the mobile node may reconfigure the localization tree by adding some new seeds and pruning some existing seeds. Note that a sleeping seed can be woken up only when the sleeping period expires. Since we can not wake up a sleeping sensor, our proactive wakeup and sleep scheme is to let the active seeds at current localization time t_i remain active at the next localization time t_{i+1}.



(a) Construct the initial localization tree (b) Proactive wakeup and sleep (when h=1)

☆ Mobile node ● Sleeping seed ○ RIS active seed □ Localizing seed △ Full duty seed

Fig.2 Main steps in P-SWIM

3.1 Initial localization tree construction

When a mobile node enters the target region, it acquires the topology around it by flooding. As illustrated in Fig.2(a), the mobile node broadcasts a REQUEST_MSG packet containing its identity NODE_ID and a counter HOP_COUNT initialized to 1. Each receiving seed maintains the minimum counter value per NODE_ID and ignores those REQUEST_MSG packets with higher HOP_COUNT values. The REQUEST_MSG packets are flooded outward with HOP_COUNT values incremented at every intermediate hop. Meanwhile, each seed replies to the mobile node with a RESPONSE_MSG packet piggybacked with seed identity SEED_ID, HOP_COUNT and the nearest intermediate seed identity PARENT_ID. The flooding ends when the HOP_COUNT value per NODE_ID at each seed is no larger than those at its neighboring seeds.

Through this mechanism, the mobile node gets the shortest distance (in hops) to all the seeds in the network. Such information can be stored by the structure of localization tree rooted at the mobile node. Each node in the localization tree is represented by a ternary item ⟨SEED_ID, HOP_COUNT, PARENT_ID⟩.

3.2 Proactive sleep and wakeup

After construction of the initial localization tree, the mobile node is aware of the topology of its neighboring seeds. Then, at time t_i, it is possible to estimate which seeds locate in the next localization area of the mobile node at time t_{i+1}. P-SWIM filters out these seeds to forewarn them to remain in activity. To distinguish the normal sleeping and active mode, the working state of each seed can be further divided into: Sleeping mode, RIS-Active mode, Full Duty mode and Localizing mode. In Sleeping mode seeds consume least energy; RIS-Active mode represents the active period in low power mode of RIS; In Full Duty mode seeds remain awake and wait for the localization request from mobile nodes at all time; In Localization mode seeds participate in the localization process and have the highest power consumption that involves communication and computation. Therefore, the localization tree includes the seeds in Full Duty mode and Localizing mode.

As demonstrated in Fig.2(b), at t_i, the mobile node is ready to proactively wakeup the seeds that reside in its localization area at t_{i+1}, i.e., h-hop neighboring seeds at t_{i+1}, denoted as W_{hi} (labeled as i since this is a value at t_i, the same as below). To find W_{hi} in the localization tree of time t_i, called LT_i, the mobile node has to first predict the seeds which locate in its possible area at t_{i+1}, denoted as T_i. We can see that these seeds are contained in the circular area with the mobile node as the origin and $r+ v_{\max}t$ as the radius.

To reduce extra power consumption we propose a seed filtering algorithm independent of location information. First, we get the average distance of per hop based on the formula by Kleinrock and Silvester in Ref.[12]:

$$d_{hop} = r\left(1+\mathrm{e}^{-N} - \int_{-1}^{1} \mathrm{e}^{-\frac{N\left(\arccos(t)-t\sqrt{1-t^2}\right)}{\pi}}\mathrm{d}t\right) \tag{1}$$

where N is the network density defined as the number of nodes in one hop communication range, i.e. $N=\frac{\pi npr^2}{l^2}$. Then, the largest layer where the seeds in T_i locate in LT_i can be obtained:

$$hop_{\max} = \left\lceil \frac{r+v_{\max}t}{d_{hop}} \right\rceil \tag{2}$$

So, we get the seeds that reside at the layer of $1\sim hop_{\max}$ in LT_i. Finally, the mobile node chooses the h-hop neighbors of the seeds T_i in as W_{hi}, i.e., the seeds that locate at the layer of $h+1\sim h+hop_{\max}$. The proposed filtering algorithm does not require location information flooding across the network. Therefore, our algorithm can quickly find the desired seeds especially in the data structure of localization tree at a low communication cost.

When S_{hi} is available, the mobile node sends a WAKEUP_MSG packet to the seeds in W_{hi}. Each receiving seed switches to Full Duty mode and is ready for participation in localization. Meanwhile, the mobile node sends a PRUNE_MSG packet to the other seeds in LT_i to notify them that the mobile node is leaving. To further minimize the power consumption, the mobile node can stop the proactive sleep and wakeup process when the h-hop connectivity is high enough to ensure a successful localization. We present the pseudo code of our proactive sleep and wakeup algorithm as illustrated in Fig.3.

```
void ProcactivelyWakeup($LT_i$) {
    $W$=F ;                    // the set of the seeds to be proactively waked up
    nConnectivity = 0;
    for hop_count from $h$+1 to $h+hop_{max}$ do {
    // get seed from Localization Tree at the given layer hop_count
    Seed_ID=GetSeedFromLayer($LT_i$, hop_count);
    AddSeed($W$, Seed_ID);
    if (nConnectivity++ >= MAX_CONNECTIVITY)    // max connectivity reaches
        break;
    }
    for each seed in $W$ {
        Seed_ID = GetNextSeed($W$);
        SendWakeupMsg(Seed_ID);// send WAKEUP_MSG to add the seed
    }
    for each seed in $LT_i$ {
        Seed_ID = Traverse($LT_i$);        // traverse Localization Tree to get each seed
        if (Seed_ID not in $W$)             // seeds not in $W$
            SendPruneMsg(Seed_ID);      // send PRUNE_MSG to prune the seed
    }
}
```

Fig.3 Algorithm of proactive wakeup and sleep

3.3 Reconfiguration of localization tree

As the sleep schedules and the mobile node moves, the localization tree varies from time to time. Therefore, we need to reconfigure the localization tree. Figure 4 shows the state transition diagram in this phase.

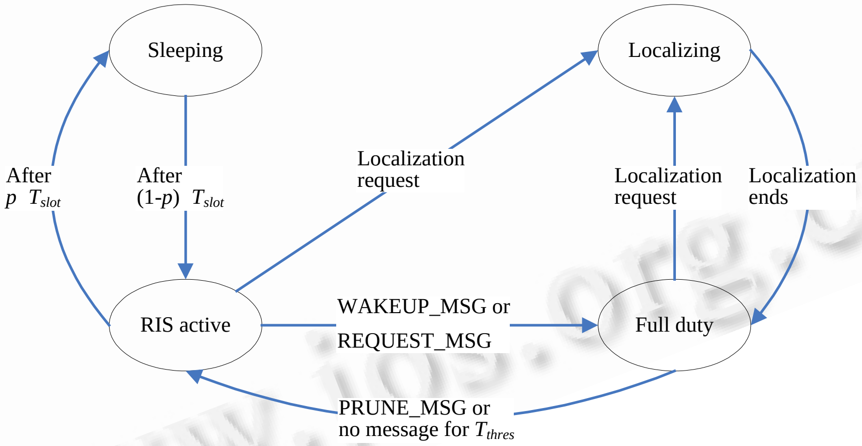


Fig.4 State transition of seeds

Since the localization tree should only include the seeds in Full Duty mode and Localizing mode, the reconfiguration involves adding the proactively woken-up seeds and pruning the seeds away from the mobile nodes. The process of the reconfiguration of localization tree is similar to that of the construction of the initial localization tree. The target node broadcasts a REQUEST_MSG packet to the network, and only the receiving seeds in Full Duty mode and Localizing mode respond the target node with a RESPONSE_MSG packet. The target node updates the corresponding ternary items using the collected information and the localization tree is then reconfigured.

4 Performance Evaluation

4.1 Simulation settings

In this section we present the simulation results in MCL Simulator[13]. In the simulations, the radio communication radius is set to 50 meters. The target network region is a free space of size 500m×500m. We use 200 seeds with uniform random distribution and 10 unknown nodes following the random waypoint mobility model[14]. Each unknown node is unaware of their velocity and direction, but has a known maximum velocity $v_{\max}$.

The node scheduling schemes that we have implemented are GAF, RIS, P-SWIM. Then, according to the range of localization area (h-hop) of mobile localization algorithms, we choose AML[15] (h=1), MCL[13] (h=2) and AMP[16] (h>2) and run them on the top of each scheme to evaluate the impact of node scheduling schemes on the localization quality. We are not only concerned with comparing their relative pros-and-cons, but also study how to tune the parameters of each scheme to better satisfy the requirements in real deployment of mobile sensor networks. The parameters for tuning are listed in Table 1.

Table 1 Parameters and packets in node scheduling schemes

Scheme	Parameters	Packet	Packet length (Byte)
GAF	Gid size, discovery time	DISCOVERY_MSG BROADCAST_MSG	4 2
RIS	Slot time, duty cycle	None	None
P-SWIM	Slot time, duty cycle	REQUEST_MSG RESPONSE_MSG WAKEUP_MSG PRUNE_MSG	2 6 2 2

We run the simulation ten rounds in each of which a mobile node performs 20 steps localization as it moves. The time interval between steps is 1 second. We record the average value as the final results.

Note that this work does not focus on improving the localization accuracy of a specific localization algorithm. Instead, we are interested in the following three performance metrics:

- Failed Steps (FS): the number of steps in which localization fails in each round.
- Seed Fidelity (SF): defined as the ratio of the number of seeds that participate in a localization process, to the number of seeds in the localization area, including the sleeping seeds and active seeds.
- Power Efficiency (PE): evaluated by the means that given a level of localization accuracy, the network overall power consumption caused by communication, sleep and activity. The corresponding control packets of each scheme are also listed in Table 1.

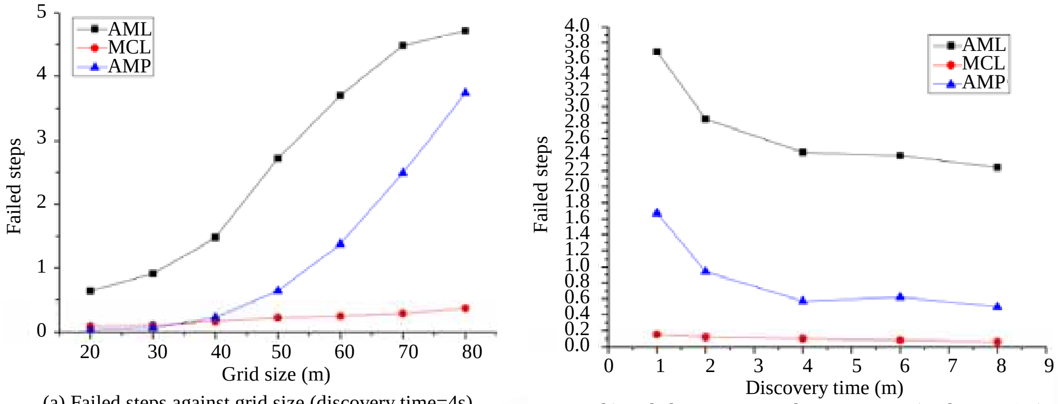
4.2 Failed steps and seed fidelity

In this part of experiment we study the Failed Steps and Seed Fidelity of localization algorithms based on three node scheduling schemes. When tuning parameters in each scheme, the $v_{\max}$ of unknown nodes is set to 10 m/s.

4.2.1 Parameter Tuning in GAF

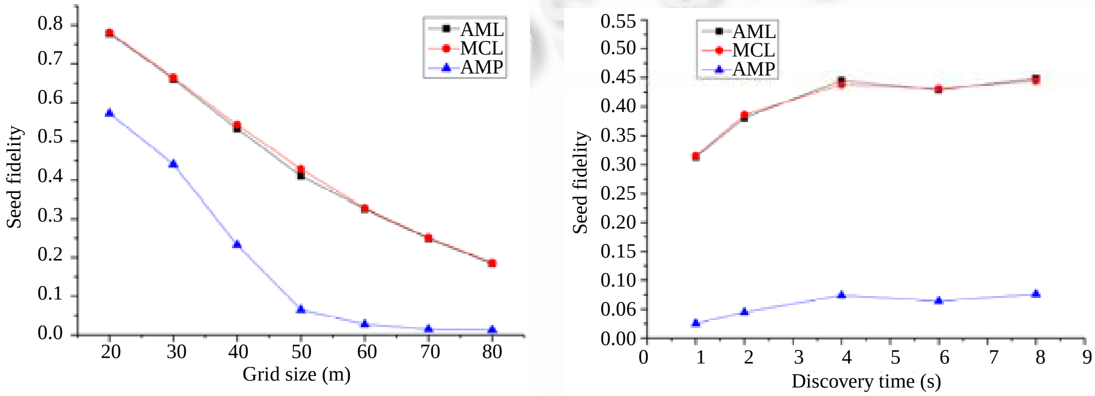
Figure 5 shows FS and SF of the network in the GAF scheme when changing its two parameters, the grid size and the discovery time. First, given a discovery time of 4s, we increase the grid size from 20m to 80m and obtain the results in Fig.5(a) and (c). The steps of failed localization in the three algorithms increase as the grid size increases. The reason is that, in GAF scheme, the number of active seeds in a single grid is fixed, then, the larger the grid is, the smaller the number of grids is, which cuts down the total number of active seeds in networks. For the same reason, larger grid size leads to a lower level of h-hop connectivity which deteriorates the seed fidelity.

Then, we evaluate the two metrics against discovery time given the grid size (50m). From Fig.5(b) and (d),



(a) Failed steps against grid size (discovery time=4s)

(b) Failed steps against discovery time (grid size=50m)



(c) Seed fidelity against grid size (discovery time=4)

(d) Seed fidelity against discovery time (grid size=50m)

Fig.5 Failed steps and seed fidelity in GAF scheme

when changing the discovery time from 2s to 10s, the number of failed steps drops and the seed fidelity has a slight increase, which means the improvement of localization quality. This is because longer discovery time will make the distribution of active seeds more frequent to change during any given period of time. Thus, mobile nodes are more likely to be tracked by widely distributed active seeds.

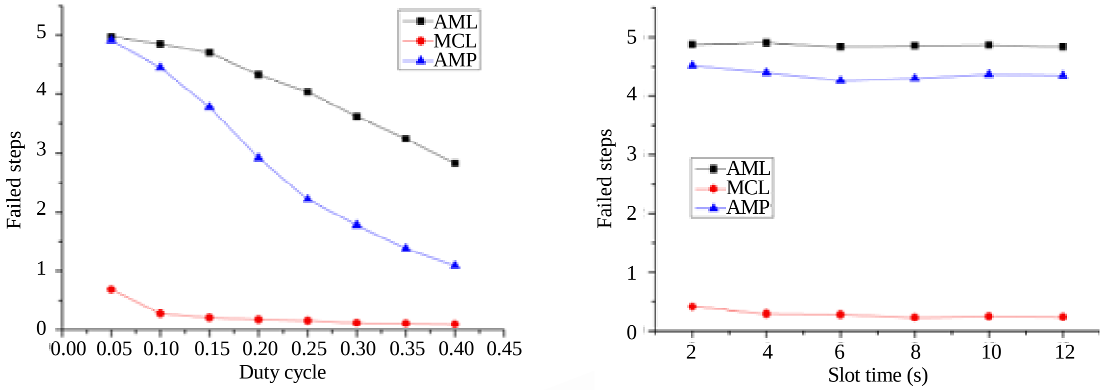
4.2.2 Parameter tuning in RIS

The performance results for RIS scheme are shown in Fig.6. First, we are given the slot time of 10s. From Fig.6(a) and (c), we observe that, as the duty cycle increases from 0.05 to 0.4, the number of failed steps drops and the seed fidelity rises gradually, which occurs because the number of active seeds is proportional to the duty cycle.

Then, in another experiment as illustrated in Fig.6(b) and (d), given the duty cycle of 0.1, the Failed steps and Seed fidelity of all the three localization algorithms remain almost steady as the time slot increases from 4s to 12s. The reason is that the length of time slot impacts the localization quality in two ways. On the one hand, longer slot time leads to increase of the time when a seed stays active, and hence increases the number of active seeds; On the other hand, the increased time slot makes the distribution of active seeds in network steady, which reduces the possibility of successful localization.

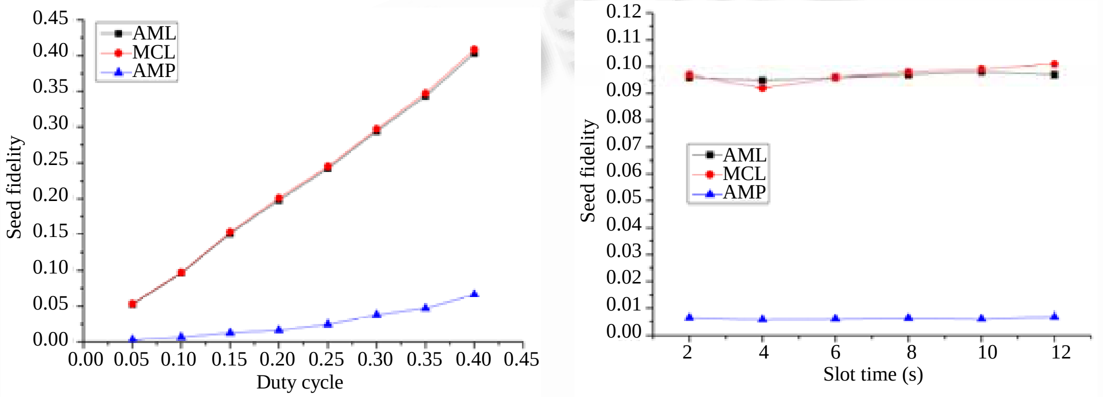
4.2.3 Parameter tuning in P-SWIM

Figure 7 illustrates the performance of localization algorithms based on P-SWIM scheme. As shown in Fig.7(a), given the slot time of 10s, the number of Failed steps drops dramatically as the duty cycle increases from



(a) Failed steps against duty cycle (slot time=10s)

(b) Failed steps against slot time (duty cycle=0.1)



(c) Seed fidelity against duty cycle (slot time=10s)

(d) Seed fidelity against slot time (duty cycle=0.1)

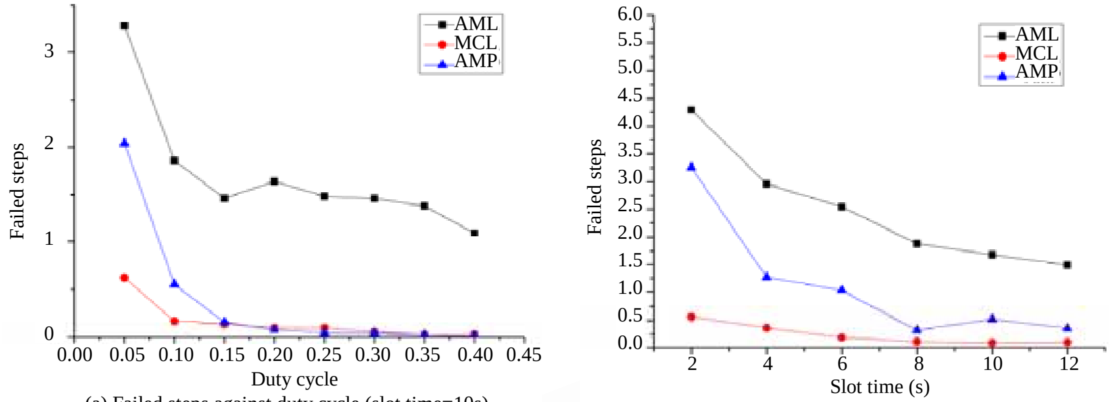
Fig.6 Failed steps and seed fidelity in RIS scheme

0.05 to 0.4. Meanwhile, as illustrated in Fig.7(c), Seed Fidelity has a sharp rise when the duty cycle increases. This happens partially because the number of active seeds increases. Moreover, the most direct reason is that P-SWIM proactively notifies the neighboring seeds to stay active and maintains a desired level of *h*-hop connectivity around the mobile node. Hence, the localization quality is improved in P-SWIM comparing with RIS under the same parameter settings.

We then evaluate P-SWIM scheme when varying the time slot at a given duty cycle. As shown in Fig.7(b) and (d), all the three algorithms achieve better localization quality when the time slot extends. The Dual-side Effect of slot time on localization quality no longer holds true in P-SWIM. For longer a slot time, an active seed has more time to be proactively woken up and wait for the approaching mobile nodes. Therefore, more seeds participate localization process, which betters the localization quality.

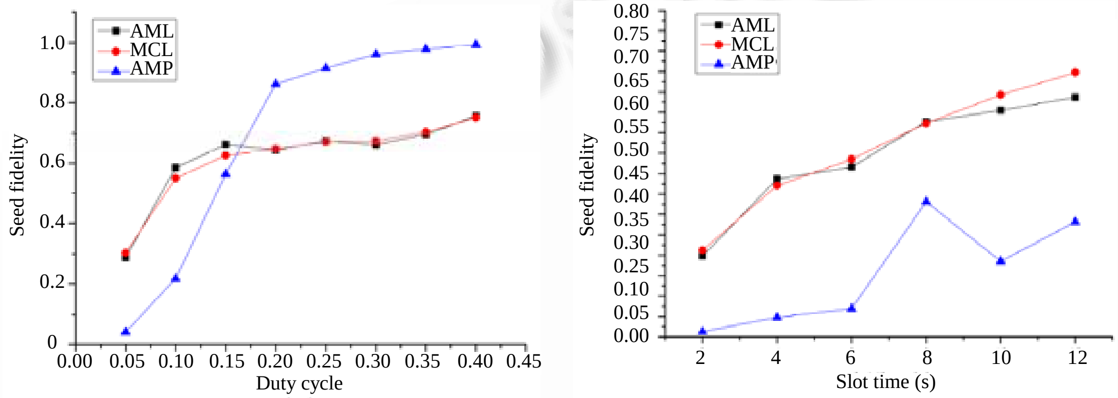
4.2.4 Comparing across the three schemes

Through these experiments we can conclude that all the three schemes can achieve better localization quality based on P-SWIM scheme than that based on GAF and RIS schemes. This happens because P-SWIM can dynamically maintain the desired *h*-hop connectivity through proactive sleep and wakeup scheduling of the exact seeds. Furthermore, by tuning the parameters of the other schemes, a given localization quality can also be obtained. Hence, our simulation results provide guidelines for handling the trade-off between localization quality and cost in real applications with different requirements.



(a) Failed steps against duty cycle (slot time=10s)

(b) Failed steps against slot time (duty cycle=0.1)



(c) Seed fidelity against duty cycle (slot time=10s)

(d) Seed fidelity against slot time (duty cycle=0.1)

Fig.7 Failed steps and seed fidelity in P-SWIM scheme

4.2.5 Comparing across all the three localization algorithms

Based on the experiments above, we can draw comparisons across all the three localization algorithms when they run on top of different node scheduling schemes. As for Failed Steps, the MCL algorithm (h=2) performs best among the three algorithms, while AML (h=1) and AMP (h>2) fail to localize in more steps. This can be expected because both AML and AMP adopt multilateration approach to calculating the position. Hence the possibility of successful multilateration is directly related to the number of active seeds that participate in localization process. Furthermore, AMP needs neighboring seeds in the range of multi-hops (h>2), so the one-hop algorithm AML can achieve better Failed Steps performance than AMP does. While, the predication-and-filtering-based algorithm MCL estimates the location by iterative sampling of the two-hop away neighbors, therefore, MCL can alleviate the impact of inadequate seeds on Failed Steps by enhancing the sample times.

When considering Seed fidelity, AMP (h>2) performs the worst, while MCL (h=2) and AML (h=1) achieve almost equal performance. This happens because the multi-hop neighboring seeds are involved into localization process in AMP, which results in the largest sum of active seeds and sleeping seeds. As the localization algorithms with limited hops, MCL and AML require less seeds from the neighborhood than AMP.

4.3 Power efficiency

Proactive wakeup and sleep in P-SWIM is based on communication across the network, hence, communication overhead has to be considered to evaluate the power efficiency of schemes. The RIS and GAF schemes with less communication cost, however, might consume more energy to achieve desired localization accuracy because more seeds need to be put into activity. In this evaluation, we measure the average power consumption only caused by node scheduling schemes when a given localization accuracy is achieved. Since the communication and computation costs of the specific localization algorithm are dependent on the intrinsic mechanism it uses, our evaluation does not include this part of consumption. In addition, our energy model is based on the measurements of Crossbow MICAz[17], 30μW for sleeping, 33mW for idle (active), 45mW for transmitting and 38mW for receiving.

Since the power consumption we evaluate is independent of localization algorithms, we use MCL algorithm to compare the power efficiency of each scheme. Given four levels of localization error *e*, we can obtain four groups of equivalent parameters (as listed in Table 2) through previous experiments. Then, the average overhead of each scheme in one round can be calculated as shown in Fig.8.

Table 2 Equivalent parameter groups of node scheduling schemes

Scheme	Parameters	Group1 (e=0.2±0.03)	Group2 (e=0.4±0.03)	Group3 (e=0.6±0.03)	Group4 (e=0.8±0.03)
GAF	Grid size	10m	20m	40m	50m
	Discovery time	10s	10s	10s	10s
RIS	Slot time	10s	10s	10s	10s
	Duty cycle	1	0.6	0.4	0.3
P-SWIM	Slot time	10s	10s	10s	10s
	Duty cycle	0.6	0.25	0.15	0.1

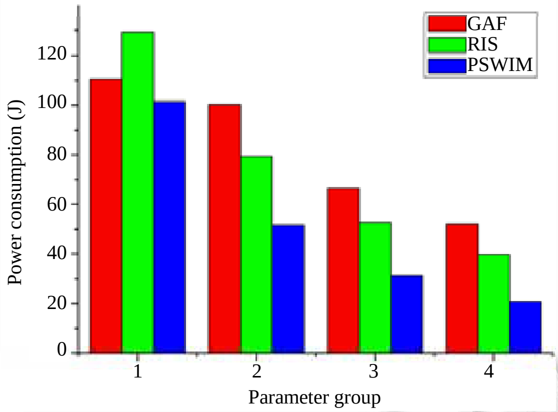


Fig.8 Power efficiency of the schemes

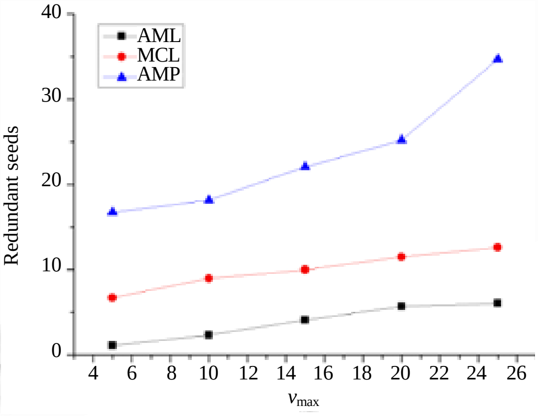


Fig.9 Redundant seeds in P-SWIM

First, we observe the power efficiency when the first parameter group is used, because the duty cycle of RIS scheme is 1, which means all seeds stay active all the time. It can be expected that the full duty RIS scheme consumes the highest energy overhead because the all-time-active seeds take much unnecessary power. When GAF and P-SWIM adopt the first parameter group, energy consumption drops 14.5% and 21.6% respectively with regard to the full duty RIS scheme.

Then, in the other three parameter groups, P-SWIM achieves the best power efficiency, saving 60.2% and 47.6% energy consumption at most comparing with GAF and RIS respectively. GAF has the poorest power saving performance because much energy is consumed during the process of grid head node competition. RIS is more power conservative than GAF since no extra communication exists.

4.4 Node velocity

In the last experiment, we evaluate the impact of the maximum velocity $v_{\max}$ of the mobile nodes on P-SWIM. The formula (2) indicates that, in P-SWIM, the node predicts its location using $v_{\max}$. Therefore, $v_{\max}$ may cause that some seeds which are not in the next localization area are proactively woken up by P-SWIM. These redundant seeds do not participate in the localization area and consume much energy due to full duty working.

Given slot time (10s) and duty cycle (0.2), when we increase $v_{\max}$ from 5m/s to 25m/s, the redundant seeds in each localization process can be obtained as shown in Fig 9. It can be observed that the number of redundant seeds rise as $v_{\max}$ increases. Redundant seeds may lead to decrease of the power efficiency. A more accurate movement predication technique to minimize the unnecessary energy caused by redundant seeds is left as future work.

5 Conclusions

In this paper we proposed a novel power management and connectivity maintenance scheme, P-SWIM. P-SWIM is designed specifically for mobile localization algorithms which need to maintain desirable network connectivity not only in the local area but also in the multi-hop area. P-SWIM can be integrated with mission-critical applications such as target tracking and emergency navigation. Simulation results indicate that localization algorithms that use P-SWIM can achieve better localization quality and power efficiency than those that adopts RIS and GAF. Furthermore, through extensive experiments, a detail study of the impact of node scheduling scheme on localization quality of localization algorithms is presented to provide efficient deployment guidelines for mobile sensor networks.

Acknowledgement The authors wish to thank the members of Research05 Group of Computer Application Institute, Beihang University for their many technical suggestions.

References:

[1] Niculescu D. Positioning in ad hoc sensor networks. IEEE Network, 2004,18(4):24−29.

[2] Ren FY, Huang HN, Lin C. Wireless sensor networks. Journal of Software, 2003,14(7):1282−1291 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/14/1282.htm

[3] Carle J, Simplot-Ryl D. Energy-Efficient area monitoring for sensor networks. Computer, 2004,37(2):40−46.

[4] Zhang P, Sadler CM, Lyon SA, Martonosi M. Hardware design experiences in ZebraNet. In: Proc. of the ACM Sensys. New York: ACM Press, 2004. 227−238.

[5] Tseng Yc, Pan MS, Tsai YY. Wireless sensor networks for emergency navigation. Computer, 2006,39(7):55−62.

[6] Langendoen K, Reijers N. Distributed localization in wireless sensor networks: A quantitative comparison. Computer Networks, 2003,43(4):499−518.

[7] Baggio A, Langendoen K. Monte Carlo localization for mobile wireless sensor networks. Ad Hoc Networks, 2008,6(5):718−733.

[8] Gui C, Mohapatra P. Power conservation and quality of surveillance in target tracking sensor networks. In: Proc. of the ACM MobiCom. New York: ACM Press, 2004. 129−143.

[9] Heinzelman WR, Chandrakasan A, Balakrishnan H. Energy-Efficient communication protocol for wireless microsensor networks. In: Proc. of the IEEE HICSS. Los Alamitos: IEEE, 2000. 223.

[10] Cerpa A, Estrin D. ASCENT: Adaptive self-configuring sensor networks topologies. IEEE Trans. on Mobile Computing, 2004,3(3): 272−285.

[11] Xu Y, Heidemann J, Estrin D. Geography-Informed energy conservation for ad hoc routing. In: Proc. of the ACM MobiCom. New York: ACM Press, 2001. 70−84.

[12] Leonard Kleinrock, Silvester J. Optimum transmission radii for packet radio networks or why six is a magic number. In: Proc. of IEEE National Telecommunications Conf. 1978. 4.3.1−4.3.5.

[13] Hu L, Evans D. Localization for mobile sensor networks. In: Proc. of the ACM MobiCom. ACM, 2004. 45−57.

[14] Camp T, Boleng J, Davies V. A survey of mobility models for ad hoc network research. Wireless Communications and Mobile Computing, 2002,2(5):483−502.

[15] Savvides A, Han CC, Strivastava MB. Dynamic fine-grained localization in ad-hoc networks of sensors. In: Proc. of the ACM MobiCom. New York: ACM Press, 2001. 166−179.

[16] Nagpal R, Shrobe H, Bachrach J. Organizing a global coordinate system from lcoal information on an ad hoc sensor network. In: Proc. of the ACM IPSN. Berlin: Springer-Verlag, 2003. 553.

[17] Crossbow. Micaz: Wireless measurement system. http://www.xbow.com.

附中文参考文献:

[2] 任丰原,黄海宁,林闯.无线传感器网络.软件学报,2003,14(7):1282−1291. http://www.jos.org.cn/1000-9825/14/1282.htm

LIU Yu-Heng was born in 1980. He is a Ph.D. candidate at BeiHang University. His current research areas are localization and power management of sensor networks.

WU Jing was born in 1979. She received her Ph.D. degree of Computer Science in 2007. Her current research areas are mobile networking and data mining.

CHEN Zhen-Yong was born in 1974. He is an associate professor at BeiHang University. His research areas are multimedia sensor networks.

XIONG Zhang was born in 1956. He is currently a professor and doctor supervisor at BeiHang University and a CCF senior member. His research areas are sensor networks and multimedia processing.
