Node Localization Algorithm Based on Multi-Communication Radius and Sparrow Search Algorithm*

PENG Duo, YANG Yawen, GAO Yuwei, WANG Chanfei*
(*School of Computer and Communication of Lanzhou University of Technology, Lanzhou Gansu* 730050, *China*)

Abstract: Aiming at the problem of low positioning accuracy of node localization algorithm in existing wireless sensor networks, a new LSSA-DV-Hop algorithm is proposed. In this algorithm, the concept of multi-communication radius is introduced to refine the number of hop nodes, and a correction factor is introduced to modify the number of hop nodes. The location problem is abstracted into a function optimization problem and the sparrow search algorithm is used to find the optimal solution of the function. At the same time, aiming at the problem that the optimization algorithm is easy to fall into the local optimal solution and the population diversity is reduced in the process of optimization, Levy is introduced and the dynamic decline factor is added to make the optimization algorithm have better global search ability in the early stage and stronger local search ability in the late stage. Simulation results show that compared with the classical DV-Hop algorithm and the existing improved algorithm, the proposed algorithm reduces the average location error by 63.23%, 48.60% and 12.92%, and the positioning accuracy is higher.

Key words: wireless sensor network; multi-communication radius; node localization; sparrow search algorithm
EEACC: 7230 **doi**: 10.3969/j.issn.1004-1699.2021.11.016

基于多通信半径和麻雀搜索的节点定位算法*

彭 铎*, 杨雅文, 高玉蔚, 王婵飞
(兰州理工大学计算机与通信学院, 甘肃 兰州 730050)

摘 要: 针对现有无线传感器网络中节点定位算法精度低的问题, 提出了一种 LSSA-DV-hop 算法。在此算法中引入了多通信半径的概念细化节点跳数, 对多跳情况引入修正因子, 对跳数进行修正。将定位问题抽象为函数寻优问题, 使用麻雀搜索算法寻求函数最优解。同时针对优化算法在寻优过程中存在的易陷入局部最优解、种群多样性减少的问题, 引入 Levy 飞行并添加动态递减因子在使其在前期有较好的全局搜索能力, 在算法后期有较强的局部寻优能力。仿真结果表明, 与经典 DV-hop 算法及现有改进算法相比提出算法定位误差平均值降低了 63.23%、48.60% 和 12.92%, 定位精度更高。

关键词: 无线传感器网络; 节点定位; 多通信半径; 麻雀搜索算法
中图分类号: TP393 **文献标识码**: A **文章编号**: 1004-1699(2021)11-1523-07

无线传感器网络(Wireless Sensor Network, WSN)是由低成本、低功耗、可自重构的传感器节点组成。并且无线传感器网络在军事、环境监测、工农业生产、医疗健康、建筑物监测、空间探测等领域有着广阔的应用前景和巨大的应用价值[1-2]。能够准确得知无线传感器网络的节点位置是将无线传感器网络技术应用于实际的前提, 因此, 如何提高无线传感器网络的定位精度是近年来的热点研究领域之一。现有的无线传感器网络定位算法主要分为距离相关算法和距离无关算法[3]。常见的距离相关算法有到达时间算法(ToA)、到达角度(AoA)、接收信号强度(RSS)等[4-5]。但由于距离相关算法需要额外的硬件支持, 因此距离无关算法应用更为广泛。

1 概述

DV-hop 算法作为距离无关算法, 具有易实现, 成本低, 不需要额外硬件设备等优点被广泛应用。近年来许多学者将与自然相关的进化算法应用于提

项目来源: 国家自然科学基金项目(61663024, 62061024); 甘肃省高校创新基金项目(2020A-021)
收稿日期: 2021-05-07 **修改日期**: 2021-08-09

高 DV-hop 算法定位精度中,如粒子群算法(PSO),鲸鱼优化算法(WOA),萤火虫算法(FA)等。以下文献采用了智能优化算法的思想:文献[6]提出了基于改进的布谷鸟算法,采用 Levy 飞行策略和柯西分布对布谷鸟算法进行改进,增强其寻优性能并应用于 DV-hop 算法中有效地提高了定位精度。文献[7]对跳距进行了修正,并且采用新型智能优化算法 runner-root 算法优化传统 DV-hop 算法,得到更加精确的未知节点坐标。文献[8]将定位问题抽象为函数寻优问题,采用并行鲸鱼优化算法提升了原算法定位精度。本文所采用的麻雀搜索算法(SSA)是新型智能优化算法,相比于其他优化算法有更强的寻优性以及收敛更快的优点;使用多种通信半径细化节点跳数以降低平均跳距误差的思想也被广泛应用。文献[9]采用双通信半径进行广播,使节点间的跳数不再为整数,使计算出的平均跳距更加精确,达到提高定位精度的目的。文献[10]利用多级通信半径修正锚节点到信邻节点的跳数信息。再根据信标节点与未知节点的距离,对平均跳距加权处理,并将每个加权后的平均跳距参与未知节点平均跳距的计算,使未知节点的平均跳距更符合实际网络情况。以上改进针对传统算法中使用平均跳距误差较大的问题对跳距进行了修正,或使用智能优化算法改善采用最小二乘法估计坐标存在的对初值敏感的问题。但以上改进仅单一的考虑了 DV-hop 算法在传播阶段或在坐标估计阶段产生误差的原因,因此本文提出了一种基于多通信半径和麻雀搜索算法的 WSN 定位方法,对算法在估计平均跳距阶段因为网络拓扑分布不均,节点间跳距误差较大的问题采用了多通信半径进行修正;以及对计算节点坐标阶段,使用最小二乘法对初值敏感、受测量误差较大的问题采用了 LSSA 智能优化算法进行修正,以此达到提高算法定位精度的目的。

1.1 DV-hop 算法

DV-hop 算法主要分为 3 个步骤[11]:

①计算最小跳数

各锚节点向整个网络广播发送包含其位置坐标的信息,节点间跳数初始化为 0,接收到信息的节点将跳数记为 1 跳,并将其转发。每转发一次跳数值加 1,每个节点记录其接收到的最小跳数值。对于接收到跳数较高的锚节点信息记为无用信息并忽略。

②估计到各锚节点的平均跳距

每个锚节点根据上一步中记录的与其他锚节点的位置信息和相距跳数,利用式(1)估算平均每跳的距离:

$$\mathrm{Hopsize}_i=\frac{\sum_{i\neq j}^{M}\sqrt{(x_i-x_j)^2+(y_i-y_j)^2}}{\sum_{i\neq j}^{M}h_{ij}} \tag{1}$$

式中:(x_i,y_i)和(x_j,y_j)是锚节点 i 和 j 的坐标,h_{ij}是锚节点 i 与 $j(i\neq j)$之间的跳数。获得各锚节点平均跳数 $\mathrm{Hopsize}_i$ 后,各锚节点向整个网络广播其平均跳距。未知节点保留接收到的第一个跳距信息并转发给邻居节点。

③未知节点位置估计

未知节点通过②获得到每个锚节点的信息,通过平均跳数和跳距计算到每个锚节点间的距离。再采用三边测量法或最大似然估计法计算其自身位置坐标。

$$d_{im}=\mathrm{Hopsize}_i\times\mathrm{hop}_{im} \tag{2}$$

式中:d_{im}和 hop_{im}为锚节点 i 和未知节点 m 之间的距离和最小跳数。

2 基于动态自适应 Levy 步长的麻雀搜索算法(LSSA)

2.1 麻雀搜索算法[12]

SSA 是一种受麻雀觅食行为和反捕食行为启发而提出的一种新型群体智能优化算法,SSA 算法相比于已有优化算法如:灰狼优化算法、粒子群优化算法、引力算法等,对单峰值函数、多峰值函数以及固定维度函数都有更好的寻优效果,即寻优精度更高,在同等迭代次数下能更快的收敛到最优解;SSA 算法相对于其他算法而言具有更好的鲁棒性和收敛速度,可以作为解决优化问题的有效工具。

SSA 算法将整个过程抽象为发现者和加入者模型,并且加入了侦查预警机制。在整个群体中,发现者搜索范围更广拥有较强的全局搜索能力,加入者为了获得更好的适应度向发现者靠近。当整个种群意识到危险时将会进行反捕食行为。

发现者数量为 20%,在迭代过程中发现者的位置更新定义为:

$$x_{id}^{t+1}=\begin{cases}x_{id}^{t}\cdot\exp\left(\dfrac{-i}{\alpha\times\mathrm{item}}\right) & R_2<\mathrm{ST}\\ x_{id}^{t}+QL & R_2\geqslant\mathrm{ST}\end{cases} \tag{3}$$

式中:t 为当前迭代次数;$item$ 为最大迭代次数;$\alpha\in(0,1]$为随机数;$R_2\in[0,1]$,$\mathrm{ST}\in[0.5,1]$分别表示安全值和预警值,Q 为服从正态分布的随机数,$\boldsymbol{L}$ 表示 $1\times d$ 的元素全为 1 的矩阵。当 $R_2<\mathrm{ST}$ 时,说明周围没有捕食者,麻雀可以安全的在当前位置觅食;当

$R_2 \geqslant ST$ 时,说明有麻雀已经意识到了危险,并发出警报,整个种群需要迅速的飞向安全区域撤离。

除发现者外其余麻雀为加入者,根据下式进行位置更新:

$$x_{id}^{t+1}=\begin{cases} Q\exp\left(\dfrac{xw_d^t - x_{id}^t}{i^2}\right) & i > \dfrac{n}{2} \\ xb_d^{t+1} + \dfrac{1}{D}\sum_{d=1}^{D}(\text{rand}\{-1,1\} \times \\ \quad | x_{id}^t - xb_d^{t+1} |) & i \leqslant \dfrac{n}{2} \end{cases} \tag{4}$$

式中:xw_d^t 表示第 t 次迭代的时候,麻雀的在 d 维最差位置;xb_d^{t+1} 表示第 $t+1$ 次迭代时麻雀在第 d 维的最佳位置;当 $i>n/2$ 时,表示第 i 个加入者未获得食物,需要去其他地方觅食;当 $i\leqslant 2/n$ 时,第 i 个加入者将在当前最优位置 xb 附近随机寻找一个位置进行觅食。

种群中负责侦查预警的麻雀占种群总数的 10%~20%,其位置按照下式进行更新:

$$x_{id}^{t+1}=\begin{cases} xb_d^t+\beta(x_{id}^t-xb_d^t) & f_i \neq f_g \\ x_{id}^t+K\left(\dfrac{x_{id}^t-xw_d^t}{|f_i-f_w|+\varepsilon}\right) & f_i = f_g \end{cases} \tag{5}$$

式中:β 表示步长控制参数,且 $\beta\sim N(0,1)$ 为一随机数;K 为 $[-1,1]$ 之间的一个随机数,控制麻雀的移动方向及步长;ε 为一最小常数;f_i 表示第 i 只麻雀的适应度值;f_g 和 f_w 分别为当前最优适应度和最差适应度值。

2.2　基于动态自适应 levy 步长的麻雀搜索算法

SSA 算法在搜索最优解时依然存在容易陷入局部最优解以及过早收敛的问题,因此提出采用动态自适应 Levy 步长改进 SSA 算法。

Levy 分布是由是由法国数学家莱维(Levy)提出的一种概率分布,Levy 飞行服从 Levy 分布,Levy 飞行可以扩大搜索空间,因此把 Levy 飞行引入智能优化算法中更容易使算法避免过早收敛。Levy 飞行位置更新式为

$$x_i^{t+1}=x_i^t+\alpha\oplus\text{Levy}(\lambda) \tag{6}$$

式中:x_i^t 为第 t 代的位置;$\oplus$为点对点乘法;Levy(λ)为随机搜索路径,并且满足:$\text{Levy}\sim u=t^{-\lambda}$,$1<\lambda\leqslant 3$,在实验中由于 Levy 分布较难实现,所以 Levy 分布通常由 Mantegna 算法来实现。其中的 α 为一固定常数,因此在搜索过程中可能存在搜寻能力较差以及精度不够高的问题,因此引入自适应动态步长 α_1 来代替 α。寻优初期粒子位置更新变化较大,需要较大步长,使粒子个体能快速寻找到最优解,寻优后期,需要较小步长来提高最优解的精度,因此需使步长的变化呈递减趋势,即 α_1 值越大则搜索步长较大,使算法更趋向于全局搜索;α_1 值越小则步长越小,使算法更趋向于局部寻优。步长改进如下式[13]:

$$\alpha_1=\alpha\exp\left(-\left|1-\frac{t}{t_{\max}}\right|\right) \tag{7}$$

则改进后的自适应 *Levy* 步长为

$$x_i^{t+1}=x_i^t+\alpha\exp\left(-\left|1-\frac{t}{t_{\max}}\right|\right)\oplus\text{Levy}(\lambda) \tag{8}$$

式中:t 为当前迭代数;$t_{\max}$ 为最大迭代次数,$\exp\left(-\left|1-\dfrac{t}{t_{\max}}\right|\right)$为动态递减因子,即 $\exp\left(-\left|1-\dfrac{t}{t_{\max}}\right|\right)$随迭代次数 t 的增加而减小,在寻优过程中 α 值较大或较小会导致整体步长增大或减小,此时添加动态因子的作用相应削弱,当 α 取值足够大或足够小时动态因子的作用则可以忽略不计,则使得在整个寻优过程中无法准确实现算法前期增大步长加强全局搜索能力,而在算法后期减小步长增强局部寻优的目的,因此本文取 $\alpha=1$,对动态因子影响最小,使成为控制步长的决定参数;

定义权重因子 f:

$$f=1-\frac{t}{t_{\max}} \tag{9}$$

SSA 算法较其他算法本身具有更好的寻优能力,因此在采用 Levy 飞行策略改进 SSA 算法时对种群中的部分个体进行扰动。对变异麻雀个体的选择,采用轮盘选择的思想,产生[0,1]之间的一个随机数 rand,定义惯性权重递减因子 f 如式(9),若 rand>f 则对当前最优解进行动态自适应 Levy 飞行变异。随迭代次数 t 的增加 f 逐渐减小,则产生随机[0,1]之间的数 rand 大于 f 的可能性就越小,即上述规则可使寻优前期对麻雀个体变异可能较大,更加有效的搜索到可能存在最优解的区域;算法后期对个体位置变异可能较小,主要在最优解可能存在的区域内进行局部搜索最优解,最大迭代次数过小可能导致算法寻优效果变差,过大会导致算法整体运行时间变长,因此本文设置最大迭代次数 $t_{\max}$ 为 50。

算法流程为:

Step 1　初始化,种群规模为 N,发现者个数为 p,侦查预警麻雀数为 s,目标函数维度 D,以及迭代次数。

Step 2　计算每只麻雀的适应度值 f_i,并对 f_i 进

行排序,选出当前最优适应度 f_g 和其对应位置 xb,以及最差适应度值 f_w 和其对应位置 xw。

Step 3 选取适应度优的前 p 个作为发现者,根据式(3)更新其位置。

Step 4 除发现者以外其余麻雀作为加入者,根据式(4)更新其位置。

Step 5 在种群中随机选取 s 只麻雀作为警戒者,并根据式(5)更新其位置。

Step 6 一次迭代完成后重新计算每只麻雀的适应度值 f_i。

Step 7 根据判断是否 rand>f 成立,选择采用式(8)进行 Levy 飞行变异,产生新解。

Step 8 是否满足停止条件,满足则退出,输出结果,否则,重复执行 Step 2~7;

3 基于多通信半径和 LSSA 的节点定位算法

3.1 多通信半径

在 DV-hop 算法中,只要在锚节点广播半径范围内的节点跳数均记为一跳。如图 1 所示,图中锚节点 A 与未知节点 B、C、D 间跳数均记为一跳,但 AB、AC、AD 的距离相差较大,采用原传播方式记录跳数存在较大误差,因此引入多通信半径进行传播。

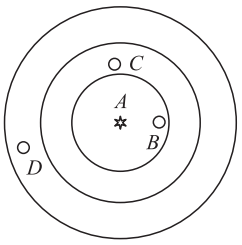


图 1 最大通信半径内节点通信图

在算法的第一阶段,锚节点传播半径为:

$$r=\frac{R}{4},\frac{2R}{4},\frac{3R}{4},R \tag{10}$$

式中:R 为最大通信半径,r 为广播半径。

3.2 多跳跳数修正

多跳传播时,已划分的单跳会影响多跳的值,减少多跳的误差,但是多跳的跳数值与实际的最优跳数依旧存在较大偏差。单一的锚节点估算出的平均跳距不能完全代表整个网络的跳距情况,并且多跳情况下,未知节点与锚节点的估算距离并不是按照直线计算,节点密度低的区域节点的折线率增大,距离误差将进一步累积,对定位效果产生影响[14]。

因此引入跳数修正因子[14],将锚节点 i,j 之间的距离定义为 d_{ij},节点 i,j 之间的真实距离与最大传播半径 R 的比值定义为相对最佳跳数 H_{ij}:

$$H_{ij}=\frac{d_{ij}}{R} \tag{11}$$

比较估算出的跳数 h_{ij} 和相对最佳跳数 H_{ij} 的差值,定义偏差系数 σ_{ij}:

$$\sigma_{ij}=\frac{h_{ij}-H_{ij}}{h_{ij}} \tag{12}$$

σ_{ij} 能够体现互相通信的锚节点间估算跳数 h_{ij} 与相对最佳跳数 H_ij 存在的差异情况。σ_{ij} 越大,标志着二者之间存在更大的偏差。在通信半径不变的情况下,估算跳数将大于或等于相对最佳跳数,利用式(13)定义差值修正系数 ω_{ij},以优化跳数信息减少误差的累积。

$$\omega_{ij}=1-\sigma_{ij}^2 \tag{13}$$

将 ω_{ij} 应用于全网跳数估计:

$$h'_{xj}=h_{xj}\omega_{ij} \tag{14}$$

式中:h_{xj} 为未知节点 x 到锚节点 j 的估计跳数,h'_{xj} 为修正后的跳数,且修正后跳数更接近于相对最佳跳数,跳数值误差将会更小。

3.3 坐标优化

无线传感器网络的定位问题可以抽象为求解函数最优值问题,确定适应度函数为:

$$f(x,y)=\sum_{p=1}^{M}\sqrt{(x-x_p)^2+(y-y_p)^2}-d_{pu} \tag{15}$$

式中:(x_p,y_p) 为已知锚节点 p 的坐标,d_{pu} 为未知节点 u 与锚节点 p 之间的距离。使用本文提出的 LSSA 算法寻求适应度函数最小值,得到的函数最小值坐标点即为优化后的未知节点坐标。

4 仿真结果分析

4.1 实验参数设置

为验证本文算法有效性,采用 MATLAB2014a 进行仿真实验。并将本文提出算法与传统 DV-hop 算法、PSO-DV-hop 算法以及文献[9]提出的双通信半径算法进行比较。设定仿真范围为 100 m×100 m 正方形区域,并在试验区域内随机生成 100 个节点,具体试验参数如表 1 所示。

表 1 网络环境及参数配置

参数	数值
网络规模	100 m×100 m
节点总数	100 个
锚节点个数	10 个~40 个
通信半径	20 m~50 m

本实验通过改变锚节点数量、节点通信半径以及节点总数进行实验,验证本文提出算法性能,并使

用所有节点的归一化平均定位误差作为衡量定位效果的标准,计算公式如下:

$$\text{error}=\frac{1}{N}\sum_{i=1}^{n}\frac{\sqrt{(x-x_i)^2+(y-y_i)^2}}{R} \tag{16}$$

式中:(x,y)为所求得未知节点坐标,(x_i,y_i)为未知节点实际坐标。R 为通信半径,N 为未知节点个数。

4.2　实验分析

设置节点总数为 100,锚节点数量为 25,节点通信半径为 30 m。本文提出算法与传统 DV-hop 算法定位效果如图 2 所示,可以看出本文提出算法估计节点位置相对传统算法更加精确,本文算法估计的部分节点位置与实际节点位置重合,个别节点仍存在误差,但整体定位误差较小。

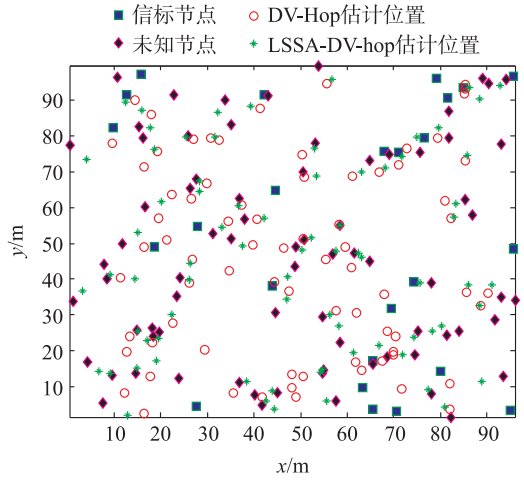


图 2　LSSA-DV-hop 定位效果

4.2.1　定位误差分析

在 100m×100m 的检测区域内随机分布 100 个节点,设置锚节点数为 20 个,通信半径为 30 m,四种算法误差如图 3 所示。

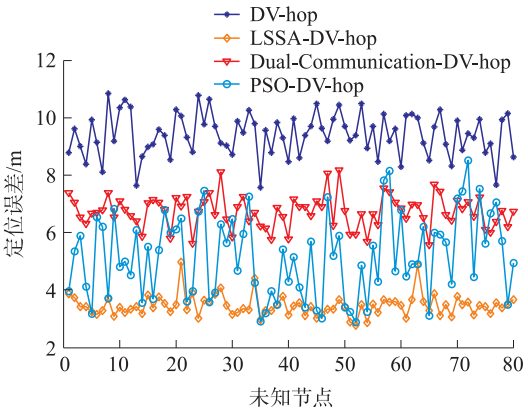


图 3　各未知节点定位误差

可以看出各节点定位误差相比于经典 DV-hop 算法、双通信半径算大以及 PSO-DV-hop 算法均为最低。DV-hop 算法定位误差平均值为 9.417 8 m,双通信半径算法定位误差平均值为 6.736 1 m,POS-DV-hop 算法定位误差平均为 5.189 4 m,本文算法 LSSA-DV-hop 定位误差平均值为 3.462 1 m。与其他三种算法相比,LSSA-DV-hop 算法定位误差分别降低了 63.23%、48.60%和 12.92%。本文所提出算法的定位精度有了明显提高。

4.2.2　锚节点数对平均定位误差影响

在检测区域随机分布 100 个传感器节点,通信半径为 30 m,锚节点数从 5 到 35 时,对比四种算法的定位精度,如图 4 所示。

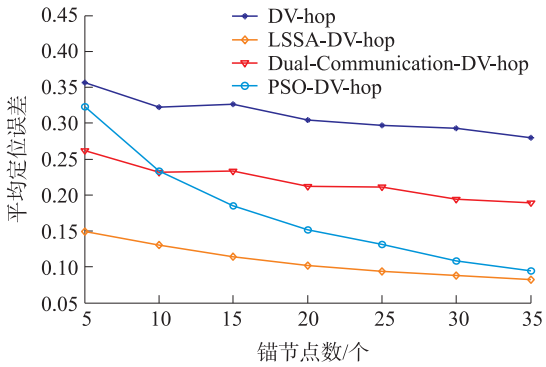


图 4　锚节点数对定位性能影响

随着锚节点数量的增加四种算法的定位误差整体呈下降趋势,这是由于随锚节点数量增加,各节点的平均跳距误差减小,未知节点和锚节点间的距离估计更加准确。相比于其余三种算法,在同等条件 LSSA-DV-hop 算法误差始终最小,并且 PSO-DV-hop 算法在锚节点数为 5 时误差为 0.322 9 误差较大。在锚节点数量为 15 以上时,误差率在 8%~12%之间,当锚节点个数为 35 时误差最小为 8.22%,此时 DV-hop 算法误差、双通信半径算法及 PSO-DV-hop 算法误差分别为 27.98%、18.90%和 9.42%,本文算法平均定位误差分别降低了 70.62%、56.50%和 12.74%。在锚节点个数为 5 个~35 个时算法的定位精度均有了明显提高。

4.2.3　通信半径对平均定位误差影响

在检测区域随机分布 100 个传感器节点,锚节点数量设置为 30 个。改变节点通信半径由 20 m 到 50 m,结果如图 5 所示。

随通信半径增加定位误差总体呈下降趋势,这是因为通信半径较小时未知节点可通信区域存在锚节点数量较少,未知节点无法准确获得自身位置。在同等条件下可以看出 LSSA-DV-hop 算法定位误差始终低于其余三种算法。平均误差在 5%~29%之间,在通信半径为 50m 时,平均误差最小为 5.38%,此时 DV-hop 算法误差、双通信半径算法和

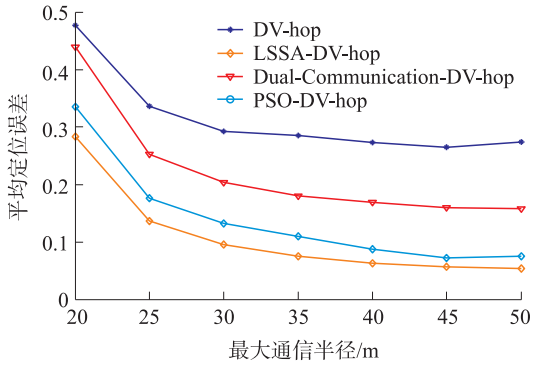


图 5 通信半径对定位性能影响

POS-DV-hop 算法误差分别为 27.43%、15.77% 和 7.52%，LSSA-DV-hop 算法平均定位误差分别降低了 80.39%、65.88%和 28.46%，定位误差有了明显的下降。

4.2.4 节点总数对平均定位误差影响

设置通信半径为 30 m，锚节点数量为总节点数的 20%，节点总数为 80 到 160 时比较三种算法定位误差，如图 6 所示。

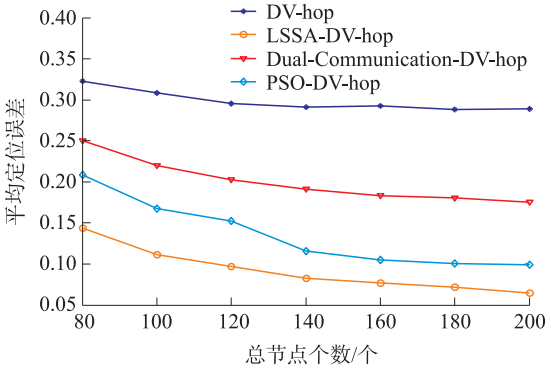


图 6 节点总数对定位性能影响

可以看出四种算法随节点总数增加定位误差呈下降趋势，这是因为随节点总数增加网络连通度提高，未知节点可以获取更多辅助定位的信息。相同条件下本文算法误差始终最低。LSSA-DV-hop 算法定位误差在 6%～14%，在节点总数为 200 时，平均误差最小为 6.51%，此时 DV-hop 算法误差、双通信半径算法和 POS-DV-hop 算法误差分别为 28.87%、17.56%和 9.89%，本文算法平均定位误差分别降低了 77.45%、62.92%和 34.18%。说明本文所提算法在各情况下定位误差明显小于 DV-hop 算法、双通信半径算法以及 PSO-DV-hop 算法。

4.2.4 算法复杂度分析

由于无线传感器网络通常资源有限的因此在设计定位算法时不能只考虑提升算法定位精度的问题，算法的复杂度也是重要因素之一。设节点总数为 n，锚节点数为 m，则 DV-hop 算法中计算节点间最小跳数的复杂度为 $o(n^3)$，计算平均跳距的复杂度为 $o(n)$，计算锚节点与未知节点间距离的复杂度为 $o(m\times n)$。[15] 文献[9]中，在原始 DV-hop 算法上增加复杂度 $o(m\times n)$；PSO-DV-hop 算法相较于 DV-hop 算法复杂度增加 $o(n^2)$。本文所提算法在多通信半径阶段复杂度为 $o(m\times n)$，麻雀搜索算法阶段复杂度为 $o(n^2)$。因此本文所提算法复杂度相较于 DV-hop 算法复杂度增加了 $o(n^2)+o(m\times n)$；相较于文献[9]算法复杂度增加了 $o(n^2)$；相较于 PSO-DV-hop 算法复杂度增加 $o(m\times n)$，但本文所算法定位误差降低明显，在各情况下定位性能稳定，明显优于 DV-hop 算法、文献[9]所提出算法的双通信半径算法以及 PSO-DV-hop 算法。

5 总结

本文通过分析现有定位算法定位精度低的原因，对产生误差的定位步骤进行优化，提出了基于多通信半径和麻雀搜索的节点定位算法。首先，在广播阶段采用多通信半径的形式细化节点间跳数，并对多跳情况产生的偏差使用跳数修正因子进行修正，使平均跳距更加精确。然后，引入了寻优能力更强的 LSSA 优化算法对未知节点定位过程进行改进，使定位算法计算出的未知节点坐标更精确，仿真结果表明本文提出的 LSSA-DV-hop 算法定位精度明显提高，具有良好的定位性能。

参考文献：

[1] Kumar, Rajesh Kumar, Sushil, et al. Geometrical Localization Algorithm for Three Dimensional Wireless Sensor Networks [J]. Wireless Personal Communications, 2014, 79(1): 249-264.

[2] 饶敬远. 基于无线传感网络的随机协同密钥预分配方案的研究[D]. 南昌：南昌大学，2017.

[3] Liu Q, Wang Z, He X, et al. Event-Based Recursive Distributed Filtering over Wireless Sensor 244 Networks [J]. IEEE Transactions on Automatic Control, 2015, 60(9): 2470-2475.

[4] Yanbin Zou, Huaping Liu. An Efficient NLOS Errors Mitigation Algorithm for TOA-Based Localization [J]. Sensor, 2020, 20(5): 1403.

[5] Tomic S, Beko M, Dinis R, et al. On Target Localization Using Combined RSS and AoA Measrements [J]. Sensors, 2018, 18(4): 1266.

[6] Cui Z H, Sun B, Wang G 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.

[7] Kanwar, Vivek Kumar, Ashok. DV-Hop-Based Range-Free Localization Algorithm for Wireless Sensor Network Using Runner-Root Optimization[J]. Journal of Supercomputing, 2020, 7: 1-18.

[8] Chai Q W, Chu S C, Pan J S, et al. A Parallel WOA with Two Communication Strategies Applied in DV-Hop Localization Method [J]. EURASIP Journal on Wireless Communications and Networking, 2020, 50: 1-10.

[9] Tuochen Li, Changzhu Wang, Qi Na. Research on DV-Hop Improved Algorithm Based on Dual Communication Radius [J]. EURASIP Journal on Wireless Communications and Networking, 2020, 113: 1-10.

[10] 邴晓瑛,徐保国. 基于多通信半径的加权 DV-Hop 改进算法[J]. 计算机应用与软件,2016,7(33):125-129.

[11] Xiao Chen, Benliang Zhang. Improved DV-Hop Node Localization Algorithm in Wireless Sensor Networks [J] International Journal of Distributed Sensor Networks, 2012, 8: 393-422.

[12] Xue J K, Shen B. A Novel Swarm Intelligence Optimization Approach: Sparrow Search Algorithm [J]. Systems Science & Control Engine ering an Open Access Journal, 2020, 8(1): 22-34.

[13] 张超,贺兴时,叶亚荣. 基于精英策略和 Levy 飞行的粒子群算法[J]. 西安工程大学学报,2018,32(6):731-738.

[14] 任克强,邓浪. 全局跳数优化与跳距误差修正的 DV-Hop 改进算法[J]. 传感技术学报,2019,32(3):438-443.

[15] 吕敬祥,龙满生,尹凯. 基于加权最小二乘优化的 DV-HOP 定位算法[J]. 传感技术学报,2020,33(3):450-455.



彭　铎(1976—),男,甘肃省兰州市人,兰州理工大学副教授,博士,硕导,主要研究方向为无线传感器网络,光纤网络,无线通信,pengduo7642@163.com;



高玉蔚(1996—),男,甘肃省兰州市人,硕士研究生,主要研究方向为无线传感器网络,1090667803@qq.com。



杨雅文(1996—),女,甘肃省兰州市人,兰州理工大学通信与信息系统专业研究生,主要研究方向为无线传感器网络,1053713244@qq.com;