

基于三角形理论的无线传感器网络定位算法*

万国峰¹, 钟俊²

(1. 西北民族大学电气工程学院, 兰州 730030; 2. 四川大学电气信息学院, 成都 610065)

摘要: 针对基于垂直平分线的区域定位算法(MBLA)存在定位精度低、迭代次数多的缺点,提出了基于三角形理论的区域定位算法(TBLA)。该算法以参与定位的两个锚节点连线作为一条边,以待定位节点与这两个锚节点的RSSI测距值作为另两条边构造三角形,然后根据三角形的形状进行定位。仿真结果表明,在相同通信半径下,TBLA定位误差只是MBLA的1/5,迭代次数减少了2/3以上,具有较高的应用价值。

关键词: 无线传感器网络; 节点定位; 垂直平分线; 三角形

中图分类号: TP393; TP301.6

文献标志码: A

文章编号: 1001-3695(2013)01-0249-03

doi:10.3969/j.issn.1001-3695.2013.01.064

Triangle-based localization algorithm for wireless sensor networks

WAN Guo-feng¹, ZHONG Jun²

(1. School of Electrical Engineering, Northwest University for Nationality, Lanzhou 730030, China; 2. College of Electric & Information Engineering, Sichuan University, Chengdu 610065, China)

Abstract: Due to the shortcoming of MBLA (midnormal-based area localization algorithm) which has low localization precision and more iterative times, this paper proposed TBLA (triangle-based localization algorithm). This algorithm used two RSSI ranging values between the node to be localized with two anchor nodes, and then constructed a triangle to locate the unknown node. Simulation result shows that TBLA's locating error is only 1/5 of MBLA's, and more importantly, the iterative times are less 1/3 of MBLA's in the same communication radius. So the application value of this algorithm has high.

Key words: wireless sensor networks (WSN); node localization; midnormal; triangle

根据定位机制,无线传感器网络定位分为两大类^[1,2]:基于测距的定位和基于非测距的定位。基于测距的定位通过测量节点间点到点的距离或角度信息,并使用三边测量法、三角测量法或最大似然估计法计算节点位置,这类算法在获得较为精确定位结果的同时产生了大量计算和通信开销,对传感器节点的能量提出了极大考验。而无须测距的定位则仅仅依靠网络连通性等信息进行定位。无须测距的定位提出的定位机制主要有质心算法、DV-Hop算法、Amorphous算法和APIT算法等。由于无须测量节点间的绝对距离或方位,降低了对节点硬件的要求,使得节点更适合于大规模传感器网络。距离无关的定位机制定位性能受环境因素的影响小,虽然定位的误差相应有所增加,但定位精度能够满足多数传感器网络应用研究的要求,是目前大家普遍重点关注的定位机制^[3]。

质心算法^[4]由于其容易实现,被广泛研究。但质心算法需要大量参考节点支撑,每个未知节点周围必须有一定数量的参考节点存在,因而锚节点密度和网络节点分布情况对定位精度的影响较大。另外,参与定位的参考节点所接收的信息大小不一样,当各个节点所接收信息的方差很大时,定位精度就会变得很低^[5]。

为了解决单跳质心算法对高锚节点密度的依赖,提高定位比率,文献[6,7]提出了多跳质心算法,并用线性回归等算法进行去中心化。不少文献将RSSI测距应用于质心定位中,如

文献[5,8~10],从而得到多种基于RSSI的加权质心定位算法。这些算法以未知节点测到的锚节点的距离(或进一步处理)作为质心定位的加权因子,收到了良好的效果,也为无线传感器网络定位提供了新思路;不过,当锚节点分布不均匀,尤其分布在区域中心时,定位覆盖率比较低。文献[10]提出了基于三角形角度的加权算法,能减小因锚节点分布不均匀而产生的定位误差。但是以上算法都比较复杂,对能量受限的无线传感器网络是一个考验。

为克服前面提到的质心算法复杂度高且定位覆盖率不高的缺点,文献[11]在分析比较基于区域的无须测距的定位算法APIT^[12]的基础上,提出了基于垂直平分线的区域定位算法。

1 相关工作

1.1 基于区域的无须测距的定位算法APIT

基于区域的无须测距的定位算法APIT需要有一定比例的传感器节点并须知道它们自身位置信息的锚节点。根据待定位节点的无线射程范围,获得能与待定位节点通信的锚节点集合。在该集合中任意选取三个锚点构成三角形,三角形数目由排列组合计算得到。待定位节点所在的区域被这些三角形分割覆盖;通过判断待定位节点是在三角形之内或之外来逐渐缩小待定位节点所在的区域。如何判断待定位点是否在由三

收稿日期: 2012-05-14; 修回日期: 2012-06-22 基金项目: 西北民族大学中央高校基本科研业务费专项资金资助项目(ZYZ2012058); 四川省科技支撑计划资助项目(2011GZ005)

作者简介: 万国峰(1970-),男,甘肃陇西人,副教授,硕士,主要研究方向为无线传感器网络、信号处理(gfwan_1970@163.com); 钟俊(1972-),男,重庆人,副教授,博士,主要研究方向为无线传感器网络、嵌入式系统。

个锚点构成的三角形之内,称为 PIT (point-in-triangulation) 判定。文献[12]中有两条定理支持这种判定,但在实际中无法用穷举的方法来判断每一个方向。因此采用近似的 PIT 判定,即如果待定位节点的邻居节点中没有比它更靠近三角形的三个顶点或更远离三角形的三个顶点,则待定位节点在该三角形内部;否则就在外部。但是这种判断方式不是 100% 准确,特别是当待定位节点在三角形的边界时,更容易出错,在某些情况下,其错误率高达 14%^[11]。

APIT 算法通过判断待定位节点是在锚节点构成的三角形之内或之外来逐渐缩小待定位节点所在的区域。因此,定位精度在很大程度上既取决于锚点密度又取决于节点密度。在每次进行 PIT 判定时,待定位节点须与邻居节点比较各自跟锚点之间的信号强度。这就会导致本来不需要定位的邻居节点也要获知锚点的信号强度,而且比较过程必然会引起信息的分享,导致不必要的通信开销,消耗大量的节点能量。这不适用于能量受限的无线传感器网络。

1.2 基于垂直平分线的区域定位算法

MBLA^[11]弥补了 APIT 算法的不足,它无须与邻居节点进行信息分享和通信,避免了能量的损耗。而 MBLA 跟邻居节点的分布无关,也不会有 PIT 定位算法中判定错误问题。具体算法如下:

MBLA 是一种基于垂直平分线的无须测距的位置估计算法。算法的核心思想是通过判断待定位节点是在两个锚点连线的垂直平分线哪一侧来进行位置估算,垂直平分线侧的判定算法称为 PIWS (point in which side) 判定。

在每次进行定位估计时,待定位节点首先获得可与之通信的锚点集合,在该锚点集合中每次任意选择两个锚点,比较待定位节点与这两个锚点的信号强度。在理想情况下,即没有任何阻挡时,待定位节点相比于信号强度弱的锚点而言离信号强度强的锚点更近。两个锚点连线的垂直平分线把整个区域分成两部分,而待定位节点一定位于信号强度强的锚点驻留的一侧,如图 1 所示,待定位节点一定位于垂直平分线的 a 侧。整个定位过程就是不断地重复这个判定过程,每一次判定都会缩小待定位节点可能的所在区域,直到所有的垂直平分线穷尽,或定位精度已达到所需精度,则算法结束。若没有达到定位精度,则求垂直平分线构成的多边形的质心。

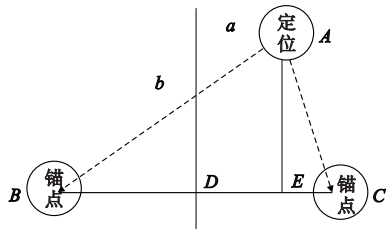


图1 PIWS判定

未知节点与锚节点的位置关系多为图 2(a)(d) 所示。可以看出,如果用文献[11]的 PIWS 判定,则每次判定后的范围比较大,故迭代的次数多,消耗的能量也多,缺点明显。本文提出一种以待定位节点与两个锚节点的 RSSI 测距值及两个锚节点的连线构成的三角形的形状,判断待定位节点位置的基于三角形理论的定位算法。

2 TBLA

TBLA 是一种基于三角形的位置估计算法。算法的核心

思想是通过未知节点与其通信的两锚节点构成的三角形的形状,判断待定位节点位置。

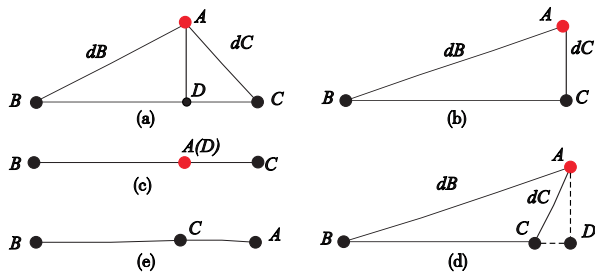


图2 定位情况图

2.1 算法描述

TBLA 在每次进行定位估计时,待定位节点首先获得可与之通信的锚点集合,根据接收到的 RSSI 值得到待定位节点与锚节点的距离,构成距离集合。在该锚点集合中每次任意选择两个锚点,与待定位节点构成三角形,如图 2 所示。

在图 2(a) 中, AD 是锚节点连线 (BC) 的垂线,当 $dB \geq dC$, 有

$$\frac{dB}{dC} \geq \frac{BD}{CD} \quad (1)$$

$$dB^2 \leq dC^2 + BC^2 \quad (2)$$

图 2(b)(c) 是 (a) 的极限, (b) 中 AC 垂直于 BC , 也满足式 (1)(2), 而 (c) 满足式 (3):

$$dB + dC = BC \quad (3)$$

图 2(d)(e) 满足式 (4), (e) 还满足式 (5):

$$dB^2 > dC^2 + BC^2 \quad (4)$$

$$dB - dC = BC \quad (5)$$

对于图 2(a)(b) 的情况, 即满足式 (2) 时, 以点 B 为基点按 $dB/(dB + dC)$ 的比例作 BC 的垂线, 把整个区域分成两部分, 那么, 未知节点肯定在锚节点 C 所在的这部分区域。而图 2(c) 中, 未知节点的位置就是垂线与 BC 的交点 (D 点)。若满足式 (4), 即如图 2(d) 的情况, 过 C 点作线段 BC 的垂线, 把区域分为两部分, 则未知节点一定在 B 点的对侧。对于满足式 (5) 的情况, 在 BC 的基础上加上 dC 的长度后得到的 A 点就是定位点。整个定位过程就是不断地重复这个判定过程, 每一次判定都会缩小待定位节点可能的所在区域, 直到所有的锚节点穷尽或定位精度已达到所需精度, 则算法结束。

2.2 算法步骤

a) 各个锚节点周期性地向周围广播自身节点 ID 和坐标信息。

b) 通过一定时间的通信, 确定在所有的锚点中哪些锚点是待定位节点可通信的, 并通过 RSSI 确定未知节点与锚点之间的距离。这些锚点构成了可通信的锚点集合, 锚点集合通过排列组合与待定位节点构造图 2 所示的三角形。

c) 三角形如果满足式 (3), 则点 D 就是待定位节点; 满足式 (5), 点 A 就是待定位节点。如果定位误差满足要求, 则转 e); 否则执行 d), 并与 f) 得到的质点配合求二者的重心点。

d) 对于满足式 (2) 的情况, 则按 $dB/(dB + dC)$ 的比例作 BC 的垂线, 把区域分成两个区域, 未知节点就在锚点 C 所在的区域。若满足式 (4), 则通过 C 点作 BC 的垂线, 把区域分成两个区域, 未知节点就在 B 点的对侧区域。

e) 聚合。重复 c), 直到所有的锚点组合都被测试过, 或已达到所需的定位精度。

f) 计算聚合之后的重叠区域的重心。

2.3 算法精度分析

假设平均可通信的锚点数为 N , 则在 MBLA 中, 垂直平分线数为 $K = C_N^2 = N(N-1)/2$ 。根据文献[11], K 条直线把一个区域分成 $S(N)_{\text{MBLA}}$ 个子区域, 而 APIT 算法的分割粒度为 $S(N)_{\text{APIT}}$, 那么 $S(N)_{\text{MBLA}} = 3S(N)_{\text{APIT}}$, 说明 MBLA 比 APIT 分割更细, 定位更准确。TBLA 对空间的分割和 MBLA 一样细, 都是 APIT 算法的 3 倍。但是, TBLA 对区域的分割是按未知节点收到的锚节点的 RSSI 的大小比例去分割的, 其更加准确, 更容易达到定位误差的要求。而对于在锚点与待定位节点之间有阻挡物时, 由于信号强度无法反映实际距离远近程度, 可能会导致单次的判定错误。但由于 TBLA 采用的是多次判定, 因此可以屏蔽掉这种单次的判定错误。

3 算法仿真及分析

本文用 MATLAB 7.10 进行仿真。选未知节点个数为 n ; 参考节点个数为 m ; 节点通信半径为 R ; 迭代次数为 k ; 节点随机布置在一个 $100\text{ m} \times 100\text{ m}$ 正方形区域内。

这里采用定位误差率来衡量定位算法的性能, 较小的定位误差说明定位算法的精确度较高。定位误差计算式为

$$\text{meanerror} = \frac{\sum_{i=1}^n \|x_{ia} - t_{ia}\|}{nR} \quad (6)$$

3.1 定位误差与迭代次数的关系

设定锚节点数 $m = 10$, 未知节点个数 $n = 40$, 所有节点通信半径 $R = 20\text{ m}$, 对 MBLA 和 TBLA 两种算法进行仿真, 结果如图 3 所示。要达到定位误差 0.50 m , MBLA 要进行 15 次以上的迭代才能得到, 而 TBLA 只需要 5 次迭代即可达到相同的精度。算法的计算工作量减少了 $2/3$ 以上, 这对于能量受限的无线传感器网络来说意义重大。

3.2 可通信锚节点数对定位误差的影响

需要定位的节点数取 40 个, 改变锚节点的数量, 得到定位误差与锚节点数的关系, 如图 4 所示。可以看出, 在锚节点数小于 3 时, 两种算法的定位误差都比较大, 但 TBLA 的误差是 MBLA 误差的 $1/3$ 左右, 这是因为 TBLA 是根据接收到的锚节点信号强度大小的比例把区域划分成两部分, 甚至在满足式 (3)(5) 时直接就得到比较准确的位置, 而 MBLA 只是以锚节点连线的平分线划分区域, 定位误差自然较大。当锚节点数大于 3, TBLA 的定位误差已经在 0.5 m 以下。随着锚节点数的增加, 区域划分越来越小, MBLA 的定位精度也提高, 二者逐渐接近, 再增加锚节点个数, 则定位精度趋于稳定, 且都能满足定位要求。

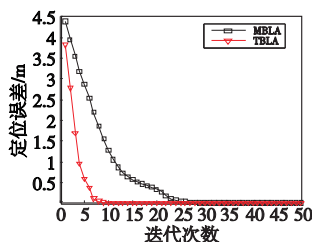


图3 定位误差与迭代次数关系

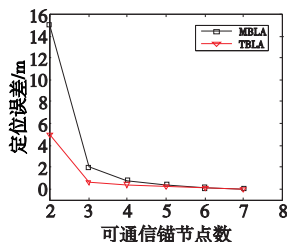


图4 定位误差与锚节点数关系

3.3 通信半径对定位精度的影响

图 5 为 100 个节点、锚节点比例为 10% 的情况。可以看出, 随着通信半径的增加, 两种算法的定位误差都有明显的下降。

这一方面是由于随着通信半径的增大, 未知节点的可定位锚节点数增加; 另一方面, 信号强度增加, 测距更加准确。但是当半径超过 30 m 后, MBLA 的定位误差稍有增加, 这是由于通信半径增大后, 两个锚节点连线的垂直平分线所分得区域增大, 定位区域也增大的原因。而 TBLA 的定位误差继续下降, 平均定位误差是 MBLA 定位误差的 $1/5$, 显示出明显的优越性。

3.4 定位误差与网络规模的关系

再来看算法对网络规模的适应程度。由图 6 可知, 在节点数小于 230 时, TBLA 的定位误差小于 MBLA, 尤其在网络规模小于 150 时更加明显, MBLA 的定位误差约是 TBLA 定位误差的 3 倍。当网络规模大于 100, TBLA 的定位误差会有所增加, 但不明显, 还是在 0.25 m 以下。但随着网络规模的增大, 节点之间通信互相干扰会增加, 从而影响测距的准确, 造成对区域划分误差的增大, 而这种干扰对 MBLA 的影响不明显。

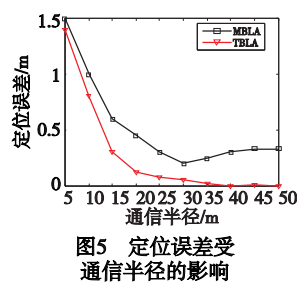


图5 定位误差受通信半径的影响

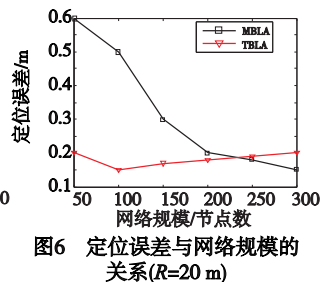


图6 定位误差与网络规模的关系($R=20\text{ m}$)

4 结束语

无线传感器网络定位误差及能量消耗是定位的重点。本文针对文献[11]提出的 MLBA 定位误差较大、迭代次数多的缺点, 提出改进算法——TBLA。仿真结果表明本文提出的算法定位精度高、迭代次数少, 适合能量受限的无线传感器网络。

参考文献:

- [1] 周正. 无线传感器网络的节点自定位技术[J]. 中兴通讯技术, 2005, 11(4): 51-56.
- [2] VO D M N, VO D, CHALLA S, et al. Nonmetric MDS for sensor localization [C]//Proc of the 3rd International Symposium on Wireless Pervasive Computing. [S. l.]: IEEE Press, 2008: 396-400.
- [3] 孙利民. 无线传感器网络[M]. 北京: 清华大学出版社, 2005: 148-155.
- [4] BULUSU N, HEIDEMANN J, ESTRIN D. GPS-less low cost outdoor localization for very small devices[J]. IEEE Personal Communications Magazine, 2000, 7(5): 28-34.
- [5] 刘运杰, 金明录, 崔承毅. 基于 RSSI 的无线传感器网络修正加权定位算法[J]. 传感技术学报, 2009, 23(5): 717-721.
- [6] 安恂, 蒋挺, 周正. 一种用于无线传感器网络的质心定位算法[J]. 计算机工程与应用, 2007, 43(20): 136-138.
- [7] 胡斌, 姚维照. 基于线性回归的无线传感器加权质心定位算法[J]. 太原理工大学学报, 2011, 42(5): 499-502.
- [8] 陈维克, 李文峰, 首珩, 等. 基于 RSSI 的无线传感器网络加权质心定位算法[J]. 武汉理工大学学报: 交通科学与工程版, 2006, 30(2): 265-268.
- [9] 周彦, 文宝, 李建勋. 无线传感器网络节点近点加权质心定位方法[J]. 计算机工程与应用, 2012, 48(1): 87-89, 93.
- [10] 刘瑾. 基于测距的无线传感器网络的定位算法的研究[J]. 航空计算技术, 2009, 39(6): 124-126.
- [11] 戴佩华, 薛小平, 邵玉华. 基于垂直平分线的区域定位算法[J]. 计算机工程, 2009, 35(2): 105-108.
- [12] RIBEIRO V J, RIEDI R H, BARANIUK R G. Locating available band-width bottlenecks[J]. IEEE Internet Computing, 2004, 8(5): 34-41.