

DOI:10.19651/j.cnki.emt.2211021

基于测距修正和蝙蝠优化的改进 DV-Hop 定位算法*

董玉 张治中 冯皎

(南京信息工程大学电子与信息工程学院 南京 210044)

摘要: 本文对 DV-Hop 算法定位误差大的问题进行研究。针对 DV-Hop 算法在求解平均跳距和未知节点位置两个阶段的缺陷,提出一种基于测距修正和蝙蝠优化的改进 DV-Hop 定位算法。首先,采用最小均方误差准则求解锚节点间的平均跳距,并添加校正因子减小测距误差;其次,利用混沌映射策略初始化种群并设置阈值 M 控制映射的次数,采用速度加权策略控制搜索的步长,增强蝙蝠算法跳出局部最优的能力;最后,使用改进蝙蝠算法确定未知节点的位置。仿真结果表明,提出的定位算法具有更高的定位精度,相比 DV-Hop 算法、BADV-Hop 算法、PSODV-Hop 算法分别提升了 32.35%、18.80%、8.16%。

关键词: 无线传感器网络;节点定位;测距修正;蝙蝠算法;DV-Hop 算法

中图分类号: TP393 **文献标识码:** A **国家标准学科分类代码:** 510.5015

Improved DV-Hop localization algorithm based on ranging correction and bat optimization

Dong Yu Zhang Zhizhong Feng Jiao

(School of Electronics and Information Engineering, Nanjing University of Information Science and Technology, Nanjing 210044, China)

Abstract: This paper focuses on the problem of large positioning error of the DV-Hop algorithm. To overcome the defects of DV-Hop algorithm when solving the average jump distance and the unknown node position, this paper proposes an improved DV-Hop positioning algorithm based on ranging correction and bat optimization. Firstly, the minimum mean squared error criterion is used to solve the average jump distance between anchor nodes while a correction factor is added to reduce the ranging error. Secondly, the chaotic mapping strategy is applied to initialize the population and set the threshold M to control the number of mappings, and besides, the speed-weighted strategy is used to control the step length of the search to enhance the ability of the bat algorithm to jump out of the local optimum. Finally, the Improved Bat Algorithm is used to determine the location of the unknown node. The simulation results show that the proposed positioning algorithm has higher positioning accuracy, which is improved by 32.35%, 18.80% and 8.16% compared with DV-Hop algorithm, BADV-Hop algorithm and PSODV-Hop algorithm, respectively.

Keywords: wireless sensor network; node positioning; ranging correction; bat algorithm; DV-Hop algorithm

0 引言

无线传感器网络(wireless sensor networks, WSN)是由大量具有数据采集、传输和处理能力的传感器节点构成,目前广泛应用于环境保护、工业监测、入侵检测、火灾探测、军事侦察、医疗监控等众多领域^[1]。在各种应用领域中知道传感器节点的位置极其必要,节点定位问题成为 WSN 的研究重点。全球定位系统(global positioning system,

GPS)在室外环境下能够达到精准的定位效果,但考虑到成本和能耗,在 WSN 中为每个传感器节点都配备 GPS 接收器是不现实的^[2]。现阶段大多数的研究都集中在 WSN 定位领域,主要依靠节点间的交互和连通性间接确定目标的位置。

WSN 定位算法根据节点间的测距方式可分为两大类,测距算法和非测距算法。前者是通过使用距离或角度信息直接计算节点间的距离,定位精度较高,但其对硬件配制要

收稿日期:2022-08-09

* 基金项目:国家自然科学基金(61501244,61501245)、江苏省自然科学基金(BK20150932)项目资助

求高,还易受多径和环境的干扰。而后者依靠节点间的交互和连通性间接测量节点间距离,因其成本低、抗噪能力强、简单易行,得到了广泛的应用。距离向量跳(distance vector hop, DV-Hop)定位算法是目前非测距定位算法中重要的研究方向,国内外学者对其定位误差大的问题,进行了深入的研究,并给出了相应的解决方案。

为了解决 DV-Hop 节点定位结果与真实连通性不一致的问题,提出了 CCDV-Hop 算法和 DCDV-Hop 算法^[3]。前者以任意两节点间的真实连通性为约束优化问题,提高了定位的性能,但其复杂度较高;而后者只考虑两跳内的真实连通度,降低了算法的复杂度,但具有一定的局限性,仅适用于分布式网络。DV-maxHop 算法^[4]通过设置控制变量 maxHop 来舍弃跳数大于预设值的锚节点,从而减小误差,提高定位的精度,但阈值设置会丢失部分锚节点信息,定位精度仍有提升空间。SRTC 算法^[5]通过信号强度反演算法校正邻居节点关系并通过新三角形质心迭代算法求解未知节点的位置,有效提高了节点的定位精度,但该算法对未知节点周边节点分布的密集程度要求较高,节点稀疏时,定位误差较大。CVX-DV-hop 算法^[6]通过矩阵变换和一阶泰勒展开将传统的节点定位问题转化为迭代凸优化问题,并采用梯度下降法求解最优解,实现了高精度和低复杂度定位,但梯度下降法收敛速度慢,求解需进行多次迭代,时间代价较大,且不能保证得到全局最小值。

传统优化算法,计算复杂度高且定位性能不佳,因此为了提高 WSN 的定位精度,许多自然启发式算法也被考虑在内成为一种合适的选择^[7]。RDEL 算法^[8]和 DECHDV-Hop 算法^[9]分别通过 RSSI 值和将离散值转换为连续值对跳数进行修正,并使用差分进化算法求解未知节点的坐标,虽然求解的未知节点的位置较精准,但差分进化算法受控制参数和进化策略的影响较大,选择不当易陷入局部最优,导致早熟收敛。OCSLC-DVHop 算法^[10]和 ISSADV-Hop 算法^[11]分别采用融合 Levy 和柯西分布的布谷鸟搜索算法以及引入柯西变异和加权反向学习的麻雀搜索算法求解位置坐标。融合各种策略的改进算法,明显提高了定位的精准性,但算法的复杂度也随之提高。GWO-DV-Hop 算法^[12]采用灰狼算法优化平均跳距,减小测距累计误差,但仍采用传统的最小二乘法求解位置坐标,定位性能存在较大的提升空间。CWDV-Hop 算法^[13]利用二维双曲线法计算未知节点位置并采用鸡群算法进行优化,提高节点定位精度,但该算法受环境因素影响较大,室内定位不精准。

综上所述,各种改进算法虽提升了定位的性能,但仍存在一定的问题。传统最小二乘法在求解未知节点的坐标时对参考方程的选择依赖性较大,极大地影响了定位的结果。单一智能优化算法虽优于最小二乘法,但无法满足全局和局部搜索的平衡,定位精准度存在较大的提升空间。为了解决上述问题,本文提出一种基于测距修正和蝙蝠优化的改进 DV-Hop 定位算法。该算法主要从跳距的计算和位

置的求解两方面进行优化,一方面通过采用最小均方误差准则求解平均跳距,同时添加校正因子降低测距的偏差,该阶段的改进算法记为 IDV-Hop(improved distance vector hop, IDV-Hop);另一方面基于增强混沌映射蝙蝠算法改进 DV-Hop 定位算法(improved distance vector hop localization algorithm based on chaotic mapping enhanced bat algorithm, ICEBADV-Hop),弥补单一智能优化算法的不足,提升算法的搜索效率,进而提高定位的精度。

1 DV-Hop 算法

1.1 DV-Hop 算法过程描述

DV-Hop 算法是由美国罗格斯大学的 Niculescu 等^[14]在 2003 年提出的一种基于多跳测距的分布式非测距算法。该算法的基本原理是首先获得未知节点到锚节点的最小跳数和平均跳距,然后将二者相乘得到两节点间的估计距离,最后使用最小二乘法确定未知节点的位置。DV-Hop 算法实现定位的过程由以下三部分组成。

1) 获取节点间最小跳数值

在 WSN 的通信半径内,所有锚节点以泛洪的方式向网络中传递包含标识、位置和跳数的数据包,未知节点接收数据包并记录当前的位置和跳数,且对其进行比较。若新接收的跳数值较大,则丢弃该数据包,否则将跳数加一传递给网络中的其它节点。

2) 估计未知节点到锚节点距离

依据最短路径法计算节点间最小跳数和锚节点间欧式距离,然后通过式(1)计算出锚节点间平均跳距 $AvgHopsize_i$ 。

$$AvgHopsize_i = \frac{\sum_{i \neq j} d_{ij}}{\sum_{i \neq j} h_{ij}} \quad (1)$$

$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \quad (2)$$

式中: d_{ij} 为锚节点 i 和 j 之间的欧式距离, (x_i, y_i) 和 (x_j, y_j) 分别为锚节点 i 和 j 的位置坐标, h_{ij} 为锚节点 i 和 j 之间的最小跳数值。

未知节点选择距其最近的锚节点作为跳距标准并通过式(3)计算到锚节点的距离。

$$\hat{d}_{ui} = AvgHopsize_i \times h_{ui} \quad (3)$$

式中: \hat{d}_{ui} 为未知节点 u 与锚节点 i 的估计距离。

3) 求解未知节点的位置

根据估计距离 \hat{d}_{ui} , 未知节点到锚节点的距离为:

$$\begin{cases} (x_1 - x_u)^2 + (y_1 - y_u)^2 = \hat{d}_{2u1}^2 \\ (x_2 - x_u)^2 + (y_2 - y_u)^2 = \hat{d}_{2u2}^2 \\ \vdots \\ (x_n - x_u)^2 + (y_n - y_u)^2 = \hat{d}_{2un}^2 \end{cases} \quad (4)$$

式中: (x_i, y_i) 和 (x_u, y_u) 分别为锚节点 i 和未知节点 u 的位置坐标。

将式(4)通过前项减后项法拆分化简后得到如式(5)所示的新线性方程组。

$$\begin{cases} 2(x_1 - x_n)x_u + 2(y_1 - y_n)y_u = x_1^2 - x_n^2 + y_1^2 - y_n^2 + \hat{d}_{un}^2 - \hat{d}_u^2 \\ 2(x_2 - x_n)x_u + 2(y_2 - y_n)y_u = x_2^2 - x_n^2 + y_2^2 - y_n^2 + \hat{d}_{un}^2 - \hat{d}_u^2 \\ \vdots \\ 2(x_{n-1} - x_n)x_u + 2(y_{n-1} - y_n)y_u = x_{n-1}^2 - x_n^2 + y_{n-1}^2 - y_n^2 + \hat{d}_{un}^2 - \hat{d}_u^2 \end{cases} \quad (5)$$

将式(5)表示成矩阵 $\mathbf{Ax} = \mathbf{b}$ 的形式, 如式(6)~(8)所示。

$$\mathbf{A} = \begin{bmatrix} 2(x_1 - x_n) & 2(y_1 - y_n) \\ 2(x_2 - x_n) & 2(y_2 - y_n) \\ \vdots & \vdots \\ 2(x_{n-1} - x_n) & 2(y_{n-1} - y_n) \end{bmatrix} \quad (6)$$

$$\mathbf{x} = [x_u, y_u]^T \quad (7)$$

$$\mathbf{b} = \begin{bmatrix} x_1^2 - x_n^2 + y_1^2 - y_n^2 + \hat{d}_{un}^2 - \hat{d}_u^2 \\ x_2^2 - x_n^2 + y_2^2 - y_n^2 + \hat{d}_{un}^2 - \hat{d}_u^2 \\ \vdots \\ x_{n-1}^2 - x_n^2 + y_{n-1}^2 - y_n^2 + \hat{d}_{un}^2 - \hat{d}_u^2 \end{bmatrix} \quad (8)$$

使用最小二乘法计算未知节点的估计位置坐标 $\hat{\mathbf{x}}$, 具体表示为:

$$\hat{\mathbf{x}} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{b} \quad (9)$$

式中: \mathbf{A}^T 是矩阵 \mathbf{A} 的转置, \mathbf{A}^{-1} 是矩阵 \mathbf{A} 的逆。

1.2 DV-Hop 算法误差分析

DV-Hop 算法误差主要包含最小跳数误差、平均跳距误差和位置求解误差, 详细介绍如下:

1) 最小跳数误差

在原始 DV-Hop 算法中, 通信范围内的所有节点都记为一跳。而实际的定位环境中, 由于节点随机分布, 导致其到中心位置的距离远近不同, 因此引起不同的跳距误差。

2) 平均跳距误差

估计节点间的距离时, 未知节点仅选择最近锚节点的平均跳距而未考虑其它节点。这种单一的选择策略具有一定的局限性, 在多跳网络中不能真实的反映跳距情况^[15]。此外, 不规则的拓扑结构会使节点的密集程度不同, 致使求解的平均跳距存在着误差。

3) 位置求解误差

传统的位置计算方法, 通过组建未知节点到锚节点的距离公式来计算节点的坐标, 使用前项减后项法引入次优

解, 造成定位的精度下降。

2 改进的 DV-Hop 定位算法

2.1 改进 DV-Hop 定位算法的基本思想

针对上述 DV-Hop 算法存在的误差, 本文提出了基于测距修正和蝙蝠优化的改进 DV-Hop 定位算法。该算法的主要内容包括, 在平均跳距计算阶段, 通过采用最小均方误差准则求解平均跳距并添加校正因子进行校正以缩小跳距误差。在节点位置估计阶段, 首先利用混沌映射策略生成初始种群并设置阈值 M 控制映射的次数, 以便增加种群多样性, 提高种群的质量。然后采用速度加权策略控制搜索的步长, 增强蝙蝠算法跳出局部最优的能力, 提高算法的搜索效率。最后利用改进的蝙蝠算法计算未知节点的位置坐标, 提高定位的精度。

2.2 平均跳距修正

在 DV-Hop 算法的第二阶段, 根据式(1)计算出的平均跳距误差较大, 本文通过添加校正因子缩小测距误差。首先, 锚节点间的平均跳距就是最小均方取极小值得到, 如式(10)所示。

$$e = \frac{1}{n-1} \sum_{i \neq j} (d_{ij} - AvgHopDist_i \times h_{ij})^2 \quad (10)$$

式中: n 为锚节点总数, e 为锚节点 i 和 j 的均方误差, 需要的平均跳距就是误差最小时的值。因此, 求误差最小的问题可以转变为求极值的问题^[16], 根据式(11)求解的平均跳距如式(12)所示。

$$\partial e / \partial AvgHS_i = 0 \quad (11)$$

$$AvgHopDist_i = \frac{\sum_{i \neq j} h_{ij} \times d_{ij}}{\sum_{i \neq j} h_{ij}^2} \quad (12)$$

式中: $AvgHopDist_i$ 为最小均方估计的平均跳距, h_{ij} 为锚节点 i 和 j 之间的最小跳数值, d_{ij} 为锚节点 i 和 j 之间的实际距离, 由式(2)计算得到。

然后, 将节点间真实距离与通过 $AvgHopDist_i$ 重新计算的估计距离做差得到误差系数 ϵ_{ij} , 随后计算校正因子 δ , 如式(13)~(14)所示。

$$\epsilon_{ij} = |d_{ij} - \hat{d}_{ij}| = |d_{ij} - AvgHopDist_i \times h_{ij}| \quad (13)$$

$$\delta = \frac{\sum_{i \neq j} \epsilon_{ij} / h_{ij}}{\sum_{i \neq j} h_{ij}} \quad (14)$$

式中: ϵ_{ij} 为误差系数, δ 为校正因子。

最后, 通过式(15)为平均跳距添加校正因子, 将校正后的

$AvgHopDist_{new}$ 与 h_{ui} 相乘, 得到估计距离 \hat{d}_{ui} , 如式(16)所示。

$$AvgHopDist_{new} = AvgHopDist_i + k\delta \quad (15)$$

$$\hat{d}_{ui} = AvgHopDist_{new} \times h_{ui} \quad (16)$$

式中: k 为平衡参数, $k \in [-1, 1]$, $AvgHopDist_{new}$ 为校正后的平均跳距, h_{ui} 为未知节点 u 到锚节点 i 的最小跳数。

2.3 改进蝙蝠算法优化节点定位

1) 蝙蝠算法

蝙蝠算法 (bat algorithm, BA) 是由英国剑桥大学的 Yang^[17] 在 2010 年提出的一种新颖的群智能类算法。该算法原理是通过模拟蝙蝠使用回声定位来捕获猎物、躲避障碍的方法来寻找最优解。蝙蝠算法具有原理简单、可调参数少、收敛速度快等优点,在迭代寻优方面优于粒子群算法、遗传算法、和声搜索算法^[18]。该算法由虚拟蝙蝠速度和位置的变化及脉冲发射率和响度的变化两部分构成。

(1) 虚拟蝙蝠速度和位置的变化

在搜寻过程中,随脉冲频率变化的虚拟蝙蝠的速度和位置更新如式(17)~(19)所示。

$$f_i = f_{\min} + (f_{\max} - f_{\min})\beta \quad (17)$$

$$v_i^t = v_i^{t-1} + (x_i^t - x_*)f_i \quad (18)$$

$$x_i^t = x_i^{t-1} + v_i^t \quad (19)$$

式中: f_i 为搜寻脉冲频率, $f_i \in [f_{\min}, f_{\max}]$; β 为服从均匀分布的向量, $\beta \in [0, 1]$; v_i^t 和 x_i^t 分别为第 t 代的飞行速度和位置, x_* 为蝙蝠的最优位置。

在局部搜索的过程中,确定了蝙蝠的最佳位置后,蝙蝠采用随机游走方式,通过式(20)更新位置。

$$x_{new} = x_{old} + \epsilon A^t \quad (20)$$

式中: ϵ 为随机数, $\epsilon \in [-1, 1]$, A^t 为蝙蝠种群的平均响度, $A^t = \langle A_i^t \rangle$ 。

(2) 脉冲发射率和响度的变化

找寻猎物时,为了扩大搜索的区域,脉冲发射率较低而响度较大;发现猎物后,为了精准的对猎物进行定位,需要调整脉冲发射率和响度,使前者升高后者减小。脉冲发射率和响度按式(21)~(22)进行更新。

$$A_i^{t+1} = \alpha A_i^t \quad (21)$$

$$r_i^{t+1} = r_i^0 [1 - \exp(-\gamma t)] \quad (22)$$

式中: A_i^t 和 A_i^{t+1} 分别为第 t 和 $t+1$ 代的响度, α 为衰减因子, $0 < \alpha < 1$, r_i^0 和 r_i^{t+1} 分别为初始脉冲发射率和第 $t+1$ 代的脉冲发射率, γ 为增加系数, $\gamma > 0$ 。

2) 改进蝙蝠算法

传统的位置求解方法,使用前项减后项法引入次优解,造成定位的精度下降。为了提高定位的精度,本文采用改进 BA 代替最小二乘法求解未知节点的坐标。BA 在寻优的过程中,蝙蝠个体逐渐向最优区移动,致使种群的多样性下降,而 BA 没有变异机制容易陷入局部最优,导致求解的准确性降低。为了解决此问题,本文通过采用混沌映射和速度加权策略进行优化,提高算法的寻优性能。

(1) 混沌映射策略

混沌序列因其具有的独特性而被广泛应用于参数优化。混沌映射模型有很多,logistic 混沌映射模型易于实现且用于参数优化时比其他模型更具优势。本文选择 logistic 混沌映射模型,混沌映射可通过式(23)实现:

$$\beta_{i+1} = \mu \beta_i \times (1 - \beta_i), i = 1, 2, \dots, S-1 \quad (23)$$

式中: μ 为控制参量, β_i 为介于(0,1)之间的随机数。

初始种群的好坏会影响 BA 的有效性,合理的种群分布可以保证更快的收敛趋势和更好的寻优性能^[19]。在 BA 中加入混沌映射以提高种群的多样性,映射过程如式(24)所示。

$$x_i^t = \beta_i x_i^t \quad (24)$$

式中: x_i^t 为混沌扰动下蝙蝠的位置, β_i 为混沌值。

(2) 速度加权策略

在进行局部搜索时,BA 易陷入局部最优,通过设置速度权重因子来控制搜索的步长,使算法跳出局部最优,增强搜索的效率。蝙蝠的速度更新如式(25)~(26)所示。

$$f_i^d = f_{\min} + (f_{\max} - f_{\min}) \times \beta \quad (25)$$

$$v_i^{t+1} = \omega \times v_i^t + (x_i^t - x_*)f_i \quad (26)$$

式中: d 为搜索空间维度, ω 为速度权重因子。

2.4 改进 DV-Hop 定位算法流程

1) 误差适应度函数

未知节点和锚节点间总距离误差越小,求解的未知节点的坐标越精准,定位问题也就转化成了求解多跳误差最小化问题。自然启发式算法对求解此类问题具有高效的优势,本文采用增强混沌映射蝙蝠算法 (chaotic mapping enhanced bat algorithm, CEBA) 求解未知节点的位置。为了评估定位精度,将适应度函数表示为:

$$Fitness = \min \left(\sum_{i=1}^n \left| \sqrt{(x_u - x_i)^2 + (y_u - y_i)^2} - \hat{d}_{ui} \right| \right) \quad (27)$$

2) ICEBADV-Hop 算法步骤及详细流程

本文提出的 ICEBADV-Hop 算法的具体实施流程如图 1 所示。

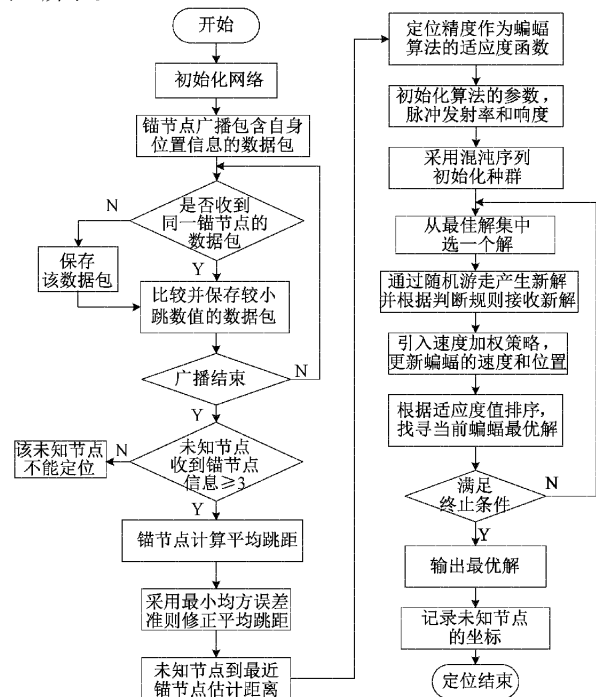


图 1 ICEBADV-Hop 定位算法的具体流程

本文提出的 ICEBADV-Hop 算法的详细实现步骤如表 1 所示。

表 1 ICEBADV-Hop 定位算法步骤

算法:ICEBADV-Hop 定位算法
输入:WSN 和 CEBA 的参数
输出:未知节点的估计位置 $\hat{\mathbf{x}} = (\hat{x}_u, \hat{y}_u)$
步骤:
1. 初始化 WSN 和 CEBA 的参数
2. 使用 Dijkstra 最短路径算法求节点间最小跳数
3. 锚节点通过式(15)代替式(1)计算修正后的平均跳距
4. 根据式(16)计算修正后的锚节点到未知节点的距离
5. 根据式(27)生成适应度函数,用 CEBA 对适用度函数进行寻优
6. 根据式(24)采用混沌映射序列初始化种群
7. 根据式(25)调整频率生成新的解,根据式(26)和(19)更新速度和位置
8. 根据式(21)~(22)调整响度和脉冲发射率
9. 更新适应度值,在每次迭代中进行排序,找出当前最优解
10. 当满足终止条件,则输出蝙蝠的最优位置,即未知节点的位置,否则继续寻优

3 仿真结果与分析

3.1 仿真参数设置

本文改进定位算法可以用于停车场车位检测,学校操场目标人员标定和环境检测点位置估计等诸多领域。为了验证改进算法 ICEBADV-Hop 的性能以及体现其应用价值。在 100 m×100 m 的正方形区域,进行停车场空余车位的检测和定位模拟,如图 2 所示。首先随机部署 100 个传感器节点,锚节点个数为 20 个,未知节点个数为 80 个,其次设置每个节点的通信半径 30 m,最后通过改进定位算法来预估未知节点的位置。为了衡量定位的精度,采用式(28)计算归一化平均定位误差。

$$ALE = \frac{\sum_{u=1}^{m-n} \sqrt{(x_u - \hat{x}_u)^2 + (y_u - \hat{y}_u)^2}}{(m-n)R} \quad (28)$$

式中: (x_u, y_u) 和 (\hat{x}_u, \hat{y}_u) 分别为未知节点 u 的实际位置和估计位置坐标, m 为节点的总数, n 为锚节点数, R 为节点通信半径。

为了验证提出的 IDV-Hop 算法和 ICEBADV-Hop 算法的定位性能,采用 MATLAB R2018b 进行仿真,并将仿真结果与算法 1(DV-Hop 算法)、算法 2(BADV-Hop 算法)和算法 3(PSODV-Hop 算法)进行比较。分析在不同锚节点比例、通信半径和节点总数下定位误差曲线变化的原因。

仿真实验中 ICEBADV-Hop 算法的网络仿真环境与 CEBA 参数设置如表 2 所示。

表 2 网络仿真环境与 CEBA 参数设置

参数	初始化值	参数	初始化值
网络模型			
拓扑结构	正方形	总节点数	100~400 个
区域边长	100 m	锚节点比例	10%~40%
运行次数	100 次	通信半径	20~45 m
CEBA			
蝙蝠种群数量 N	30	初始脉冲发射率 r^0	0.01
响度衰减因子 α	0.5	最大声波响度 A	0.25
脉冲发射率增加系数 γ	0.5	最大脉冲发射率 r	0.5
速度权重因子 ω	0.2	最大迭代次数	50
最小脉冲频率 f_{\min}	0	最大脉冲频率 f_{\max}	2
混沌映射控制参量 μ	4	阈值 M	100

3.2 仿真结果分析

1) 节点定位误差对比

设置 WSN 区域的边长为 100 m,节点总数为 100 个,锚节点个数为 20,通信半径为 30 m。仿真的网络节点分布图、DV-Hop 算法和 ICEBADV-Hop 算法节点误差分布分别如图 2~4 所示。由图中可以看出,本文提出的 ICEBADV-Hop 算法具有更高的定位精度。

2) 锚节点对定位误差的影响

设置 WSN 区域为 100 m×100 m,节点总数为 100 个,通信半径 30 m,仿真了锚节点比例从 10%增加到 40%时不同算法的定位误差对比图,如图 5 所示。

由图 5 可以看出,锚节点比例从 10%增加到 40%时,定位误差曲线的变化趋势均从明显下降逐渐走向平稳。这是因为锚节点比例的增加可以使网络的连通度得到提

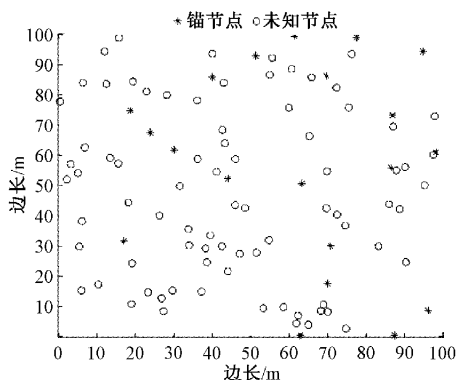


图 2 节点分布

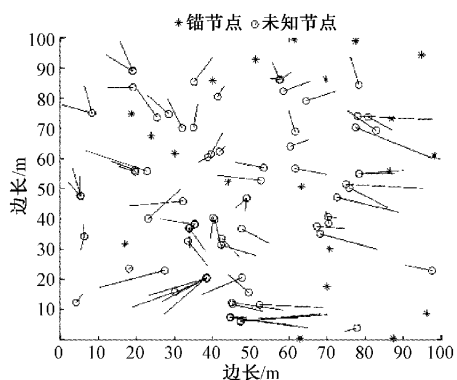


图 3 DV-Hop 算法节点误差分布

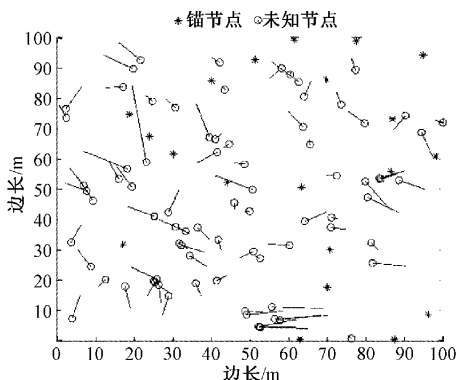


图 4 ICEBADV-Hop 算法节点误差分布

高,未知节点能够获得更多定位所需的信息,以致定位误差明显减小。当锚节点比例为 25% 时,ICEBADV-Hop 算法的定位精度比算法 1、算法 2、算法 3 分别提高了 30.89%、17.38%、7.71%。由此可知,在不同的锚节点比例下,ICEBADV-Hop 算法定位误差最小,这是因为在算法的第二阶段采用最小均方估计平均跳距并添加校正因子进行校正,这减小了测距的误差,提高了定位的精度。

3) 通信半径对定位误差的影响

设置 WSN 区域为 $100\text{ m} \times 100\text{ m}$, 节点总数为 100 个, 锚节点比例为 20%, 仿真了通信半径从 20 米增加到 45 米时不同算法的定位误差对比图, 如图 6 所示。

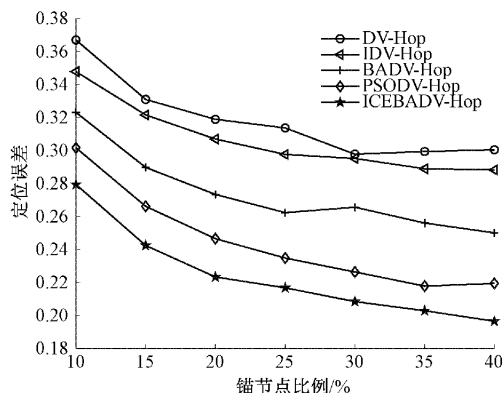


图 5 不同锚节点比例下定位误差对比图

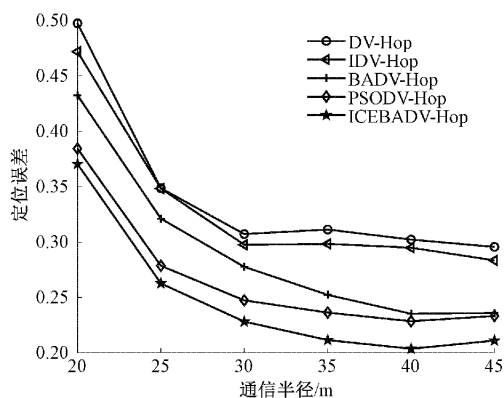


图 6 不同通信半径下定位误差对比图

由图 6 可以看出,各算法的定位误差随通信半径的增加呈下降趋势,这是因为通信半径的增加会使参与定位的锚节点数增多,导致节点间的跳数减少,使得测距更准确,定位更精准。当通信半径为 35 m 时,相比算法 1、算法 2、算法 3, ICEBADV-Hop 算法的定位精度分别提高了 32.05%、16.20%、10.49%。由此可知,在不同通信半径下,ICEBADV-Hop 算法在这几种算法中定位误差最小。

4) 节点总数对定位误差的影响

设置 WSN 区域为 $100\text{ m} \times 100\text{ m}$, 锚节点比例为 20%, 通信半径为 30 m, 仿真了节点总数从 100 个增加到 400 个时不同算法的定位误差对比图, 如图 7 所示。

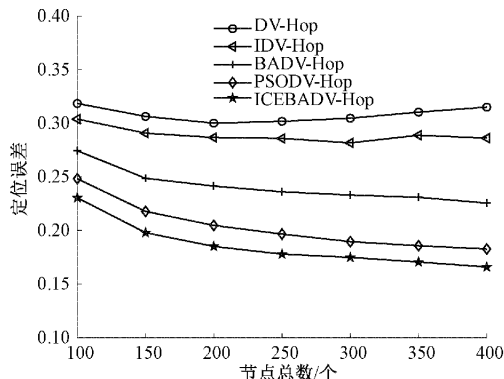


图 7 不同节点总数下定位误差对比图

由图 7 可以看出,各算法的定位误差曲线随节点总数的增加从明显下降逐渐趋向平稳。这是因为节点总数的增加使网络的连通性得到提升,这减少了节点间的距离和跳数,使计算的平均跳距更加准确,定位更加精准。但当节点总数增加到足够大时,继续增加会使 WSN 中的节点密度过高,节点间通信会产生较大的能耗,增加了非必要成本,降低了定位的性能。因此应该根据实际情况选择合适的节点总数。当节点总数为 250 个时,ICEBADV-Hop 算法的定位精度比算法 1、算法 2、算法 3 分别提高了 41.09%、24.66%、9.56%。由此可见,在不同节点总数下,ICEBADV-Hop 算法在这几种算法中定位误差最小且能够在较少的节点总数下达到较高的定位精度。

4 结 论

本文针对 DV-Hop 算法在求解平均跳距和未知节点位置两个阶段的缺陷,提出了 ICEBADV-Hop 算法。首先采用最小均方误差准则求解锚节点间的平均跳距,并添加校正因子缩小跳距误差;然后通过混沌映射和速度加权策略增强算法的效率;最后利用 CEBA 确定未知节点的位置以提高定位精度。实验仿真结果表明,本文提出的 ICEBADV-Hop 算法在不同锚节点比例、通信半径和节点总数下均具有较好的定位性能,与 DV-Hop 算法、BADV-Hop 算法、PSODV-Hop 算法相比,具有更高的定位精度。

参考文献

- [1] HARIKRISHNAN R, JAWAHAR S, KUMAR V, et al. A comparative analysis of intelligent algorithms for localization in wireless sensor networks [J]. *Wireless Personal Communications*, 2016, 87 (3): 1057-1069.
- [2] 彭宇, 王丹. 无线传感器网络定位技术综述[J]. *电子测量与仪器学报*, 2011, 25(5): 389-399.
- [3] GUI L, XIAO F, ZHOU Y, et al. Connectivity based DV-Hop localization for internet of things[J]. *IEEE Transactions on Vehicular Technology*, 2020, 69(8): 8949-8958.
- [4] SHAHZAD F, SHELTAI T R, SHAKSHUKI E M. DV-maxHop: A fast and accurate range-free localization algorithm for anisotropic wireless networks [J]. *IEEE Transactions on Mobile Computing*, 2017, 16 (9): 2494-2505.
- [5] 高森, 王迪, 鲁国阳, 等. 信号强度反演的质心迭代定位算法[J]. *中国测试*, 2021, 47(12): 92-97.
- [6] JIN Y, ZHOU L, ZHANG L, et al. A novel range-free node localization method for wireless sensor networks [J]. *IEEE Wireless Communications Letters*, 2022, 11(4): 688-692.
- [7] SINGH A, SHARMA S, SINGH J. Nature-inspired algorithms for wireless sensor networks: A comprehensive survey[J]. *Computer Science Review*, 2021, 39: 100342.
- [8] 褚银菲, 吕惠芳. 基于测距修正的差分进化的 DV-Hop 定位算法[J]. *国外电子测量技术*, 2021, 40(6): 33-37.
- [9] CUI L, XU C, LI G, et al. A high accurate localization algorithm with DV-Hop and differential evolution for wireless sensor network[J]. *Applied Soft Computing*, 2018, 68: 39-52.
- [10] CUI Z, SUN B, WANG G, et al. A novel oriented cuckoo search algorithm to improve DV-Hop performance for cyber-physical systems[J]. *Journal of Parallel and Distributed Computing*, 2017, 103: 42-52.
- [11] 印雷, 顾德, 刘飞. 基于改进麻雀搜索算法优化的 DV-Hop 定位算法[J]. *传感技术学报*, 2021, 34(5): 670-675.
- [12] KAUR A, KUMAR P, GUPTA G P. Nature inspired algorithm-based improved variants of DV-Hop algorithm for randomly deployed 2D and 3D wireless sensor networks[J]. *Wireless Personal Communications*, 2018, 101(1): 567-582.
- [13] CHEN J, ZHANG W, LIU Z, et al. CWDV-Hop: A hybrid localization algorithm with distance-weight DV-Hop and CSO for wireless sensor networks[J]. *IEEE ACCESS*, 2021, 9: 380-399.
- [14] NICULESCU D C S, NATH B. DV based positioning in ad hoc networks[J]. *Telecommunication Systems*, 2003, 22(1): 267-280.
- [15] 宋跃才, 林海涛, 卞媛, 等. 基于测距修正和蜜獾优化的改进 DV-Hop 定位算法 [J]. *电子测量技术*, 2022, 45(9): 147-153.
- [16] HADIR A, ZINE-DINE K, BAKHOUYA M, et al. An optimized DV-hop localization algorithm using average hop weighted mean in WSNs[C]. *The 2014 5th Workshop on Codes, Cryptography and Communication Systems(WCCCS 2014)*, IEEE, 2014: 25-29.
- [17] YANG X. A new metaheuristic bat-inspired algorithm[M]. *Nature inspired cooperative strategies for optimization (NICSO 2010)*, Springer, 2010: 65-74.
- [18] KHAN K, SAHAI A. A comparison of BA, GA, PSO, BP and LM for training feed forward neural networks in e-learning context [J]. *International Journal of Intelligent Systems and Applications*, 2012, 4(7): 23.
- [19] SAYED G I, HASSANIEN A E, AZAR A T. Feature selection via a novel chaotic crow search algorithm [J]. *Neural computing and applications*, 2019, 31(1): 171-188.

作者简介

董玉(通信作者), 硕士研究生, 主要研究方向为无线传感器网络、无线通信信道建模。

E-mail: 1370856944@qq.com

张治中, 博士, 教授, 主要研究方向为物联网、通信网测试、移动通信与信息处理。

E-mail: zhangzz@nuist.edu.cn

冯姣, 博士, 副教授, 主要研究方向为移动通信、通信网络协议及资源分配、无线通信信道。

E-mail: jiao.feng@nuist.edu.cn