國立臺灣大學工學院工程科學與海洋工程學系

碩士論文

Department of Engineering Science and Ocean Engineering

College of Engineering

National Taiwan University

Master Thesis

Maximizing Network Lifetime with Adaptive Beacon Duty Scheduling

延長網路存活時間之適應性錨節點睡眠排程

周元超

Yuan-Chao Chou

指導教授：丁肇隆 博士、張瑞益 博士

Advisors: Dr. Chao-Lung Ting and Dr. Ray-I Chang

中華民國 101 年 7 月

July, 2012

誌謝

時光飛逝，兩年多的碩士生涯即將劃上句點。本論文的完成要感謝許多人的協助與幫忙，首先最感謝我的指導教授丁肇隆博士及張瑞益博士。丁老師與張老師帶領著懵懵懂懂的我進入了無線感測網路的專業領域，且總是不厭其煩地給予許多建議與教誨；從兩位老師身上我學習到了做研究應有的精神，對學問的嚴謹態度更是我學習的最佳典範，無論在生活上或是研究上皆受益良多，在此對兩位老師致上由衷的謝意。另外還要感謝口試委員王家輝博士以及林正偉博士對於論文的指導與建議，使得此論文能夠更臻完善。

感謝研究夥伴們在這兩年間的照顧與陪伴。感謝學長姐榮棄、忠原、育正、孟翰、建彰、佳穎、任芯總能在我迷惘時適時地替我解惑；感謝一起打拼的戰友承鴻、志永、孃瑩、景陽、肇普、昱瀚、仁峰、柏沇及學弟妹涵仁、億鑫、之盈、東輝、禹豪在研究上的切磋砥礪及課餘的嘻笑打鬧，真的很開心能夠認識你們。

感謝弈勳、光宏、詠翔、書蔚、博範、彥廷，碩士生涯中有你們的相挺與陪伴，為這兩年的研究生活提供源源不絕的活力。女朋友雅鈴在背後的默默支持更是我前進的動力，沒有雅鈴的體諒、包容，相信這兩年時光描繪的會是幅很煎熬的光景。

感謝我最親愛的家人們，李春菊女士與弟弟周華展，謝謝你們給我的支持與包容，讓我能順利完成碩士學位。最後，謹以此篇論文獻給所有關心我的人。

周元超　謹致於

國立台灣大學　工科所　資訊與網路實驗室

中華民國　101 年　7 月

中文摘要

定位技術已在無線感測器網路(WSNs)中被廣泛應用於找出節點之未知位置。一般在執行定位技術的過程中，會大量佈置錨節點(位置已知之節點)來協助推算其他節點之位置；然而，同時啟動所有的錨節點並不能明顯地增加定位準確率，反而會增加額外的能量成本與頻寬成本。在這樣的情況下，通常只需要同時啟動一部分的錨節點就能達到準確率的要求，因此為了減少成本並延長網路存活時間，本論文提出了可使錨節點自己安排工作周期的 Adaptive Beacon Duty Scheduling (ABDS)演算法。ABDS 會在線上即時量測錨節點位置之效益(在此位置啟動錨節點後可能帶給定位準確率多少助益)，並且根據此量測結果挑出對定位準確率最有效益的那些錨節點以啟動之，以將同時啟動的錨節點數量最小化。由於在前人的相關研究中，並未實際去量測不同錨節點位置的不同效益，因此 ABDS 可更佳地適應充滿無法預測之雜訊的真實環境。此外，為了在 ABDS 中精確地量測錨節點位置之效益，我們觀察到錨節點對其覆蓋範圍產生的定位效能改善量事實上是非均勻分布的，並提出了尚未被討論過的 Distribution-Adapted Grid (DAG)量測法以適應此現象。與前人的方法相比，使用了 DAG 量測法的 ABDS 可以減少 10%的錨節點使用量，並且延長 54%的存活時間。

關鍵字：無線感測器網路、定位、錨節點位置、工作排程、睡眠排程。

ABSTRACT

Within typical localization processes in wireless sensor networks (WSNs), beacon nodes which know their locations will broadcast information for localizing an unknown location. Although beacon nodes are massively deployed in the terrain, only a fraction of the beacon nodes are required to be active for satisfying accuracy requirement. Too many active beacon nodes may bring the system with little improvement on localization accuracy but waste of both costs of energy and bandwidth. To reduce the costs and prolong the system lifetime, we propose the Adaptive Beacon Duty Scheduling (ABDS) algorithm that can self-configure beacon duty. ABDS can turn on only the minimum set of beacon nodes in a same time according to the online-measured effectiveness of beacon locations (the effect of activating a beacon node at the location for improving localization performance), which is not considered in previous methods. Moreover, to precisely measure the effectiveness of beacon locations in ABDS, we need to realize the fact that a beacon node actually contributes non-uniformly distributed impact within its coverage. This Distribution-Adapted Grid (DAG) measurement that can adapt the non-uniformly distributed impact was not discussed in previous methods. Compared to the previous methods, ABDS with the usage of DAG measurement can reduce 10% beacon usage and provide 54% longer lifetime.

Keywords: Wireless sensor networks, localization, beacon location, duty scheduling, sleep scheduling.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1 Introduction

1.1 Motivation

In wireless sensor networks (WSNs), both network operations and most application level tasks require the help of localization algorithms to acquire knowledge of the physical locations of devices [1], [2], [3], such as event detection [4], routing [5] and coverage [6]. These localization algorithms usually make use of beacon (anchor) nodes, whose locations are known prior to perform the localization algorithm, for the purpose of estimating unknown locations of other sensor nodes [7], [8], [9]. Locations are then computed by proximity-based approaches, range-based approaches, or angle-based approaches [3] according to the information gathered from beacon nodes. The accuracy of location estimations may increase as a function of the number of covered beacon nodes. However, deploying too many beacon nodes brings costs of bandwidth and excessive power consumption with limited improvement on localization accuracy, and thus leads to shortened system lifetime. Therefore, a scheduling scheme which considers effectiveness of beacon locations (the effect of activating a beacon node at the location for improving localization performance) to schedule their duty cycle is useful for the purpose of increasing system lifetime while maintaining required localization accuracy. Most of existing scheduling algorithms are designed for maintaining sensing coverage or connectivity [10], [11], [12], [13]. Only a few papers are proposed to schedule beacon duty cycle, such as [14], [15], [16], [17]. However, most of them did not consider the impact of beacon node deployment, which has been identified as a significant factor that has strong influence on localization accuracy [18]. Moreover, they are designed to control the density of active beacon nodes. It is not friendly for users (in this thesis, “users” are the people that apply a localization algorithm and a beacon duty scheduling algorithm to construct their localization system) to set a desired control point of localization accuracy. In [15], the authors propose a scheduling algorithm that considers theoretical location error estimation of beacon nodes, which is not able to adapt noisy environment in real world. In addition, it is specifically designed for distance-based localization and cannot be applied to other kinds of localization algorithm.

1.2 Contributions

As demonstrated in [18], a good beacon deployment can meet the localization performance requirement with fewer active beacon nodes. The costs of power and bandwidth can be reduced and the system lifetime can be prolonged if only fewer beacon nodes are active to strobe at the same time. In real world, noise is inevitable and unpredictable in the localization systems. Therefore, a scheduling method that can dynamically adapt the noisy environment is necessary for both the purposes of selecting minimum active set of beacon nodes to reserve energy and controlling localization accuracy to user defined control point.

In this thesis, we propose Adaptive Beacon Duty Scheduling (ABDS) algorithm which can prolong system lifetime while maintaining required localization accuracy. The fundamental limitation of previous approaches is that they basically miss the actual impact of beacon location in real world. They do not take into account effectiveness of beacon location on reducing localization error that cannot be predicted a priori. For achieving efficient scheduling, only the minimum set of beacon nodes with best effect on localization are expected to be activated in the same time for reducing power consumption and packet traffic. Therefore, an improved measurement to the effectiveness of beacon locations that can more precisely dig out beacon nodes with better effect on localization is also developed in this thesis. Empirical manners that can adapt to terrain conditions are favored in real world because it is hard to build a model to fit a certain environment. To our knowledge, little literatures have been published into this area, such as [18], [19]. In [18], Max and Grid take localization error and regional cumulative localization error as the measurements to the effectiveness of beacon locations respectively. In [19], Greedy addresses the beacon placement as a set cover problem, and takes coverage degree as the measurement to the effectiveness of beacon locations. We observed that after activating an additional beacon node, the impact on localization performance does not uniformly distribute in its coverage. The region closer to the beacon has better chance to reduce localization error. This phenomenon was not discussed in previous studies and should be overcome to get a more precise measurement to the effectiveness of beacon locations. Consequently, we improve the measurement in Grid and propose the Distribution-Adapted Grid (DAG) measurement for designing minimum active beacon nodes deployment in beacon duty scheduling. The scheduling algorithm meets following design goals.

- Maximize system lifetime
- Distributed
- Adaptive to noisy environment in real world
- Can be applied to any types of localization algorithm
- User defined localization accuracy requirement

1.3 Thesis Organization

The remainder of this thesis is organized as follows. Chapter 2 discusses related researches about beacon duty scheduling, measuring effectiveness of beacon locations, and localization algorithm. The DAG measurement is developed in Chapter 3. In Chapter 4, the ABDS algorithm is developed. Chapter 5 shows the performance evaluation of proposed methods. Finally, the conclusions are made in Chapter 6.

Chapter 2 Related Works

In this section, related works are presented in three parts. Section 2.1 introduces existing algorithms for scheduling duty cycle of beacon nodes. Two referred works on the measurement to the effectiveness of beacon locations are introduced in Section 2.2. To explain the phenomenon of non-uniformly distributed impact produced by beacons on localization, two localization algorithms are introduced in Section 2.3.

2.1 Beacon Duty Scheduling Algorithms

In [16], STROBE algorithm is proposed to schedule beacon duty by tuning the operation beacon density. STROBE adopts a scheduled rendezvous scheme that activates all beacon nodes up at the same time. Beacon nodes then exchange the information of activation distribution with their neighbors. If the local density of active beacon nodes is under a user defined threshold, the beacon node remains active for transmitting beacon signal to maintain the active density requirement. If the density exceeds the threshold, the beacon node takes a probability of excess density (e.g., the threshold is 7 and the density for now is 9, then the probability is (9-7)/7) to sleep for the purpose of reducing density to the threshold. The state transition of STROBE is shown in Fig. 2.1.

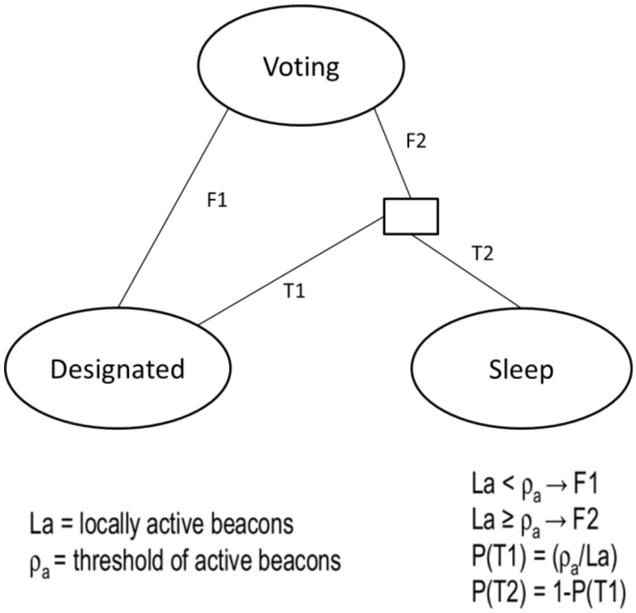


Fig. 2.1. The state transition of STROBE.

In [17], in order to spread energy consumption over beacon nodes for loading balance, the authors proposed E-STROBE, which extends STROBE to consider the ratio of remaining energy as a factor in making decision to active or sleep. E-STROBE takes a probability of the ratio of remaining energy before entering into the active state. Fig. 2.2 illustrates the state transition of E-STROBE.

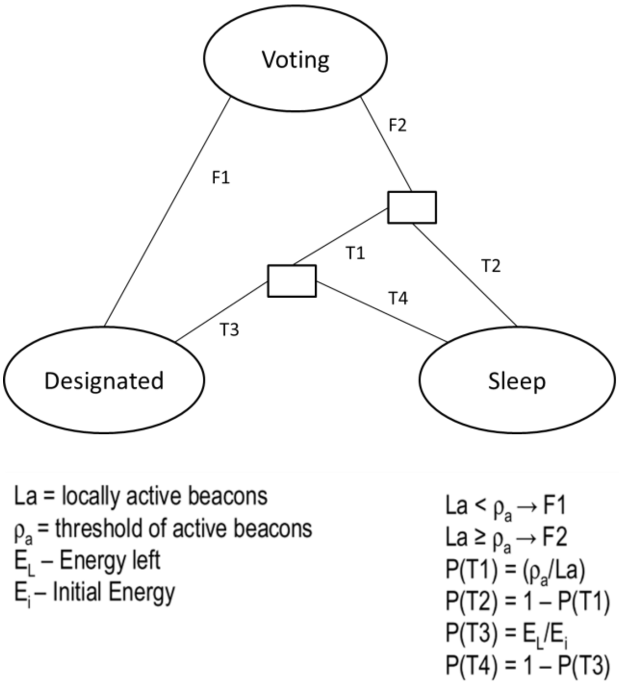


Fig. 2.2. The state transition of E-STROBE.

In [14], the authors proposed a beacon duty scheduling algorithm inspired by Span [10]. The algorithm fuses the parameters of active density and remaining energy into a delay time. After a beaconing or sleeping period, each beacon node transits the state to calculate the delay time. When the delay timer expires, the beacon node checks the active density in its neighborhood. If the density does not satisfy a user defined threshold, the beacon node is activated to transmit beacon signal, otherwise it is turned off to sleep for reserving energy.

In [15], the authors of [14] improve their scheduling algorithm by considering the theoretical location error estimation [20] rather than the density of beacon nodes as the design parameter. The location error estimation, which is specifically designed for range-based localization algorithms, makes use of the CRLB that places a lower bound on the variance of unbiased estimators [21] and was derived for position estimation in [22]. If the estimated location error is above a user defined threshold, the beacon node is activated to transmit beacon signal, otherwise it is turned off to sleep.

2.2 Related Works on the Measurement of the Effectiveness to Beacon Locations

In the Max algorithm [18], localization error (distance between estimated location and actual location) is taken to be the measurement to evaluate the effectiveness of a beacon location. The idea is that a location with larger localization error has larger room to improve localization performance and thus larger benefit gained by placing a beacon node on this location. By this measurement, beacons are placed in an incremental manner. Every time the effectiveness for localization performance at each point on the terrain is measured, and then an additional beacon is placed on the point with highest localization error. The steps to incrementally place beacons are described as following.

1. The environment terrain is divided into $Step \times Step$ squares.
2. Measure the effectiveness for reducing localization error at each point in the terrain that corresponds to a square corner.
3. Add an additional beacon at the point that has the highest measurement value among all points.

Fig. 2.3 illustrates the Max algorithm.

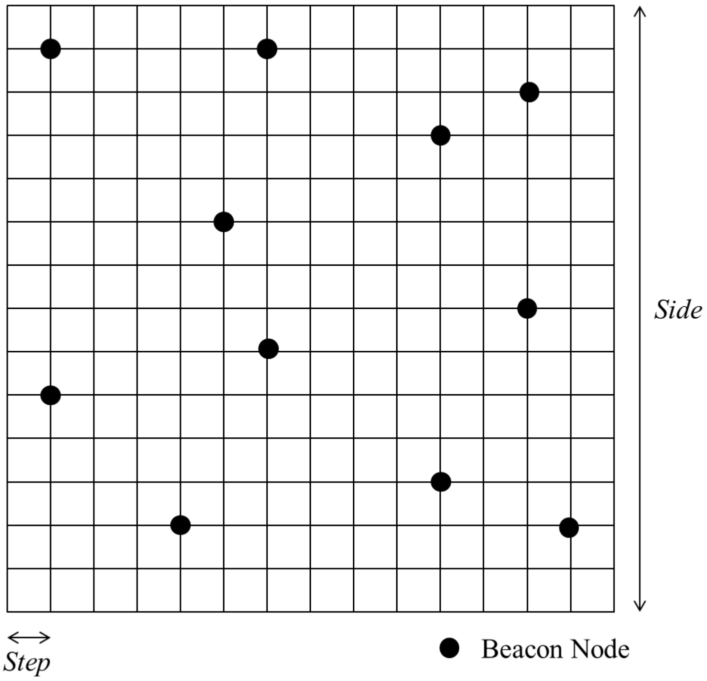


Fig. 2.3. Illustration of the Max algorithm.

There is an assumption that locations with high localization error are spatially correlated in Max. However, a point may get very high localization error while the localization error at other points close to it remains low, and that adding an additional beacon affects the localization error on its nearby region not just the point where it is placed. Based on these observations, the authors proposed the Grid algorithm [18]. In Grid, a 2-dimensional rectangular sliding window called "*grid*" with side length *gridSide=2R* (*R* is the ideal radius of communication range) is set up and the localization errors that lie in the grid are summed up to be the measurement at the center of this grid. The Grid measurement is taken to incrementally place beacons (by the steps described in **Max beacon placement algorithm**) in the Grid algorithm, as illustrated in Fig. 2.4

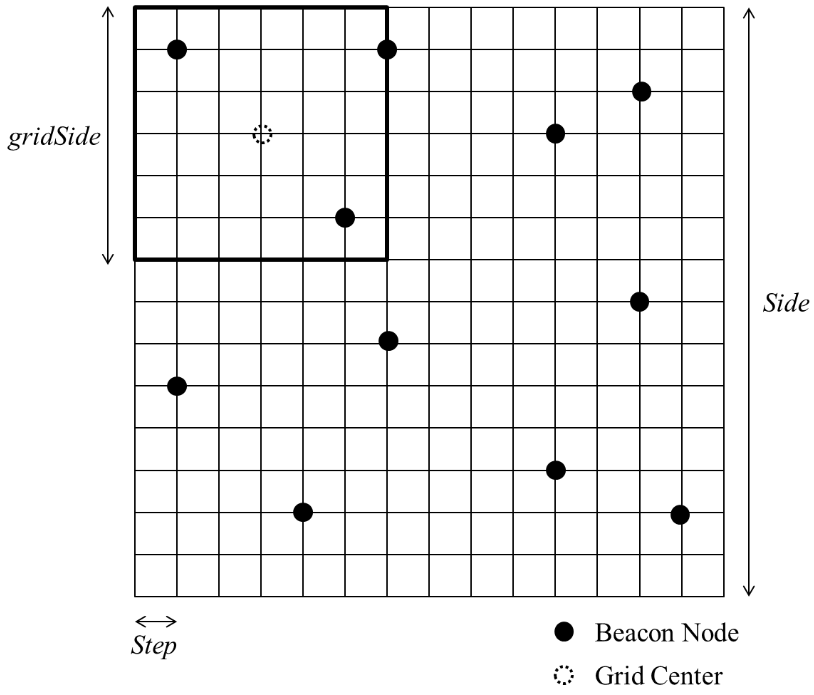


Fig. 2.4. Illustration of the Grid algorithm.

It is a proper concept in Grid to cumulate regional localization error as the measurement to decide beacon locations. However, the Grid measurement overestimates the ability of a beacon to improve the localization performance on its nearby region in such a way that beacon node resources are misspent. The observation of this phenomenon and the proposed measurement to more precisely estimate regional localization impact are described in Chapter 3.

2.3 Localization Algorithms

Connectivity based localization. Connectivity based localization [8] adopts beacon nodes to periodically transmit beacon signal with a time period *T*. Nodes with unknown location (blind node) listen for a period $t >> T$ to evaluate connectivity. A beacon node is considered as connected on the understanding that the packet-receive-ratio from the beacon node in the period *t* exceeds a threshold. After appraising the connectivity of beacon nodes, the blind node estimates its location (*X*X_{est}, Y_{est}) as the centroid of the locations of all connected beacon nodes with the centroid formula shown in Eq. (2.1).

$$\left(X_{est}, Y_{est}\right) = \left(\frac{X_1 + \ldots + X_N}{N} \ \frac{Y_1 + \ldots + Y_N}{N}\right) \tag{2.1}$$

Pattern matching localization. Based on taking received signal strength indices (RSSIs) from beacons as the feature vector of a location, pattern matching localization estimates an unknown location with similar features [7]. This localization algorithm consists of two phases, namely training phase and locating phase. In the training phase, RSSIs on training locations are recorded with the location coordinates, and the collected data are used to build a localization model. In the locating phase, a blind node collects the RSSIs from beacons to be the feature vector of its location, and then inputs the feature vector into the localization model to estimate the unknown location.

Chapter 3 Distribution-Adapted Grid Measurement

In order to select out a minimum set of active beacon nodes with best effect on localization to satisfy the requirement of localization accuracy, a precise measurement to the effectiveness of beacon locations is necessary for efficient beacon duty scheduling. Researchers have proposed methods for the measurement in beacon placement algorithms [17], [18]. However, the Grid measurement has a defect and solutions have not been proposed, hence for better predicting the effectiveness of beacon locations, an improved measurement (i.e., Distribution-Adapted Grid or DAG) is proposed in this chapter.

3.1 The Problem of Predicting the Effectiveness of a Beacon Location

This thesis addresses the problem of predicting the effectiveness of a beacon location as follows. A deployment B_a^n that consists of n active beacon nodes $\{b_a^1,...,b_a^n\}$ with Cartesian coordinates $\{\mathbf{z}_a^1=\left(x_1,y_1\right)\in\mathbb{R}^2,...,\mathbf{z}_a^n\}$ that know their locations a priori and have an ideal communication radius R exists in a two-dimensional squared terrain $T=\left[0,side\right]\times\left[0,side\right]\subset\mathbb{R}^2$ divided into $step\times step$ squares as Fig. 2.3. We denote an active beacon node b_a^i located at $\mathbf{z}_a^i$ by $b_a^i(\mathbf{z}_a^i)$ and the location $\mathbf{z}_a^i$ of an active beacon node b_a^i by $\mathbf{z}_a^i(b_a^i)$. The localization error ε at a location $\mathbf{z}=\left(x,y\right)\in T$ is the Euclidean distance between $\mathbf{z}$ and the estimated location $\hat{\mathbf{z}}=\left(x,y\right)\in T$ at it, i.e. $\varepsilon(\mathbf{z},B_a^n)=d_{\mathbf{z},\hat{\mathbf{z}}}=\left\|\mathbf{z}-\hat{\mathbf{z}}\right\|=\sqrt{(x-x)^2+(y-y)^2}$. The mean localization error ε_{mean} on T with a certain active beacon deployment B_a^n is

$$\varepsilon_{mean}(B_a^n) = \frac{\sum_{x=0}^{side/step} \sum_{y=0}^{side/step} \varepsilon\left((x,y), B_a^n\right)}{(\frac{side}{step}+1)^2} . \tag{3.1}$$

A measurement to the effectiveness of beacon locations $\psi(\mathbf{z}, B_a^n)$ takes the active beacon deployment and measurable information of the location (e.g., measured RSSIs, measured distances, packet receiving ratios, etc.) as inputs to predict a location's effectiveness for activating a beacon node to reduce the mean localization error on the terrain. That is, for a perfect measurement $\psi(\mathbf{z}, B_a^n)$, given an existing beacon deployment B_a^n and $B_a^{n+1} = \left\{B_a^n, b_a^{n+1}\left(\mathbf{z}_a^{n+1}\right)\right\}$,

$$\psi(\mathbf{z}_{n+1}, B_a^n) \propto \Delta\varepsilon_{mean} , \tag{3.2}$$

$$\text{where } \Delta\varepsilon_{mean} = \varepsilon_{mean}(B_a^n) - \varepsilon_{mean}(B_a^{n+1}) . \tag{3.3}$$

The precision of a $\psi(\mathbf{z}, B_a^n)$ can be evaluated by the reduction of mean localization error produced via activating a beacon node at the location $\mathbf{z}_i$ with maximum $\psi(\mathbf{z}_i, B_a^n)$, i.e.,

$$\Delta\varepsilon_{mean}(B_a^n, B_a^{n+1}) = \varepsilon_{mean}(B_a^n) - \varepsilon_{mean}(\left\{ B_a^n, b_a^{n+1}(\mathop{\arg\max}_{\left\{\mathbf{z}_i \in T - \mathbf{z}(B_a^n)\right\}} \psi(\mathbf{z}_i, B_a^n)) \right\}) . \tag{3.4}$$

3.2 Developing Distribution-Adapted Grid Measurement

According to the definition proposed in Section 3.1, the measurements used in Max and Grid can be written down as Eq. (3.5) and Eq. (3.6).

$$\psi_{Max}(\mathbf{z}, B_a^n) = \varepsilon(\mathbf{z}, B_a^n) . \tag{3.5}$$

$$\psi_{Grid}(\mathbf{z}, B_a^n) = \sum_{x=\mathbf{z}.x-\frac{R}{step}}^{\mathbf{z}.x+\frac{R}{step}} \sum_{y=\mathbf{z}.y-\frac{R}{step}}^{\mathbf{z}.y+\frac{R}{step}} \varepsilon\left((x,y), B_a^n\right) \tag{3.6}$$

Although the Grid measurement ψ_{Grid} can be used to design appropriate beacon locations when beacon density is low, it starts to mismeasure the effectiveness of beacon locations when the beacon density rises and the ability of a beacon to improve the localization performance on its nearby region decays. To consider regional localization error is reasonable. However, we observed that the improvement on localization performance in the coverage of a beacon node does not distribute uniformly. The location where an additional beacon is activated holds best effect of improving localization accuracy. This was observed from both connectivity based localization and pattern matching localization.

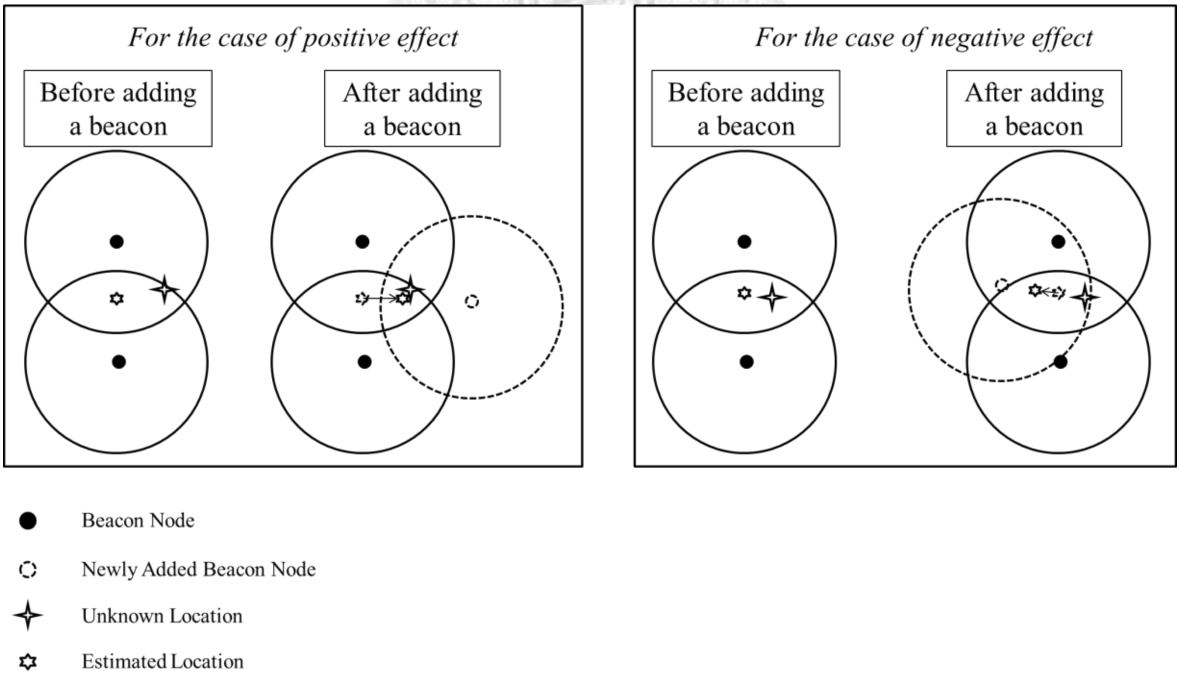


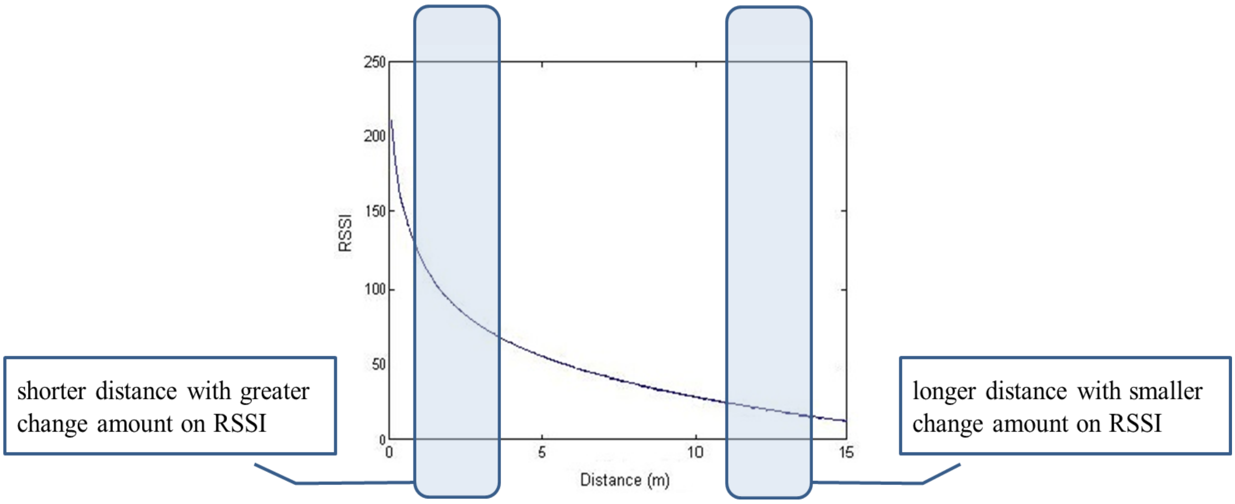
Fig. 3.1. Illustration of the positive effect and negative effect of adding a beacon node in connectivity based localization.

For the case of connectivity based localization, the estimated location of a blind node with unknown location is the centroid of connected beacons. Due to the fact that activating an additional beacon that covers it will pull the estimated location to be closer to the new beacon, both positive effect and negative effect on the nearby region of a new beacon are possible to occur, as illustrated in Fig. 3.1. Therefore, only the locations of added beacons always gains improvement on localization performance.

For the case of pattern matching localization, the information of locations hides in RSSI features, hence the pattern matching localization performs better with more dissimilar RSSI features. According to the path loss model without the noise term, signal strength decreases with the increase of distance in a logarithmic fashion [23], as described in Eq. (3.7).

$$\left[\frac{P_r(d_0)}{P_r(d)}\right]_{dB} = -10\beta\log(\frac{d}{d_0}), \tag{3.7}$$

where $P_r(d)$ is the mean received power at distance d, $P_r(d_0)$ is the received power at the close-in reference distance d_0 and β is the environment-dependent parameter. The closer the distance between transmitter and receiver is, the greater the change of RSSI is. Fig. 3.2 illustrates this phenomenon.



shorter distance with greater change amount on RSSI

longer distance with smaller change amount on RSSI

Fig. 3.2. Illustration of the impact of RSSI variation.

Even if noise exists in the path loss model in real world, a location closer to the beacon has better chance to get a more distinguishable feature and thus better performance of pattern matching localization.

To confirm our observations, Fig. 3.3 shows the distribution of averaged improvements on localization error in the coverage of beacons over iterations in an incremental beacon placement. The point where a beacon is placed holds best effect to improve localization performance as our inference. The ability of a beacon to improve the localization performance on its neighboring region decreases with the increase of the distance from the beacon.

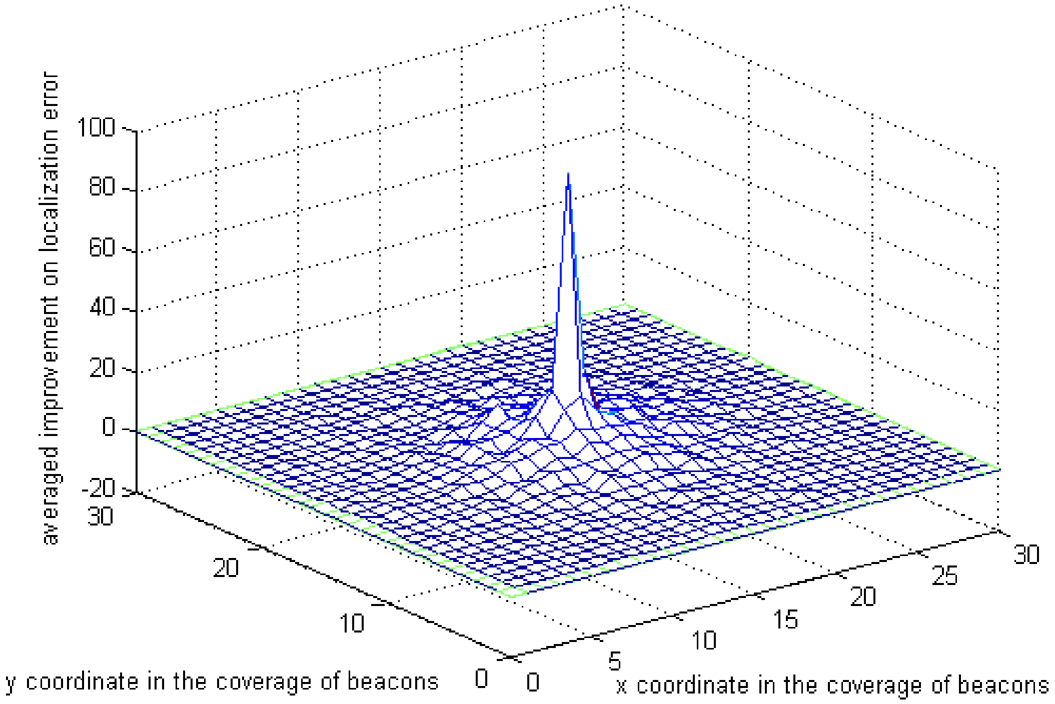


Fig. 3.3. The distribution of averaged improvements on localization error in the coverage of beacons.

To adapt this centralized-distributed improvement on localization and precisely measure the effectiveness of beacon locations, we suggest that the region closer to a beacon should take a heavier weight in computing the regional localization error over the beacon coverage. As a result, the Distribution-Adapted Grid (DAG) measurement that can more properly measure the effectiveness of a beacon location is proposed in this thesis. Two famous centralized distributions, namely Cauchy and Gaussian, are considered to weight the regional error. Fig. 3.4 shows the lateral view of the improvement distribution shown in Fig. 3.3 and the weight distributions generated by Cauchy and Gaussian.

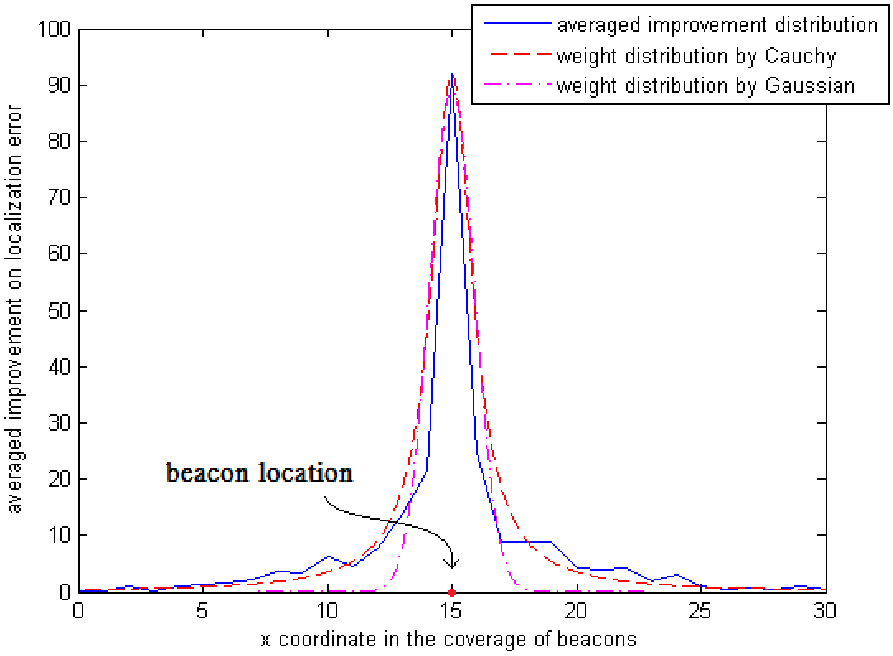


Fig. 3.4. The averaged improvement distribution and weight distributions.

The Cauchy distribution is more approximative to the improvement distribution, and thus adopted to generate a centralized weight distribution to adjust the cumulation process in regional localization error to fit the improvement distribution in beacon coverage. The three-parameter Cauchy distribution is defined by

$$f\left(x;x_0,\gamma,I\right)=I\left[\frac{\gamma^2}{(x-x_0)^2+\gamma^2}\right], \tag{3.8}$$

where γ is the scale parameter which specifies the half-width at half-maximum, I is the height of the peak, and x_0 is the location of the peak of this distribution. Because the purpose of this distribution function here is to generate weight distributions, I is taken with 1, and (x-x_0) can be replaced by the distance D between beacon location and the location which contributes its localization error to cumulate regional error. Accordingly, the Cauchy-form weight distribution function here is defined by

$$w\left(D;\gamma\right)=\frac{\gamma^2}{(D)^2+\gamma^2}\,. \tag{3.9}$$

Fig. 3.5 demonstrates a weight distribution generated by Eq. (3.9) with $\gamma = 1$.

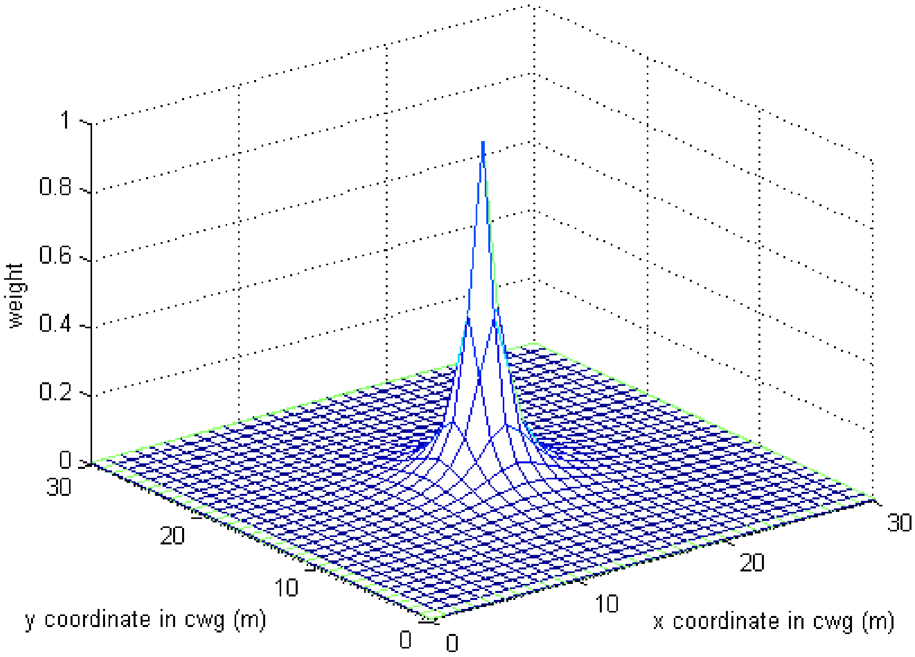


Fig. 3.5. The plot of Cauchy-form weight distribution with γ = 1.

To compute the DAG measurement, the weight distribution is applied to weighting the regional localization error. A 2-dimensional rectangular sliding window called "*centralized-weighting grid* (*cwg*)" with side length *gridSide=2R* is set up and the localization errors that lie in the *cwg* are summed up with multiplying the corresponding weights obtained by substituting the distance from the center of *cwg* into Eq. (3.9). This weighted regional localization error is taken to be the DAG measurement, as defined by

$$\psi_{DAG}(\mathbf{z}, B_a^n) = \sum_{x=\mathbf{z}.x-\frac{R}{step}}^{\mathbf{z}.x+\frac{R}{step}} \sum_{y=\mathbf{z}.y-\frac{R}{step}}^{\mathbf{z}.y+\frac{R}{step}} w\left(\left\|(x,y)-\mathbf{z}\right\|;\gamma\right)\cdot\varepsilon\left((x,y),B_a^n\right). \tag{3.10}$$

In this chapter, we observe that a beacon node contributes a centralized-distributed localization improvement in its coverage. Its location holds the best effect to reduce localization error, while the amount reduced on neighboring region decreases with the increase of the distance from the beacon node. It can result bias in measuring the effectiveness of a beacon location if one does not consider the regional error or consider the regional error as uniformly distributed. According to the impact distribution shown in Fig. 3.3, we select the Cauchy distribution, which can fit the distribution best, to design the DAG measurement. DAG can consider the real impact distribution in a proper manner and thus measure the effectiveness of a beacon location more precisely.

Previous measurements were implemented in the manner of incremental beacon placement (i.e., given an initial beacon deployment, then iteratively place a beacon node at the location with greatest measured effectiveness). Therefore, to evaluate and compare the performance of our DAG with previous methods, it will be applied to design beacon deployments by incrementally placing beacon nodes in Chapter 5. According to Eq. (3.2), a measurement is more precise if the mean localization error reduced by activating an additional beacon node with biggest value is greater. Accordingly, the localization accuracy can be achieved with fewer active beacon nodes.

Chapter 4 Adaptive Beacon Duty Scheduling

Most of previous scheduling algorithms that consider beacon duty cycle take active beacon density as the design parameter to adjust beaconing duty and be provided for users to set desired control points, such as [14], [16], [17]. Nevertheless, two drawbacks majorly exist in such manners.

- They do not consider the impact of beacon location, which has been identified as a significant factor for localization accuracy [18]. The algorithms make beacon nodes detect active neighbors and compute local density (density in their radio coverage) when the beaconing (sleeping) timers expire, and then select beacon nodes to be turned on or turned off to satisfy the density requirement according to random factors rather than the impact of beacon nodes. Therefore, some beacon nodes located at the locations without any benefit on reducing localization error (e.g., the location near to an active beacon node) may be turned on to transmit beacon signal, and thus the costs of energy and bandwidth are wasted.
- Density is not an intuitional parameter for setting a desired control point. Expressly, one wants to set the requirement of localization accuracy when he is building a localization system with a beacon duty scheduling algorithm. The relation between density and accuracy depends on the localization algorithm used, environmental factors, and power settings of beacon nodes. It is inconvenient to readjust the density requirement for desired accuracy whenever these conditions change.

In [15], a location error estimation method is introduced to evaluate the localization error against probability at a beacon location [20], [21], [22]. However, the scheduling algorithm attempts to assign more beacon duty to the beacon nodes placed at a location with lower estimated error. In the formula of delay time [15],

$$delay_i = \left[\left[\frac{d_i^{max}(P)}{D} \right] + \left[1 - \frac{E_r}{E_t} \right] + R \right] \times T \,, \tag{4.1}$$

a smaller ratio term of location error estimation will result in a shorter delay time, and thus make the beacon node earlier to wake up to preempt a beacon duty. As a result, beacon nodes located at the locations with less effectiveness on reducing localization error are easier to be turned on, hence the costs of energy and bandwidth rise. Although the estimation formula considers two environment-dependent parameters, they are fixed that do not adapt to the variations in environments. Furthermore, the environment-dependent impact of beacon locations is not considered The Gaussian noise variable introduced in the estimation is used to describe the spatially distributed noise [23], whereas the formula considers it in a time-dependent manner. Moreover, the location error estimation is derived for distance-based localization algorithms, hence cannot be applied to localization algorithms in other types.

To effectively reduce cost of beaconing and maximize system lifetime, the impact of beacon locations must be adaptively considered in scheduling beacon duty cycle. In this chapter, we use the DAG measurement developed in Chapter 3 to design an adaptive beacon duty scheduling algorithm.

4.1 Problem of Beacon Duty Scheduling

Beginning from following the notations defined in Section 3.1, we introduce other notations to address the problem of beacon duty scheduling. For a given wireless network N consists of beacon nodes b^i in $B \subset N$ that know their locations $\left\{\mathbf{z}^1 ... \mathbf{z}^q\right\}$ and blind nodes in $U \subset N$ with unknown locations, select a set of active beacon nodes $B_a^n \subseteq B$ to transmit beacon signal in a period T_B, while the amount of mean localization error $\varepsilon_{mean}(B_a^n)$ remains below a distance threshold D. Other beacon nodes $B_s^m \subset B$ in the sleeping state turn off their radio transceiver to reserve energy. Neighbors of a beacon node b^i is denoted by $N(b^i) = \left\{b^j \in B \middle| P_r(d_{b_i,b_j}) > P_{threshold}\right\}$, which are the beacon nodes that receive the signal from b^i with RSSI greater then received power threshold $P_{threshold}$.

4.2 Developing Adaptive Beacon Duty Scheduling

In this section, we propose Adaptive Beacon Duty Scheduling (ABDS) algorithm, which attempts to achieve following design goals.

- *Maximize system lifetime*: A beacon duty scheduling algorithm must be able to find out redundant beacon nodes that have less effectiveness on reducing localization error and make them sleep to reduce power consumption.
- *Distributed*: The scheduling algorithm should be distributed that needs only local information obtained by 1-hop broadcast from neighbors for two reasons: WSNs are usually constructed in an ad-hoc manner, hence it should be able to adapt churn (nodes joining or leaving) without coordination provided by a central server; information delivery by multi-hop flooding can exponentially increase energy consumption that is unfavorable for battery-supported WSNs.
- *Adaptive to noisy environment in real world*: Noise in measured signal for localization systems is inevitable and unpredictable before applying the

system into the working environment (e.g., measure the RSSIs at various distances to evaluate noise strength) in real world. In addition, the environmental conditions may change and disturbances may occur when the system is working. Therefore, an on-line adaptive manner is attractive in this scenario.

- *User defined localization accuracy requirement*: The mean localization error in a terrain should be maintained to satisfy a required threshold. The intuitional parameter for setting requirement for a localization system is localization error. Accordingly, a beacon duty scheduling algorithm must be able to control localization error.

ABDS mainly consists of decision stage and execution stage. In the decision stage, information about local mean error, remaining energy, and effectiveness on reducing localization error are computed and exchanged between neighboring beacon nodes in B. After making the decision, beacon nodes join the active beacon set B_a^n to take the duty to transmit beacon signal or join the sleeping beacon set B_s^m to reserve energy in the execution stage. When the timer of execution period expires, a decision is remade for rotating the beacon duty. To apply DAG measurement $\psi_{DAG}(\mathbf{z}, B_a^n)$, localization errors at neighboring beacon nodes are required, and therefore the decision stage is composed of three phases. First, activity information about one-hop neighbors is collected. The localization error at beacon nodes' location can thus be computed and exchanged between neighboring beacon nodes in the second phase. With the knowledge of localization errors on one-hop neighbors, beacon nodes can then locally compute DAG measurement to evaluate their effectiveness on reducing localization error. The value of DAG measurement is fused with the ratio of remaining energy and exchanged in the third phase. Finally, the decision can be made according to the comparison of active priority with neighbors. The state transition of ABDS can be illustrated by Fig. 4.1.

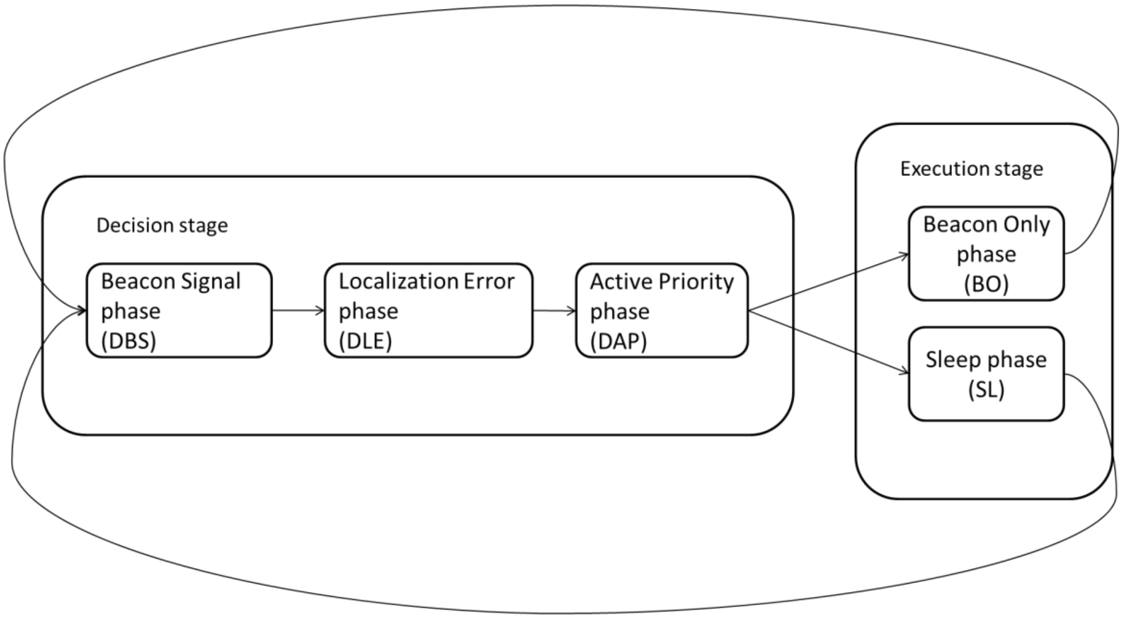

Decision stage
Beacon Signal phase (DBS)
Localization Error phase (DLE)
Active Priority phase (DAP)
Execution stage
Beacon Only phase (BO)
Sleep phase (SL)

Fig. 4.1. The state transition of ABDS.

Follows describe the detail of each phase in ABDS.

- *Decision stage-Beacon Signal (DBS) phase*: All beacon nodes $b^i \in B$ start with $state^i = active$ and set a timer T_{DBS}. Active beacon nodes $b_a^i \in B_a^n$ broadcast advertisements in a beaconing interval T_B to announce their activity. All b^i turn on the radio transceiver to listen for advertisements from their neighboring active beacon nodes and construct active neighbor list $N_a(b^i) = \left\{ b_a^j \in B_a^n \middle| P_r(d_{b^i,b_a^j}) > P_{threshold} \right\}$. When the timer T_{DBS} expires, all b^i enter into next phase.
- *Decision stage-Localization Error (DLE) phase*: All beacon nodes b^i set a timer T_{DLE}. All b^i estimate their location according to the information of

active neighbor list $N_a(b^i)$ and compute localization error of $\varepsilon(\mathbf{z}^i, N_a(b^i))$, and then broadcast the value of $\varepsilon(\mathbf{z}^i, N_a(b^i))$ to neighboring beacon nodes in $N(b^i)$ in the beaconing interval T_B and listen for $\left\{\varepsilon(\mathbf{z}^j, N_a(b^j)) \middle| b^j \in N(b^i)\right\}$. When the timer T_{DLE} expires, all b^i enter into next phase.

- *Decision stage-Active Priority (DAP) phase*: All beacon nodes b^i set a timer T_{DAP}. According to the localization errors distributed in $N(b^i)$, i.e. $\left\{\varepsilon(\mathbf{z}^j, N_a(b^j)) \middle| b^j \in N(b^i)\right\}$, all b^i can compute

$$\psi_{DAG}(\mathbf{z}^i) = \sum_{b^j \in N(b^i)} w\left(d_{b^i,b^j};\gamma\right) \cdot \left(\varepsilon(\mathbf{z}^j, N_a(b^j)) - \varepsilon_{threshold}\right), \tag{4.2}$$

where $\varepsilon_{threshold}$ is the user defined mean error threshold. A fused active priority AP can then be computed by

$$AP^i = \psi_{DAG}(\mathbf{z}^i) \cdot \left[ef \cdot \left(\frac{E_r^i}{E_i^i} \right) + \left(1 - ef\right) \right], \tag{4.3}$$

where E_r^i is the remaining energy of b^i, E_i^i is the initial energy of b^i at time 0, and ef in the range $[0,\ 1]$ is an energy factor that decide what level should the term of energy ratio $\frac{E_r^i}{E_i^i}$ be considered. For a b^i with $AP^i > 0$, it has higher priority to take a beacon duty with higher AP^i for the reason of activating beacon nodes as few as possible. Otherwise for $AP^i < 0$, it has higher priority to sleep with higher AP^i (less effect on reducing localization error) for inactivating beacon nodes as many as possible. Therefore, the energy ratio term is considered in this way to make a beacon node with less energy get a chance to sleep more easily. All b^i broadcast their AP^i in T_B and listen for $\left\{AP^j \middle| j \in N(i)\right\}$. When the timer T_{DAP} expires, all b^i check their local mean error for making decision by

$$\varepsilon_{mean}^{i} = \frac{\sum\limits_{b^j \in \left\{N(b^i)\quad b^i\right\}} \varepsilon(\mathbf{z}^j, N_a(b^j))}{\left\|\left\{N(b^i)\quad b^i\right\}\right\|}. \tag{4.4}$$

If the state of b^i is active and the local mean error ε_{mean}^{i} is smaller than $\varepsilon_{threshold}$, b^i may be a redundant active beacon node, hence it then compare its AP^i with other neighbors that have same conditions ($state^j = active$ and $\varepsilon_{mean}^{j} < \varepsilon_{threshold}$) and set its state to be asleep if AP^i is the greatest one. Otherwise, if the state of b^i is asleep and the local mean error ε_{mean}^{i} is greater than $\varepsilon_{threshold}$, b^i is a candidate to be active to reduce localization error, hence it then compare its AP^i with other neighbors that have same conditions ($state^j = asleep$ and $\varepsilon_{mean}^{j} > \varepsilon_{threshold}$) and set its state to be active if AP^i is the greatest one. This can be expressed by

$$\begin{aligned} &state^i = asleep \Leftrightarrow \left(state^i = active\right) \wedge \left(\varepsilon_{mean}^{i} < \varepsilon_{threshold}\right) \\ &\wedge \left(\forall b^j \in N(b^i) \middle| \left((state^j = active) \wedge (\varepsilon_{mean}^{j} < \varepsilon_{threshold})\right) : AP^i > AP^j\right) \end{aligned} \tag{4.5}$$

and

$$\begin{aligned} &state^i = active \Leftrightarrow \left(state^i = asleep\right) \wedge \left(\varepsilon_{mean}^{i} > \varepsilon_{threshold}\right) \\ &\wedge \left(\forall b^j \in N(b^i) \middle| \left((state^j = asleep) \wedge (\varepsilon_{mean}^{j} > \varepsilon_{threshold})\right) : AP^i > AP^j\right). \end{aligned} \tag{4.6}$$

After making the decision, b^i enters into execution stage to transmit beacon signal or sleep according to the decision. If $state^i = active$, b^i joins B_a^n

and broadcasts beacon signal in next stage. Otherwise, if $state^i = asleep$, b^i joins B_s^m and sleep for reserving energy.

- *Execution stage-Beacon Only (BO) phase*: All $b_a^i \in B_a^n$ set a timer T_{BO} at the start time in BO. All b_a^i periodically transmit beacon signal at intervals T_B and sleep for the remainder of the intervals. When the timer T_{BO} expires, all b_a^i transition back to the DBS phase.
- *Execution stage-Sleep (SL) phase*: All $b_s^i \in B_s^m$ set a timer T_{SL} at the start time in SL and then go to sleep. When the timer T_{SL} expires, all b_s^i transition back to the DBS phase.

Chapter 5 Evaluations

Simulations and evaluations of proposed DAG measurement and ABDS algorithm are carried out in MATLAB 7.11.0 with a wireless sensor network simulated by the typical shadowing propagation model [23]. To confirm our observations in Section 3.2, the DAG measurement is evaluated on both connectivity based localization and pattern matching localization. To compare with previous methods, the ABDS algorithm is evaluated on connectivity based localization for the comparison with STROBE and E-STROBE, and on maximum-likelihood estimator (MLE) for the comparison with Gribben’s method in [15].

5.1 Environment Model

The log-normal shadowing model is adopted to generate simulated terrains with real-world noise condition. Based on the path loss model as defined in Eq. (3.7), the Gaussian random variable with zero mean and standard deviation σ_{db} (shadowing deviation) $X_{db} \sim N\left(0, \sigma_{db}\right)$ is added to make the propagation model noisy. It reflects the variation of the mean received power at certain distance. The overall log-normal shadowing model is represented by

$$\left[\frac{P_r\left(d_0\right)}{\overline{P_r\left(d\right)}}\right]_{dB} = -10\beta\log\left(\frac{d}{d_0}\right) + X_{dB}. \tag{5.1}$$

The mean received power from beacons at each point on a random generated terrain is calculated by Eq. (5.1) and applied to perform further simulations.

5.2 Evaluation of the DAG Measurement

To be compared with previous measurements proposed in beacon placement algorithms, proposed DAG measurement is applied to incrementally place beacon nodes to design beacon deployments. Not only referred works in [18], but also a recent empirical method namely Greedy [19], which allows adaptation to terrain conditions and takes connected beacon density as the measurement to incrementally place beacons to design beacon deployment, was carried out in simulations and compared with proposed DAG measurement.

5.2.1 Localization Algorithms

The incremental beacon placement process was performed on connectivity based localization and pattern matching localization. Connectivity based localization algorithm computes the centroid of connected beacons as the estimated location for a blind node. If the mean received power exceeds the receiving threshold $P_{threshold}$, a blind node is identified as connected with the beacon node. Subsequent to computing mean received power by Eq. (5.1), connectivity is evaluated by applying following condition,

$$\overline{P_r(d)} > P_{threshold} \,. \tag{5.2}$$

In the pattern matching localization algorithm, the k-nearest neighbor (KNN) algorithm, which uses Euclidian distance to find out *k* nearest patterns and select the mode to be the output, is applied to extract signal feature of locations. This localization algorithm comprises training phase and locating phase. The half of data points (feature vector of signal strengths and corresponding location coordinate) are uniformly chosen to be the training data set to establish feature database amid the training phase. The training data selection is illustrated in Fig. 5.1. In the locating phase, an unknown location is estimated by the KNN operation on its signal strength feature.

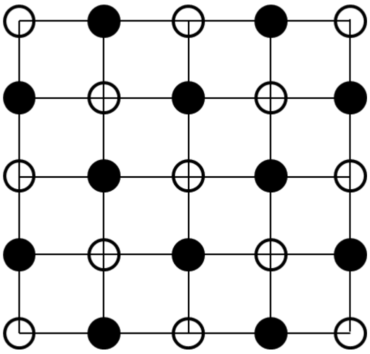


Fig. 5.1. Solid circles mark the training data points.

5.2.2 Simulation Parameters

The values of log-normal shadowing model parameters are chosen from the ranges of their typical values in indoor environments [23]. $P_r(d_0)$ is obtained by the experiment in an indoor environment. The parameters and corresponding values used in this simulation are summarized in Table 5.1. They do not exactly reflect the details of a real environment, but are representative of the range of environments in which the algorithms may be applied.

Table 5.1. Parameters and their values used in the simulations of incremental beacon placement.

Parameter	*Side*	*Step*	*R*	*gridSide*	Υ	*k*	β	d_0	$P_r(d_0)$	σ_{dB}
Value	100 m	1 m	15 m	30 m	1	5	1.8	1 m	-54 dbm	5.5

5.2.3 Performance Metrics

For the simulations performed with connectivity based localization, Eq. (3.1) is used to evaluate and compare the performances of various measurements used to design beacon deployment. However, according to the fact that the locations of training data are known, for the simulations with pattern matching localization, only the data points which are not in the training data set are chosen to examine the mean localization error. Therefore, following equation is used to examine the mean localization error in the pattern matching localization.

$$\varepsilon_{mean}(B_a^n) = \frac{\sum_{x=0}^{side/step} \sum_{y=0}^{side/step - x \bmod 2} \varepsilon\left((x, 2y + x \bmod 2), B_a^n\right)}{\left\lceil \left(\frac{side}{step} + 1\right)^2 \Big/ 2 \right\rceil} \tag{5.3}$$

5.2.4 Simulation Flow

DAG, Max, Grid, and Greedy were evaluated in the simulations carried out in MATLAB to incrementally place beacons (by the steps described in Section 2.2) respectively. In each of simulation rounds, initially 20 beacons are randomly placed in the terrain. After examining the localization errors on the terrain, the location without a beacon that has highest measurement value is chosen to place an additional beacon. Then the mean localization error is re-examined. Fig. 6 shows the flow chart to illustrate the flow of incremental beacon placement. There is a random factor in the initial state (i.e., random beacon placement). To characterize the statistical significance of our simulation results, each simulation executes for 10 times with different random initial beacon placements.

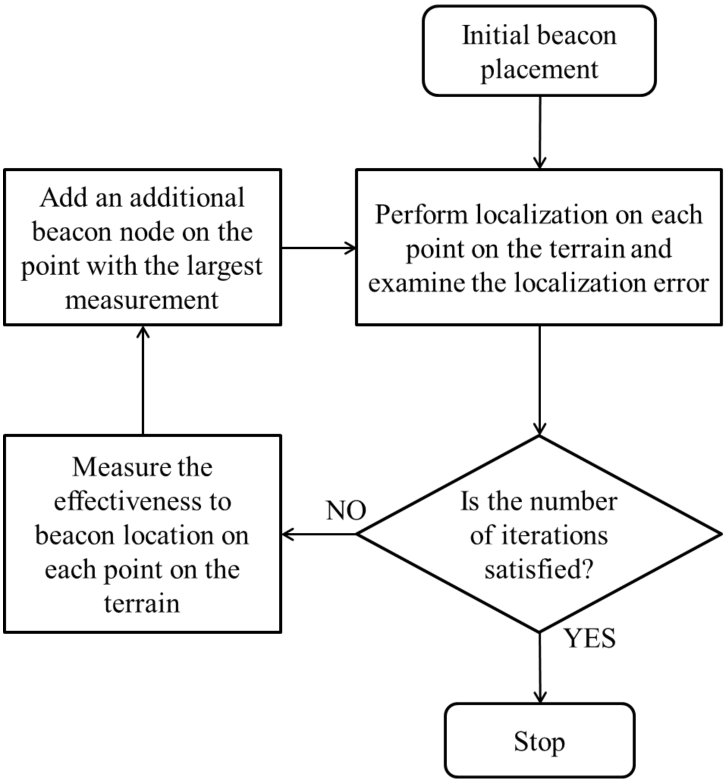

Initial beacon placement
Perform localization on each point on the terrain and examine the localization error
Is the number of iterations satisfied?
NO
YES
Stop
Measure the effectiveness to beacon location on each point on the terrain
Add an additional beacon node on the point with the largest measurement

Fig. 5.2. Flow chart of incremental beacon placement.

5.2.5 Simulation Results

Fig. 5.3 shows the final placements of Max, Grid, DAG, and Greedy. According to the centralized-distributed improvement on localization, the Grid measurement overly expects the ability of a beacon node to reduce the localization error in neighboring region. Therefore, many beacon nodes will be designed to place at locations with no any benefits on localization. This overestimation flaw of Grid causes a locally dense placement behavior that extremely squanders on beacon resource.

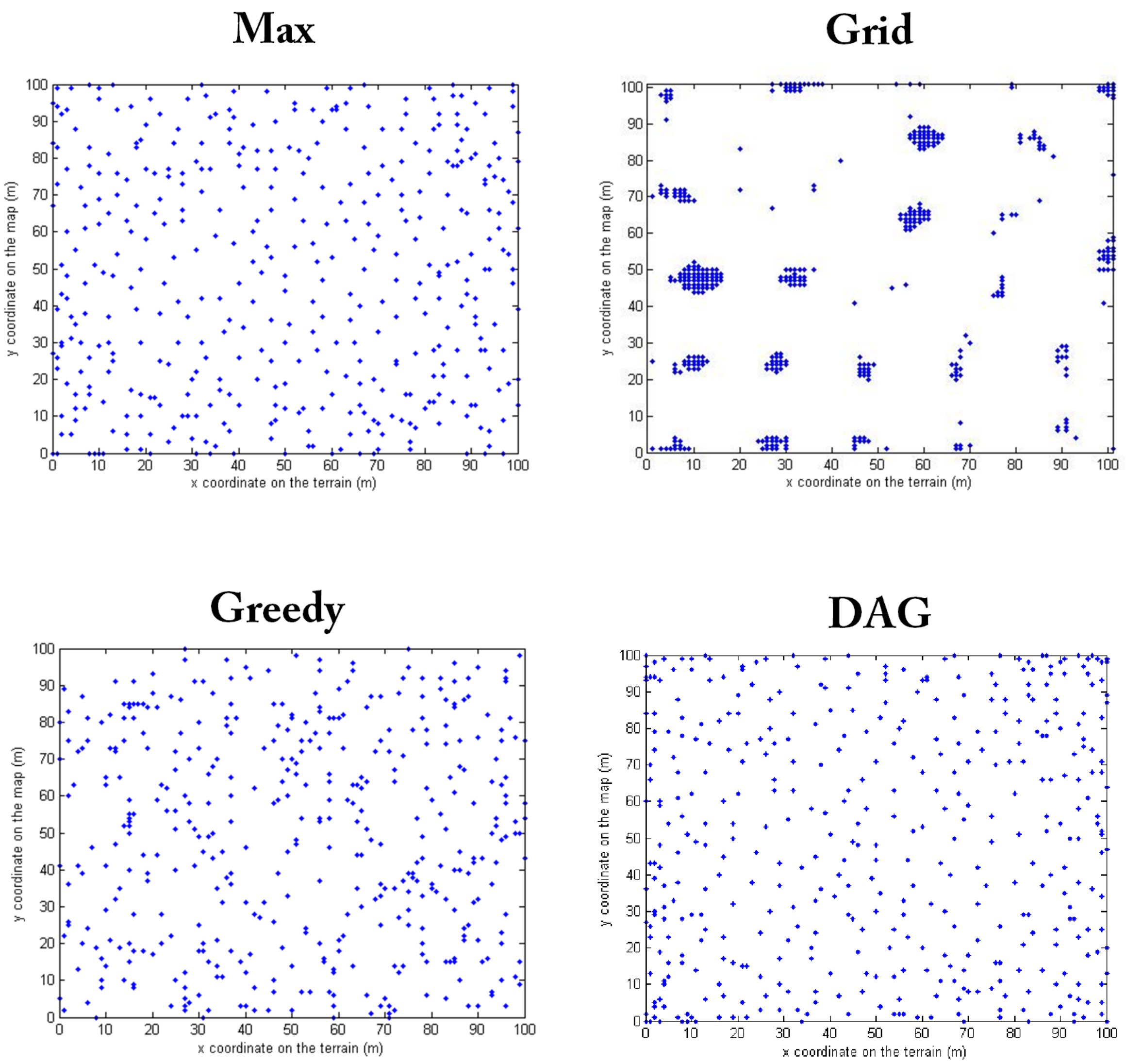


Fig. 5.3. The final beacon placement of Max, Grid, Greedy, and DAG in a 2-dimentional space.

Fig. 5.4 and Fig. 5.5 show the simulation results of incrementally placing beacons on connectivity based localization and pattern matching localization respectively. The averages and 95% confidence intervals are plotted in the figures.

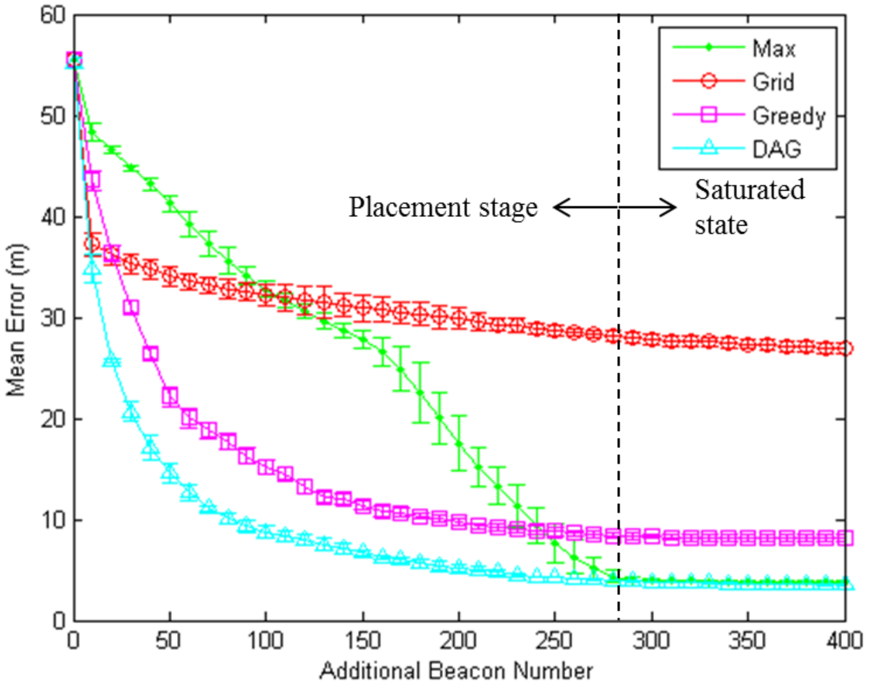


Fig. 5.4. Performance of the measurements to incrementally place beacon nodes with connectivity based localization.

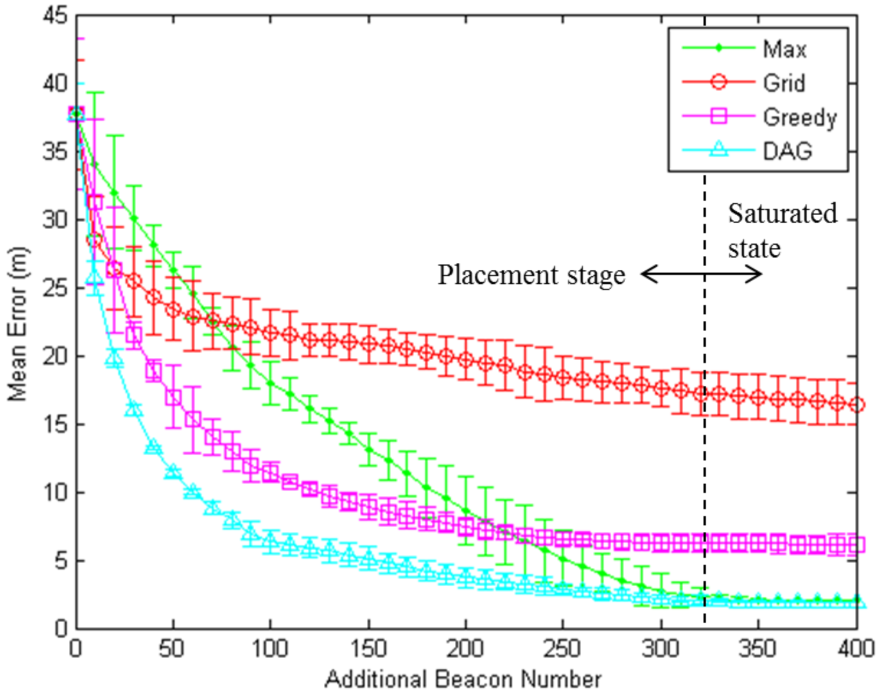


Fig. 5.5. Performance of the measurements to incrementally place beacon nodes with pattern matching localization.

As shown in Fig. 5.4, Grid provides better performances than Max at early placement stage. That confirms the idea of considering regional localization error. However, because of the locally dense placement behavior, Grid quickly starts to converge and provides the saturated localization performance much worse than Max. Although Greedy and GWG have similar trends on reducing localization error, DAG performs about two times better than Greedy at saturated state. DAG surpasses other three methods at early placement stage and provides the saturated performance same as Max (around mean error of 4 meters) with less additional beacon number to save 30% beacons. Moreover, for the device shortage scenario, it reduces 76% localization error compared to Max when only half of the beacons at saturated state (around 150 beacons) are placed. This phenomenon also exists in the pattern matching localization as shown in Fig. 5.5. DAG can reduce 25% usage of beacons and reduce 61% localization error with half of the placed beacons at saturated state.

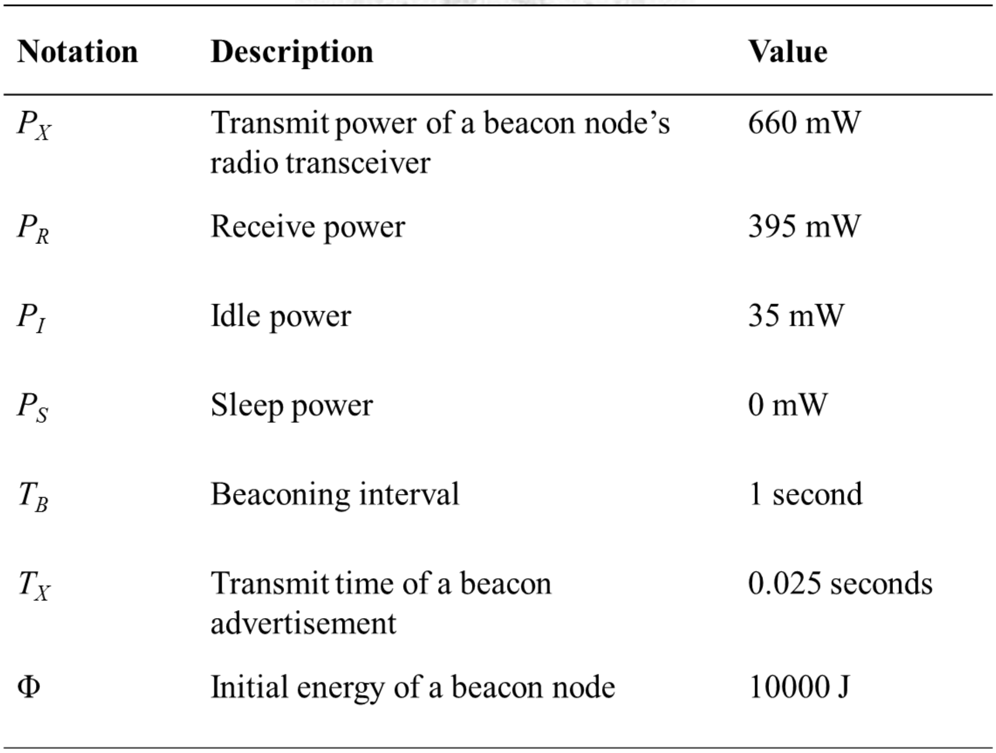
5.3 Evaluation of the ABDS algorithm

To evaluate the performance of ABDS, STROBE, E-STROBE, and Gribben's method are also implemented in the simulated wireless sensor network to be compared with ABDS. The simulated network is composed of 100 beacon nodes uniformly deployed in a square area of size 100 m × 100 m in a grid manner. Beacon duty scheduling algorithms are performed on each beacon node. The environmental condition (noise distribution) is same across simulations of various beacon duty scheduling algorithms. Simulation methodologies are described in following sections.

5.3.1 Energy Model

In addition to the signal propagation model in Section 5.1 and corresponding values in 5.2.2, an energy model is introduced to simulate the energy usage of beacon duty scheduling algorithms. We use the same energy model in [16], as summarized in Table 5.2. This energy model only characterizes the energy usage of the radio transceiver and does not model the energy usage of local computation, because typical computational costs are much lower than communication costs and thus negligible [15], [16], [17], [24].

Table 5.2. Parameter settings of the energy model in the simulations of beacon duty scheduling algorithms.



Notation	Description	Value
P_X	Transmit power of a beacon node's radio transceiver	660 mW
P_R	Receive power	395 mW
P_I	Idle power	35 mW
P_S	Sleep power	0 mW
T_B	Beaconing interval	1 second
T_X	Transmit time of a beacon advertisement	0.025 seconds
Φ	Initial energy of a beacon node	10000 J

5.3.2 Localization Algorithms

STROBE, E-STROBE, and our ABDS are evaluated and compared on connectivity based localization. To evaluate and compare Gribben's method, however, it was specifically designed for range-based localization methods. The input for localizing a blind node is the estimated distance from beacon nodes, which is derived from Eq. (5.1) and given by

$$\tilde{d}_{i,j} = d_0 \left(\frac{P_r(d_0)}{P_r\left(d_{i,j}\right)} \right)^{\frac{1}{\beta}} . \tag{5.4}$$

Accordingly, the scheduling algorithm cannot be applied on proximity-based localization. It was evaluated on MLE [22], which is

$$\hat{\mathbf{z}}_i = \underset{\{\mathbf{z}_i \in T\}}{\arg\min} \sum_{j \in B_a^n} \left(\ln \frac{\tilde{d}_{i,j} / C^2}{d^2\left(\mathbf{z}_i, \mathbf{z}_j\right)} \right)^2 , \tag{5.5}$$

$$C = \exp\left[\frac{1}{2} \left(\frac{\ln(10)}{10} \frac{\sigma_{db}}{\beta} \right)^2 \right] \tag{5.6}$$

for a node i with connected active beacon nodes B_a^n. Therefore, ABDS is also implemented on MLE in another simulation for the comparison with Gribben’s method.

5.3.3 Parameter Setting of Algorithms

Due to the fact that all beacon nodes are active to exchange information for making decision at decision stage and only a fraction of beacon nodes are active at execution stage, system lifetime is sensitive to the ratio of time of execution to time of decision. Higher ratio can result in longer system lifetime. However, a long time of execution stage can bring the system low response to variations. Accordingly, these scheduling algorithms need to set a same time of execution stage for fair comparison. Table 5.3, Table 5.4, and Table 5.5 summarize the parameters in ABDS, STROBE, E-STROBE, and Gribben’s method, and corresponding values set in the simulations.

Table 5.3. Parameters of ABDS and corresponding values.

Notation	Description	Value
T_{DBS}	Time of beacon signal phase in decision stage	5 seconds
T_{DLE}	Time of localization error phase in decision stage	T_{DBS}
T_{DAP}	Time of active priority phase in decision stage	T_{DBS}
T_{BO}	Time of beacon only phase in execution stage	$1000T_{DBS}$
T_{SL}	Time of sleep phase in execution stage	T_{BO}

Table 5.4. Parameters of STROBE and E-STROBE, and corresponding values.

Notation	Description	Value
T_V	Time of voting state	5 seconds
T_D	Time of designated state	$1000T_V$
T_S	Time of sleep state	T_D

Table 5.5. Parameters of Gribben's method and corresponding values.

Notation	Description	Value
$delay_{max}$	Maximum delay time	5 seconds
T_R	Time of reference (for beaconing and sleeping)	$1000delay_{max}$
α	Scaling factor for error estimation	6.9
β	Scaling factor for error estimation	139.9
γ	Scaling factor for error estimation	-2.6

5.3.4 Performance Metrics

During the simulations, as in [16], following metrics are measured periodically for a snapshot interval 100 seconds.

- *Mean localization error*: The mean localization error is computed by Eq. (3.1) according to currently active beacon nodes at the instant.
- *Percentage of active beacon nodes*: This is the percentage of total beacon nodes that are actively sending beacon signal at any given instant of time (either in decision stage or execution stage).

Other two metrics not a function of time are used.

- *System lifetime*: A beacon duty scheduling system is dead if the moving average of mean localization error exceeds an error threshold. The system lifetime is the time elapsed since the start before the system has died.
- *Mean active ratio*: This is the mean active beacon ratio in the duration of system lifetime.

In addition, we take the snapshot of active beacon node distribution to understand the outcome of beacon duty distribution produced by various scheduling algorithms.

5.3.5 Simulation Results

- ABDS, STROBE, and E-STROBE on connectivity based localization

ABDS, STROBE, and E-STROBE are simulated on connectivity based localization. The error threshold for ABDS is set to be 12 m. The density threshold for STROBE and E-STROBE is set to be 4 (4 active beacon nodes in a coverage) to control the mean error to be under 12 m. Snapshots of the distribution of active beacon nodes in ABDS, STROBE, and E-STROBE are shown in Fig. 5.6. Because STROBE and

E-STROBE only consider the local density of neighboring active beacon nodes, the nodes that should take active beaconing duty are randomly selected out from candidates. As a result, some beacon nodes at the location with minimum benefits on localization (such as very close to other active beacon nodes) are designed to be active. This phenomenon results in the holes of active beacon distribution and waste of power resource as described in Section 5.2.

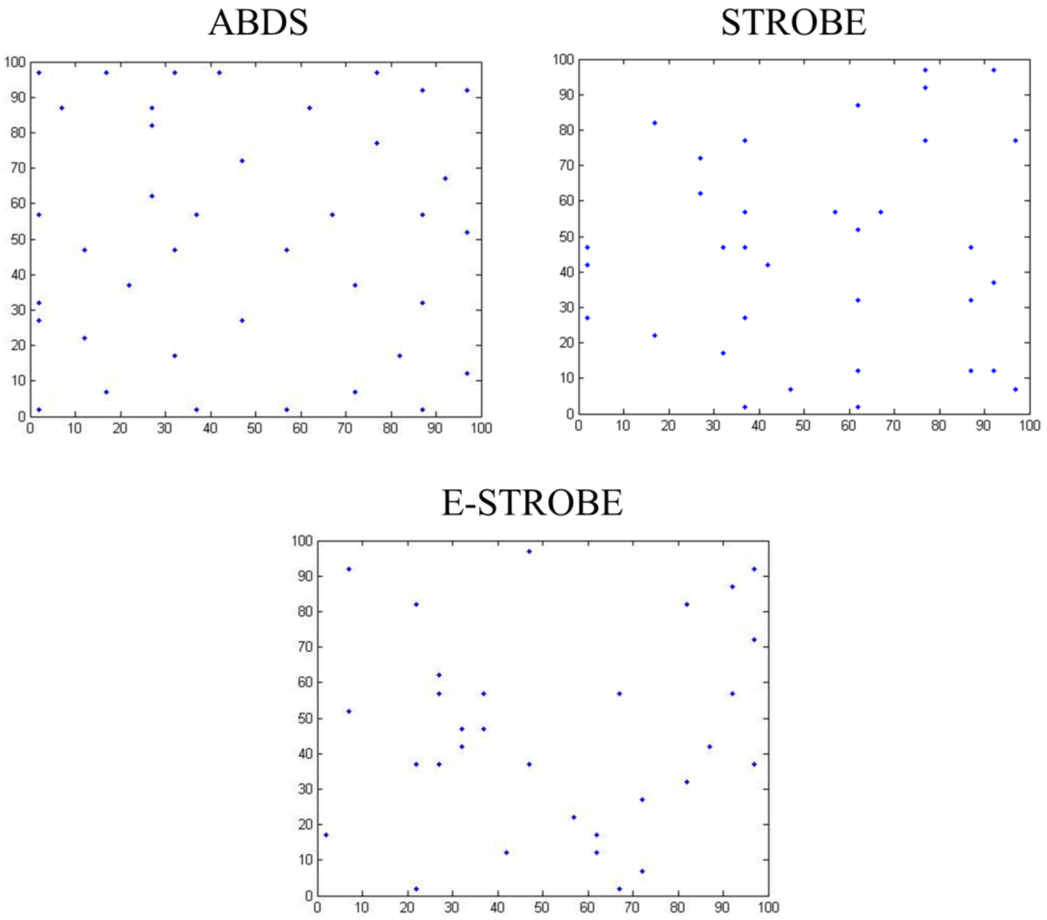
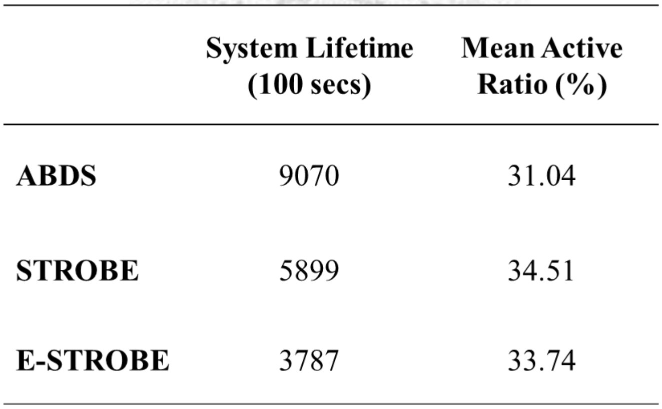


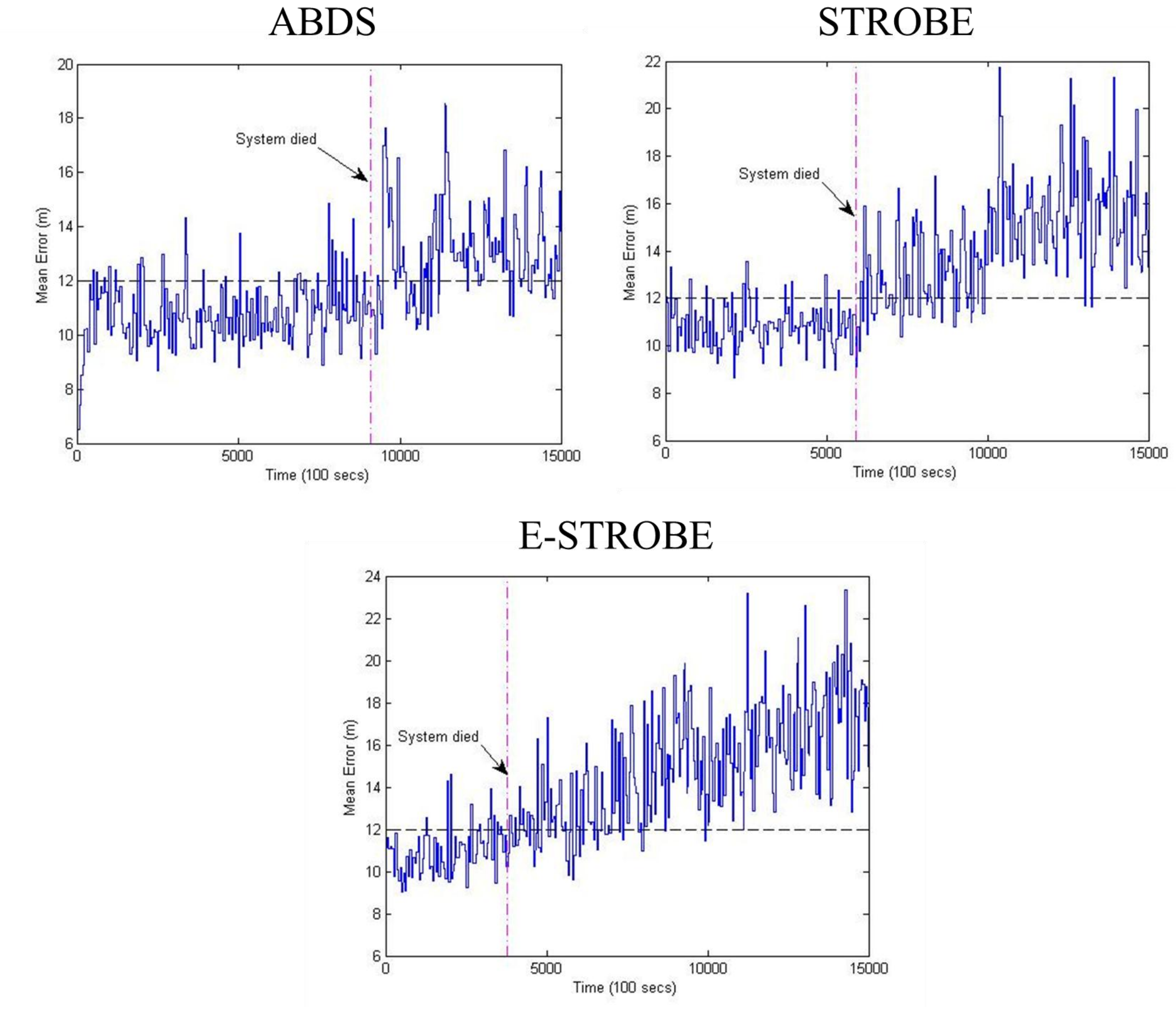
Fig. 5.6. Snapshots of active beacon map of ABDS, STROBE, and E-STROBE on connectivity-based localization.

ABDS considers the effectiveness of beacon locations to assign beaconing duty rather than by random. Therefore, ABDS can design more efficient distribution of active beacon nodes with less node resource. Fig. 5.7 plots the mean localization error of these three algorithms and Fig. 5.8 plots the active beacon ratio. Table 5.6 summarizes the corresponding system lifetime and mean active ratio. As shown in Table 5.6, ABDS reduces 10 % beacon usage (mean active ratio) and prolongs 54% lifetime compared to STROBE. E-STROBE considers an energy threshold independently after checking the density threshold in its scheduling strategy. A beacon node may be turned off even it should be active to maintain the density threshold. As shown in Fig. 5.7 and Fig. 5.8, this method incrementally decreases the active ratio and increases the mean error with the consumption of energy. Therefore, it quickly makes the system dead.

Table 5.6. Performance comparison of ABDS, STROBE, and E-STROBE.



	System Lifetime (100 secs)	Mean Active Ratio (%)
ABDS	9070	31.04
STROBE	5899	34.51
E-STROBE	3787	33.74



ABDS

STROBE

E-STROBE

Fig. 5.7. Mean localization error of ABDS, STROBE, and E-STROBE on connectivity-based localization.

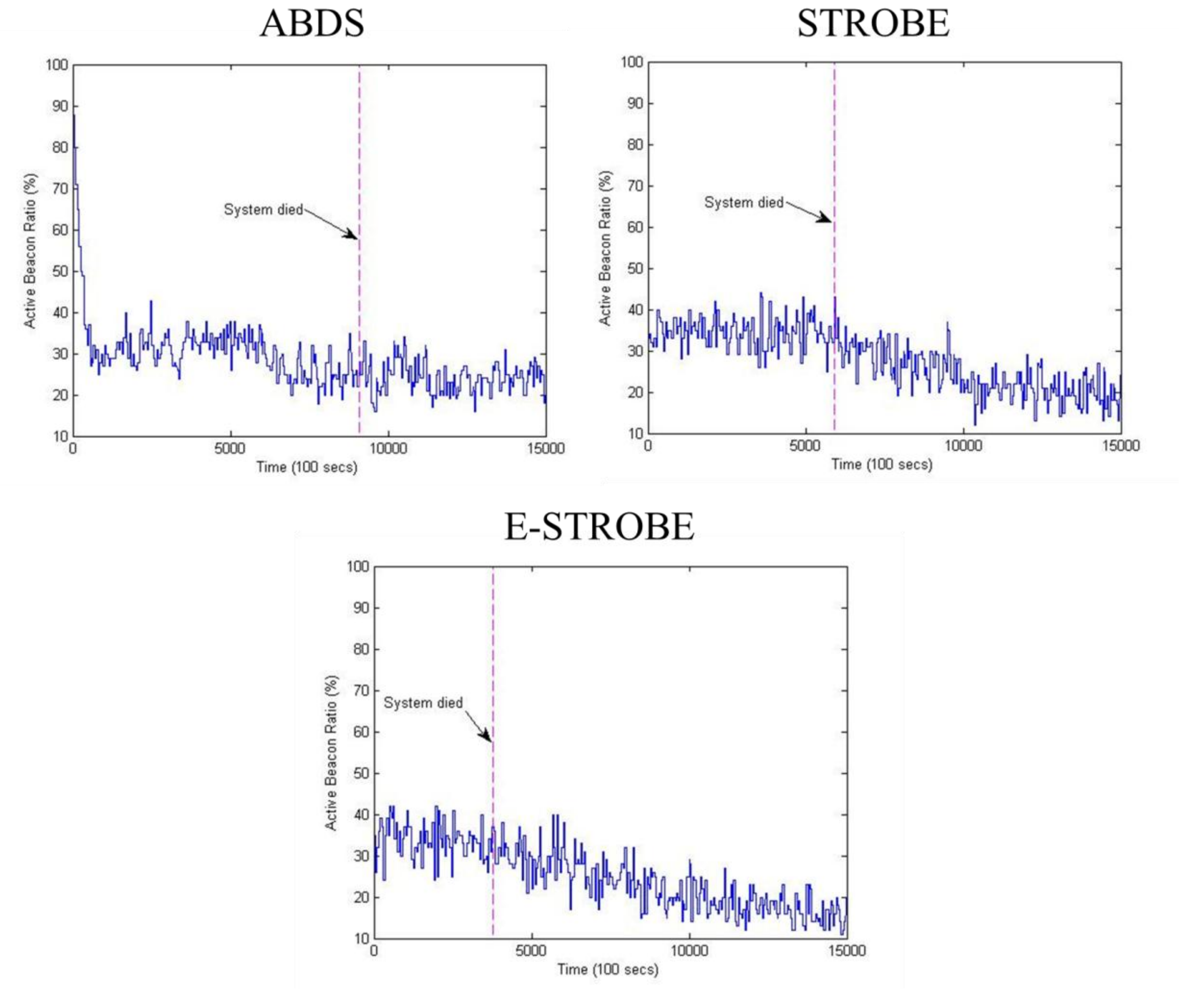
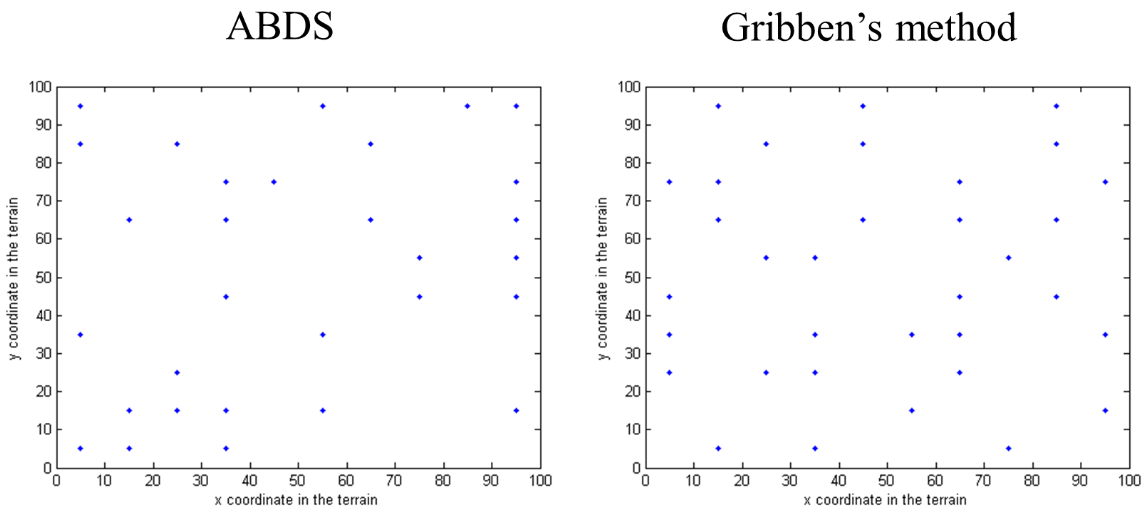


Fig. 5.8. Active beacon ratio of ABDS, STROBE, and E-STROBE on connectivity-based localization.

- ABDS and Gribben’s method on MLE localization

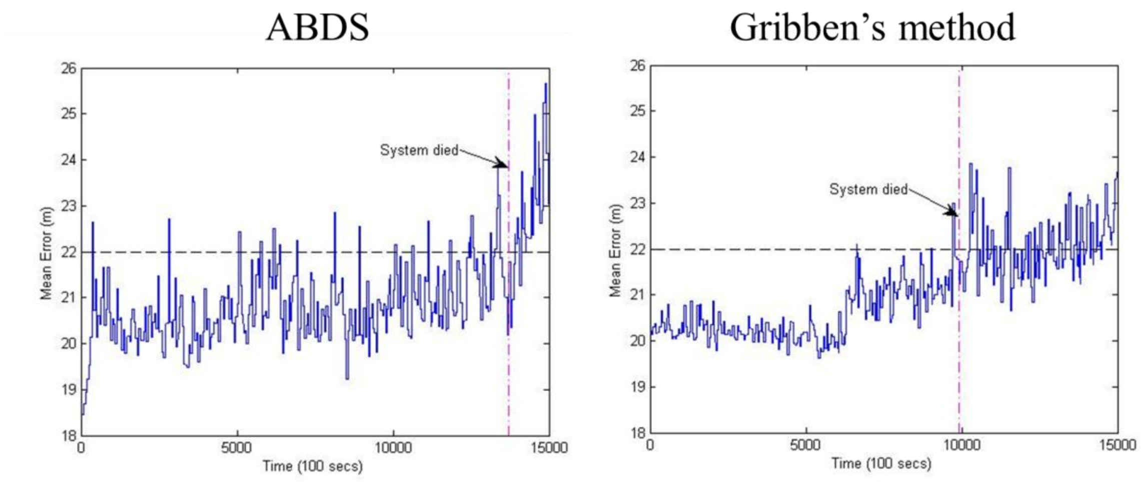
Because Gribben’s method is specifically designed for range-based localization approaches, ABDS and Gribben’s method[15] are compared in the simulations on MLE localization. The error threshold is set to be 22 m. Snapshots of the distribution of active beacon nodes in ABDS and Gribben’s method are shown in Fig. 5.9. The location impact of beacon nodes is also considered in Gribben’s method, so it does not activate beacon nodes very closely.


ABDS

Gribben’s method

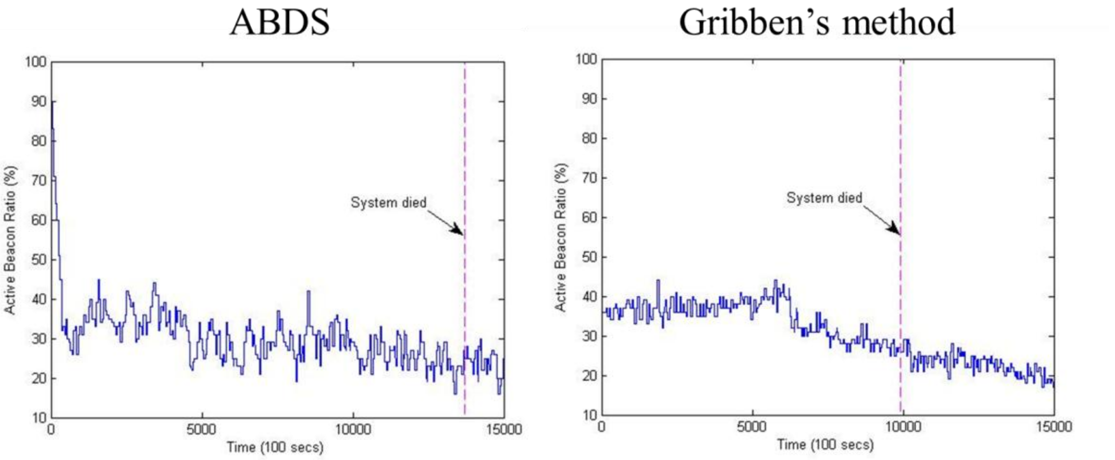
Fig. 5.9. Snapshots of active beacon map of ABDS and Gribben’s method on MLE localization.

However, this method does not consider real noise distribution in its model. As a result, it cannot adapt the real environment very well. As shown in Fig. 5.10 and Fig. 5.11, Gribben’s method activates too many beacon nodes and thus shortens the system lifetime. Table 5.7 summarizes the system lifetime and mean active ration of ABDS and Gribben’s method. ABDS provides 38% longer lifetime and 12% fewer beacon usage.


ABDS

Gribben’s method

Fig. 5.10. Mean localization error of ABDS and Gribben’s method on MLE localization.



ABDS

Gribben’s method

Fig. 5.11. Active beacon ratio of ABDS and Gribben’s method on MLE localization.

Table 5.7. Performance comparison of ABDS and Gribben’s method.

	System Lifetime (100 secs)	**Mean Active Ratio (%)**
ABDS	13718	30.65
Gribben’s method	9916	34.69

Chapter 6 Conclusions

In this thesis, in order to efficiently schedule beacon duty cycle, we designed a measurement to the effectiveness of beacon locations. We face the improvement distribution on localization performance in the coverage of an additional beacon. We propose DAG to adapt the non-uniform improvement distribution to avoid the overestimation and locally dense placement occurred in the method of Grid. Our approach adopts Cauchy distribution to generate a centralized weight distribution. The weights are applied to cumulate regional localization error. The purpose is to more precisely measure the effectiveness of beacon locations and thus design better beacon deployments (with fewer beacon nodes deployed). To evaluate DAG, we compare it with the classical Max, Grid, and Greedy methods in the experiments of incremental beacon placement. The experiments were carried out in a simulated indoor environment built by the log-normal shadowing model. The results demonstrate that the placement of additional beacons with the employment of DAG can reduce 30% beacon usage. When only half of these beacons at saturated state were placed, DAG reduces 76% localization error.

ABDS for scheduling beacon duty cycle is then designed by the usage of DAG. ABDS attempts to achieve following design goals.

- Maximize system lifetime
- Distributed algorithm
- Adaptive to noisy environment in real world
- Can be applied to any types of localization algorithm
- User defined localization accuracy requirement

In real world, noise makes beacon nodes at various locations have different effectiveness to reduce localization error. This impact of beacon location is not considered in previous scheduling algorithms. In ABDS, neighboring beacon nodes exchange information to compute the DAG measurement and find out the efficient beacons. As a result, the requirement of localization error can be satisfied by fewer active beacon nodes and the system lifetime can be prolonged. Compared to previous method, ABDS can reduce 10% beacon usage and provide 54% longer lifetime.

REFERENCES

[1] J. Yick and B. Mukherjee, D. Ghosal. “Wireless sensor network survey”, Computer Networks, Vol. 52, No. 12, pp. 2292-2330 (2008).

[2] S. Gezici. “A survey on wireless position estimation”, Wireless Personal Communications, Vol. 44, No. 3, pp. 263-282 (2008).

[3] J. Wang, R. K. Ghosh, and S.K. Das. “A survey on sensor localization”, Journal of Control Theory and Applications, Vol. 8, No. 1, pp. 2-11 (2010).

[4] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Anderson. “Wireless sensor networks for habitat monitoring”, In Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications, pp. 88-97 (2002).

[5] M. Mauve, A. Widmer, and H. Hartenstein. “A survey on position-based routing in mobile ad hoc networks”, IEEE network, Vol. 15, No. 6, pp. 30-39 (2001).

[6] S. Meguerdichian, F. Koushanfar, M. Potkonjak, and M. B. Srivastava. “Coverage problems in wireless ad hoc sensor networks”, In IEEE INFOCOM, Vol. 3, pp. 1380-1387 (2001).

[7] P. Bahl and V. N. Padmanabhan. “Radar: An in-building user location and tracking system”, In IEEE INFOCOM, Vol. 2, pp. 775-784 (2000).

[8] N. Bulusu, J. Heidemann, and D. Estrin. “GPS-less low cost outdoor localization for very small devices”, IEEE Personal Communications, Vol. 7, No. 5, pp. 28-34 (2000).

[9] S. H. Fang, T. N. Lin, and P. C. Lin. “Location fingerprinting in a decor-related space”, IEEE Transactions on Knowledge and Data Engineering, Vol. 20, No. 5, pp. 685-691 (2008).

[10] B. Chen, K. Jamieson, H. Balakrishnan, and R. Morris. “Span: An Energy-Efficient Coordination Algorithm for Topology Maintenance in Ad Hoc Wireless Networks”, Wireless Networks, Vol. 8, No. 5, pp. 481-494 (2002).

[11] G. Anastasia, M. Contib, M. D. Francescoa, and A. Passarellab. “Energy conservation in wireless sensor networks: A survey”, Ad Hoc Networks, Vol. 7, No. 3, pp. 537-568 (2009).

[12] E. Bulut and I. Korpeoglu, I. “Sleep Scheduling with Expected Common Coverage in Wireless Sensor Networks”, Wireless Networks, Vol. 17, No. 1, pp. 19-40 (2011).

[13] S. Mahfoudh and P. Minet. “Survey of Energy Efficient Strategies in Wireless Ad Hoc and Sensor Networks”, International Conference on Networking, pp. 1-7 (2008).

[14] J. Gribben, A. Boukerche, and R. W. N. Pazzi. “Scheduling for Scalable Energy-Efficient Localization in Mobile Ad Hoc Networks”, In Seventh Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communication and Networks (SECON), pp. 1-9 (2010).

[15] J. Gribben and A. Boukerche, “Localization scheduling in wireless ad hoc networks”, IEEE 35th Conference on Local Computer Networks, pp. 512-519 (2010).

[16] N. Bulusu, J. Heidemann, D. Estrin, and T. Tran. “Self-configuring Localization Systems: Design and Experimental Evaluation”, ACM Transactions on Embedded Computing Systems (TECS), Vol. 3, No. 1, pp. 24-60 (2003).

[17] A. Tarnacha and T. L. Porta. “E-STROBE: An adaptive beacon activation algorithm for sensor localization”, Vehicular Technology Conference, Vol. 5, pp. 2951-2955 (2003).

[18] N. Bulusu, J. Heidemann, and D. Estrin. “Adaptive beacon placement”, In Proceedings of Distributed Computing Systems, pp. 489-498 (2001).

[19] V. S. Rani and S. V. Raghavan. “A greedy approach to beacon placement for localization”, IEEE IFIP Wireless Days, pp. 1-5 (2008).

[20] J. Gribben and A. Boukerche. “Probabilistic Estimation of Location Error in Wireless Ad Hoc Networks”, In IEEE Global Communications Conference (GLOBECOM), pp. 1-5 (2010).

[21] S. M. Kay. “Fundamentals of statistical signal processing: estimation theory”, Prentice-Hall Signal Processing Series (1993).

[22] N. Patwari, A. O. Hero, M. Perkins, N. S. Correal, and R. J. O’Dea. “Relative location estimation in wireless sensor networks”, IEEE Transactions on Signal Processing, Vol. 51, No. 8, pp. 2137-2148 (2003).

[23] T. S. Rappaport. “Wireless communications: principles and practice, 2nd Ed”, Prentice Hall (2002).

[24] G. J. Pottie and W. J. Kaiser. “Wireless integrated network sensors”, Communications of the ACM, Vol. 43, No. 5, pp. 51-58 (2000).