

一种低复杂度定位算法的设计与实现

黄沛昱 蒋子泉

(重庆邮电大学宽带泛在接入技术研究所 重庆 400065)

摘要 目前,基于定位技术的应用已经成为最具发展潜力的物联网业务之一。物联网感知层传感节点受自身功耗和处理能力的限制,无法直接求解无线定位算法方程。对传统三边测量定位算法进行改进,通过几何近似求解的方法,只需要进行简单的比较、加减、乘除运算,即可获取节点的位置信息。该算法在求解过程中引入了一定的误差值,但大幅降低了定位算法的求解复杂度,为低运算能力的微控制器直接定位提供了一种可行的解决方案。经过定位实验评估,该算法在文中实验条件下的平均定位误差为 1.18m,可满足多数中小场景范围定位的要求。

关键词 物联网,三边定位算法,定位误差,微控制器

中图分类号 TP393 **文献标识码** A

Design and Implementation of Low Complexity Positioning Algorithm

HUANG Pei-yu JIANG Zi-quan

(Broadband Ubiquitous Network Research Laboratory, Chongqing University of Posts and Telecommunications, Chongqing 400065, China)

Abstract Currently, applications based on location technology have become one of the most development potential Internet of Things(IOT) business. Because IOT sensing layer sensor nodes are limited by their own power consumption and processing capability, they can not directly solve the equation of wireless positioning algorithm. This paper modified traditional trilateral positioning algorithm by the method of geometric approximate calculation. New algorithm process only needs to make simple comparison, addition, subtraction, multiplication, and division operations to get location information of the node. It introduces a certain error value in the solving process, but significantly reduces solving complexity and providing a feasible solution for the low operational capability processor to directly calculate positioning information. Positioning experiment evaluation shows that the positioning average error of the algorithm is 1.18m, and it can satisfy the most small and medium-sized scene range positioning requirements.

Keywords Internet of things, Trilateral positioning algorithm, Positioning error, Microcontroller

1 引言

根据评估机制的不同,无线定位算法可分为基于测距算法和基于非测距算法两大类。其中在测距算法中,通常采用接收信号强度指示(Received Signal Strength Indicator, RSSI)、到达时间差(Time Difference Of Arrival, TDOA)、到达时间(Time Of Arrival, TOA)、到达角度(Angle Of Arrival, AOA)等方式,并根据信号在空间中的传播模型获得信标节点到待测节点的距离或角度等信息,通过三边测量定位、三角测量定位等定位算法来计算待测节点的坐标^[1]。

在实际应用中,受物联网感知层节点功耗和处理能力的影响,节点无法利用 RSSI 等信息和已有定位算法直接算出自身的坐标,通常是将节点感知到的 RSSI 值、到达时间值通过通信链路发送到具有高性能处理器的设备上(如 PC),进而通过相应算法计算目标坐标,完成定位、跟踪等具体应用。所以传统的定位方式直接应用到物联网中,具有通信开销大、定位运算复杂,以及需要借助设备实现等缺点。因此,在传统定

位方式的理论基础上,本文提出一种新的适用于低功耗、低处理能力节点直接求解的定位算法。该算法为特定定位场景特别是物感知层联网中常用的感知设备定位提供了一种可行的解决方案。

2 算法理论依据

在测距定位技术中,三边测量法(Trilateration)是最常见的定位方法之一,也是其他很多定位算法的理论基础。其原理如图 1 所示。

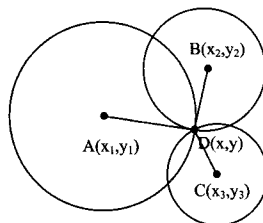


图 1 三边测量法

到稿日期:2012-07-25 返修日期:2012-11-13 本文受国家自然科学基金(61001105,61102151,61271261)资助。

黄沛昱(1981—),女,硕士,讲师,主要研究方向为电子技术,E-mail:huangpy@cqupt.edu.cn;蒋子泉(1987—),男,硕士,主要研究方向为嵌入式操作系统软件。

已知A、B、C 3个节点的坐标分别为 (x_1, y_1) 、 (x_2, y_2) 、 (x_3, y_3) ，且它们到定位目标节点的距离分别为 d_1 、 d_2 、 d_3 ，假设目标节点的坐标为 $D(x, y)$ ，可建立方程组：

$$\begin{cases} (x-x_1)^2 + (y-y_1)^2 = d_1^2 \\ (x-x_2)^2 + (y-y_2)^2 = d_2^2 \\ (x-x_3)^2 + (y-y_3)^2 = d_3^2 \end{cases} \quad (1)$$

求解方程组(1)，利用最小二乘估计得到 $D(x, y)$ 点的坐标：

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 2(x_1-x_3) & 2(y_1-y_3) \\ 2(x_2-x_3) & 2(y_2-y_3) \end{bmatrix}^{-1} \cdot \begin{bmatrix} x_1^2 - x_3^2 + y_1^2 - y_3^2 + d_3^2 - d_1^2 \\ x_2^2 - x_3^2 + y_2^2 - y_3^2 + d_3^2 - d_2^2 \end{bmatrix} \quad (2)$$

该方程有唯一解的情况是3个圆交于一点，实际情况下几乎不可能出现，且该方程组的求解代价非常高，微控制器无法计算。在实际定位测距中，通过信标节点到未知节点的RSSI值换算出来的距离值并不是精确的，由于多径效应等因素使得电磁信号在空间中衰减，计算出的距离往往不等于两个节点间的实际距离。所以实际的模型通常采用三角质心算法，如图2所示。

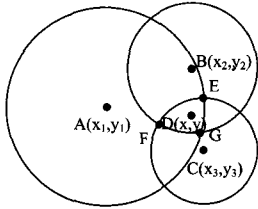


图2 质心定位算法

3圆的重叠区域由弧EF、FG、EG所围成，由于D一定在A、B、C相应的通信范围内，因此D点的实际位置必定在该重叠区域内。用 $\triangle EFG$ 的质心 D' 坐标来估计实际D的位置。通过先求3圆两两交点，并进一步通过计算判断E、F、G3点坐标来求解该算法，进而求得质心 D' 坐标。该算法更适用于实际应用并且一定程度上降低了算法复杂度，但是计算代价仍然较高。

本文算法的思想是：以满足三边定位条件为前提，首先用两圆的相交区域的中点作为三角形的顶点，那么这3个相交区域的中点所围成的三角形的质心度就用来表示实际未知节点D的估计坐标，如图3所示。

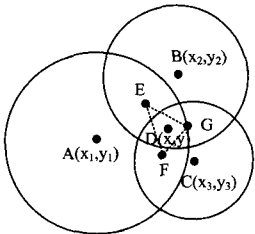


图3 低运算复杂度的定位算法

由于求解图中坐标E、F、G的复杂度仍然较高，不适用于低运算能力的微控制器求解，因此进一步采用定位圆的坐标横移和竖移的方法获得E、F、G节点的近似坐标。经算法求解分析可得，本文算法大约仅需要300~400个指令周期(假设与或非运算占用3个指令周期)，且避免了开平方运算的出现，为低运算能力的微控制器直接计算出估计坐标提供了可

能^[2-5]。

3 算法原理

假设已获取了3个定位圆的定位信息，包括距离矢量(即定位圆半径)和锚节点坐标，且默认定位圆满足三边定位的基本条件：任意两圆不存在相离、包含的情况。则任意两定位圆的关系可分为相交、相切两种情况。记两圆分别为K、N，圆心分别为 $k(x_k, y_k)$ 、 $n(x_n, y_n)$ ，半径分别为 R_k 、 R_n 。假设对两圆的位置以及大小做出规定： $x_k < x_n$ 。为了求解圆K、N相交区域的中点坐标，本文思想是分别独立求解该中点的横、纵坐标。对于横坐标：先把圆N垂直移动到和K同一纵坐标的位置，产生新的圆 N_x (图中虚线圆所示)，该算法求解过程并不需要实际移动圆而增加复杂度。平移后的 N_x 与K的位置关系可能是相交、内切以及包含，具体存在如图4、图5所示的几种情况。

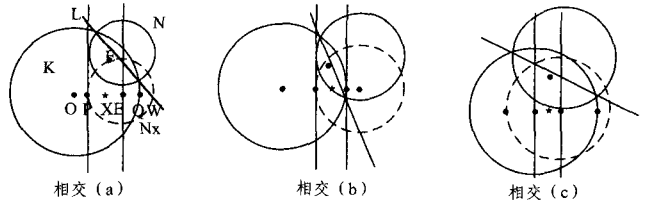


图4 平移后 N_x 与N相交

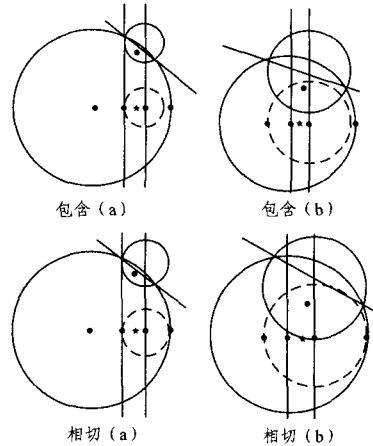


图5 平移后 N_x 与N包含及相切

记K、N相交区域的中点为E，交点所在的直线为L，记录 $x_k, x_n, x_k + R_k, x_n - R_n$ 4个值并按从小到大排序为O、P、Q、W，取中间两个值，分别记为P、Q，如图4相交(a)标示，那么可得到两个结论：无论N平移后的圆 N_x 与K相交、相切还是包含，E点的横坐标 x_E 都必然落在P、Q之间，即 $P \leq x_E \leq Q$ 。由于在实际定位中，感知节点的通信范围有限，因此可以取 $(P+Q)/2$ 作为 x_E 的估计坐标。

两圆相交区域中点的纵坐标估计值可以采用类似的方法，通过横移N使得N的圆心平移到了与K相同的横坐标来近似求得，进而推广到两圆相交的所有位置和大小情况以及平移的方法，可以得到更为一般的结论：

- 两圆相交区域的中点横坐标值必然介于左圆右边界横坐标、右圆左边界横坐标、左圆圆心横坐标、右圆圆心横坐标4个值从大到小(或从小到大)排列后中间两值的范围之间。
- 两圆相交区域的中点纵坐标值必然介于上圆下边界纵坐标、下圆上边界纵坐标、上圆圆心纵坐标、下圆圆心纵坐标

4 个值从大到小(或从小到大)排列后中间两值的范围之间。

对上述一般性结论的几种代表性情况给予证明。

如图 6 所示,记两圆圆心分别为 D 、 E , 圆心连线 L 与两圆相交区域边界的交点为 A 、 B , AB 中点为 C , 半径分别为 R_e 、 R_d 。

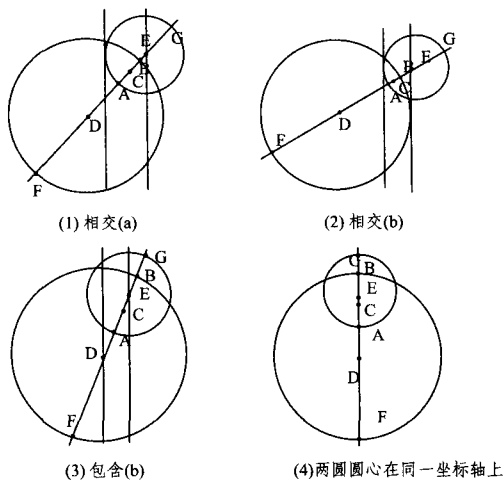


图 6 几种情况的证明

(1) 相交(a)

证明:

$$\because AE=AG/2, AC=AB/2, AG>AB$$

$$\therefore AE>AC \Rightarrow X_c < X_e$$

$$\because X_e - R_e < X_a < X_c$$

$$\therefore X_e - R_e < X_c < X_e, \text{得证。}$$

(2) 相交(b)

证明:

$$\because X_a < X_c < X_b, X_a > X_e - R_e, X_b < X_d + R_d$$

$$\therefore X_e - R_e < X_c < X_d + R_d, \text{得证。}$$

(3) 包含(b)

证明:

$$\because AE=AG/2, AC=AB/2, AG>AB$$

$$\therefore AE>AC \Rightarrow X_c < X_e$$

$$\because DB=FB/2, CB=AB/2, FB>AB$$

$$\therefore CB < DB \Rightarrow X_c > X_d$$

$$\therefore X_d < X_c < X_e, \text{得证。}$$

(4) 两圆圆心在同一坐标轴上

证明:

$$\because E, C, D \text{ 在同一 } y \text{ 轴上}$$

$$\therefore X_e = X_c = X_d, \text{得证。}$$

所以通过上述结论可以将求解相交区域中点坐标的复杂计算转为计算 4 个指定值中中间两值的中点来简单地求解图 3 中 E 、 F 、 G 的估计坐标, 从而计算出质心 D 的坐标, 这在很大程度上降低了算法求解的复杂度。

4 算法过程

(1) 采集满足三边定位条件的锚节点坐标以及距离量。

建立信标节点集合: $node = \{A, B, C\}$;

建立信标节点位置信息集合: $position = \{(x_a, y_a), (x_b, y_b), (x_c, y_c)\}$;

建立信标节点距离量集合: $RSSI = \{R_a, R_b, R_c\}$ 。

(2) 将 A 、 B 、 C 两两取出进行计算。

a) 比较两定位圆圆心横坐标, 计算左圆右边界横坐标、右圆左边界横坐标、左圆圆心横坐标、右圆圆心横坐标 4 个值;

b) 将这 4 个值采用冒泡排序从小到大进行排列, 排列后的顺序记为 O 、 P 、 Q 、 W ;

c) 计算 $X_{ij} = (P+Q)/2$ (i, j 为定位圆下标, 取 A 、 B 、 C);

d) 比较两定位圆圆心纵坐标, 计算上圆下边界纵坐标、下圆上边界纵坐标、上圆圆心纵坐标、下圆圆心纵坐标 4 个值;

e) 将这 4 个值采用冒泡排序从小到大排列, 排列后顺序记为 K 、 L 、 M 、 N ;

f) 计算 $Y_{ij} = (L+M)/2$;

g) 将 (X_{ij}, Y_{ij}) 作为相交区域中点的估计坐标, 若计算出所有中点估计坐标, 进行 (3), 否则重复 (2)。

(3) 计算中点围成的三角形质心, 将其作为待测节点的估计坐标。

$$\begin{cases} X = (X_{AB} + X_{BC} + X_{AC})/3 \\ Y = (Y_{AB} + Y_{BC} + Y_{AC})/3 \end{cases} \quad (3)$$

算法流程图如图 7 所示。

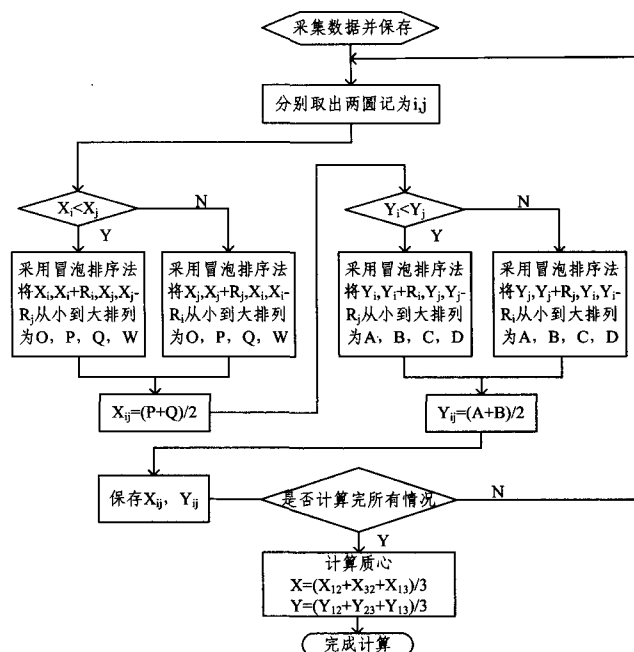


图 7 算法流程图

5 算法验证

本文采用 C#.NET 平台对算法进行仿真实验。从理想参考节点、理想距离量、算法复杂度、定位精度等参数对所提出的机制进行分析。

假设计算得到的测试点位置为 (x, y) , 真实位置为 (x', y') , 那么定位误差 E 为^[6-8]:

$$E = \sqrt{(x-x')^2 + (y-y')^2} \quad (4)$$

n 个测试点的误差分别为 $E_1, E_2, \dots, E_n$, 那么定义平均误差 $\bar{E}$ 为:

$$\bar{E} = (E_1 + E_2 + \dots + E_n)/n \quad (5)$$

如图 8 所示, 在一个 $20\text{m} \times 12\text{m}$ 的矩形区域范围内布置 8 个锚节点作为参考节点。

试验 1 假设待测节点所接收到的距离量 d 为理想情况, 即接收到的 3 个距离量 d 来自所有距离量中最小的 3 个且 d 无误差引入。

该试验用于测试本文算法对于理想情况下引入的误差情况,经过计算可得 10 个待测点在理想情况下的 $\bar{E}$ 为 1.18m,其中最大误差为 2.40m,最小误差为 0.06m,基本满足小范围内的定位需求。

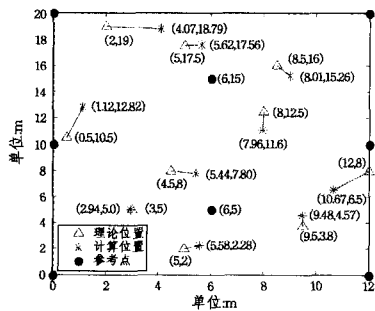


图 8 算法仿真

试验 2 如考虑距离量 d 的引入误差对定位精度的影响,以试验 1 条件为基础,分别采用节点通信距离的 5%、10%、20%、30%作为误差范围的最大值,对每个取值情况分别进行 15 次试验,并对每次的平均误差取平均值,那么误差变化如图 9 所示。

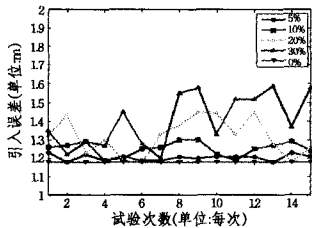


图 9 误差变化曲线

算法仿真的效果如图 10 所示。

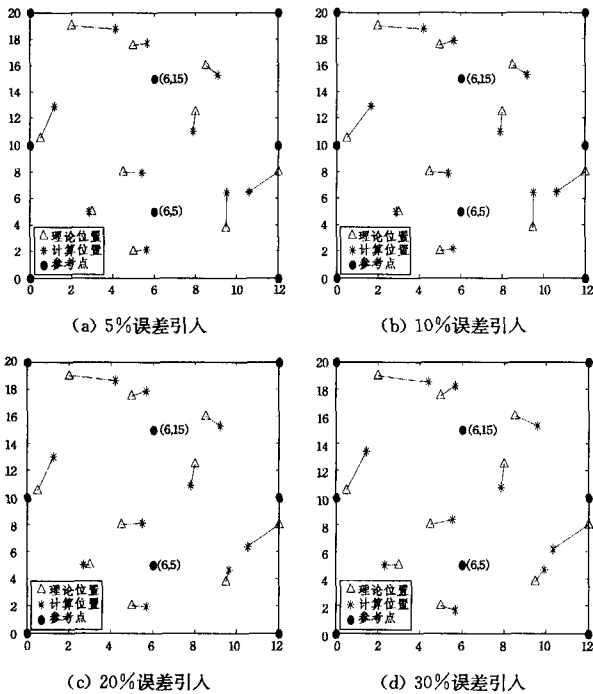


图 10 距离量引入误差对精度的影响

由图 9 及图 10 可见, d 的引入误差分别取 5%、10%、20%、30%时,算法定位误差分别为 1.19m、1.25m、1.33m、1.55m。所以在 d 较为理想的情况下,距离量的引入误差对算

法影响并不十分明显。
 试验 3 在试验 1 的基础上增加了 5 个非理想参考点,目标点与估计点的关系如表 1 所列。

表 1 引入非理想锚节点情况

目标节点	参考节点 1	参考节点 2	参考节点 3	估计坐标	误差
(5,2)	(12,0)	(6,5)	(12,10)	(8.31,4.30)	4.03
(9.5,3.8)	(0,0)	(12,0)	(6,5)	(7.81,2.64)	2.05
(9.5,3.8)	(6,5)	(10,10)	(12,10)	(8.34,7.42)	3.80
(8,12)	(6,5)	(10,10)	(12,10)	(10.19,9.54)	3.29
(0.5,10.5)	(0,0)	(6,5)	(10,10)	(5.67,5.33)	6.306

可见,当引入非理想的参考节点时,误差值急剧增加,使得估计值不满足定位的需求。因此获得理想的参考节点的距离量,对定位具有非常重要的影响。故要求算法一开始获得的距离量为最小的 3 个值。

试验 4 取参考点坐标(6,5)、(0,10)、(12,10),它们所围成的三角形质心坐标为(6,8.33)。图 11 中 Line1、Line2 分别表示沿质心到坐标(0,12)、(12,9.5)方向上偏离质心的距离 d 与定位误差的关系。

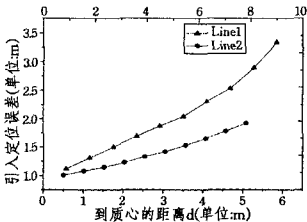


图 11 偏离质心的距离 d 与定位误差的关系

可见,沿某方向下,定位误差随着目标节点与参考节点围成的三角形的质心偏离的距离增大而增大。进一步,记相邻参考节点坐标围成的三角形为 $\triangle ABC$,质心为 D 。通过试验 1、试验 2 及试验 3 的测试点可以观察到,当目标节点越靠近 D 时,引入的误差值越小;当目标节点靠近 $\triangle ABC$ 的边和顶点时,误差变化得越快,误差值越大;当移动超过 $\triangle ABC$ 范围外很大时,误差基本变得不可接受。所以,根据这一情况,在实际环境的布点中,要使定位的区域尽可能地包含在参考坐标围成的三角形内。

同时,为解决实际待测节点出现在参考节点围成的三角形边界附近误差较大的情况,可以对算法做适当的修改。取一个阈值,近似地认为小于该阈值的情况为两定位圆内切或者外切,目标节点在切线上,那么定位误差主要由阈值的大小决定,从而减小了文中平移方法引入的误差。这要求在算法的实现上首先判断两两定位圆是否相切或近似相切的情况,如果相切,则可按照以下公式计算切点值作为近似估计值^[9,10]。

- 通过外切坐标公式求切点 $S(x,y)$ 的坐标:

$$\begin{cases} x=(x_1+x_2 * r_1 / r_2) / (1+r_1 / r_2) \\ y=(y_1+y_2 * r_1 * r_2) / (1+r_1 / r_2) \end{cases} \quad (6)$$
- 通过内切坐标公式求内切点 $T(x,y)$ 的坐标:

$$\begin{cases} x=x_2-(x_1-x_2) * r_2 / (r_1-r_2) \\ y=y_2-(y_1-y_2) * r_2 / (r_1-r_2) \end{cases} \quad (7)$$

但修改过的算法由于引入了大量浮点数的除法、乘法等运算以及圆位置关系的判断,因此增加了算法计算的复杂度。以上文所述的处理器为例,该算法在最差的情况下需增加约

(下转第 146 页)

- [5] Chen G, Gray R S. Simulating non-scanning worms on peer-to-peer networks[C]//Proceedings of the 1st International Conference on Scalable Information Systems. Hong Kong, China, 2006
- [6] Xie C, Yin Z Y. The Research of Worms in P2P Networks[C]//Proceedings of Computational Intelligence and Natural Computing International Conference. 2009; 389-392
- [7] 夏春和, 石昀平, 李肖坚. 结构化对等网中的 P2P 蠕虫传播模型研究[J]. 计算机学报, 2006, 29(6): 952-959
- [8] 罗兴睿, 姚羽, 高福祥. 基于纯 P2P 原理的蠕虫传播模型的研究[J]. 通信学报, 2006, 27(11): 53-58
- [9] Fan X, Xiang Y. Modeling the Propagation of Peer-to-Peer Worms under Quarantine[C]//Proceedings of Network Operations and Management Symposium (NOMS). Osaka, Japan; IEEE Press, 2010; 942-945
- [10] Fan X, Xiang Y. Modeling the Propagation Process of Topology-Aware Worms: An Innovative Logic Matrix Formulation[C]//Proceedings of Network and Parallel Computing 2009 Sixth IFIP International Conference. Gold Coast, QLD; IEEE Press, 2009; 182-189
- [11] Fan X, Xiang Y. Propagation Modeling of Peer-to-Peer Worms [C]//Proceedings of 24th IEEE International Conference on Advanced Information Networking and Applications. 2010; 1128-1135
- [12] Fan X, Guo W W, Looi M. Modeling and Simulating the Propagation of Unstructured Peer-to-Peer Worms[C]// Proceedings of Computational Intelligence and Security 2011 Seventh International Conference. Hainan, China; IEEE Press, 2011; 573-577
- [13] Shin J, Kim T, TAK S. A Reputation Management Scheme Improving the Trustworthiness of P2P Networks[C]//Proceedings of Convergence and Hybrid Information Technology 2008 International Conference. Daejeon, Korea; IEEE Press, 2008; 92-97
- [14] 冯朝胜, 秦志光, Cuthbert L, 等. P2P 网络中沉默型蠕虫传播建模与分析[J]. 计算机研究与发展, 2010, 47(3): 500-507
- [15] Yang W, Li Y. P2P worm Propagation modeling and analysis under dynamic quarantine defense[C]//Proceedings of e-Business and Information System Security 2010 2nd International Conference. Wuhan, Huhei; IEEE Press, 2010; 1-4
- [16] Yu W, Boyer C, Chellappan S, et al. Peer-to-Peer System-based Active Worm Attacks; Modeling and Analysis[C]//Proceedings of IEEE International Conference on Communications. 2005; 295-300
- [17] Jafarabadi A, Azgomi M A. An SIR model for the propagation of topology-aware active worms considering the join and leave of hosts[C]//Proceedings of Information Assurance and Security 7th International Conference. 2011; 204-209
- [18] Gao C X, Zhang F Y, Xin Y, et al. Research on Worm's Propagation and Defense Model in Different P2P Networks[J]. Journal of Beijing University of Posts and Telecommunications, 2006, 29(22): 49-53
- [19] 邓映秩. 对等网络蠕虫及其防御技术研究[D]. 成都: 电子科技大学, 2007

(上接第 125 页)

8000 个指令周期, 但根据处理器对浮点数乘除运算的支持程度不同, 所消耗的指令周期差别也很大。以试验 1 条件为基础, 10 个待测点在理想情况下的 $\bar{E}$ 为 0.686m, 其中最大误差为 1.62m, 最小误差为 0.05m。可见修改过的算法进一步提高了定位精度。

6 算法复杂度分析

本文的算法复杂度与传统的质心定位算法复杂度相似, 但是本文算法通过判断语句与简化的目标节点估计方法, 大幅降低了求解的复杂性。若取 3 个满足定位条件的锚节点确定目标节点的估计坐标, 利用传统的质心算法平均需要做 66 次加法、88 次减法、92 次乘法、32 次除法、12 次开平方运算、3 次比较判断才能计算出估计节点坐标; 本文算法则至多需要 10 次加法、无减法、无次乘法、2 次整除运算、6 次移位运算、无开平方运算、42 次比较判断。对于目前常用的一些低处理能力 MCU, 开平方运算以及浮点数除法运算需要占用大量的 MCU 指令周期, 甚至于很多低运算能力 MCU 都无法进行开方运算。该方法一方面避免了开方运算以及浮点数除法运算的出现, 从而大幅减小了 MCU 的运算量, 提高了运算速率, 更重要的是为低运算能力的处理器通过距离量直接求解定位估计坐标提供了可能性, 实现了降低通信开销的目的。

结束语 本文意在提出一种适用于物联网感知层节点低运算能力微处理器也能直接计算目标节点位置的计算方法。该方法通过多层判断语句的嵌套, 分类计算目标节点的估计目标值, 来达到快速定位、降低通信开销的目的。试验表明, 该方法有效地改善了算法的运算量, 从而达到了降低功耗的目的, 并且保证引入误差控制在要求范围内, 能够满足多数的

短距离无线定位的需求。下一步还将分析如何通过 RSSI 值在低运算的处理器下计算通信距离, 从而实现低处理能力的处理器直接实现定位应用的功能。

参考文献

- [1] 刘锋, 章登义. 基于 RSSI 的无线传感器网络质心定位算法[J]. 计算机科学, 2012, 39(B06): 96-98
- [2] 夏心江, 胡钢, 王烨华. 基于同心圆定位算法的改进算法研究[J]. 计算机科学, 2012, 39(6): 68-71
- [3] 苟胜难. 基于改进的 RSSI 无线传感器网络节点定位算法研究[J]. 计算机应用研究, 2012, 29(5): 1867-1869
- [4] Vivekanandan V, Wong V W S. Concentric anchor beacon location algorithm for wireless sensor networks[J]. IEEE Transactions on Vehicular Technology, 2007, 56(5): 2733-2744
- [5] Bahl P, Padmanabhan V N. RAFA: An In Building RF-based User Location and Tracking System [C] // Proc. IEEE Joint Conf. IEEE Computer Communications Societies (INFOCOM). Telaviv, Israel, Mar. 2000; 775-784
- [6] Bergamo P, Mazzini G. Location in Sensor Networks with Fading and Mobility[C]//Proc. IEEE Int. Symp. Personal, Indoor Mobile Radio Communications (PIMRC). Lisbon, Portugal, Sep. 2002; 750-754
- [7] 李娟, 王珂, 李莉. 基于锚圆交点加权质心的无线传感器网络定位算法[J]. 吉林大学学报: 工学版, 2009, 39(6): 1649-1653
- [8] 朱剑, 赵海, 徐久强, 等. 无线传感器网络中的定位模型[J]. 软件学报, 2011, 22(7): 1612-1625
- [9] 顾晶晶, 陈松灿, 庄毅. 基于无线传感器网络拓扑结构的物联网定位模型[J]. 计算机学报, 2010, 9: 1548-1556
- [10] 王琦. 基于 RSSI 测距的室内定位技术[J]. 电子科技, 2012, 6: 50-54