

基于改进果蝇算法的无线传感网络布局研究

吴良超¹, 郭 星^{1,2}

(1 安徽大学 计算智能与信号处理重点实验室, 安徽 合肥 230039;

2 安徽大学 计算机科学与技术学院, 安徽 合肥 230601)

摘 要: 提出一种改进的果蝇算法, 并将其应用于 WSN 问题. 果蝇算法具有参数少、结构简单、全局搜索能力强、收敛速度快等特点, 将改进的果蝇算法用于 WSN 节点优化部署, 可以快速得到解集, 并且具有很高的覆盖率. 通过仿真实验, 可以看出改进的果蝇算法相比于原始果蝇算法及其他智能算法具有更快的收敛速度和更高的收敛精度, 结果更接近理论值.

关键词: WSN; 果蝇算法; 传感器; 收敛; 覆盖率

中图分类号: TP31

文献标识码: A

文章编号: 1000-7180(2016)12-0152-04

Research on Wireless Sensor Network Layout Based on Improved Fruit Fly Optimization Algorithm

WU Liang-chao¹, GUO Xing^{1,2}

(1 Key Laboratory of ICSP, Ministry of Education, Anhui University, Hefei 230039, China;

2 School of Computer Science and Technology, Anhui University, Hefei 230601, China)

Abstract: This paper presents an improved algorithm for FOA, and apply it to WSN problem. The FOA has the characteristics of less parameters, simple structure, strong global search ability and fast convergence speed, the improved algorithm is used to optimize the WSN nodes deployment, which can quickly get the solution set, and has a high coverage rate. Through simulation experiments, we can see that the improved algorithm has faster convergence speed and higher convergence precision compared to the original FOA and other intelligent algorithms, and the results are more close to the theoretical value.

Key words: WSN; FOA; sensor; convergence; coverage rate

1 引言

WSN 是由一组传感器节点以自组织的方式构成的有线或无线网络, 其目的是协作地感知、采集和处理网络覆盖的地理区域中感知对象的信息, 并发布给观察者. 在某个固定大小的监测区域内, 当传感器数量一定时, 如何快速找到节点部署位置, 使网络覆盖率达到最大, 是目前研究的热门方向.

目前, 已经有粒子群算法^[1-2]、遗传算法等智能算法应用于节点部署的优化, 但这些算法存在结构复杂、收敛速度慢和精度不高等问题. 因此本文提出一种改进的果蝇算法, 并应用于 WSN 节点的部署,

可以在固定大小检测区域内, 快速搜索到最优的节点部署位置, 使网络覆盖率达到最高, 实现快速高效解决部署问题的目标.

2 果蝇算法

果蝇优化算法^[3](Fruit Fly Optimization Algorithm, FOA) 是一种全局优化算法. 由于算法具有参数少、速度快、精度高等特点, 因此广泛用于解决寻优问题. 但是果蝇算法也存在着一些问题, 在算法的优化过程中, 可能会陷入局部最优, 使得算法的收敛速度下降, 收敛精度较低. 本文提出了一种改变步长的果蝇算法(Change the Step of FOA, CSFOA), 通

过改变步长的方式解决这些问题,通过实验结果体现改进算法的优越性。

2.1 基本果蝇算法

果蝇拥有优于其他物种的感官,它可以收集空气中的气味分子搜寻远处的食物,然后聚集同伴朝食物方向并拢。FOA 算法就是模拟果蝇群体的觅食过程而衍生的优化算法^[6],其步骤可以归纳如下。

① 初始化种群大小 Sizepop, 最大迭代次数 Maxgen, 随机初始化果蝇群体位置 X_axis, Y_axis 。

② 果蝇个体随机向各个方向的位置进行搜寻

$$\begin{cases} X_i = X_axis + H_r \\ Y_i = Y_axis + H_r \end{cases} \quad (1)$$

式中, H_r 为在固定区间 $[-H, H]$ 之间的随机数, 即

$$H_r = H \times [2 \times \text{rand}(1, 1) - 1] \quad (2)$$

式中, H 为果蝇每次搜索的固定步长。

③ 计算果蝇个体与原点的距离 d , 再将距离的倒数做为果蝇个体的味道浓度判定值 S_i

$$\begin{cases} d = \sqrt{X_i^2 + Y_i^2} \\ S_i = 1/d \end{cases} \quad (3)$$

④ 将味道浓度判定值 S_i 代入适应度函数 (Fitness function), 求出果蝇个体的味道浓度

$$\text{Smell}_i = \text{Function}(S_i) \quad (4)$$

⑤ 找出果蝇群体中 Smell_i 最佳的果蝇

$$[\text{bestSmell} \text{ bestIndex}] = \min(\text{Smell}_i) \quad (5)$$

⑥ 记录并保留最佳味道浓度值 bestSmell_i 与 X, Y 坐标。

$$\begin{cases} \text{Smellbest} = \text{bestSmell} \\ X_axis = X(\text{bestIndex}) \\ Y_axis = Y(\text{bestIndex}) \end{cases} \quad (6)$$

⑦ 开始迭代, 重复执行 ② ~ ⑤, 并判断最佳味道浓度是否优于前一次迭代得到的味道浓度, 并且当前迭代次数小于最大迭代数 Maxgen, 则执行 ⑥, 否则, 算法结束。

2.2 改进步长果蝇算法

在原始的果蝇算法中, 步骤 ② 果蝇的搜寻步长固定, 即果蝇每次在固定的范围内搜索, 产生随机结果。显然, 在果蝇的群体数一定时, H 的值越大, 果蝇搜索的范围越大, 全局搜索的能力越强, 收敛速度越快, 但收敛精度会降低。 H 的值越小, 搜索的精度会提高, 局部搜索能力变强, 但易快速陷入局部最优^[4-5]。因此, 在运用果蝇算法解决 WSN 问题时, 需要通过改变步长的方式, 使算法既能具有很强的全局搜索能力, 又不会陷入局部最优而无法跳出, 从而降低了收敛速度和收敛精度。

通过实验发现, 将迭代过程分成两个阶段。

(1) 在前一半的迭代过程中, 通过增加果蝇的移动步长使果蝇群体达到快速收敛, 不易陷入局部最优。公式如下:

$$\begin{cases} X_i = X_axis + (\text{rand}() \times 2 - 1) \times \\ \quad (2R \times \frac{m}{\text{Maxgen}} + 1) \\ Y_i = Y_axis + (\text{rand}() \times 2 - 1) \times \\ \quad (2R \times \frac{m}{\text{Maxgen}} + 1) \end{cases} \quad (7)$$

式中, $X_i \in [1, 50], Y_i \in [1, 50], R$ 为传感器有效半径。

(2) 后一半的迭代过程中, 步长在 $[-2, 2]$ 之间跌宕, 既能满足在小范围搜索达到高精度, 又能使算法不易陷入局部最优。公式如下:

$$\begin{cases} X_i = X_axis + (\text{rand}() \times 2 - 1) \times \\ \quad (\sin(m) + 1) \\ Y_i = Y_axis + (\text{rand}() \times 2 - 1) \times \\ \quad (\sin(m) + 1) \end{cases} \quad (8)$$

式中, $X_i \in [1, 50], Y_i \in [1, 50]$ 。

3 WSN 节点模型

在一个 $a \times b$ 的二维区域内, 每个接收点的位置可以表示为 (a_i, b_i) 。在这个二维区域内随机部署 N 个发射器, 已知发射器的有效半径为 R , 则每组发射器的集可以表示为

$$W = (\omega_1, \omega_2, \dots, \omega_i, \dots, \omega_N) \quad (9)$$

接收点 (a_k, b_k) 可以收到 ω_i 发射信号的概率为

$$P_i = \begin{cases} 1, d \leq R \\ 0, d > R \end{cases} \quad (10)$$

式中, d 为传感器到 k 点的距离。该接收点 k 能接收到这组发射器信号的概率为

$$P(a_k, b_k) = 1 - \prod_{i \in N} (1 - P_i) \quad (11)$$

最后, W 发射器对这片区域的信号覆盖率为

$$G_{(W)} = \frac{\sum P(a_k, b_k)}{a \times b} \quad (12)$$

4 改进果蝇算法的应用

将 CSFOA 应用于解决 WSN^[6] 问题的算法步骤如下:

① 随机初始 Sizepop 个果蝇的位置 $(X_axis_{(t)}, Y_axis_{(t)})$, 其中 $t \in [1, \text{Sizepop}]$, 最大迭代数 Maxgen。

② 根据式(10) ~ (12) 计算出 $G_{(W)}$ 的最大值,

并记录果蝇的位置.

[bestSmell bestIndex] = max(G_(w)) (13)

③ 进入第一阶段寻优, 每个果蝇个体根据式(7) 确定搜索步长, 随机向各个方向搜寻.

④ 循环执行步骤 ②, 找出其中覆盖率最大的果蝇, 并记录此果蝇的位置做为此次搜索确定的发射器位置, 也是下次搜索的果蝇初始位置, 所有果蝇向此位置聚集, 保存其覆盖率.

$$\begin{cases} \text{bestG} = \text{bestSemll} \\ X_axis = \text{bestIndex}(X) \\ Y_axis = \text{bestIndex}(Y) \end{cases} \quad (14)$$

⑤ 迭代执行步骤 ③④, 直至满足最大迭代次数 Maxgen/2, 并记录最终结果.

⑥ 进入第二阶段寻优, 此时需要缩小步长, 使果蝇个体在小范围震荡搜索, 提高精确度, 每个果蝇个体根据式(8) 确定搜索步长, 随机向各个方向搜寻.

⑦ 循环执行步骤 ②, 找出其中覆盖率最大的果蝇, 并记录此果蝇的位置作为此次搜索确定的发射器位置, 也是下次搜索的果蝇初始位置, 所有果蝇向此位置聚集, 保存其覆盖率.

$$\begin{cases} \text{bestG} = \text{bestSemll} \\ X_axis = \text{bestIndex}(X) \\ Y_axis = \text{bestIndex}(Y) \end{cases} \quad (15)$$

⑧ 迭代执行步骤⑥⑦, 直至迭代此时达到 Maxgen, 并记录最终结果. 此时得到最大覆盖率和对应的发射器摆放位置.

基于 CSFOA 算法的 WSN 问题工作流程如图 1 所示.

5 仿真实验

5.1 实验参数设置

实验测试平台 Window XP, Matlab7, 机器主频为 3.3 GHz, 内存为 2 GB.

考虑到监测区域大小、传感器数量和传感器有效覆盖半径对测试结果的影响, 实验分为 2 组分别进行实验仿真, 每组实验的果蝇种群数量 Sizpop=50, 迭代次数 Maxgen=200, 每组实验运行 50 次, 最后对实验数据进行平均. 各组的实验参数如表 1 所示.

表 1 分组实验参数

实验组号	实验参数		
	监测区域	传感器数量/个	监测半径/m
1	40 m×40 m	20	4
2	50 m×50 m	25	5

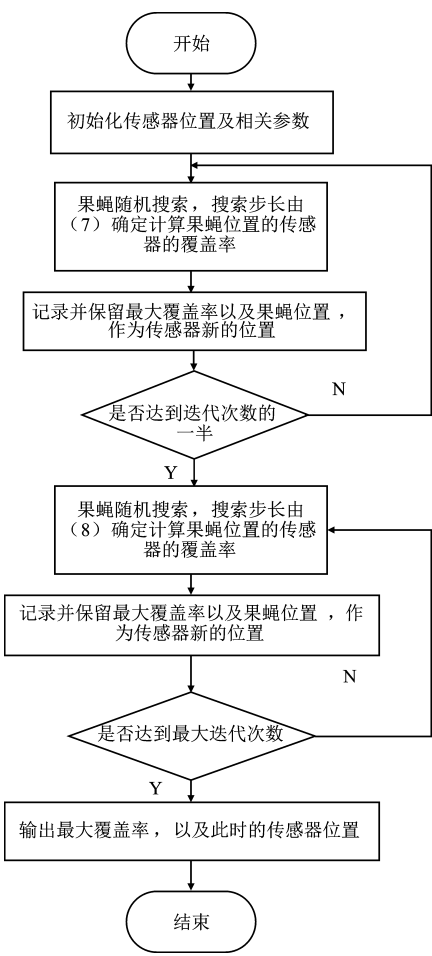


图 1 算法流程图

5.2 实验结果分析

如图 2 所示, 在 40×40 的监测区域内中放置 20 个半径为 4 的传感器, 理论的覆盖率为 62.8%. 在 Sizpop=50,

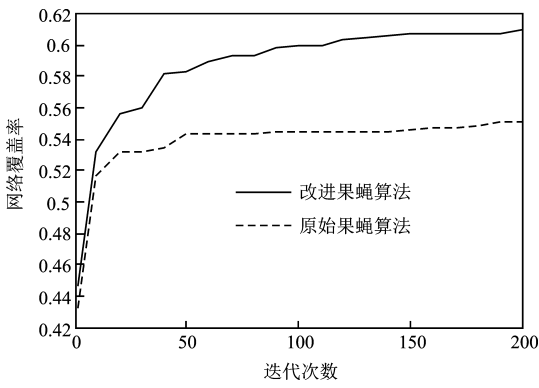


图 2 第 1 组 CSFOA 与原始果蝇算法的覆盖率

Maxgen=200 的条件下, 改进的果蝇算法得到的最大覆盖率为 60.9%, 达到理论值的 96.8%. 而原始果蝇算法得到的最大覆盖率为 55.1%, 只达到了理论值的 87.5%. 如图 3 所示, 监测区域为 50×

50,放置 25 个半径为 5 的传感器,理论覆盖率为 78.5%,CSFOA 在运行 200 次后得到的最大覆盖率为 76.3%,达到理论值的 97%,而原始果蝇算法得到的最大覆盖率为 66.1%,只达到理论值得 84.2%. 由此看出改进的果蝇算法相比于原始果蝇算法具有更高的收敛精度. 从实验结果中可以看出,原始果蝇算法在前期就很快陷入了局部最优的错误解中,从而使收敛精度不高. 改进的果蝇算法由于在前期步长递增,所以在很小的迭代次数时就可以高度收敛,在迭代后半段,通过使步长在小范围跌宕,这样可以使算法避免陷入局部最优的同时,在小范围内搜索,得到更高精度的解.

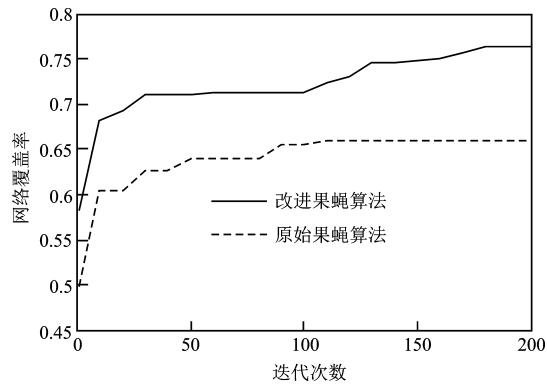


图 3 第 2 组 CSFOA 与原始果蝇算法的覆盖率

如图 4 所示,在 40×40 的监测区域,放置 20 个半径为 4 的传感器,迭代次数 $\text{Maxgen}=200$,通过改变果蝇种群数量,可以看出种群数量越大,前期的收敛越快,但随着迭代次数的增长,数量小和数量大的果蝇群收敛精度逐渐接近,最后得出的结果几乎一致.

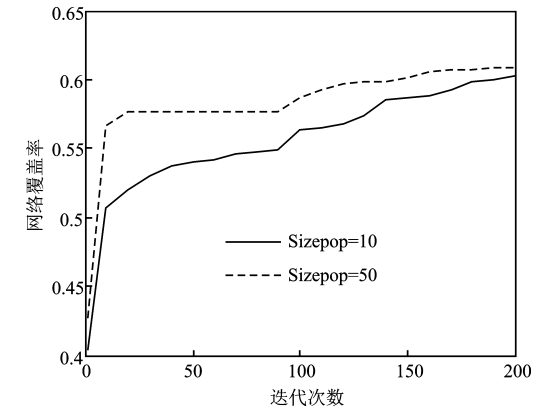


图 4 种群数目不同的覆盖率

如图 5 所示,在 40×40 的监测区域,放置 20 个半径为 4 的传感器,最后找到的传感器在监测区域的部署,覆盖率为 60.9%.

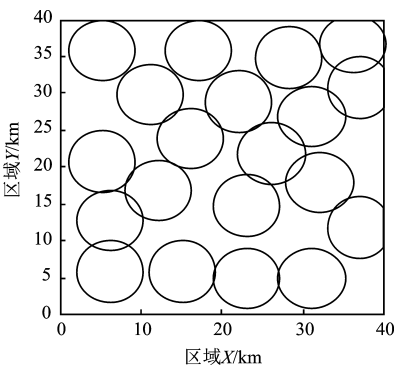


图 5 算法最优节点部署图

为了验证 CSFOA 算法的优越性,在参数相同的情况下(区域面积为 40×40 ,节点半径为 5,迭代次数为 200 次),对果蝇算法、CSFOA、PSO 算法、遗传算法,蛙跳算法进行了仿真实验,仿真结果如表 2 所示(设理论覆盖度为 1).

从表 2 可以看出,在相同条件下,CSFOA 的网络覆盖度最高,可以达到理论值的 96.8%,比原始果蝇算法高了 9.3%,与 PSO 算法比较,提高了 8.4%,比遗传算法高了 6.5%,比蛙跳算法高了 17.2%. 该算法能获得相对于其他算法更高的覆盖率. 由于该算法紧密结合 WSN 模型特点,所以与其他算法相比,优化效果更明显.

表 2 算法覆盖率比较

覆盖算法	覆盖率/%
果蝇算法	87.5
CSFOA	96.8
PSO 算法	88.4
遗传算法	90.3
蛙跳算法	79.6

6 结束语

本文对果蝇算法进行了改进,提出了基于改变步长的果蝇算法,该步长在迭代前期随迭代次数递增而递增,在迭代后期步长在小范围内跌宕变化,使算法在前期具有很快的收敛速度,后期有很好的收敛精度. 并将改进的算法应用于 WSN 问题,找出最优覆盖度下的传感器部署位置. 通过实验比较了相同环境下,几种优化算法的 WSN 问题覆盖率,结果显示 CSFOA 具有最高的覆盖率. 为了更贴近实际情况,接下来的研究主要是考虑改进算法,使其在传

(下转第 160 页)

参考文献:

[1] 朱炼,孙枫,夏芳莉,等. 图像融合研究综述. 传感器与微系统[J]. 2014,33(2): 1000-9787.

[2] 苗启广,叶传奇,汤磊,李伟生. 多传感器图像融合技术与应用[M]. 西安:西安电子科技大学出版社,2014.

[3] Ren X, Malik J. Learning a classification model for segmentation[C]// IEEE International Conference on Computer Vistion, Nice, France:IEEE,2003: 10-17.

[4] Fachao Qin, Jiming Guo, Fengkai Lang. Superpixel segmentation for polarimetric SAR imagery using local iterative clustering[J]. IEEE Geoscience and Remote Sensing Letters,2015,12(1):13-17.

[5] Bai X D, Cao Z G, Wang Y, et al. Image segmentation using modified SLIC and Nystr m based spectral clustering[J]. Optik,2014,125(16):4302-4307.

[6] Jinyu Cong, Benzheng Wei, Yilong Yin, et al. Performance evaluation of simple linear iterative clustering algorithm on medical image processing[J]. Bio-Medical Materials and Engineering,2014,24(6):3231-3238.

[7] Altan-Ulzii, Moonon Jianwen Hu. Multi-focus image fusion based on NSCT and NSST[J]. Sens Imaging, 2015,16(1):1-16.

[8] Yang Lei. Technique for image de-noising based on non-subsampled shearlettran sform and improved intuitionistic fuzzy entropy[J]. Optik, 2015,126:446-453.

[9] Zhang Baohua, Lu Xiaoqi, Pei Haiquan, et al. A fusion algorithm for infrared and visible images based on saliency analysis and non-subsampled Shearlet transform [J]. Infrared Physics & Technology, 2015, 73 (1): 286-297.

[10] Qu Guihong, Zhang Dali, Yan Pingfan. Information measure for performance of image fusion [J]. Electronics Letters, 2002, 38(7): 313-315.

[11] Zhao Hengjun, Shang Zhaowei, Tang Yuanyan, et al. Multi-focus image fusion based on the neighbor distance [J]. Pattern Recognition, 2013, 46 (3): 1002-1011.

作者简介:

苗 娜 女,(1991-),硕士研究生. 研究方向为信息融合. E-mail:591447907@qq. com.

杨 静 男,(1970-),副教授. 研究方向为信息融合.

(上接第 155 页)

传感器半径不同的情况下依然实用.

参考文献:

[1] 宁剑平,王冰,李洪儒,等. 递减步长果蝇优化算法及应用[J]. 深圳大学学报:理工版, 2014, 31 (4): 367-373.

[2] 徐富强,陶有田,吕洪升. 一种改进的果蝇优化算法[J]. 苏州大学学报(自然科学版), 2014, 29 (1): 16-23.

[3] 韩俊英,刘成忠. 自适应调整参数的果蝇优化算法[J]. 计算机工程与应用, 2014,50(7):50-55.

[4] 马超,董玲. 果蝇优化算法(FOA)步长改进及其多元函数最优化方法[J]. 数学学习与研究(教研版),

2013,13(1):90-92.

[5] 宋明智,杨乐. 基于改进自适应 PSO 算法的 WSN 覆盖优化方法[J]. 计算机应用研究, 2013, 30 (11): 3472-3475.

[6] 汪成亮,王强. 基于活动预测和能耗均衡的 WSN 路由算法[J]. 北京航空航天大学学报, 2014, 40(1):10-17.

作者简介:

吴良超 男,(1992-),硕士研究生. 研究方向为智能算法改进及应用. E-mail:wukantian@163. com.

郭 星 男,(1983-),博士,讲师. 研究方向为模式识别与信号处理.