



专题：工业互联网平台及安全

基于 IPSO-MC 融合算法的无线传感器节点定位

李勇燕¹, 吴坚平²

(1. 绍兴文理学院上虞分院, 浙江 绍兴 312300; 2. 浙江大学, 浙江 杭州 310058)

摘要: 针对无线传感器网络 (WSN) 中的节点定位精度不足的问题, 提出一种基于优化粒子算法与膜计算 (improved particle swarm optimization by membrane computing, IPSO-MC) 的融合算法。采用 Kent 映射初始化种群和引入领域粒子提高粒子群全局优化能力, 通过权重因子和非线性极值扰动提高粒子群的局部优化能力, 利用 Levy 飞行机制优化个体位置, 借助膜计算的进化规则最终获得粒子群算法的最优解。仿真实验表明, 与鸡群算法、改进的粒子群算法和膜计算的定位算法相比, 本文算法在参考节点比例指标对比中分别提高 3.24%、5.12% 和 8.15%、在节点个数指标对比中分别提高 2.26%、7.82% 和 9.81% 以及在通信半径指标对比中分别提高 2.15%、5.5% 和 7.5%, 说明该算法在节点定位方面具有良好的效果。

关键词: 无线传感器网络; 节点定位; 粒子群算法; 膜计算

中图分类号: TP391

文献标识码: A

doi: 10.11959/j.issn.1000-0801.2020063

Wireless sensor node localization based on IPSO-MC

LI Yongyan¹, WU Jianping²

1. University of Shaoxing at Shang Yu, Shaoxing 312300, China

2. Zhejiang University, Hangzhou 310058, China

Abstract: To solve the problem of insufficient node positioning accuracy in wireless sensor networks, an algorithm based on improved particle swarm optimization by membrane computing (IPSO-MC) was proposed. Kent mapping was used to initialize the population and domain particles were introduced to improve the global optimization of the particle swarm. The weight factor and nonlinear extreme value perturbation were used to improve the local optimization ability of the particle swarm, and the Levy flight mechanism was used to optimize the individual position. Finally, the optimal solution of the particle swarm algorithm was obtained by the evolutionary rules of the membrane computing. Simulation experiments show that compared with the chicken flock algorithm, the improved particle swarm algorithm and the membrane computing, the proposed algorithm improves 3.24%, 5.12% and 8.15% in the comparison of reference node ratio indicators, and the increase in the number of nodes indicators by 2.26%, 7.82% and 9.81%, and the comparison of communication radius indicators increased by 2.15%, 5.5% and 7.5%, respectively. This indicates that the algorithm has a good effect in node localization.

Key words: wireless sensor network, node location, particle swarm algorithm, membrane computing

收稿日期: 2020-02-09; 修回日期: 2020-03-11

基金项目: 浙江省重点研发计划基金资助项目 (No.2018C03052)

Foundation Item: The Key Research and Development Projects of Zhejiang Province of China (No.2018C03052)



1 引言

无线传感器网络 (wireless sensor network, WSN) 一直以来都是当前信息网络技术研究的热点, 它具有成本低、体积小、传输距离远等优点, 广泛地应用在工业、军事、医疗监测等方面^[1]。作为无线传感器网络中的关键技术, 节点定位一方面能够帮助人们实时获得所需的位置信息, 另一方面能够确保无线网络中的各种路由设备以及相关的覆盖技术能够正常地运行。因此研究无线传感器网络中的节点定位具有非常重要的意义。

针对如何提高节点定位精度的问题, 国内外学者提出了不同的改进措施。参考文献[2]提出一种基于距离估计的改进蒙特卡罗定位算法, 虽能保证更高的定位精度, 减少无效定位的节点数目, 但提高了算法复杂度。参考文献[3]提出基于测距修正的无迹卡尔曼滤波优化定位算法, 能够有效地降低算法平均误差率, 但会产生额外的能耗。参考文献[4]提出一种带混沌映射的无线传感器网络蝴蝶优化定位算法, 提高了节点定位的准确率和精度, 但也提高了算法的定位时间。参考文献[5]提出一种自适应罚函数优化粒子群的定位算法, 具有更好的定位精度和收敛速度, 但提高了算法复杂度。参考文献[6]提出一种基于果蝇优化算法的无线传感器网络节点定位方法, 具有较好的定位精度, 但增加了运行时间。参考文献[7]提出一种基于差分优化的无线传感器定位算法, 在一定程度上提高节点精度, 会产生额外的能耗。参考文献[8]提出基于极限学习机和粒子群算法融合的定位算法, 节点定位精度得到了明显的改善, 但增加了算法复杂度。参考文献[9]提出采用本地化节点定位方法提升整个无线传感器网络算法的性能, 但缺乏与其他算法效果的对比。参考文献[10]提出一种基于飞蛾算法的无线传感器网络定位方法, 降低了节点定位能耗, 但提高了算法复杂度。参考文献[11]提出基于教学方法的无线传感器节

点的定位研究, 与基于 GA 相比, 粒子群优化 (particle swarm optimization, PSO) 算法的定位效果有了明显的改进, 但提高了算法复杂度。参考文献[12]提出一种新型的用于无线传感器节点定位的算法, 能够减少通信时间和能量消耗, 但增加了对本地硬件的能耗。

从以上研究现状中发现目前大部分无线传感器的节点定位算法还是依靠仿生学算法完成定位, 这说明采用该类型算法得到的定位结果是有效的。

2 定位节点模型构建

无线传感器网络中的节点定位一般分为测距和非测距两种类型, 其中非测距定位以简单易实现、定位精度高等优点被广泛采用。它主要使用最小二乘法计算未知节点坐标, 但定位精度容易受到来自累计误差因素的影响。针对这个问题, 本文构建了一种非测距的定位节点模型, 对可能存在的误差进行分析。以二维平面为例进行研究, 设定 (x, y) 为未知节点坐标, (x_i, y_i) 为参考节点坐标, 未知节点与参考节点之间对应的距离为 d_i , 表达如式 (1):

$$\begin{cases} (x_1 - x)^2 + (y_1 - y)^2 = d_1^2 \\ (x_2 - x)^2 + (y_2 - y)^2 = d_2^2 \\ \vdots \\ (x_n - x)^2 + (y_n - y)^2 = d_n^2 \end{cases} \quad (1)$$

将式 (1) 前 $n-1$ 个方差减去最后一个方程, 得到式 (2):

$$\begin{cases} 2(x_1 - x_n)x + 2(y_1 - y_n)y = x_1^2 - x_n^2 + y_1^2 - y_n^2 - d_1^2 + d_n^2 \\ 2(x_2 - x_n)x + 2(y_2 - y_n)y = x_2^2 - x_n^2 + y_2^2 - y_n^2 - d_2^2 + d_n^2 \\ \vdots \\ 2(x_{n-1} - x_n)x + 2(y_{n-1} - y_n)y = x_{n-1}^2 - x_n^2 + y_{n-1}^2 - y_n^2 - d_{n-1}^2 + d_n^2 \end{cases} \quad (2)$$

使用线性方程组表示 $AL = b$, 得:

$$A = \begin{bmatrix} 2(x_1 - x_n) & 2(y_1 - y_n) \\ \vdots & \vdots \\ 2(x_{n-1} - x_n) & 2(y_{n-1} - y_n) \end{bmatrix} \quad (3)$$

$$L = \begin{bmatrix} x \\ y \end{bmatrix} \quad (4)$$

$$b = \begin{bmatrix} x_1^2 - x_n^2 + y_1^2 - y_n^2 + d_1^2 - d_n^2 \\ \vdots \\ x_{n-1}^2 - x_n^2 + y_{n-1}^2 - y_n^2 + d_{n-1}^2 - d_n^2 \end{bmatrix} \quad (5)$$

在实际环境过程中肯定会存在一些误差因素 ε ，并且这些因素给节点定位带来很大的干扰，因此必须考虑，将以上的线性方程组修正为 $AL + \varepsilon = b$ ，通过最小二乘法：

$$L = (A^T A)^{-1} A^T b \quad (6)$$

从式 (6) 中发现参数 b 对 L 的求解具有非常重要的影响，而影响参数 b 的因素是 d_n ，当 d_n 值比较大时，用于计算具体节点坐标的最小二乘法无法有效应用。针对这个问题，将误差较大环境下的定位问题转换为函数约束的优化问题，节点间的距离测量误差为：

$$|r_i - d_i| < \varepsilon_i \quad (7)$$

其中， ε_i 为节点之间的测距误差，而 r_i 是传感器网络中参考节点与未知节点的实际距离， $i \in [1, n]$ ，因此优化问题转换为式 (8)：

$$\begin{cases} d_1^2 - \varepsilon_1^2 \leq (x - x_1)^2 + (y - y_1)^2 \leq d_1^2 + \varepsilon_1^2 \\ d_2^2 - \varepsilon_2^2 \leq (x - x_2)^2 + (y - y_2)^2 \leq d_2^2 + \varepsilon_2^2 \\ \vdots \\ d_n^2 - \varepsilon_n^2 \leq (x - x_n)^2 + (y - y_n)^2 \leq d_n^2 + \varepsilon_n^2 \end{cases} \quad (8)$$

设 $f_i = \sum_{i=1}^n \sqrt{(x - x_i)^2 + (y - y_i)^2 - d_i^2}$ ， f_i 表示未知节点与参考节点之间的测量误差，因此目标函数为：

$$f(x, y) = \sum_{i=1}^n \sqrt{(x - x_i)^2 + (y - y_i)^2 - d_i^2} \quad (9)$$

其中， $f(x, y)$ 的值越小，表示求解的坐标值与实际值就越接近。因此可以将无线传感器网络中的节点定位问题转换为多维约束优化方程，通过仿生学算法寻找最优特性，求解所获得的节点具体位置逐步迭代到最小差值，直到获得最佳值。

3 相关算法

3.1 膜计算

1998 年，欧洲科学院院士 Paum G^[14] 受到生物细胞的启发，提出了膜计算 (membrane computing, MC) 的概念，该模型是从细胞处理化学物质的机理中抽象出来的一种计算模型。

一个膜模型由以下元素组成：

$$F = (V, T, \mu, w_1, \dots, w_m, (R_1, \rho_1) \cdots (R_m, \rho_m)) \quad (10)$$

其中， V 是字母表， T 是输出字母表， μ 表示膜结构， w_m 表示膜内对象， R_m 表示进化规则有限集合， ρ_m 为规则执行顺序。该算法优势在于膜内的区域都是单独运行的，膜内交流仅仅发生在相邻区域，膜间计算能够适合各种智能优化算法并能够有效地避免陷入局部最优。

3.2 粒子群算法

Eberhart 和 Kennedy 在 1995 年根据生物界中的鸟类在觅食过程中的行为和特点提出了粒子群优化 (particle swarm optimization, PSO) 算法，它是一种智能种群的随机优化算法。它能够有效地获得全局最优解。该算法具有实现简单、收敛速度快、参数少、效率高等优点。使用 D 维向量描述粒子 i 的信息，位置描述为 $x_i = (x_{i1}, x_{i2}, \dots, x_{iD})^T$ ，速度描述为 $v_i = (v_{i1}, v_{i2}, \dots, v_{iD})^T$ ，因此速度和位置的更新方程为：

$$\begin{aligned} v_{id}^{k+1} &= \omega \times v_{id}^k + c_1 \times r_1 \times (p_{bestid}^k - x_{id}^k) + \\ &c_2 \times r_2 \times (g_{bestid}^k - x_{id}^k) \end{aligned} \quad (11)$$

$$x_{id}^{k+1} = x_{id}^k + v_{id}^{k+1} \quad (12)$$

其中， p_{best} 代表个体极值点， g_{best} 代表全局极值点， ω 为权重因子， c_1 和 c_2 表示学习因子， r_1 和 r_2 表示 (0,1) 之间的随机数。

4 基于 IPSO-MC 的无线传感节点定位设计

4.1 改进的粒子群算法——IPSO

(1) 种群初始化和领域粒子

本文采用 Kent 映射方法对粒子群算法种群进



行初始化,在一定程度上能够避免盲目搜索而导致的最优解可能出现的偏差问题,从而有效保持种群多样性。Kent 映射如式 (11) 所示。同时为了提高粒子群的全局优化能力,引入领域粒子概念使得粒子在进化过程中收到来自其他粒子对其的影响。领域粒子设定如式 (13),更新后的速度公式如式 (14):

$$x_{n+1} = \begin{cases} x_n / \delta & , 0 < x_n \leq \delta \\ (1 - x_n) / (1 - \delta) & , \delta < x_n \leq 1 \end{cases} \quad (13)$$

$$v_{id}^{k+1} = \omega \times v_{id}^k + c_1 \times r_1 \times (p_{bestid}^k - x_{id}^k) + c_2 \times r_2 \times (g_{bestid}^k - x_{id}^k) + c_3 \times r_3 \times (n_{bestid}^k - x_{id}^k) \quad (14)$$

其中, ω 为权重因子, c_1 、 c_2 和 c_3 为学习因子, n_{best} 表示领域粒子,通过式 (15) 进行更新:

$$n_{bestk} = \min \left| \frac{\sum_{i=1}^n f(i)}{n} - f(k) \right| \quad (15)$$

(2) 权重因子和非线性极值扰动

在基本的粒子群算法中,粒子速度公式中的权重因子一般设置为某个固定的值,显然这样的设置具有一定的不合理性,也不符合粒子在迭代过程中出现的状态变化的情况。本文将权重因子的设置与迭代次数进行综合考虑,设置权重因子变化的范围,使得权重因子的数值随着迭代次数的不断增加而实现动态的改变,设置如式 (16) 所示。此外粒子在迭代过程中容易陷入局部最优而导致该粒子周围聚集着大量的其他粒子,降低了整个算法的个体寻优的能力。此外,本文引入极值扰动因子的概念,有效地保证粒子具有“活力”,避免算法在局部获得最优解之后陷入停滞,促使粒子能够搜索到更多的未知区间,提高搜索能力。扰动算子设置如式 (17):

$$\omega = \omega_{\min} + (\omega_{\max} - \omega_{\min}) \times \frac{t}{t_{\max}} \quad (16)$$

$$\text{dis} = \text{dis}_{\max} - \frac{(\text{dis}_{\max} - \text{dis}_{\min})}{\sum_{t=1}^{\max} t} (t_{\max} - t) \quad (17)$$

其中, $\omega_{\max}$ 为权重最大值, $\omega_{\min}$ 为权重最小值,设

置 $t_{\max}$ 为最大迭代次数, t 为当前迭代次数。dis 为扰动算子, $\text{dis}_{\max}$ 和 $\text{dis}_{\min}$ 分别表示扰动算子的范围,因此粒子的速度更新为:

$$v_{id}^{k+1} = \omega \times v_{id}^k + c_1 \times r_1 \times \text{dis} \times (p_{bestid}^k - x_{id}^k) + c_2 \times r_2 \times (g_{bestid}^k - x_{id}^k) + c_3 \times r_3 \times (n_{bestid}^k - x_{id}^k) \quad (18)$$

(3) 个体位置优化

Edwards^[13]对特定动物的活动特点进行研究得出了符合 Levy 飞行特征的结论,即这种飞行特征能够满足在小范围内的局部搜索要求,又能满足大范围内的全局搜索要求,有效地平衡局部和全局之间的关系。Levy 飞行步长变化的分布密度函数可以近似表示为:

$$\text{Levy}(s) \sim |s|^{-1-\beta}, 0 < \beta \leq 2 \quad (19)$$

$$s = \mu / |v|^{1/\beta} \quad (20)$$

$$\mu \sim N(0, \sigma_\mu^2), v \sim N(0, \sigma_v^2) \quad (21)$$

其中:

$$\sigma_\mu = \left\{ \frac{\Gamma(1+\beta) \sin(\pi\beta/2)}{\Gamma[(1+\beta)/2] \beta / 2^{(\beta-1)/2}} \right\}^{1/\beta}, \sigma_v = 1 \quad (22)$$

粒子群算法与其他种群智能算法一样也存在类似的 Levy 飞行特征,容易陷入局部最优,因此觅食个体更新行为中引入基于 Levy 飞行机制,式 (23) 如下:

$$x_{id}^{k+1} = x_{id}^k \times a(t) \times \text{sign}(\text{rand}) \oplus s + v_{id}^{k+1} \quad (23)$$

其中, rand 为 $[-1,1]$ 之间的随机数; $\text{sign}(\text{rand})$ 为 Levy 飞行方向,如式 (24) 所示; $a(t)$ 为尺度系数,如式 (25) 所示。

$$\text{sign}(\text{rand}) = \begin{cases} 1, & \text{rand} \geq 0 \\ -1, & \text{rand} < 0 \end{cases}, -1 \leq \text{rand} \leq 1 \quad (24)$$

$$a(t) = a_{\text{init}} \cdot \exp(-t / (f_{\text{obj}}^{\text{best}}(x_i') - f_{\text{obj}}(x_i'))) \quad (25)$$

其中, $f_{\text{obj}}(x_i')$ 和 $f_{\text{obj}}^{\text{best}}(x_i')$ 分别表示当前个体适应度和最优适应度值, t 为当前迭代次数, a_{init} 为初始尺度系数。

4.2 融合的 IPSO-MC 算法

膜计算中的主膜只有一个,而辅助膜具有多

个, 主膜的作用在于收集最优的粒子以及进行粒子群算法的局部优化, 辅助膜进行全局优化。膜计算具体的过程是首先通过编码规则将粒子个体随机分配到不同的膜中, 其次通过进化规则, 将辅助膜中获得的全局优化的结果送入主膜, 不断地进行迭代直到达到最大迭代次数, 最终获得整个膜系统的最优粒子。

(1) 编码规则

本文采用如图 1 所示的膜结构, 初始化膜的结构表达为 $[1, [6] [6] [3] [4] [5] [5] [3]] 1$ 。该膜结构中包含表层膜 1, 主膜为 6, 辅助膜为 3、4、5, 其中表层膜主要收集来自主膜抛弃的适应度差的粒子。将粒子随机分配到膜结构中, 保证主膜和辅助膜中至少包含一个粒子 (除表层膜以外), 其中, $q_i (i=1, 2, 3, \dots, n)$ 表示辅助膜中的粒子个体。设搜索空间的维数为 d , 粒子的位置 $x'_i = (x'_{i1}, x'_{i2}, \dots, x'_{id})$ 将被看作膜系统处理的对象, 而粒子的速度表示当前个体的状态, 因此种群中的所有粒子对应的解集合就是膜系统的对象集合, 在整个种群的搜索空间内, 多重集合表示为 $w_i = (x_{i1}, x_{i2}, \dots, x_{im})$, 其中 i 表示膜内的粒子数, $x'_{i1}, x'_{i2}, \dots, x'_{id}$ 代表第 i 个膜内的算法迭代中第 t 代中的每一个粒子对应的解。

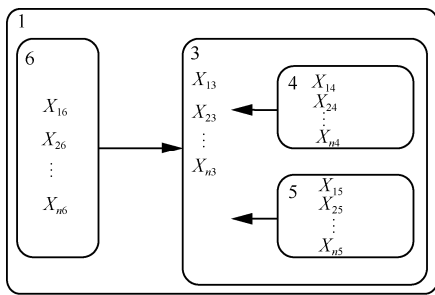


图 1 基于改进的粒子群算法的膜结构

(2) 进化规则

为了更好地提高算法性能, 找到更好的最优解。在辅助膜中, 对粒子群算法进行种群初始化以及引入领域粒子, 增强种群的多样性以获得更好的全局最优解; 同时主膜中对粒子群算法采

用权重因子和非线性极值扰动提高算法的局部搜索性能和收敛速度以获得局部最优解。主膜与辅助膜之间通过信息因子相互交流, 及时比较两种膜中产生的最优解, 最终产生整体算法的最优解。辅助膜内的优质粒子被当作信息传递给主膜的进化规则是在粒子按照式 (18) 和式 (23) 更新速度和位置之后, 根据适应度值由高到低的排序结果, 将适应度值高的粒子个体送入主膜中不断地进行迭代, 并不断地更新主膜的最优粒子个体。设维度为 $d (d < n)$ 、辅助膜为 L , 主膜到辅助膜的操作进化规则如下:

$$\begin{aligned} X'_{1L} \cdots X'_{nL} \cdots X'_{dL} &\rightarrow \\ X'_{1L} \cdots X'_{nL} \cdots X'_{dL} (X'_{1L} \cdots X'_{nL} \cdots X'_{dL}) \end{aligned} \quad (26)$$

4.3 算法流程

步骤 1 建立节点定位模型, 使用 IPSO-MC 算法解决该模型转化成寻优问题; 将每一个节点对应粒子群算法中的一个粒子, 定义粒子群算法和膜计算所需要的相关参数, 定义膜计算的进化规则, 将粒子划分到膜结构中, 保证辅助膜中都包含一个粒子。

步骤 2 根据主膜和辅助膜中的规则, 初步设定粒子的局部最优解和全局最优解。

步骤 3 在每个辅助膜中根据粒子的适应度值进行排序, 将各自膜中适应度值高的粒子送入主膜中。

步骤 4 主膜在每一次迭代中对各个辅助膜中送来的适应度值高的优质粒子进行重新排序, 丢弃劣质粒子给表层膜, 并更新主膜中的粒子适应度值。

步骤 5 判断算法是否达到最大迭代次数, 如果达到则停止迭代, 转步骤 6, 否则转步骤 2。

步骤 6 输出最优粒子即最优节点定位结果。

5 仿真实验与分析

为了进一步说明本文算法在节点定位中的效果, 将本文算法 (IPSO-MC) 与改进的粒子群算



法 (IPSO)、膜计算的定位算法 (MC) 和参考文献[15]的鸡群算法 (CSO) 进行对比。硬件平台为 CPU 为酷睿 i5, 内存为 8 GB DDR3, 硬盘容量为 500 GB, 操作系统为 Window 7, 仿真平台 MATLAB2010。根据无线传感器节点定位的特点, 从参考节点比例、节点密度和通信半径 3 个指标进行对比。设定 PSO 的惯性权重为 0.5, c_1 、 c_2 均为 0.5, ω 设为 0.5, IPSO 算法中的 c_1 、 c_2 为 1, ω 设为 1, a_{init} 为 1, MC 算法中辅助膜为 3 个, IPSO-MC 中的 δ 为 0.4, dis_{max} 和 dis_{min} 分别为 1 和 0.1, ω_{max} 和 ω_{min} 分别为 1 和 0.1, c_1 、 c_2 、 c_3 都设置为 0.5, 辅助膜为 3 个, rand 为 1, a_{init} 为 1。实验对比了参考节点比例、节点密度和通信半径对定位误差的影响。无线传感器网络中的环境相关参数见表 1。

表 1 仿真环境参数

参数名称	参数数值
区域大小	10 m×10 m
总节点数/个	50
参考节点数/个	20
通信半径/m	30
节点分布	随机分布

仿真实验中将误差结果取平均值, 对比 3 种定位算法性能的优劣, 误差平均值计算式如下:

$$E_{\text{average}} = \frac{\sum_{i=1}^N \sqrt{(\bar{x}_i - x_i)^2 + (\bar{y}_i - y_i)^2}}{N \times R} \quad (27)$$

其中, $(\bar{x}_i, \bar{y}_i)$ 为估算坐标, (x_i, y_i) 为实际坐标, N 为未知节点的节点个数, R 为通信半径。

(1) 参考节点比例的影响

图 2 显示了 4 种算法在不同的参考节点比例下的定位误差变化, 从图 2 中发现随着参考节点的比例升高, 4 种算法的定位误差都在逐渐变小, 但 IPSO-MC 算法在定位误差的效果上表现更好, 定位精度相比 CSO 算法、IPSO 算法和 MC 算法分别提高了 3.24%、5.12%和 8.15%。从总体上看,

IPSO-MC 算法在参考节点比例方面能够有效地提高定位精度。

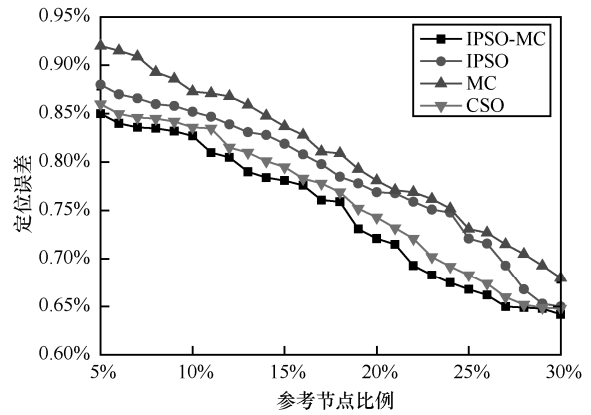


图 2 IPSO-MC 与 3 种典型算法在不同参考节点比例下的定位误差对比

(2) 节点个数的影响

图 3 显示了 4 种算法在不同的节点个数下的定位误差变化, 随着节点数量逐渐变多, 4 种定位算法的误差都在不同程度地减少, IPSO-MC 算法的定位精度相比 CSO 算法、IPSO 算法和 MC 算法分别提高了 2.26%、7.82%和 9.81%。尤其是在节点个数在 20~30 时, IPSO-MC 算法误差下降得最为明显。当节点总数增加到 45 以后, 定位误差并没有大幅度降低, 曲线趋于稳定, 究其原因主要是节点个数加大, 增加了节点之间的相互通信, 导致损失了更多的能量。当待测区域节点数量较小时, IPSO-MC 算法具有明显的优点和良好的稳定性, 因此在节点密度方面, IPSO-MC 算法能够有效地提高定位精度。

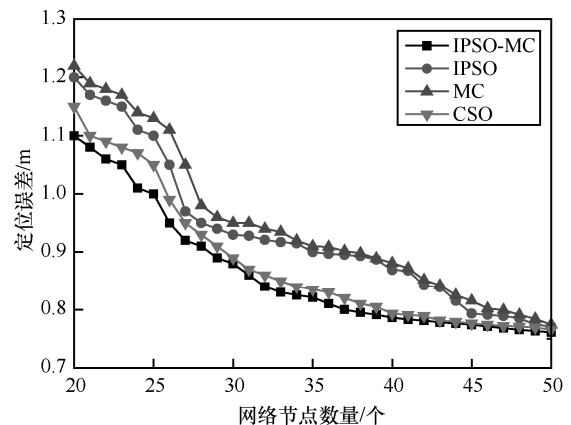


图 3 IPSO-MC 与 3 种典型算法在不同的节点个数下的定位误差对比

(3) 通信半径的影响

图4显示了4种算法在不同的通信半径下的定位误差变化,随着通信半径的逐渐增大,IPSO-MC算法的定位精度相比CSO算法、IPSO算法和MC算法分别提高了2.15%、5.5%和7.5%,这说明通信半径对定位误差的影响较大,当通信半径较小时,锚节点的数量较小,定位效果不理想。但是IPSO-MC算法相比IPSO算法和MC算法具有更高的精度,这说明通信半径的大小对IPSO-MC算法的影响较小。

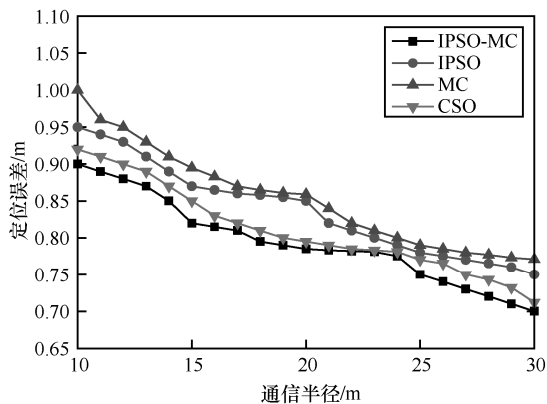


图4 IPSO-MC与3种典型算法在不同的通信半径下的定位误差对比

6 结束语

针对无线传感器网络中节点定位误差大的缺点,本文提出了一种IPSO-MC融合算法,该算法采用使用Kent映射初始化种群和引入领域粒子提高粒子群全局优化能力,使用权重因子和非线性极值扰动提高粒子群的局部优化能力,使用Levy优化个体位置,并借助膜计算的进化规则最终获得粒子群算法的最优解。仿真实验中IPSO-MC算法相比于CSO算法、IPSO算法、MC算法能够有效地提高节点定位精度。

下一步将考虑在三维空间下的节点定位的效果,使得该算法的应用更加广泛。

参考文献:

[1] PARULPREET S, ARUN K, ANI K, et al. Computational intel-

ligence techniques for localization in static and dynamic wireless sensor networks-a review[M]//Computational Intelligence in Sensor Networks. Heidelberg: Berlin Springer, 2019: 25-54.

[2] 王灵娇, 梁雅媚, 郭华. 基于距离估计的无线传感器网络移动节点定位研究[J]. 云南大学学报(自然科学版), 2019, 41(3): 476-483.

WANG L J, LIANG Y M, GUO H. Study on mobile nodes location based on distance estimation for wireless sensor network[J]. Journal of Yunnan University (Natural Sciences), 2019, 41(3): 476-483.

[3] 田丽芳, 徐慧娟. 无线传感器网络中基于测距修正的定位算法[J]. 传感技术学报, 2017, 30(10): 1572-1577.

TIAN L F, XU H J. Localization algorithm based on modifying average hop distance in wireless sensor networks[J]. Journal of Transduction Technology, 2017, 30(10): 1572-1577.

[4] 李田来, 刘方爱. 带混沌映射的WSN蝴蝶优化定位算法[J]. 计算机工程与设计, 2019, 40(6): 1729-1733.

LI T L, LIU F A. Butterfly optimization localization algorithm with chaotic map in wireless sensor networks[J]. Computer Engineering and Design, 2019, 40(6): 1729-1733.

[5] 刘宏, 韩亚波, 张时斌. 基于自适应罚函数优化粒子群的WSN定位算法[J]. 传感技术学报, 2018, 31(8): 1253-1257, 1265.

LIU H, HAN Y B, ZHANG S B. WSNs Localization algorithm based on self-adaptive penalty function optimization particle swarm optimization[J]. Journal of Transduction Technology, 2018, 31(8): 1253-1257, 1265.

[6] 张健, 郭星, 李炜. 基于果蝇优化算法的WSN节点定位研究[J]. 微电子学与计算机, 2018, 35(4): 89-92.

ZHANG J, GUO X, LI W. Research of WSN node localization based on fruit fly optimization algorithm[J]. Microelectronics & Computer, 2018, 35(4): 89-92.

[7] 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.

[8] PHOEMPHON S, SO-IN C, NIYATO D T. A hybrid model using fuzzy logic and an extreme learning machine with vector particle swarm optimization for wireless sensor network localization[J]. Applied Soft Computing, 2018(65): 101-120.

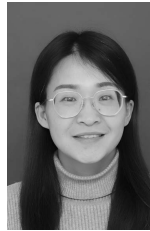
[9] CHEIKHROUHO O, BHATTI G M, ALROOBAEA R, et al. A hybrid DV-hop algorithm using RSSI for localization in large-scale wireless sensor networks[J]. Sensors, 2018, 18(5): 1469-1482.

[10] STRUMBERGER I, TUBA E, BACANIN N, et al. Wireless sensor network localization problem by hybridized moth search algorithm[C]//Proceeding of 2018 14th International Wireless Communications & Mobile Computing Conference (IWCMC). Piscataway: IEEE Press, 2018: 316-321.

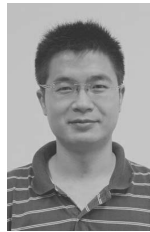


- [11] SHARMA G. Novel DV-hop localization algorithm using teaching learning based optimization for wireless sensor networks[J]. Telecommunication Systems, 2018, 67(2): 163-178.
- [12] KUMAR S, LOBIYAO D K. Novel DV-hop localization algorithm for wireless sensor networks[J]. Telecommunication Systems, 2017, 64(3): 509-524.
- [13] EDWARDS A M, PHILLIPS R A, WATKING N W, et al. Revisiting levy flight search patterns of wandering albatrosses, bumblebees and deer[J]. Nature, 2007(449): 1044-1048.
- [14] GH P. Computing with membranes[J]. Journal of Computer and System Science, 2000, 61(1): 108-143.
- [15] 李鹏, 陈桂芬, 胡文韬. 基于鸡群算法的无线传感器网络定位研究[J]. 传感技术学报, 2019, 32(6): 866-871, 891.
- LI P, CHEN G F, HU W T. Research on wireless sensor network location based on chicken swarm optimization[J]. Journal of Transduction Technology, 2019, 32(6): 866-871, 891.

[作者简介]



李勇燕（1981—），女，绍兴文理学院上虞分院讲师，主要研究方向为无线传感器。



吴坚平（1983—），男，浙江大学博士生，主要研究方向为复杂异构数据融合。