

# 基于边框定界的 WSN 分布式全搜索定位算法

姚英彪, 曾嵘, 易志强

(杭州电子科技大学 通信工程学院, 浙江 杭州 310018)

**摘 要:** 提出一种基于边框定界的 WSN 分布式全搜索定位算法。该算法通过节点测距得到邻居节点的坐标和距离信息, 然后通过边框定界方法确定节点存在的位置区域, 最后将位置区域网格化, 并用全搜索方法在该区域搜索最佳估计点, 最佳估计点的坐标即为节点的定位坐标。该算法应用到网络时需运行多轮, 通过逐步求精得到节点的定位坐标。仿真实验表明该算法达到当前其他复杂定位算法的性能。

**关键词:** 无线传感器网络; 边框定界; 定位算法; 全搜索

中图分类号: TP393

文献标识码: A

文章编号: 1000-436X(2012)Z2-0135-06

## Bounding box based distributed search localization algorithm for WSN

YAO Ying-biao, ZENG Rong, YI Zhi-qiang

(College of Telecommunication Engineering, Hangzhou Dianzi University, Hangzhou 310018, China)

**Abstract:** A bounding box based distributed full search localization algorithm for WSN (wireless sensor networks) was proposed. Firstly, the coordinates and distances of neighbor nodes were obtained by ranging. Secondly, the possible node position area was determined based on bounding box method. Finally, the area was meshed and the full search method was employed to search the best estimation point, whose coordinates are the node's localization coordinates. In real WSN, the algorithm need run several rounds to obtain node's localization coordinates by stepwise refinement. Simulation results show that the proposed algorithm achieves the comparable performance with other up-to-date complex localization algorithms.

**Key words:** wireless sensor networks; bounding box; localization algorithm; full search

### 1 引言

无线传感器网络(WSN, wireless sensor network)是指由大量传感器节点组成的以无线方式通信的自组织网络, 其应用的关键技术之一是节点定位技术<sup>[1]</sup>; 如环境监测、地震救灾、目标检测与跟踪等应用, 这些应用都要求传感器节点知道自己的位置信息。

根据定位过程中是否要测距可以将定位技术分为基于测距 (range-based) 和无需测距 (range-free) 两类。基于测距的定位算法需要测量未知节点与已

知节点的距离或者角度等信息, 然后利用三边定位法、三角定位法计算出未知节点的位置<sup>[2~4]</sup>; 基于测距的定位算法代表有接收信号强度指示(RSSI)、到达时间(TOA)、到达时间差(TDOA)、到达角度(AOA)等。无需测距的定位算法不需要测量节点间的距离或角度, 而是利用网络连通性等信息计算节点位置<sup>[5~7]</sup>; 无需测距定位算法的代表有质心定位算法、DV-Hop 算法、凸规划算法、APIT 算法等。基于测距的定位算法的性能一般高于无需测距算法的性能, 但是无需测距的定位算法不需要专门的硬件用于测距, 节点成本和功耗更具优势。

根据定位过程中主要计算的完成地点可以将定位技术分为集中式定位算法与分布式定位算法<sup>[9, 10]</sup>。集中式定位算法需要每个节点将本地测距信息或连接信息发送给 Sink 节点, 由 Sink 节点计算每个节点的位置, 计算完成后再将位置信息发给未知节点。集中式算法的优点是能从全局处理数据、定位更精确, 缺点是需要大量的消息传输, 导致网络能耗和带宽的消耗; 分布式定位算法是未知节点利用本地测距信息或连接信息进行位置计算, 因而它的性能可能不如集中式定位算法, 但它是能量和带宽有效的, 而且能很好地随网络规模缩放。在实际的大规模 WSN 定位中, 分布式算法更有吸引力。

本文提出一种基于测距的 WSN 全搜索分布式定位算法, 该算法在网络中通过多轮运行、逐步求精得到节点定位坐标。每一轮运行过程中, 首先通过节点测距得到邻居节点的坐标和距离信息, 然后通过边框定界方法确定节点可能存在的位置区域, 最后通过全搜索方法在该区域搜索最佳估计点, 该点坐标即为节点本轮的定位坐标。

## 2 基于边框定界的 WSN 分布式全搜索定位算法

### 2.1 平面定位问题数学模型

设节点  $S_i$  的坐标未知, 它有  $m$  个已知坐标为  $(x_j, y_j), j=1, 2, \dots, m$  的一跳邻居节; 另外, 通过测距知道节点  $S_i$  与节点  $S_j$  的距离为  $d_{ij}$ , 那么节点  $S_i$  的定位坐标  $(\hat{x}_i, \hat{y}_i)$  可以通过最小化下式进行估计。

$$\arg \min_{(\hat{x}_i, \hat{y}_i)} \sum_{j=1}^m \left( \sqrt{(\hat{x}_i - x_j)^2 + (\hat{y}_i - y_j)^2} - d_{ij} \right)^2, j=1, 2, L, m \quad (1)$$

上式的典型求解方法有似然估计算法、基于泰勒展开的最小二乘算法<sup>[9]</sup>、线性规划算法<sup>[11]</sup>等。这些算法都涉及到复杂的矩阵计算, 且在有误差的条件下某些算法的误差会累计<sup>[12]</sup>, 造成最后的定位误差特别大。基于这样考虑, 本文采用搜索算法的思想<sup>[8]</sup>, 提出基于边框定界的 WSN 分布式全搜索定位算法。

### 2.2 边框定界全搜索定位算法思想

边框定界全搜索算法的思想可以用下面 3 步来概括。

**Step 1** 用边框定界思想确定未知节点的可能存在的区域, 如图 1 所示。即以已知节点为中心,

以与未知节点的 2 倍测量距离为边长, 画正方形, 所有正方形重合的区域为未知节点可能存在的区域。从图 1 中可以看到未知节点的坐标应该满足

$$\begin{cases} x_i \in [\max(x_j - d_{ij}), \min(x_j + d_{ij})] \\ y_i \in [\max(y_j - d_{ij}), \min(y_j + d_{ij})] \end{cases}, j=1, 2, L, m \quad (2)$$

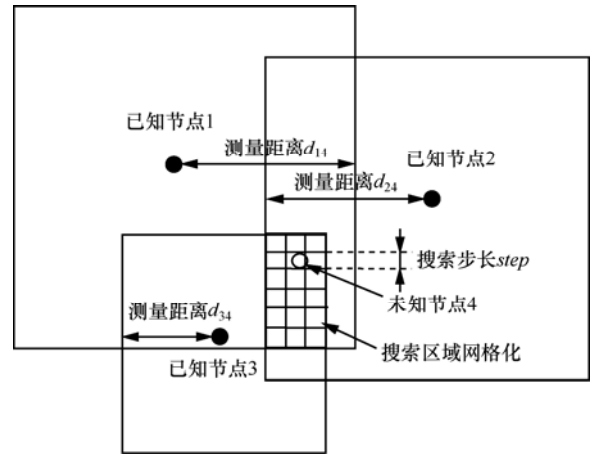


图 1 边框定界法确定搜索区域

在实际的 WSN 应用中, 由于存在测距误差, 可能使得  $d_{ij} < d_{rij}$  ( $d_{rij}$  为真实距离), 这时  $S_i$  必定不在图 1 所示的区域, 从而  $S_i$  的坐标不满足式(2); 或者式(2)给出的区域超过了网络的物理区域  $[0, x_{\max}]$  或  $[0, y_{\max}]$ 。考虑到这点后, 可以用下式确定节点  $S_i$  可能存在的区域 ( $\alpha \geq 1$ , 它用来控制区域大小), :

$$\begin{cases} x_i \in [x_L = \max(x_j - \alpha * d_{ij}, 0), x_H = \min(x_j + \alpha * d_{ij}, x_{\max})] \\ y_i \in [y_L = \max(y_j - \alpha * d_{ij}, 0), y_H = \min(y_j + \alpha * d_{ij}, y_{\max})] \end{cases} j=1, 2, L, m \quad (3)$$

**Step 2** 将搜索区域网格化。式(3)已经确定了搜索矩形区域的大小, 然后利用下式将搜索区域网格化,  $step$  为搜索步长 (或网格大小)。

$$\begin{aligned} & \text{for } x = x_L : step : x_H \\ & \text{for } y = y_L : step : y_H \end{aligned} \quad (4)$$

**Step 3** 在每个网格点上计算目标函数, 如下式所示。目标函数最小的点即为节点  $S_i$  的最佳估计点, 该点坐标即为估计坐标  $(\hat{x}_i, \hat{y}_i)$ 。

$$f(x, y) = \sum_{j=1}^m \left( \sqrt{(x - x_j)^2 + (y - y_j)^2} - d_{ij} \right)^2, j=1, 2, L, m \quad (5)$$

从上述边框定界全搜索定位算法的思想可以看到, 该算法的复杂度主要取决于 2 个方面: 搜索区域的网格大小和节点的邻居节点数。搜索区域网

格小，则每次搜索点数多，算法复杂度变高，定位精度也高；反之，搜索区域网格大，则每次搜索点数少，算法复杂度降低，定位精度也降低。节点邻居节点数多，则目标函数计算复杂度提高，定位精度也随之提高；反之亦然。本文在算法研究中，将  $\alpha$  设为 1.5、网格大小  $step$  设为  $0.025R$ 、参与搜索的邻居节点集合 ( $S\_NeiG$ ) 的最大元素数设为 7，即  $m$  最大值为 7；另外为实现网络节点的快速定位，将  $S\_NeiG$  的最少元素设为 2，即只要有 2 个邻居节点，就采用上面搜索算法进行节点坐标定位。

### 2.3 该算法在网络中的应用

由于本文算法是一个典型的分布式算法，初始时节点的已知坐标的邻居节点偏少，甚至没有，因此要实现整个网络的节点定位，算法必须按轮运行多轮，每轮需定位节点与已知位置的邻居节点进行测距（无额外的消息传输），然后采用边框定界全搜索算法定位自己的坐标，最后通过多轮运行逐步提高网络节点定位覆盖率和定位精度。实际仿真中发现，一般运行 10~30 轮左右即可以达到较好的定位性能。

本文算法在运行过程中将整个网络中的节点分为 4 类：锚节点、高精度定位节点、低精度定位节点、未知节点。锚节点是网络初始时知道自己坐标的节点；低精度定位节点是指节点定位过程中发现它的定位坐标与其邻居节点坐标之间的距离  $d_{Lij}$  和这 2 个节点的测量距离  $d_{ij}$  存在矛盾，即  $d_{Lij} \geq \beta d_{ij}$ 。考虑到有  $d_{ij}$  误差，本文算法仿真中取  $\beta=1.5$ ；高精度定位节点是指节点定位过程中没有发现矛盾。初始时网络中只存在锚节点和未知节点，随着定位算法的运行，网络中出现高精度定位节点和低精度定位节点。

具体的基于边框定界的 WSN 分布式全搜索定位算法如下。

**Step1** 每轮开始已知节点向周围节点广播自身的信息，广播  $T(T \geq 1)$  次。广播信息主要包括节点 ID、节点坐标信息、测距信息等内容。已知节点由锚节点和已经定位成功的普通节点（包括高精度定位节点和低精度定位节点）组成。

**Step2** 节点  $S_i$ （非锚节点）接收邻居节点  $S_j$  的广播信息，记录成功接收到邻居节点消息的次数  $C_{ij}$  和测距结果  $d_{ij}$ ，然后将这些信息存储在本地，如果  $C_{ij} \geq T/2$ ，则认为  $S_j$  是  $S_i$  的邻居节点，最终得到节点  $S_i$  的邻居节点集合及测距信息。

**Step3** 每轮广播结束后，节点  $S_i$  按下面步骤进行基于边框定界的全搜索定位。

1) 从节点  $S_i$  的邻居节点集合中确定  $S\_NeiG$ 。如果节点  $S_i$  的实际邻居节点多于 7 个，从中挑选定位等级最高的 7 个邻居节点构成  $S\_NeiG$ ，否则  $S\_NeiG$  由全部邻居节点组成。挑选顺序为锚节点、高精度定位节点、低精度定位节点，同级的情况下随机挑选。

2) 采用边框定界法确定节点  $S_i$  的搜索区域。当搜索区域出现矛盾，即  $x_L \geq x_H$  或  $y_L \geq y_H$  时，说明节点  $S_i$  的  $S\_NeiG$  有定位误差较大邻居节点存在，此时节点  $S_i$  广播此消息，并结束自己的定位。 $S\_NeiG$  中的节点收到该消息后，若自己的定位等级为高精度定位节点，则随机降级为低精度定位节点。

3) 在搜索区域内搜索目标函数最小点。当节点平均目标函数值小于某个值（仿真中取 2）或节点  $S_i$  不为高精度定位节点时，更新节点  $S_i$  的坐标。节点平均目标函数值  $\bar{f}$  定义为 ( $n$  为  $S_i$  的邻居节点集合  $S\_NeiG$  的元素数)

$$\bar{f} = f(x, y)_{\min} / n \quad (6)$$

4) 依次计算节点  $S_i$  与其邻居节点  $S_j$  的距离  $d_{Lij}$ 。如果  $d_{Lij} \geq 1.5 \times d_{ij}$ ，设置节点  $S_i$  的定位等级为低精度定位节点，否则为高精度定位节点；另外，如果  $S_j$  为高精度定位节点，则随机降级为低精度定位节点。

**Step4** 每轮计算结束后，在新的一轮中已定位节点广播自己的定位信息，开始新一轮的算法运行。

## 3 仿真实验

### 3.1 测距模型及参数定义

在本文算法仿真实验中，节点的测距模型如式(7)所示。

$$d_{ij} = d_{r_{ij}} \times |(1 + n_f \times randn)| \quad (7)$$

其中， $d_{ij}$  为节点  $S_i$  与节点  $S_j$  的测量距离， $d_{r_{ij}}$  为实际距离， $n_f$  代表测距噪声水平， $randn$  服从  $N(0, 1)$  分布。当  $d_{ij} > R$  ( $R$  为节点通信半径) 时，认为测距失败，即 2 个节点不是邻居节点；反之，当  $d_{ij} \leq R$  测距成功，2 个节点是邻居节点，记录 2 个节点的  $d_{ij}$ 。

为避免混淆，先给出几个本文将要使用的性能

指标的定义。

**定义 1** 测距误差。平均的测量距离与实际距离的差值的绝对值相对于节点无线射程  $R$  的大小。计算公式为

$$\varepsilon_{\text{range}} = \frac{1}{N} \sum_{(i,j)} \frac{|d_{ij} - d_{r_{ij}}|}{R} \quad (8)$$

**定义 2** 定位误差。网络中平均的节点定位坐标与实际坐标的距离相对于节点无线射程  $R$  的大小。计算公式为

$$\varepsilon_{\text{est}} = \frac{1}{n} \sum_{i=1}^n \frac{\sqrt{(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2}}{R} \quad (9)$$

**定义 3** 定位覆盖率。定位完成后，已定位节点（高精度定位节点与低精度定位节点数目）相对于网络初始未知节点数目的比值。

### 3.2 实验结果

为了验证本文算法的有效性，本文在 Matlab 环境下进行了实验。仿真区域为  $50\text{m} \times 50\text{m}$  的正方形，总节点 100 个，实验结果是 30 次实验的平均结果。

#### 1) 定位误差随测距误差的变化。

仿真实验环境为：9 个锚节点，91 个未知节点；锚节点均匀分布（人工布置），分别在  $[5, 5], [5, 25], [5, 45], [25, 5], [25, 25], [25, 45], [45, 5], [45, 25], [45, 45]$ ，未知节点随机生成；所有节点的通信半径  $R=11\text{m}$ 。仿真中通过改变  $n_f$  得到不同的测距误差，由此得到的测距误差、定位误差随  $n_f$  的变化情况（算法运行 20 轮）如图 2 所示。从图 2 中可以看到，定位误差随着测距误差增长而增长，但增长速度比测距误差慢；当测距误差达到  $0.128R$  时，定位误差只有  $0.062R$  左右，取得了不错的定位性能。另外，当测距误差为零时，定位误差仍然存在，这主要是由于搜索区域的网格精度（取为  $0.025R$ ）造成。

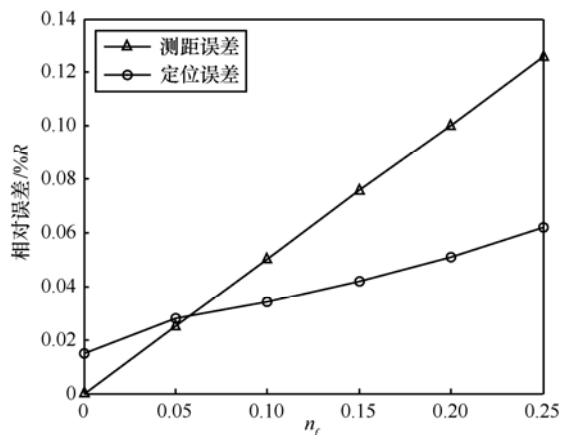


图 2 测距误差对定位误差的影响

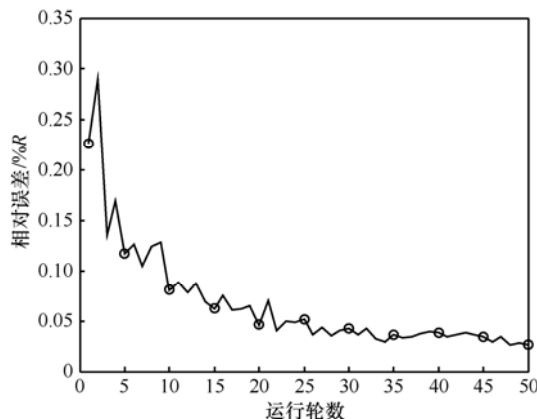


图 3 定位误差随运行轮数的变化

图 3、图 4 为  $n_f=0.25$  时，一个实验场景的定位误差、各种类型节点数随着算法运行轮数的变化情况。从图 3 中可以看到，随着算法运行轮数增多，定位误差逐渐下降，但下降过程不是严格单调的，当运行到 21 轮以后，定位误差基本已经在  $0.05R$  以下，后面继续运行对降低定位误差影响不大。

从图 4 中可以看到，本实验场景中当算法运行到第 7 轮时就实现了全部节点的定位。同时，由于算法中存在对高精度定位节点（节点类型 1）随机降级为低精度定位节点（节点类型 2）的降级机制，

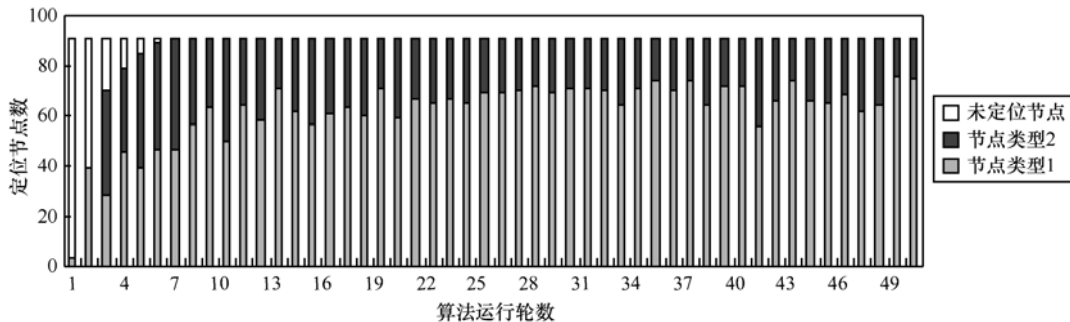


图 4 各种类型节点数随着算法运行轮数的变化

这使得在整个场景中这 2 类节点一直存在。

2) 锚节点随机分布情况下, 定位误差随锚节点数量的变化。

由于 WSN 在某些应用中, 锚节点无法人工部署, 因此定位算法在锚节点随机分布情况下的性能也很重要。此时实验仿真环境为: 100 个节点随机分布在 50m×50m 的区域, R=11m,  $n_f=0.25$ 。30 个实验场景的平均结果如图 5 所示。从图 5 可以看到, 在锚节点随机分布情况下, 9 个锚节点时定位误差为 0.25R 左右, 远大于手工分布的 0.062R; 随着锚节点数量的增加, 定位误差迅速下降, 当锚节点数量达到 20% 以上后, 取得了和手工分布 9 个锚节点差不多的定位精度。出现这种情况的原因在于锚节点随机分布时, 较少的锚节点无法保证在 WSN 中均匀分布, 从而造成部分区域的节点的定位误差较大, 而随着锚节点增加到 20% 以后, 基本能保证每个未知节点都至少能收到一个锚节点的定位信息, 从而提高了定位性能。仿真结果说明当锚节点数大于 20% 以后, 本文算法可以应用到锚节点随机分布的场景。

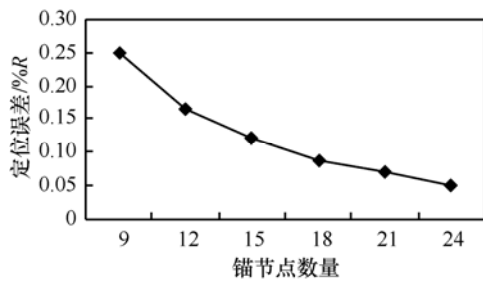


图 5 定位误差随锚节点数量变化关系

### 3.3 与其他算法的结果比较

文献[10]也采用式(7)作为测距模型, 与本文不同的是, 该文不仅假设测距有误差, 也假设初始锚节点的位置也存在误差。提出利用 ESOCP (edge-based SOCP) 算法得到节点的初始位置, 利用 NCSG(NonConvex sequential greedy)算法优化节点的位置。人工布置 8 个锚节点 (无本文人工布置的中间节点[45, 45]), 在锚节点位置误差小于 0.2R 条件下, 定位误差与  $n_f$  的关系如图 6 所示。

从图 6 与图 2 的结果对比中可以看到, 在假设锚节点位置无误差条件下, 本文结果优于文献[10]的最优结果 (ESOCP+NCSG)。在同样假设锚节点位置误差小于 0.2R 条件下,  $n_f$  从 0.05 变化到 0.25 时得到的平均定位误差为: 0.183 3R、0.186 2R、

0.181 6R、0.191 1R、0.196 1R。在  $n_f$  较小时定位误差明显大于文献[14]的结果, 而在  $n_f=0.25$  时, 定位误差已经差不多。主要原因在于本文算法没有对锚节点的坐标进行求精, 因此在  $n_f$  较小时, 定位误差主要决定于锚节点的位置误差。综上所述, 本文认为本文算法取得了与当代复杂算法相当的定位性能。

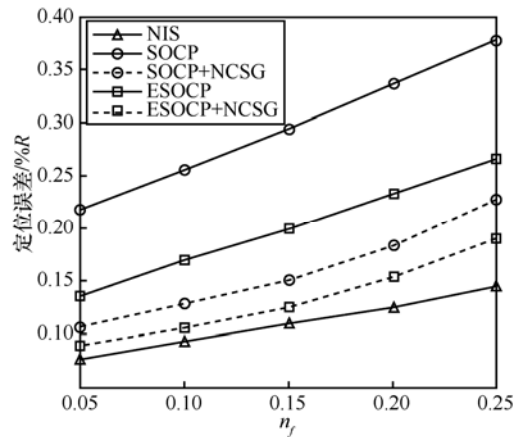


图 6 文献[10]的结果

## 4 结束语

定位技术是无线传感器网络实际应用的关键技术之一。本文在边框定界方法的基础上, 提出一种基于全搜索的 WSN 分布式定位算法, 该算法具有通信开销小、易于理解、易于工程实现的优点, 并且定位性能与算法复杂度可以通过调整搜索步长和参与搜索的邻居节点数进行控制, 因此对工程实践应用具有一定参考意义。在下一步工作中, 需要重点解决快速搜索算法问题以降低算法实现的复杂度。

### 参考文献:

- [1] STOJIMENOVIC I. Handbook of Sensor Networks Algorithms and Architectures [M]. John Wiley & Sons Inc publication, 2005, 277-310.
- [2] ZHANG S G, CAO J N, CHEN L J, et al. On accuracy of region-based localization algorithms for wireless sensor networks[A]. Proc of the IEEE 6th International Conference on Mobile Adhoc and Sensor Systems[C]. Macau, China, 2009. 30-39.
- [3] CHEN Y F, SHU L, LI M C, et al. The insights of DV-based localization algorithms in the wireless sensor networks with duty-cycled and radio irregular sensors[A]. Proc of 2011 IEEE International Conference on Communications[C]. Kyoto, Japan, 2011. 1-6.

- [4] LI X, SHI H C, SHANG Y. A sorted RSSI quantization based algorithm for sensor network localization[A]. Proc of the 11th Int Conf on Parallel and Distributed Systems[C]. Los Alamitos, CA, USA, 2005. 557-563.
- [5] NICOLESCU D, NATH B. Ad-hoc positioning systems(APS)[A]. Proc of the IEEE Global Telecommunications Conference[C]. San Antonio, USA, 2001. 2926-2931.
- [6] DOHERTY I, PISTER K S, GHAOUI L E. Convex position estimation in wireless sensor networks[A]. Proc of the IEEE INFOCOM 2001[C]. Anchorage, USA, 2001. 1655-1663.
- [7] HE T, HUANG C D, BLUM B M. Range-free localization schemes in large scale sensor networks[A]. Proc of the 9th Annual Int Conf on Mobile Computing and Network[C]. New York, USA, 2003. 81-95.
- [8] 朱博, 陈曙. 一种无线传感器网络质心定位改进算法[J]. 传感技术学报, 2010, 23(6): 868-872.
- ZHU B, CHEN S. An improved centroid localization algorithm for wireless sensor network[J]. Chinese Journal of Sensors and Actuators, 2010, 23(6): 868-872.
- [9] 罗旭, 柴利, 杨君. 无线传感器网络 SL-n 迭代定位算法[J]. 通信学报, 2011, 32(5): 129-138.
- LUO X, CHAI L, YANG J. SL-n iterative localization algorithm in wireless sensor networks[J]. Journal of Communications, 2011, 32(5): 129-138.
- [10] SHI Q J, HE C, CHEN H Y, *et al.* Distributed wireless sensor network localization via sequential greedy optimization algorithm[J]. IEEE Trans Signal Process, 2010, 58(6): 3328-3340.
- [11] 王珊珊, 殷建平, 张国敏, 蔡志平. 求解无线传感器网络定位问题的线性规划算法[J]. 计算机研究与发展, 2009, 46(5): 705-712.
- WANG S S, YIN J P, ZHANG G M, *et al.* A linear programming algorithm for wireless sensor networks localization[J]. Journal of

Computer and Development, 2009, 46(5): 705-712.

- [12] 王小平, 罗军, 沈昌祥. 无线传感器网络定位理论和算法[J]. 计算机研究与发展, 2011, 48(3): 353-363.

WANG X P, LUO J, SHEN C X. Theory and algorithms on localization in wireless sensor networks[J]. Journal of Computer and Development, 2011, 48(3): 353-363.

#### 作者简介:



姚英彪 (1976-), 男, 湖北松滋人, 博士, 杭州电子科技大学副教授, 主要研究方向为无线传感器网络、嵌入式系统设计等。



曾嵘 (1976-), 男, 江苏淮安人, 博士, 杭州电子科技大学副教授, 主要研究方向为移动通信关键传输技术。



易志强 (1976-), 男, 重庆人, 博士, 杭州电子科技大学讲师, 主要研究方向为 SoC 设计和数字通信。