

# 改进的人工蜂群优化 DV-Hop 定位算法

刘燕\*, 高丽

兰州交通大学电子与信息工程学院, 甘肃 兰州 730070

**摘要** 针对距离向量跳段(DV-Hop)算法在多边定位中误差较大的问题,详细分析了定位误差形成的原因,并提出了一种改进的人工蜂群优化 DV-Hop 定位算法。该算法引入数学优化模型,使用改进的区域限定人工蜂群算法对该模型进行目标寻优,优化多边定位的执行过程。实验结果表明,改进的人工蜂群算法可减少多边定位阶段的误差和计算量,且定位效果较好。

**关键词** 无线传感器网络; 人工蜂群; 距离向量跳段(DV-Hop)算法; 节点定位

**中图分类号** TN929.5; TP212.9; TP18

**文献标志码** A

**doi:** 10.3788/LOP57.192303

## Optimized DV-Hop Localization Algorithm Based on Improved Artificial Bee Colony

Liu Yan\*, Gao Li

College of Electronic and Information Engineering, Lanzhou Jiaotong University, Lanzhou, Gansu 730070, China

**Abstract** Aiming at the large error problem of distance vector-hop (DV-Hop) algorithm in multilateral positioning, the cause of positioning error is analyzed in detail, and an improved artificial bee colony optimization DV-Hop localization algorithm is proposed in this work. The algorithm introduces a mathematical optimization model, uses an improved area-limited artificial bee colony algorithm to optimize the model, and optimizes the execution process of multilateral positioning. Experimental results demonstrate that the modified artificial bee colony algorithm can reduce the error and the amount of calculation in the multilateral positioning stage, and the positioning effect is good.

**Key words** wireless sensor networks; artificial bee colony algorithm; distance vector-hop (DV-Hop) algorithm; node localization

**OCIS codes** 280.4788; 150.1135

## 1 引言

无线传感器网络(WSN)是信息传感技术的一次技术革命,集合了低功耗嵌入式技术、无线通信技术、微电子系统等。WSN由大量静态和移动传感器节点组成,具有自组织感知、计算和无线通信能力,可实现网络覆盖区域的采集、监控功能,具有成本低、功耗低、组织灵活、部署迅速等特点,为人类的生产和生活提供了极大的便利,广泛应用于环境监测、国防、目标追踪等领域。

WSN定位算法可分为基于距离和无距离的定位。基于距离的定位算法包括到达时间(TOA)<sup>[1]</sup>、

到达时间差(TDOA)<sup>[2]</sup>、到达角(AOA)<sup>[3]</sup>和接收信号强度(RSSI)<sup>[4]</sup>等指标。而无距离算法仅依赖于网络拓扑和节点的连通性,具有实现简单、功耗低的特点,受到了广泛关注。典型的无距离算法有近似三角形内点测试(APIT)算法<sup>[5]</sup>、Amorphous算法<sup>[6]</sup>、质心算法<sup>[7]</sup>、多维定标节点定位(MDS-MAP)算法<sup>[8]</sup>和距离向量跳段(DV-Hop)<sup>[9]</sup>算法。

DV-Hop算法是一种基于距离矢量路由原理的分布式定位算法,具有简单易实现的特点,在无距离算法中应用最广泛。但DV-Hop算法的定位精度不高,主要误差来自于锚节点(AN)与未知节点(UN)之间的距离估计以及根据估计距离使用最大

收稿日期: 2019-12-30; 修回日期: 2020-02-21; 录用日期: 2020-03-20

\* E-mail: 1002406287@qq.com

似然估计或最小二乘(LS)法计算的节点位置。改进的 DV-Hop 定位算法存在复杂度高、成本开销大与定位精度低等问题,因此,人们在 DV-Hop 算法的第三阶段利用进化算法,如遗传算法<sup>[10]</sup>、粒子群优化算法<sup>[11]</sup>、布谷鸟搜索算法<sup>[12]</sup>优化多目标定位的执行过程,但群体智能进化算法存在早熟收敛或收敛速度慢、容易陷入局部最优等问题。

为了提高 DV-Hop 定位算法的精度和解决人工蜂群(ABC)算法的收敛问题,本文提出了一种基于区域限定蜜源更新的 ABC 算法。在 ABC 算法初始化蜜源阶段,根据锚节点信息对未知节点的可能范围进行区域划分,再使用智能算法进行目标寻优,获得未知节点的估计坐标。同时仿真分析了不同实验条件下本算法的性能。

## 2 理论背景

### 2.1 DV-Hop 算法

DV-Hop 算法的核心理论是用锚节点与未知节点间的平均跳距和相应跳数的乘积估计二者间距离,主要步骤如下。

1) 锚节点广播自身的消息包,以获得从锚节点到其他节点的最小跳数。

所有锚节点通过广播的方式向邻居节点广播自身的消息包,包含锚节点的位置和初始设置为 0 的跳数,当消息包被其他节点重新广播时,跳数会增加 1。如果一个节点从同一个锚节点接收到另一条信息,但跳数与保存的数量相比较,则忽略该信息,所有节点都能得到每个锚节点的位置以及从自身到其他锚节点的最小跳数。

2) 计算所有锚节点的平均每跳距离(AHD)并将其广播到网络。

根据第  $i$  个锚节点获得的其他锚节点的位置和相应的最小跳数,计算第  $i$  个锚节点的 AHD,可表示为

$$D_i = \frac{\sum_{j=1, j \neq i}^N \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}}{\sum_{j=1, j \neq i}^N h_{ij}}, \quad (1)$$

式中,  $N$  为 AN 的总数,  $(x_i, y_i)$  和  $(x_j, y_j)$  分别为第  $i, j$  个 AN 的坐标,  $h_{ij}$  为第  $i, j$  个 AN 之间的最小跳数。所有锚节点的 AHD 都通过泛洪在网络中广播,使未知节点可由其最近的锚节点接收。

3) 执行定位计算

未知节点到锚节点的估计距离等于步骤 2) 中

的 AHD 与最小跳数的乘积,使用最大似然估计法或 LS 法可获得未知节点的坐标。

设  $(x_u, y_u)$  为第  $u$  个 UN 的坐标,  $d_{ui}$  为第  $u$  个 UN 到第  $i$  个 AN 的估计距离,可表示为

$$d_{ui} = D_u \times h_{ui}, \quad (2)$$

式中,  $h_{ui}$  为第  $u$  个 UN 到第  $i$  个 AN 的最小跳数,  $D_u$  为第  $u$  个 UN 保存的 AHD。所有锚节点的实际坐标及其到未知节点的估计距离,可表示为方程组

$$\begin{aligned} \sqrt{(x_u - x_1)^2 + (y_u - y_1)^2} &= d_{u1} \\ \sqrt{(x_u - x_2)^2 + (y_u - y_2)^2} &= d_{u2} \\ &\vdots \\ \sqrt{(x_u - x_N)^2 + (y_u - y_N)^2} &= d_{uN}. \end{aligned} \quad (3)$$

可将(3)式转换为矩阵形式  $\mathbf{AX} = \mathbf{B}$ , 其中  $\mathbf{A}$ ,  $\mathbf{B}$  和  $\mathbf{X}$  可表示为

$$\begin{aligned} \mathbf{X} &= \begin{bmatrix} x_u \\ y_u \end{bmatrix} \\ \mathbf{A} &= \begin{bmatrix} 2(x_1 - x_N) & 2(y_1 - y_N) \\ 2(x_2 - x_N) & 2(y_2 - y_N) \\ \vdots & \vdots \\ 2(x_{N-1} - x_N) & 2(y_{N-1} - y_N) \end{bmatrix}. \\ \mathbf{B} &= \begin{bmatrix} x_1^2 - x_N^2 + y_1^2 - y_N^2 - d_{u1}^2 + d_{uN}^2 \\ x_2^2 - x_N^2 + y_2^2 - y_N^2 - d_{u2}^2 + d_{uN}^2 \\ \vdots \\ x_{N-1}^2 - x_N^2 + y_{N-1}^2 - y_N^2 - d_{u(N-1)}^2 + d_{uN}^2 \end{bmatrix} \end{aligned} \quad (4)$$

用 LS 法计算该矩阵方程,得到第  $u$  个 UN 的估计位置,可表示为

$$\mathbf{X} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{B}. \quad (5)$$

从 DV-Hop 算法的执行过程来看,误差主要来源于锚节点的 AHD 和最小跳数的准确性、计算未知节点的 AHD、多目标定位的优化。为了减少误差,实验使用改进的 ABC 算法优化目标定位阶段的定位精度。

### 2.2 人工蜂群算法

ABC 算法是一种通过群体中个体间的协作寻找最优解的智能算法,文献[13]证明了 ABC 算法的全局收敛性。在 ABC 算法中,人工蜂群由雇佣蜂、旁观蜂和侦察蜂组成。雇佣蜂进行食物源的挖掘,旁观蜂根据轮盘赌选择法随机选择一个食物源,侦察蜂的首要任务是在搜索陷入局部最优时寻找新的食物源。随机食物源的位置对应优化问题的随机

解,食物源的花蜜量表示适应度。雇佣蜂与食物源一一对应,雇佣蜂的数量与旁观蜂相同,侦察蜂的数量只有一只。假设在  $D$  维空间中,蜜源数量为  $X_{SN}$ ,第  $n$  个食物源的位置为  $(x_{n1}, x_{n2}, \dots, x_{nD}), n \leq X_{SN}$ ,则人工蜂群寻找最佳食物源的步骤如下。

1) 雇佣蜂阶段。一个雇佣蜂对应一个食物源,每个雇佣蜂在当前食物源周围进行邻域搜索,产生新的食物源,然后根据贪婪准则选择更好的食物源。

2) 旁观蜂阶段。雇佣蜂通过摇摆舞将食物源信息传达给旁观蜂,旁观蜂根据得到的信息通过概率选择食物源进行邻域搜索。食物源的花蜜质量越高,被选择的概率越大,同时产出新的食物源,并根据贪婪准则选择一个比旧食物源质量更好的食物源。

3) 侦查蜂阶段。如果某一食物源没有通过预先设定的循环次数得到改善,表明该食物源已经耗尽,此时雇佣蜂成为侦察蜂,随机产生新的食物源。

在 ABC 算法中,每一个雇佣蜂和旁观蜂都会在其当前位置的邻域内,通过搜索方程生成一个新的食物源,可表示为

$$v_{nw} = x_{nw} + \varphi \cdot (x_{nw} - x_{kw}), \quad (6)$$

式中,  $x_{nw}$  为原食物源的位置,  $v_{nw}$  为邻域搜索的新食物源位置,  $\varphi$  为  $[-1, 1]$  范围内的随机数,  $w \in \{1, 2, \dots, D\}; n, k \in \{1, 2, \dots, X_{SN}\}$ , 且  $n \neq k$ 。

搜索过程中,雇佣蜂通过摇摆舞分享食物源的信息后,旁观蜂根据轮盘赌选择法获得食物源。与食物源相关的选择概率可表示为

$$P_n = \frac{f_n}{\sum_{k=1}^{X_{SN}} f_k}, \quad (7)$$

式中,  $f_n$  为食物源  $n$  的花蜜质量,对应优化问题中的适应度值,可表示为

$$f_n = \begin{cases} \frac{1}{(1+F_n)}, & F_n \geq 0 \\ 1 + \text{abs}(F_n), & F_n < 0 \end{cases}, \quad (8)$$

式中,  $F_n$  为与  $f_n$  对应食物源  $n$  的目标函数值,  $\text{abs}()$  为取绝对值的函数。

如果一个食物源在限定时间内没有更新,侦察蜂会生成一个新的食物源,可表示为

$$x'_{nw} = l_w + \phi \cdot (u_w - l_w), \quad (9)$$

式中,  $x'_{nw}$  为随机生成的新食物源位置,  $\phi$  为  $[0, 1]$  范围内的随机数,  $l_w$  和  $u_w$  分别为  $w$  维度上坐标值的上限和下限。

ABC 算法是搜索解空间最优解的一种迭代算

法,在优化节点定位问题时可避免对初始值的依赖性,且具有很强的全局最优性。但与其他进化算法一样,ABC 算法也存在收敛速度慢、早熟等问题,很难得到全局最优解。

### 3 改进的人工蜂群优化多边定位算法

#### 3.1 人工蜂群算法的改进

群体智能算法在多目标定位中可以显著提高定位精度,ABC 算法相比其他智能算法具有更强的全局最优性,不易陷入局部最优,但在定位过程中收敛速度慢。为了进一步提高 ABC 算法的定位性能,加快收敛速度,同时避免算法在迭代后期陷入局部最优,提出了基于区域限定的人工蜂群(RABC)算法,采用多种策略对 ABC 算法进行优化,有效改善了 ABC 算法的定位性能和定位精度,具体步骤如下。

1) 基于区域限定的蜜源更新及雇佣蜂搜索范围

初始蜜源范围对 ABC 算法的收敛速度和寻优结果影响较大,使用随机初始化方法,可在一定程度上保证种群的随机分布及多样性,但会影响算法的收敛速度。因此在初始化蜜源阶段,根据锚节点信息,利用 Max-Min 思想对未知节点的可能范围进行区域划分,改变雇佣蜂的搜索区域,以提高收敛速度。首先获取三个锚节点到未知节点的估计距离,然后以三个锚节点为圆心,锚节点到该未知节点的距离为半径画圆,用这些圆的外接正方形作为一个矩形区域,然后在该矩形区域内随机生成初始蜜源,如图 1 所示。其中,阴影部分为初始蜜源区域,即雇佣蜂的搜索区域。

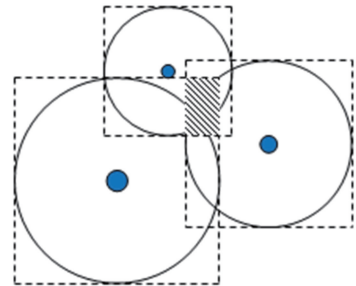


图 1 初始蜜源区域示意图

Fig. 1 Schematic diagram of initial nectar source area

2) 扰动频率

限定初始蜜源的位置区域后,可大大提高算法的收敛速度,但在寻优的迭代后期会存在蜜源位置相似度高,导致位置更新速度变慢,搜索能力下降。为了增加搜索后期的蜜源多样性,同时避免

ABC算法出现早熟和陷入局部最优,在位置更新过程中增加扰动频率,引入可修改控制参数( $X_{MR}$ )以提高收敛速度<sup>[14]</sup>。位置更新过程可表示为

$$v_{nw} = x_{nw} + (\varphi - \phi)(x_{nw} - x_{kw}) \times X_{MR} \quad (10)$$

(10)式可修改所有位置的参数,但较低的 $X_{MR}$ 会导致搜索寻优过程缓慢,而较高的 $X_{MR}$ 会增加寻优解的多样性。

### 3.2 算法模型

假设种群大小为 $S$ ,雇佣蜂和观察蜂的规模为 $0.5S$ 。在定位过程中,每个蜜源可表示为一个二维坐标,第 $n$ 个蜜源的位置坐标为 $(x_{n1}, x_{n2})$ 。采用ABC算法执行节点定位时,可用第 $u$ 个UN到第 $n$ 个AN的估计距离与实际距离存在的误差作为目标函数,可表示为

$$F_{un} = \sum_{i=1}^N \left| \sqrt{(x_{n1} - x_i)^2 + (x_{n2} - y_i)^2} - d_{ui}^2 \right|, \quad (11)$$

式中, $(x_i, y_i)$ 为锚节点的位置坐标, $F_{un}$ 为第 $u$ 个UN下第 $n$ 个食物源的目标函数。与 $F_{un}$ 对应的蜜源适应度可表示为

$$f_{un} = \begin{cases} \frac{1}{(1+F_{un})}, & F_{un} \geq 0 \\ 1 + \text{abs}(F_{un}), & F_{un} < 0 \end{cases} \quad (12)$$

改进后算法的具体步骤如下。

1) 初始化种群。在限定区域内随机确定蜜源位置,同时计算每个蜜源的适应度,蜂群由雇佣蜂和观察蜂组成。

2) 雇佣蜂在各自的蜜源附近搜索新的蜜源。

3) 基于贪婪选择策略确定是否更新原蜜源。

4) 计算每个蜜源在所有蜜源中所占的比重,选择比重较大的作为观察蜂,观察蜂在其附近根据贪婪准则搜索新的蜜源。如果新蜜源的适应度高于原蜜源,则更新位置,否则保持原位置。

5) 若该蜜源经过若干次搜索后适应度仍没有提高,则放弃该蜜源,并由侦察蜂随机选择一个新蜜源。

6) 记录每次迭代的最优解,重复步骤2)~步骤6),直至满足循环条件,算法结束。

### 3.3 实验结果与讨论

实验在内存为8 G,CPU为Inter Core i7,操作系统为Windows10的计算机上进行,仿真软件为MatlabR2018。在尺寸为100 m×100 m的区域内随机生成并部署所有传感器节点,随机选取部分传

感器节点作为锚节点。为了分析改进的ABC算法在随机分布节点下的定位精度和算法复杂度,并验证实验数据的有效性和普遍性,在相同条件下,进行了50次独立仿真,每次仿真都重新随机分布传感器节点。

假设 $(x_u^{\text{true}}, y_u^{\text{true}})$ 为未知节点的实际坐标, $(x_u^{\text{est}}, y_u^{\text{est}})$ 为相应的估计坐标, $R$ 为传感器节点的最大通信半径, $X_{\text{num}}$ 为未知节点的总数,一次仿真的网络标准化平均相对定位误差可表示为

$$\epsilon_{\text{network}} = \frac{\sum_{u=1}^{X_{\text{num}}} \sqrt{(x_u^{\text{true}} - x_u^{\text{est}})^2 + (y_u^{\text{true}} - y_u^{\text{est}})^2} / R}{X_{\text{num}}} \quad (13)$$

为了测试RABC算法在多目标定位中的性能,将仿真结果与DV-Hop和ABC算法进行对比,同时比较和分析定位误差与节点总数 $X_{\text{SN}}$ 、锚节点总数 $X_{\text{AN}}$ 和最大通信半径 $R$ 的变化关系。在不同节点总数条件下,设置锚节点总数为30,最大通信半径为30 m。在不同锚节点个数条件下,设置节点总数为100,最大通信半径为30 m。在不同最大通信半径条件下,设置节点总数为100,锚节点总数为30。

实验中RABC算法的参数设置如表1所示,其中, $T_{\text{Max}}$ 为最大迭代次数, $C$ 为单个食物源的最大迭代次数,仿真时分别设置为10和100。此外,ABC算法的参数与RABC的参数相同。

表1 RABC算法的参数

Table 1 Parameters of RABC algorithm

Parameter	Value
$T_{\text{Max}}$	10
$S$	50
$D$	2
$X_{\text{MR}}$	2
$C$	10/100

按照表1中的参数得到 $C=10$ 时,50次独立仿真下 $\epsilon_{\text{network}}$ 的均值随节点总数 $X_{\text{SN}}$ 、锚节点总数 $X_{\text{AN}}$ 和最大通信半径 $R$ 的变化曲线,如图2所示。可以发现,在最大迭代次数为10时,RABC算法的性能明显优于DV-Hop和ABC算法,原因是ABC算法的收敛速度慢。在图2(a)中,当节点总数增加时,DV-Hop的 $\epsilon_{\text{network}}$ 均值总体呈上升趋势,而ABC和RABC两种算法的均值略有下降,在传感器节点过多时均值略有上升,但总体变化不明显,这表明节点总数对三种算法的影响不大。在图2(b)中,

所有算法的  $\epsilon_{network}$  均值随锚节点总数的增加而减小,这表明增加锚节点的总数有助于提高未知节点的定位精度,这也是图 2(a)中所有算法在锚节点不变时未知节点数量过多会导致  $\epsilon_{network}$  均值增大的原因。图 2(c)为  $\epsilon_{network}$  均值随最大通信半径  $R$  的变化曲线,可以发现,DV-Hop、ABC 和 RABC 算法的曲线变化规律相同。当最大通信半径增大到 35 m 时,ABC 算法的收敛速度明显加快;当通信半径大于 35 m 时,ABC 算法的定位精度优于 DV-Hop 算法。图 3 为根据表 1 的参数得到  $C=100$  时,50 次独立仿真下  $\epsilon_{network}$  均值随节点总数  $X_{SN}$ 、锚节点总数  $X_{AN}$  和最大通信半径  $R$  的变化曲线。

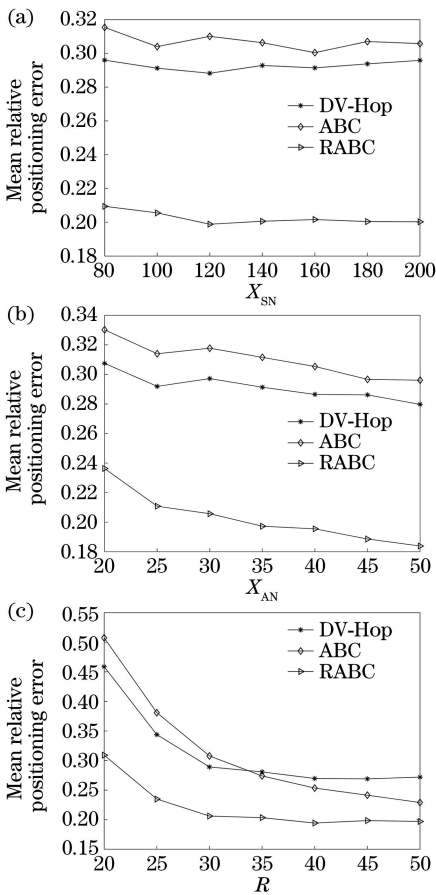


图 2 平均  $\epsilon_{network}$  随不同参数的变化曲线。  
(a)  $X_{SN}$ ; (b)  $X_{AN}$ ; (c)  $R$  ( $C=10$ )

Fig. 2 Average  $\epsilon_{network}$  change curves with different parameters. (a)  $X_{SN}$ ; (b)  $X_{AN}$ ; (c)  $R$  ( $C=10$ )

从实验结果可以看出,在最大迭代次数为 10 时,相比 ABC 算法,RABC 算法的误差降低了 4%~10%,但此时 ABC 算法没有收敛<sup>[15]</sup>。在最大迭代次数为 100 次时,相比 DV-Hop 算法与 ABC 算法,RABC 算法的误差分别降低了 11%和 6%左右。综上所述,增加迭代次数可加快 ABC 算法的收

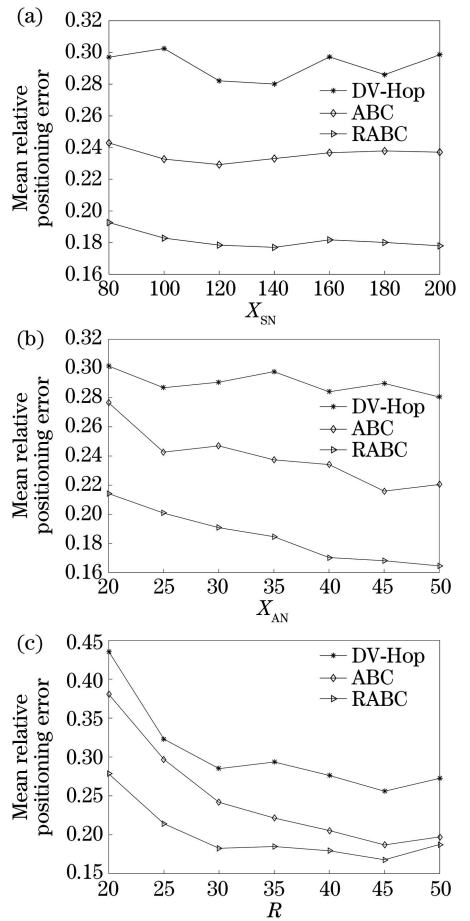


图 3 平均  $\epsilon_{network}$  随不同参数的变化曲线。

(a)  $X_{SN}$ ; (b)  $X_{AN}$ ; (c)  $R$  ( $C=100$ )

Fig. 3 Average  $\epsilon_{network}$  change curve with different parameters. (a)  $X_{SN}$ ; (b)  $X_{AN}$ ; (c)  $R$  ( $C=100$ )

敛速度,得到的寻优结果也更接近真实值。且 RABC 算法的定位精度始终优于 DV-Hop 算法和 ABC 算法,表明了本算法的有效性。

## 4 结 论

针对 WSN 节点定位中 DV-Hop 算法定位误差较大的问题,在多边定位阶段引入数学优化模型,提出了一种基于 ABC 算法的改进优化策略。仿真结果表明,区域限定策略可以有效提高 ABC 算法在定位执行过程中的收敛速度;提高扰动频率也可以加快后期收敛和寻优结果的多样性。与原始 ABC 算法相比,在同等计算量下本算法的定位精度有显著提高。但由于条件限制,本算法只在最大迭代次数为 10 和 100 时对结果进行了对比分析,未能分析算法的最优最大迭代次数。其次,每次仿真实验的节点分布不同,个别节点使用传统算法的性能可能优于改进的 ABC 算法,如何均衡区分这部分节点以获

得性价比更高的定位性能是未来研究的方向。

### 参 考 文 献

- [1] Yang J. TOA based joint time synchronization and localization for WSN [J]. Journal of Nanjing University of Posts and Telecommunications (Natural Science), 2016, 36(4): 77-82.  
杨健. 基于 TOA 的无线传感器网络时间同步与定位联合算法[J]. 南京邮电大学学报(自然科学版), 2016, 36(4): 77-82.
- [2] Zhang J, Wang G. Enhanced semidefinite relaxation method for TDOA/FDOA-based source localization in wireless sensor networks [J]. Chinese Journal of Sensors and Actuators, 2018, 31(12): 1912-1918.  
张杰, 王刚. 无线传感器网络中基于 TDOA/FDOA 的增强半正定松弛定位算法研究[J]. 传感技术学报, 2018, 31(12): 1912-1918.
- [3] Yang C C, Chen J H, Liu D L, et al. Indoor joint localization algorithm based on time and angle of arrival[J]. Laser & Optoelectronics Progress, 2019, 56(22): 222802.  
杨超超, 陈建辉, 刘德亮, 等. 基于到达时间和到达角度的室内联合定位算法[J]. 激光与光电子学进展, 2019, 56(22): 222802.
- [4] Ye Z W, Ye H Y, Nie X Y, et al. High-accuracy visible light positioning method based on received signal strength indicator [J]. Chinese Journal of Lasers, 2018, 45(3): 0306002.  
叶子蔚, 叶会英, 聂翔宇, 等. 基于接收信号强度检测的高精度可见光定位方法[J]. 中国激光, 2018, 45(3): 0306002.
- [5] Dai T H, Li H. Node localization of wireless sensor networks based on improved APIT algorithm [J]. Transducer and Microsystem Technologies, 2016, 35(1): 135-138.  
戴天虹, 李昊. 基于改进 APIT 算法的无线传感器网络节点定位[J]. 传感器与微系统, 2016, 35(1): 135-138.
- [6] Nagpal R, Shrobe H, Bachrach J. Organizing a global coordinate system from local information on an Ad Hoc sensor network[J]. Information Processing in Sensor Networks, 2003, 2634: 333-348.
- [7] Kaur A, Kumar P, Gupta G P. A weighted centroid localization algorithm for randomly deployed wireless sensor networks [J]. Journal of King Saud University-Computer and Information Sciences, 2019, 31(1): 82-91.
- [8] Shang Y, Ruml W. Improved MDS-based localization[C]// INFOCOM 2004. Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies, March 7-11, 2004, Hong Kong, China. New York: IEEE, 2004, 4: 2640-2651.
- [9] Niculescu D, Nath B. DV based positioning in Ad Hoc networks [J]. Telecommunication Systems, 2003, 22(1/2/3/4): 267-280.
- [10] Lin F D, Chen J P, Ding K, et al. Optimization of DV-Hop algorithm based on genetic algorithm and binary ant colony algorithm [J]. Instrument Technique and Sensor, 2019(1): 86-90, 96.  
林凤德, 陈佳品, 丁凯, 等. 基于遗传算法和二进制蚁群算法的 DV-Hop 定位算法的优化[J]. 仪表技术与传感器, 2019(1): 86-90, 96.
- [11] Singh S P, Sharma S C. Implementation of a PSO based improved localization algorithm for wireless sensor networks [J]. IETE Journal of Research, 2019, 65(4): 502-514.
- [12] Goyal S, Patterh M S. Wireless sensor network localization based on cuckoo search algorithm [J]. Wireless Personal Communications, 2014, 79(1): 223-234.
- [13] Bansal J C, Gopal A, Nagar A K. Analysing convergence, consistency, and trajectory of artificial bee colony algorithm [J]. IEEE Access, 2018, 6: 73593-73602.
- [14] Tsai H C. Artificial bee colony directive for continuous optimization [J]. Applied Soft Computing, 2020, 87: 105982.
- [15] Zhu Z W, Song L P. Research on improved WSN positioning based on hybrid artificial bee colony and particle swarm [J]. Modern Electronics Technique, 2019, 42(23): 6-10, 16.  
朱正伟, 宋丽萍. 基于混合人工蜂群粒子群改进的 WSN 定位研究[J]. 现代电子技术, 2019, 42(23): 6-10, 16.