

一种分布式无锚点定位算法的优化与实现

孙 昱¹, 董淑福¹, 温 东²

(1. 空军工程大学电讯工程学院, 陕西西安, 710077; 2. 海军潜艇学院, 山东青岛, 266071)

摘要 符合分布式和无锚点特点的定位算法一直是传感器网络节点定位技术的一个重要发展方向。通过对大量无锚点定位算法的分析, 提出了一种符合分布式特点且定位精度较高的无锚点定位算法。为进一步提高该算法的节点定位精度, 建立了该算法的误差模型, 通过对其分析, 提出了减少目标节点定位误差的方法。为衡量节点估计位置的准确程度, 首先提出了2个指标——定位等级与可信度; 然后以原算法为基础, 利用提出的指标, 按照误差分析得出的结论, 设计了一种从目标节点邻居表中筛选出高精度邻居节点的优化机制, 从而提高了目标节点的定位精度。计算机仿真分析表明, 优化后的算法计算节点估计位置的有效性和可靠性均高于原算法, 进而证明了这种优化设计的可行性。

关键词 传感器网络; 分布式; 无锚点定位; 误差分析; 节点定位算法

DOI 10.3969/j.issn.1009-3516.2012.02.017

中图分类号 TP393 **文献标识码** A **文章编号** 1009-3516(2012)02-0079-05

节点定位问题一直是传感器网络研究的热点问题^[1-2]。现在人们研究该问题大多考虑的是有锚节点的定位算法^[3], 而对无锚节点的定位算法研究较少。实际上, 无锚点的定位算法对无线传感器网络的定位问题具有十分重要的意义, 因为在某些特殊的应用场合中, 传感器网络中所有节点的位置均是随机的, 无法事先获取。

最早的无锚点定位算法是 Priyantha 等人提出的 AFL 算法^[4], 可是它并非完全的分布式算法。ABC 算法虽然具有无锚点、分布式的特征, 而且步骤简单, 但它的定位误差较大^[5]。Shang 等人根据多维标度原理, 在无锚点的情况下也能进行定位, 但它属于集中式定位^[6]。鉴于上述几种算法的定位误差均比较大^[7], 文献[8]基于计算几何理论^[9], 提出了一种新的分布式无锚点定位算法, 并取得到了较好的效果。其主要是运用分步循环求精的思想: 首先建立一个由虚拟锚点组成的坐标系, 然后对全网节点进行粗定位; 当粗定位完成后, 再利用质点弹簧模型对全网进行更细致的定位以得到更准确的节点位置。

本文首先建立上述算法的误差模型, 然后依据误差分析得出的结果, 设计一种优化算法, 以减少原算法^[8]的定位误差, 提高其定位精度。

1 原算法误差分析

在利用文献[8]中算法计算目标节点位置时, 由于所采用的计算公式是估算公式, 无法精确地计算出目标节点的位置, 因此得到的节点位置总是存在着系统误差; 同时, 在原算法中, 用于计算目标节点位置的邻居节点是随机的, 所以, 估算出的目标节点位置还有随机误差。

假设用于计算目标节点位置的邻居节点有 n 个, 其编号为 $1, 2, \dots, n$, 记其编号的集合 $C = \{1, 2, \dots, n\}$ 。设这 n 个邻居节点的位置的系统误差的极限误差分别是 e_1, e_2, \dots, e_n , 随机误差的极限误差分别是 $\delta_1, \delta_2, \dots$,

* 收稿日期: 2011-05-25

基金项目: 国防预研基金资助项目(9140A06050610JB15)

作者简介: 孙 昱(1989-), 男, 江西吉安人, 硕士生, 主要从事移动传感器网络研究。

E-mail: 458657673@qq.com

δ_n ,按照极限误差的合成公式^[10],可得目标节点的总极限误差 $\Delta_{\text{总}}$ 如下:

$$\Delta_{\text{总}} = \pm t \sqrt{\sum_{i=1}^n \left(a_i \frac{e_i}{t_i} \right)^2 + \sum_{i=1}^n \left(a_i \frac{\delta_i}{t_i} \right)^2} + R \quad (1)$$

式中: a_i 为误差传递系数; t_i 为置信系数; t 为总误差的置信系数; R 为各随机误差间的协方差之和。考虑到随机误差具有补偿性,即计算所用的邻居节点数越多,得到的目标节点位置的随机误差越小,因此,目标节点的总极限误差修正为:

$$\Delta_{\text{总}} = \pm t \sqrt{\sum_{i=1}^n \left(a_i \frac{e_i}{t_i} \right)^2 + \frac{1}{n} \sum_{i=1}^n \left(a_i \frac{\delta_i}{t_i} \right)^2} + R \quad (2)$$

根据式(2),若要减少目标节点的总极限误差,可以适当增加用于计算目标节点位置的邻居节点的个数。假设用于计算目标节点位置的邻居节点中有 k 个节点的极限误差远大于其余邻居节点的极限误差,这些节点的编号构成集合 $C_k = \{m_1, m_2, \dots, m_k\}$,其余邻居节点的编号构成的集合为 $C_{n-k} = C - C_k$ 。即对于 $\forall i \in C_k, j \in C_{n-k}$ 均有 $e_i \gg e_j, \delta_i \gg \delta_j$ 。此时,根据式(2)可得目标节点位置的总极限误差为:

$$\Delta_{\text{总}} \approx \pm t \sqrt{\sum_{i=1}^k \left(a_{m_i} \frac{e_{m_i}}{t_{m_i}} \right)^2 + \frac{1}{n} \sum_{i=1}^k \left(a_{m_i} \frac{\delta_{m_i}}{t_{m_i}} \right)^2} + R \quad (3)$$

即在计算中,一个或几个大的误差对计算结果的精度影响举足轻重,若能适当控制最大误差分量,就能大大提高计算结果的精度。而在原算法计算目标节点位置的邻居节点中,有一些邻居节点的位置由于误差积累次数过多,其误差远大于其它邻居节点的位置误差。因此,对这类邻居节点应予以控制,即不应让其参与对目标节点位置的计算。

2 优化方案

2.1 相关概念

1)计算表:即用于计算目标节点坐标的邻居节点的集合,它是邻居表的一个子集。

2)定位等级:节点的定位等级为其计算表中最底的节点定位等级值加1,通过该指标来衡量误差积累的个数。并规定:虚拟锚点的定位等级为0,即认为虚拟锚点的坐标是精确的,没用误差积累。例如,如果一个节点的计算表中均为锚点,那么它的定位等级为1。

3)可信度:用来衡量节点估计位置的可信程度。该指标在文献[8]中,是通过检验某一节点与其邻居表中的节点之间的通信链接关系来反映该节点估计位置的正确性程度。而在本文中,则是通过检验与其计算表中的节点之间的通信链接关系来反映其估计位置的正确性程度。当一个节点的可信度高于阈值时,认为其定位比较精确可靠。

2.2 第1阶段算法

现在,对于一个节点的估计位置,可以使用定位等级和可信度来衡量它的准确度:①对于定位等级不同的节点,定位等级小的节点其估计位置的准确度高。②对于定位等级相同的节点,可信度高于阈值的节点其估计位置的准确度高。

按照误差分析中得出的结论,容易得出目标节点的计算表应尽量符合以下3个原则,按优先级由高到低排列为:目标节点计算表中的邻居节点的定位等级应尽可能高;目标节点计算表中的定位等级最低的邻居节点应尽可能可信;目标节点计算表中的邻居节点数应越多越好。

按照上述原则,当目标节点收到一个定位等级高于其计算表中的最低定位等级的邻居节点信息时,它应该进行如下操作:

1)判断将该节点加入计算表后,是否可以提高自己的定位等级;

2)判断将该节点加入计算表后,是否可以使自己的计算表满足表中最低定位等级的邻居节点全部可信的条件;

3)判断是否需要将该邻居节点加入到自己的计算表中以增加计算表中的邻居节点数目。

当目标节点收到一个定位等级等于其计算表中的最低定位等级的邻居节点信息时,其操作为:①判断将该节点加入计算表后,是否可以使自己的计算表满足表中最低定位等级的邻居节点全部可信的条件;②判断

是否需要将该邻居节点加入到自己的计算表中以增加计算表中的邻居节点数目。

具体算法流程图见图 1。

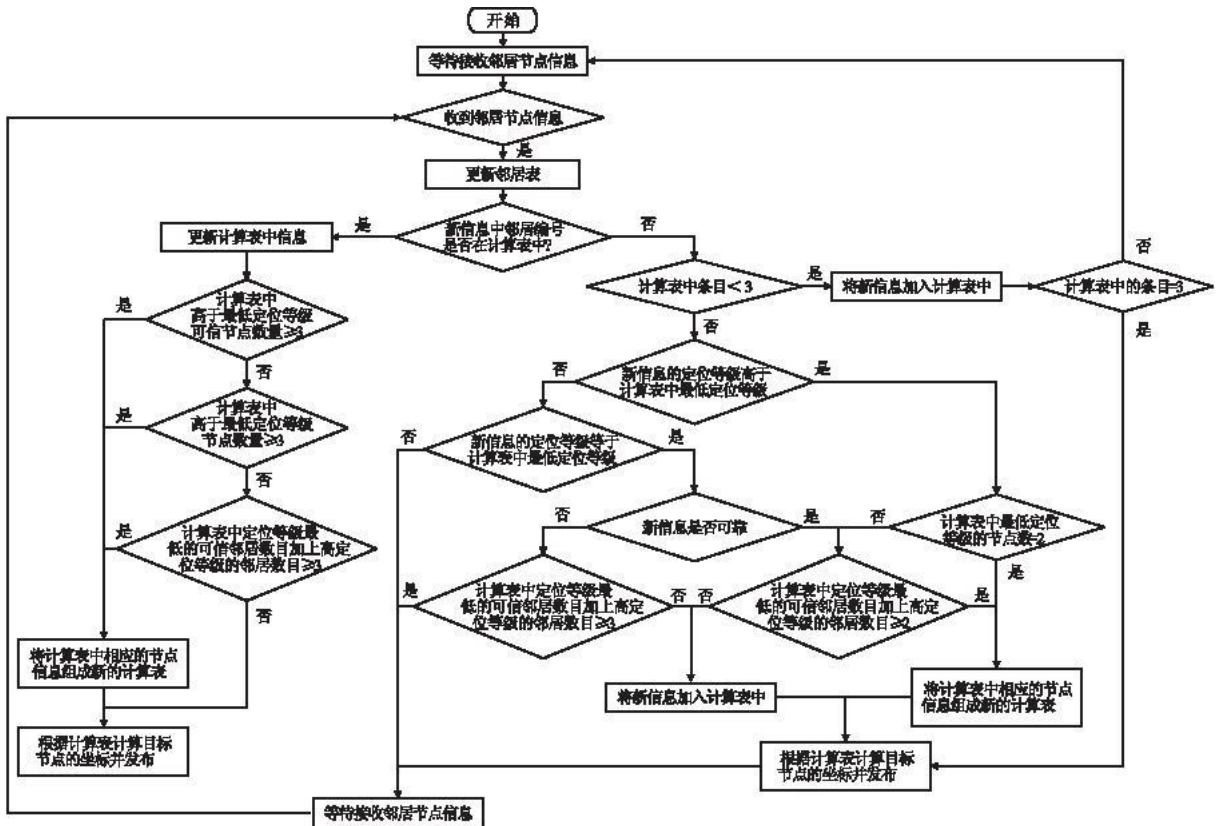


图 1 优化算法第 1 阶段

Fig. 1 The first stage of the optimized algorithm

由于计算表中的邻居节点坐标都是该目标节点所能接触到的范围内最准确的,因此,经过第一阶段的计算后,所有节点均可以获得一个比较准确的初始坐标。而且,没有必要像原算法那样执行位置变异策略。这也是该优化方案的一个优势。

2.3 第 2 阶段算法

相应地,在整个网络进行第 2 阶段的精确定位时,与原算法使用邻居表中的节点位置不同,优化算法使用的是计算表中的节点位置,具体算法流程如下:

首先目标节点根据计算表,利用弹簧模型算出它的新位置,如果新位置的能量小于原位置,则继续迭代,直至该目标节点能量最低。当该目标节点的邻居表发生更新时,如果新信息中的邻居节点编号同时也在计算表中,则更新计算表,并启动新一轮的迭代过程。在这里有两点需要解释:一是选用计算表而非邻居表来进行迭代是基于计算表中的节点坐标更为准确的考虑。因为邻居表中的某些邻居坐标甚至是依赖于该目标节点进行定位的,如果利用它们来进行迭代,不仅运算量更大,而且定位的精确度将下降。另外,在第 2 阶段运行的时候,每个节点的计算表中的邻居编号是不会改变的,因为这些计算表中的邻居节点是在第 1 阶段遴选出来的。而第 2 阶段除了可能更新计算表中某些邻居节点的坐标值外,是不会变换计算表的。

3 仿真分析

3.1 优化算法的有效性分析

在 100 m × 100 m 的平面区域随机布撒 50 个节点,节点间的通信距离为 40 m,然后按照算法利用 Matlab 编程进行仿真,所得结果如图 2 所示。图中的圆圈表示的是节点真实位置,而三角形表示按原算法估计出的节点位置,星号表示的是按优化算法估计出的节点位置。从图中可以很明显的看出,按优化算法估计出的节点位置绝大部分和真实的节点位置相差无几。经过计算,在本仿真中原算法和优化算法的平均节点误差分别为 17.69 m 和 8.29 m。事实上,在进行的大量仿真过程中发现,采用优化算法后,只要经过足够多次的第

2 阶段的更精确的定位,几乎百分之百的估计位置都和真实位置重合;而对于原算法,由于其没有较好的考虑误差积累以及如何消除的问题,进行一定次数的迭代计算后,整个拓扑趋于稳定且离真实位置有一定的误差,即使再增加第 2 阶段的迭代次数,整个拓扑结构的变化也很微小。

因此,与原算法相比,本优化算法在估计节点的实际位置方面更为精确、有效。

3.2 优化算法的可靠性分析

考虑到实际情况中所用节点一旦确定下来为某种具体的产品,其理想通信距离是固定值,不会发生改变,因此,本文更多的是从节点密度的方面来验证该算法的可靠性。在 $100\text{ m} \times 100\text{ m}$ 的平面区域分别布撒 40、50、60、70 个节点,分别计算原算法和优化算法估计出的节点位置与真实节点位置的平均误差距离。仿真计算结果见图 4,原算法的平均误差距离为 35 m 左右,而优化算法的平均误差距离仅为 5 m 左右。在大量的仿真实验中,我们还发现,如果节点密度过大,原算法的平均误差距离会急剧增加,也就是说,原算法只适用于节点密度适中的情况,而优化后的算法却能以一个可以接受的误差距离很好地适应大节点密度情况。

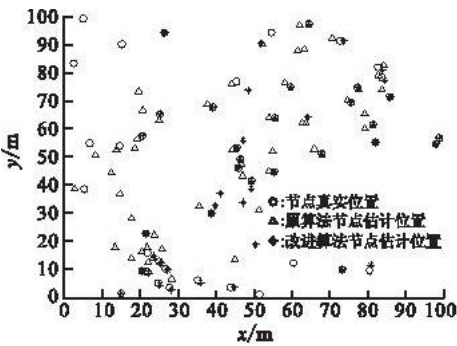


图 2 各算法估计位置比较

Fig. 2 Location comparison of the algorithms

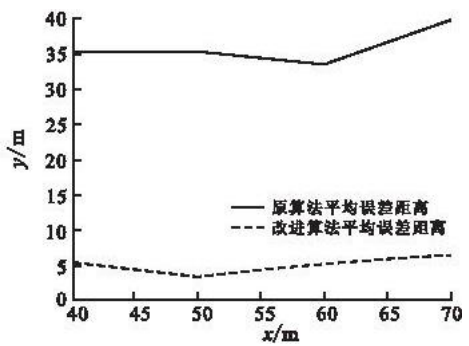


图 3 不同节点密度的平均误差距离

Fig. 3 Average error distance under different node density

当然,本优化算法也存在一个不足,即和原算法一样,要求节点的连接度要大于 3 以进行精确定位,否则,对于不满足要求的节点,定位误差则较大。因此,在节点间的通信距离一定的条件下,本算法不能适应节点密度过低的情况。

4 结束语

本文通过对一种分布式无锚点定位算法的误差分析,提出了一种提高其定位精度的优化方案。该方案首先提出了节点定位等级和可信度的概念,进而利用相应的指标来筛选目标节点邻居表中的邻居节点以使目标节点的计算表最优,从而提高了目标节点的定位精度。最后,通过仿真实验验证了该方案的有效性和可靠性。此方案还可以应用于类似的传感器网络的节点定位当中。

参考文献(References):

- [1] 王福豹,史龙,任丰原. 无线传感器网络中的自身定位系统和算法[J]. 软件学报, 2005, 16(5): 857-968.
WANG Fubao, SHI Long, REN Fengyuan. Self-localization systems and algorithms for wireless sensor networks [J]. Journal of software, 2005, 16(5): 857-968. (in Chinese)
- [2] 崔逊学,方红雨,朱徐来. 传感器网络定位问题的概率特征[J]. 计算机研究与发展, 2007, 44(4): 630-635.
CUI Xunxue, FANG Hongyu, ZHU Xulai. Probabilistic character for localization problem in sensor networks [J]. Journal of computer research and development, 2007, 44(4): 630-635. (in Chinese)
- [3] Langendoen K, Reijers N. Distributed localization in wireless sensor networks: a quantitative comparison [J]. Computer networks, 2003, 43(4): 499-518.
- [4] Priyantha N B, Balakrishnan H, Demaine E, et al. Anchor free distributed localization in sensor networks [R]. MIT laboratory for computer science tech report, #892.
- [5] Savarese C, Rabaey J M, Beutel J. Locationing in distributed ad-hoc wireless sensor networks [C]//Proc of the 2001 IEEE int conf acoustics, speech and signal processing. Piscataway, NJ: IEEE press, 2001: 2037-2040.

- [6] Shang Y, Ruml W, Zhang Y. Localization from mere connectivity [C]//Proc of the 4th ACM int symp on mobile ad hoc networking & computing. New York: ACM, 2003: 201 – 212
- [7] 许秀兰, 赵仕俊. 无线传感器网络中锚节点无关定位算法研究[J]. 计算机工程与设计, 2007, 28(14): 1000 – 7024.
XU Xiulan, ZHAO Shijun. Research of anchor – free localization algorithm in wireless sensor networks [J]. Computer engineering and design, 2007, 28(14): 1000 – 7024(in Chinese)
- [8] 崔逊学, 刘建军, 樊秀梅. 传感器网络中一种分布式无锚点定位算法[J]. 计算机研究与发展, 2009, 46(3): 425 – 433.
CUI Xunxue, LIU Jianjun, FAN Xiumei. A distributed anchor – free localization algorithm in sensor networks [J]. Journal of computer research and development, 2009, 46(3): 425 – 433. (in Chinese)
- [9] 周培德. 计算几何——算法设计与分析[M]. 北京:清华大学出版社, 2005.
ZHOU Peide. Computational geometry – algorithm analysis and design [M]. Beijing: Tsinghua university press, 2005. (in Chinese)
- [10] 沙定国. 误差理论与测量不确定度评定[M]. 北京:中国计量出版社, 2003.
SHA Dingguo. The error theory and measurement uncertainty[M]. Beijing: China metrology press, 2003. (in Chinese)
- (编辑:徐楠楠)

The Optimization and Realization of A Distributed Anchor – free Location Algorithm

SUN Yu¹, DONG Shu – fu¹, WEN Dong²

(1. Telecommunication Engineering Institute, Air Force Engineering University, Xi'an 710077, China; 2. Navy Submarine Academy, Qingdao 266071, Shandong, China)

Abstract: The location algorithm having the distributed and anchor – free features is always one of the important developing directions in the node location techniques of sensor network. On the basis of analyzing a great many anchor – free location algorithm, the article finds out a distributed anchor – free location algorithm with which the location obtained is quite accurate. In order to further improve its location accuracy, the article establishes an error model of the algorithm and then proposes a method of reducing the target node positioning error by analyzing this error model. And to measure the accuracy of the estimated location of node, the article puts forward two targets, positioning level and credibility. According to the conclusion from error analysis, the article designs an optimization mechanism with the usage of the targets proposed on the basis of the original algorithm, which can filter out high – precision neighbor nodes from the neighbor table of a target node, and thereby the positioning accuracy of the target node is improved. The computer simulation shows that the availability and reliability of estimated location which are calculated by the optimized algorithm are higher than those calculated by the original algorithm, thus the feasibility of this optimization is verified.

Key words: sensor network; distributed; anchor – free location; error analysis; node location algorithm