

DOI:10.19651/j.cnki.emt.2210598

# 基于断路器柔性装配的移动机器人路径规划\*

王凌浩<sup>1,2</sup> 舒亮<sup>1,2</sup> 钱祺<sup>1,2</sup>

(1.浙江省低压电器工程技术研究中心 温州 325207; 2.温州大学乐清工业研究院 温州 325606)

**摘要:** 针对移动机器人在断路器柔性装配过程中的路径长度较长、转折点较多等效率问题,提出了一种BAS算法与PSO算法结合的路径优化方法。利用天牛个体更新方式与群体学习相结合,采用自适应步长衰减策略以及动态权重变更策略,实现全局路径规划寻优。为了验证BSO算法的有效性,通过三种不同的测试函数比较性能以及仿真地图进行对比,最后将该算法通过ROS应用到实际地图上。实验结果表明,相比于GA-PSO算法、AIW-PSO算法、BAS算法,路径长度优化率分别提升了7.7%、14.8%与12.5%,转折点优化率分别提升提升了25%、57.1%与40%。综上所述,本文所提出的融合算法能够有效地解决装配过程中的效率问题,提高断路器柔性装配产线效益。

**关键词:** 移动机器人;路径优化;BSO算法;断路器

**中图分类号:** TP242;TM561 **文献标识码:** A **国家标准学科分类代码:** 510.8050

## Path planning of mobile robot based on flexible assembly of circuit breaker

Wang Linghao<sup>1,2</sup> Shu Liang<sup>1,2</sup> Qian Qi<sup>1,2</sup>(1. Low Voltage Apparatus Technology Research Center of Zhejiang, Wenzhou 325207, China;  
2. Yueqing Institute of Industrial Research, Wenzhou University, Wenzhou 325606, China)

**Abstract:** Aiming at the problems of long path length and more turning points of mobile robots in the flexible assembly process of circuit breaker, a path optimization method combining BAS algorithm and PSO algorithm is proposed. By combining the individual update method of beetles with group learning, the adaptive step size decay strategy and the dynamic weight change strategy are adopted to realize the optimization of global path planning. In order to verify the effectiveness of the BSO algorithm, three different test functions are used to compare the performance and the simulation map, and finally the algorithm is applied to the actual map through ROS. Experimental results show that compared with GA-PSO algorithm, AIW-PSO algorithm and BAS algorithm, the optimization efficiency of path length is increased by 7.7%, 14.8% and 12.5%, and the optimization efficiency of the number of turns is increased by 25%, 57.1% and 40%, respectively. In summary, the fusion algorithm proposed in this paper can effectively solve the efficiency problem in the assembly process and improve the efficiency of the flexible assembly line of the circuit breaker.

**Keywords:** mobile robots; path optimization; beetle swarm optimization algorithm; circuit breaker

## 0 引言

断路器是电气行业中的重要元件,随着智能化的生产,一条自动化产线每天能生产几千只断路器。由于断路器内部零件比较复杂,传统方式上料、成品装配方式多为人工运输,造成较大的人力消耗,存在运输时间过长、成本增加、效率低下等问题。

当前移动机器人技术飞速发展,作为一种运输体,移动机器人在更多产业中得到广泛应用<sup>[1]</sup>。随着十二五计划的

开展,越来越多的工厂进行智能制造,移动机器人作为智能制造的重要环节,在智能制造、智能仓储环节<sup>[2]</sup>发挥着重要作用。将移动机器人与断路器生产结合起来,对于提高生产效率有着重要研究意义。移动机器人在断路器柔性装配产线中,需要负责断路器零件的上料以及成品断路器仓储的工作,因此移动机器人在柔性装配车间的工作可以视为移动机器人多约束条件下的全局路径规划问题。除了必要的路径长度以及转折次数问题,还要包括移动机器人与柔性装配单元之间的安全距离以及符合移动机器人的运动学约束<sup>[3]</sup>。

收稿日期:2022-07-07

\* 基金项目:浙江省重点研发计划项目(2021C01046)资助

全局路径规划就是机器人目的地出发,找到一个能安全无碰撞的到达目的地路径的规划<sup>[4]</sup>。移动机器人在识别障碍物后永远不会离开它,找到一条合适的路径才能使机器人安全、有效的避开障碍物,在确保机器人运动过程安全的同时也能提高路径规划的效率。路径规划的方法起源于 20 世纪 60 年代,起初科学家提出了许多技术路线,其中包括人工势场法<sup>[5]</sup>、Dijkstra<sup>[6]</sup>以及图解法,但由于这些方法的计算效率低下、且容易陷入局部最小值的危险,从而演变出一些启发式算法<sup>[7]</sup>,例如现在的仿生学算法<sup>[8]</sup>、神经网络算法<sup>[9]</sup>以及强化学习<sup>[10]</sup>等算法,这类算法在路径规划上具有很好的找到满意解并且有很好的收敛性。全秋娟等<sup>[11]</sup>在粒子群算法 (particle swarm optimization, PSO) 的基础上,提出了一种能够动态调整权重的粒子群算法,能够根据粒子的适应度值动态地调整算法中惯性权重和学习因子的取值,其中惯性权重采用非线性指数递减,有利于平衡算法的全局搜索与局部搜索能力,但是算法较为单一,容易陷入局部最优值。Lin 等<sup>[12]</sup>提出了一种用天牛须算法 (beetle antennae search, BAS) 进行路径规划的方法,该算法能够有效提高机器人路径规划的速度、避障能力和环境适应性,相比于其他算法,可以更好的在静态环境中规划路径。通常采用某种单一智能算法时,移动机器人在搜索路径的过程中往往容易陷入局部最优,造成路径长度冗长、转折点较多、转折角度过大等问题,因此许多研究人员将不同的算法融合起来,利用不同算法之间的优点,提升路径寻优效率。赵辉等<sup>[13]</sup>提出一种智能启发的 BAS 算法和 A\* 算法相结合的算法,简称 BASA\* 算法,在考虑障碍物与机器人间一定安全距离的基础上,减少路径长度。但此方法在复杂环境中寻优效率较低,不利于效益的提高。Wu 等<sup>[14]</sup>提出一种基于 BAS 算法和人工势场算法 (artificial potential field, APF) 的混合算法,称为 BAS-APF 方法。通过建立势场,加快了收敛速度,避免了 APF 容易陷入局部极小值的缺陷,解决了路径规划中路径长度冗长的问题。胡章芳等<sup>[15]</sup>提出一种改进 PSO 优化算法,通过遗传算法 (genetic algorithm, GA) 的选择、交叉、变异加强局部优化,最后通过改进的粒子群更新公式对粒子进行更新,实现机器人路径的优化,但优化后的路径转折点仍较多。

目前采用的路径规划方法虽然能一定程度上的解决路径长度较长、转折点较多的问题,但仍存在效率低下,优化不明显等问题。针对移动机器人在断路器柔性装配中装配效率的问题,本文提出一种启发式搜索算法 BAS 算法<sup>[16]</sup>与智能群体算法 PSO 算法<sup>[17]</sup>结合的算法,简称天牛群算法 (beetle swarm optimization, BSO)。在保持与生产单元一定距离为约束条件的前提下,利用天牛独特的更新方式以及粒子群寻优的高效性,能够有效地减少路径长度与转折次数,并且可以应用到复杂的装配环境当中,计算效率较高。最终,通过 ROS 系统中的 Slam 算法对实际地图进行构图,并在构造好的地图上分别运行本文融合算法与

文献<sup>[11]</sup>、<sup>[12]</sup>、<sup>[15]</sup>中的 3 种算法,通过对 4 种算法下的路径进行对比,从而验证本文所提出的融合算法在解决断路器柔性装配问题上的有效性。

## 1 断路器柔性装配车间搭建

### 1.1 断路器内部构造

断路器内部的零件种类十分复杂,其中包括壳体、磁系统、手柄、灭弧室、手柄、大 U、热系统等,从上料到成品运输需要移动机器人多次参与运输,断路器零件模型如图 1 所示。

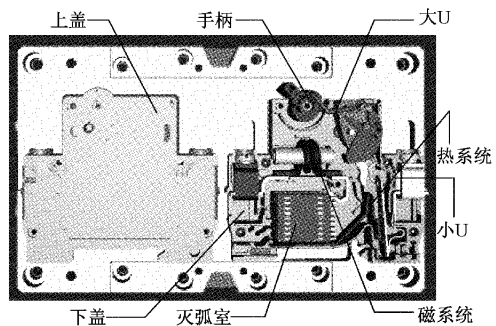


图 1 断路器零件模型

### 1.2 柔性装配平台

柔性装配平台包括一台 OMRON 并联机器人,一台 EPSON 六轴机器人,一台 EPSON 大四轴机器人,一台全向移动机器人,两台 ROKAE 六轴机器人。产线布局以及装配流程如图 2 所示。

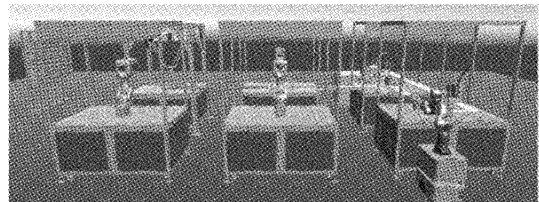


图 2 断路器柔性装配产线

移动机器人将装配所用到的断路器零件夹取到并联机器人工作台上,并联机器人通过视觉对运输来的零件进行分拣,将分拣好的零件通过六轴机器人夹取到载具上,通过大四轴机器人对空壳断路器进行装配后,最后移动机器人将装配好的成品断路器运输到货架上完成装配,装配流程如图 3 所示。

## 2 目标函数的建立

移动机器人的目标函数是评价其路径优化好坏的标准,本文针对断路器装配路径规划,目的为提高装配效率,即在保证安全距离与平滑的条件下,路径代价越小越好,其中约束条件包括安全距离与平滑性,如图 4 所示。

本文对计算路径代价选用的是欧氏距离,假设起始点为  $st_0$ ,目标点设为  $st_{i+1}$ ,路径 ST 的总长度计算公式为:

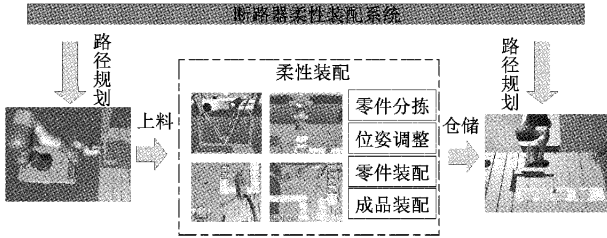


图3 断路器柔性装配流程

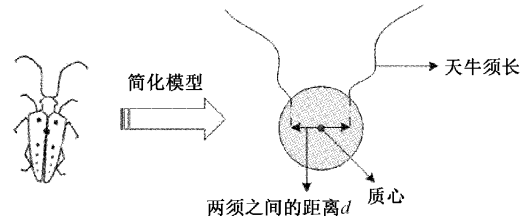


图5 天牛须算法数学模型

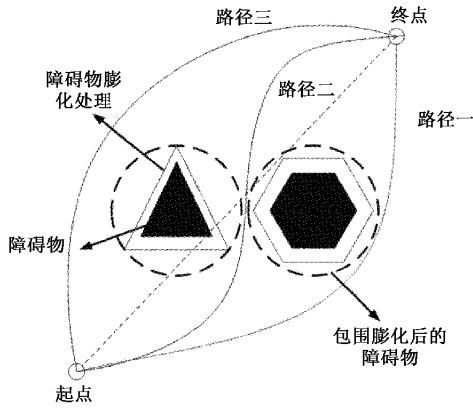


图4 路径规划路线

$$S(ST) = \sum_i^N d(st_i, st_{i+1}) \quad (1)$$

在坐标系  $0-XY$  当中,  $d(st_i, st_{i+1})$  的计算公式可以表示为:

$$d(st_i, st_{i+1}) = \sqrt{(x_{st_{i+1}} - x_{st_i})^2 + (y_{st_{i+1}} - y_{st_i})^2} = \sqrt{\frac{d(st_i, st_{i+1})^2}{n+1} + (y_{st_{i+1}} - y_{st_i})^2} \quad (2)$$

其中,  $n$  表示起始点到目标点线路的段数,  $d(st_i, st_{i+1})$  表示点  $st_i$  点  $st_{i+1}$  间的距离。

### 3 算法优化与融合

#### 3.1 天牛须算法

BAS算法具有较强的局部寻优能力,并能根据两个触角的信息浓度差来调节寻优方向,能够跳出局部最优,提升寻优能力。通过天牛两根触角对食物浓度进行更新,根据天牛自身左右须对食物判断的浓度差来决定天牛下一步行驶的方向,如果左须的浓度大于右须的浓度,那么天牛个体将向偏左的方向行驶,反之,则向偏右侧行驶。

BAS算法的构造比较简单,其中包含天牛须长以及两须之间的距离  $d$ ,以及天牛搜寻食物的步长因子  $s$ ,天牛的步长跟须长决定了一开始的搜索效率,随着食物浓度的越来越大,天牛的步长会根据搜索能力不断变化。天牛的简化模型如图5所示。

通过简化后的数学模型,可以清晰地对天牛的参数进行设置,由于天牛的觅食方向是位置的,所以首先需要对天牛寻路的方向做归一化处理,公式如下:

$$\vec{b} = \frac{rands(n,1)}{|rands(n,1)|} \quad (3)$$

其中,  $n$  代表维度,  $rands$  代表随机函数,一般来说取值范围为  $[0,1]$ 。天牛的左右须长表示如下:

$$\begin{aligned} x_l &= x + d\vec{b} \\ x_r &= x - d\vec{b} \end{aligned} \quad (4)$$

其中,  $x$  是天牛的质心坐标,  $d$  是天牛两须之间的距离。

左右须参数确定后,通过比较左右须的适应度值来判断下一步前进的方向,同时进行位置的更新。公式如下:

$$x^{t+1} = x^t - s^t \vec{b} \text{sign}(f(x_{rt}) - f(x_{lt})) \quad (5)$$

#### 3.2 自适应步长衰减策略

随着迭代次数的增多,天牛搜寻步长也会慢慢缩短,如果天牛的须长过长,则很可能会错过最优解,而如果天牛须长过短,则很容易进入局部的最优。所以为增强算法的搜索能力,可以使用自适应衰减策略来计算步长因子,公式如下:

$$\begin{aligned} d^t &= d^{t-1} \epsilon \\ s^t &= s^{t-1} \epsilon \\ \epsilon &= 1 - 0.08 \end{aligned} \quad (6)$$

其中,  $d^t$  为迭代  $t$  次后的两须之间的距离,  $s^t$  为迭代  $t$  次后的步长因子,衰减系数  $\epsilon$  在  $[0,1]$  区间内取值,本文取  $\epsilon = 0.92$ 。

#### 3.3 天牛群算法

对于标准BAS算法而言,与PSO算法相似,也是通过迭代来进行寻找最优解,虽然算法的收敛速度很快,并且规划出的路径避免了因为转折点过多造成多余路径,不容易陷入局部最优,但存在缺陷,BAS算法中只有一个天牛个体,天牛虽然朝着食物浓度高的方向行走,但不能保障在有限的迭代内搜索到整个空间的食物,因此搜索空间相对于群智能算法存在局限性。因此为了解决这一问题,通过与其他优化算法相结合,可以提高BAS算法的性能,弥补不足。本文通过与PSO算法结合,并结合步长衰减策略,将粒子群的每个粒子当作天牛个体,根据天牛须算法来更新个体的适应度值,根据PSO算法来更新群体,通过天牛个体独特的寻优方式以及粒子群的迭代更新速度,能够很好的找到一条适合移动机器人行走的路径,并且能够避免陷入局部最优。BSO算法的流程如下所示。

步骤 1)初始化参数;

步骤 2)计算每个天牛须的位置以及左右须位置的适应度值;

步骤 3)根据计算出左右须的适应度值,步长的调整,更新学习因子与权重,更新每只天牛个体的速度和位置;

步骤 4)判断路径是否为可行路径,如若不是,返回步骤 2);

步骤 5)更新天牛群体的全局最优位置 gbest 与个体最优位置 pbest;

步骤 6)判断是否达到最优解,如若不是,返回步骤 3); BSO 算法具体流程如图 6 所示。

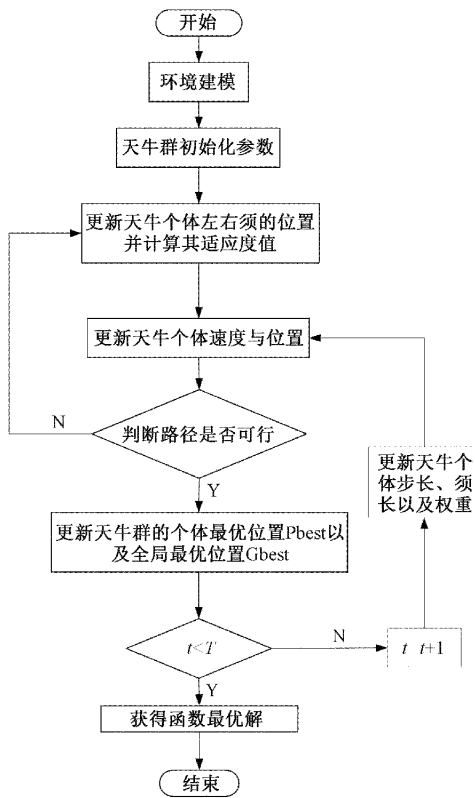


图 6 BSO 算法流程

BSO 算法的速度和位置更新公式如下:

$$\begin{aligned}
 vb_i &= s^t \bar{b} \text{sign}((f(x_n) - f(x_u))) \\
 v_i^{t+1} &= v_i^t + c_1 \cdot \text{rand} \cdot (Pb_i^t - x_i^t) + \\
 &c_2 \cdot \text{rand} \cdot (Pg_i^t - x_i^t) + c_3 \cdot \text{rand} \cdot vb_i \\
 x_i^{t+1} &= x_i^t + v_i^{t+1} \quad (7)
 \end{aligned}$$

其中,  $t$  是迭代次数,  $f(x_n)$  和  $f(x_u)$  分别代表天牛右须和左须的适应度函数,  $vb_i$  是天牛更新率,  $v_i^{t+1}$  与  $x_i^{t+1}$  分别为第  $t$  次迭代后更新的速度和位置。  $s^t$  是步长因子, 影响天牛搜索的步长,  $Pb_i^t$  为迭代  $t$  次后的个体最优值,  $Pg_i^t$  为迭代  $t$  次后的全局最优值,  $c_1, c_2, c_3$ , 分别为 3 个学习因子,  $\text{rand}$  为随机函数。

### 3.4 动态权重策略

针对于 BSO 算法前期收敛快,后期收敛较慢的问题,提出一种动态权重变更策略,通过对收敛速率的判断来改变权重的大小,该方法能够有效提高 BSO 算法的全局搜索能力。

$$\begin{aligned}
 \omega_f &= \omega_l \left( \frac{f - \bar{f}}{\Delta f} + 1 \right) \\
 \omega &= \begin{cases} \omega_{\max}, \omega_f > \omega_{\max} \\ \omega_f, \omega_{\max} \geq \omega_f \geq \omega_{\min} \\ \omega_{\min}, \omega_f < \omega_{\min} \end{cases} \quad (8)
 \end{aligned}$$

其中,  $\bar{f}$  为平均适应度值,  $f$  第  $n$  次迭代时的适应度值,  $\omega_l$  为标准权重值。  $\omega_{\max}$  和  $\omega_{\min}$  一般取为 2.0 和 0.8,  $\Delta f$  为最大适应度值和最小适应度值的差。

## 4 仿真分析

为了验证 BSO 算法的有效性,本文在 Matlab2016(b) 下进行仿真实验,分别在以下两个实验内进行对比验证,观察算法的优越性。天牛群的种群规模大小  $P$  设为 200,迭代次数  $N$  设置为 100 次,天牛须长间距  $d$  设置为 4,步长衰减系数  $\epsilon$  为 0.92,步长大小设为 4,天牛群速度更新的权重系数  $c_1=2.0; c_2=1.5; c_3=0.8$ ;

实验一:通过测试函数来验证算法的优越性,本文选择了一个单峰函数和两个多峰函数来比较 3 种算法的最小值,通过最小值的比较来验证 BSO 算法的优越性。

单峰测试函数:

1) Sphere model

函数表达式:

$$f(x) = \sum_{i=1}^n x_i^2 \quad (9)$$

全局最优值:

$$\min f(x) = 0 \quad (10)$$

函数特点:可以验证函数收敛情况

仿真结果如图 7 所示,从 3 条曲线对比可以得到,BSO 算法的收敛值为  $1.51 \times 10^{-4}$ ,GA-PSO 算法的收敛值为  $1.59 \times 10^{-3}$ ,AIW-PSO 算法的收敛值为  $5.77 \times 10^{-3}$ ,BAS 算法的收敛值为  $2.09 \times 10^{-4}$ ,通过验证,BSO 算法的收敛效果要比其他 3 种算法要优越。

多峰测试函数:

2)Rastrigin model

函数表达式:

$$f(x) = 10d + \sum_{i=1}^d [x_i^2 - 10\cos(2\pi x_i)] \quad (11)$$

全局最优值:

$$\min f(x) = 0 \quad (12)$$

函数特点:来验证算法的实用性

仿真结果如图 8 所示,从 3 条曲线对比可以得到,BSO 算法的收敛值为 0.25,GA-PSO 的收敛值为 1.03,AIW-PSO 的收敛值为 2.24,BAS 算法的收敛值为 3.08,通过验



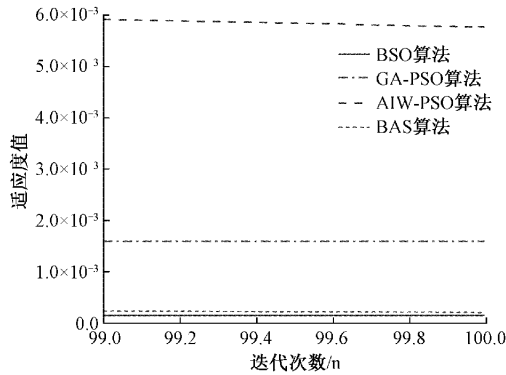


图7 Sphere model 迭代曲线

证,BSO 算法的实用性要比其他 3 种算法要好。

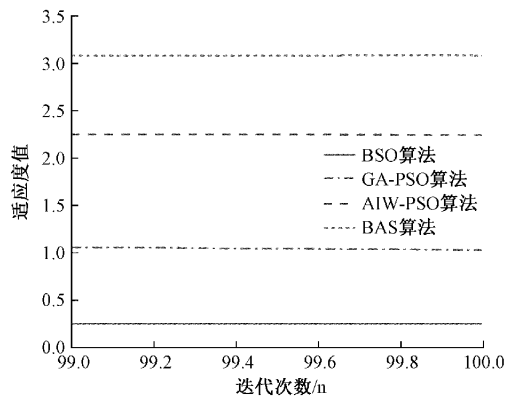


图8 Rastrigin model 迭代曲线

3) Griewank model

函数表达式:

$$f(x) = \sum_{i=1}^d \frac{x_i^2}{4000} - \prod_{i=1}^d \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1 \quad (13)$$

全局最优值:

$$\min f(x) = 0 \quad (14)$$

函数特点:检测算法跳出局部的能力

仿真结果如图 9 所示,从 3 种算法的曲线可以看出,BSO 算法的收敛值为  $9.68 \times 10^{-5}$ ,GA-PSO 算法的收敛值为  $1.42 \times 10^{-4}$ ,AIW-PSO 算法的收敛值为  $3.84 \times 10^{-3}$ ,BAS 算法的收敛值为  $1.31 \times 10^{-3}$ ,所以可以得出 BSO 算法跳出局部最优的能力要优于其他 3 种算法。

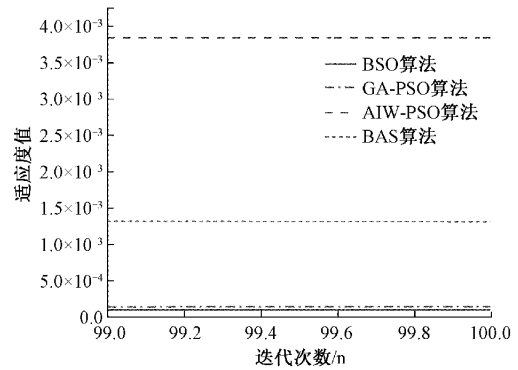


图9 Griewank model 迭代曲线

为了防止实验的偶然性,本文对 3 种测试函数的性能测试进行多次实验,通过得到的平均值来验证算法的结论的可靠性,分别对 4 种算法的测试进行 10 次验证,表 1 是 10 次验证的最优值、最差值以及平均值所测得数据如表 1 所示。

表 1 3 种测试函数测试结果

函数名称	特征值	BSO	GA-PSO	AIW-PSO	BAS
Sphere	最优值	$1.05 \times 10^{-5}$	$1.78 \times 10^{-4}$	$6.28 \times 10^{-4}$	$1.64 \times 10^{-4}$
	最差值	$2 \times 10^{-4}$	$3.2 \times 10^{-2}$	$3.76 \times 10^{-2}$	$9.57 \times 10^{-2}$
	平均值	$9.72 \times 10^{-5}$	$4.32 \times 10^{-3}$	$1.67 \times 10^{-2}$	$2.56 \times 10^{-2}$
Rastrigin	最优值	$6.72 \times 10^{-3}$	0.93	$0.63 \times 10^{-2}$	1.03
	最差值	10.70	3.22	4.67	14.99
	平均值	1.92	1.75	3.46	3.75
Griewank	最优值	$1.05 \times 10^{-5}$	$1.42 \times 10^{-4}$	$5.44 \times 10^{-5}$	$1.31 \times 10^{-3}$
	最差值	$1.83 \times 10^{-4}$	$1.21 \times 10^{-3}$	$3.84 \times 10^{-3}$	$5.54 \times 10^{-2}$
	平均值	$3.92 \times 10^{-5}$	$6.75 \times 10^{-4}$	$7.62 \times 10^{-4}$	$1.59 \times 10^{-2}$

通过 3 个测试函数最终测试结果可以看出,在对比了 GA-PSO 算法、AIW-PSO 算法、BAS 算法后,BSO 算法最优值的收敛值都要小于其他 3 种算法,最差值、平均值整体优于其他 3 种算法,从而可以说明 BSO 算法的收敛效果更优,性能更加稳定,实用性更高。

实验二:本文在基于圆形障碍物的地图下进行仿真实验,分别在简单与复杂环境下进行实验,其中简单环境地

图大小设置为  $10 \text{ m} \times 10 \text{ m}$ ,障碍物的数量设置为 3 个,复杂环境地图的大小设置为  $30 \text{ m} \times 30 \text{ m}$ ,障碍物的数量设置为 12 个,且均匀地分布在地图上,起点设置为  $[0,0]$ ,终点分别设置为  $[10,10]$ , $[30,30]$ 。简单环境下的路线图如图 10 所示。

从仿真结果可以看出 BSO 算法相比于其他两种算法路径更加的平滑,路径长度更短,为了更直观地看出优化

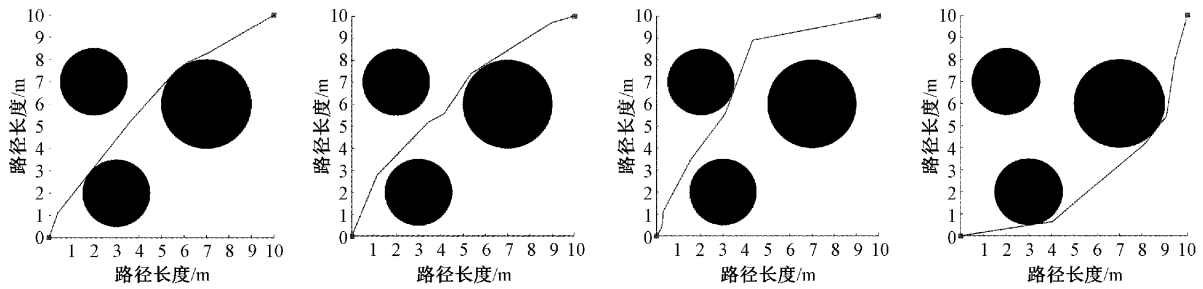


图 10 4 种算法在简单环境下的路线图

结果,通过迭代曲线来比较路径长度的大小,如图 11 所示。

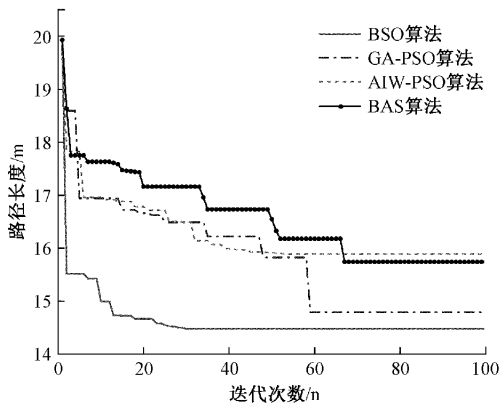


图 11 简单环境路径长度迭代曲线

通过简单环境下的迭代曲线可以发现,BSO 算法优化

后的路径长度为 14.47 m,达到稳定的迭代次数为 30 次,GA-PSO 算法的路径长度为 14.79 m,达到稳定的迭代次数为 59 次,AIW-PSO 算法的路径长度为 15.89 m,达到稳定的迭代次数为 53 次,BAS 算法的路径长度为 15.74 m,达到稳定的迭代次数为 67 次。通过最优适应度值的比较可以得出,相比与 GA-PSO 算法,路径长度减少了 0.32 m,性能提升 2.2%,迭代次数减少 29 次,性能提升 49.2%。相比与 AIW-PSO 算法,路径长度减少了 1.42 m,性能提升 8.9%,迭代次数减少 23 次,性能提升 43.4%。相比于 BAS 算法,路径长度减少了 1.27 m,性能提升 8.1%,迭代次数减少 37 次,性能提升 55.2%。

复杂环境下的路径规划如图 12 所示,从仿真结果可以看出 BSO 算法相比于 AIW-PSO 算法与 BAS 算法的转折点更少,路径更加的平滑,路径长度更短,优化效果明显,4 种算法路径长度大小的迭代曲线如图 13 所示。

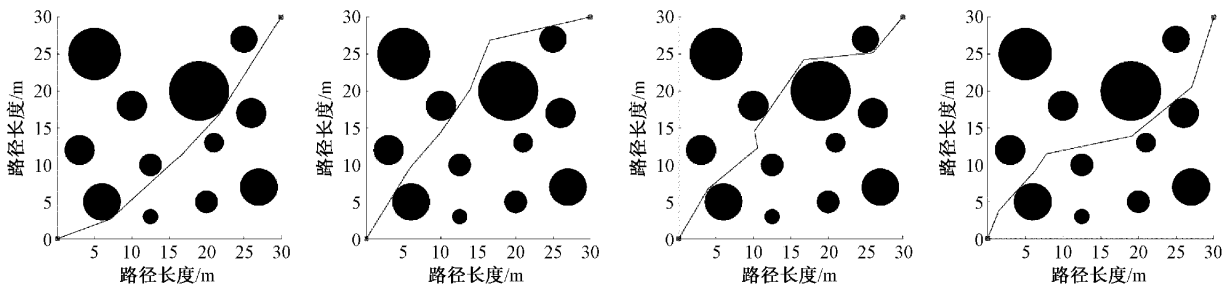


图 12 4 种算法在复杂环境下的路线图

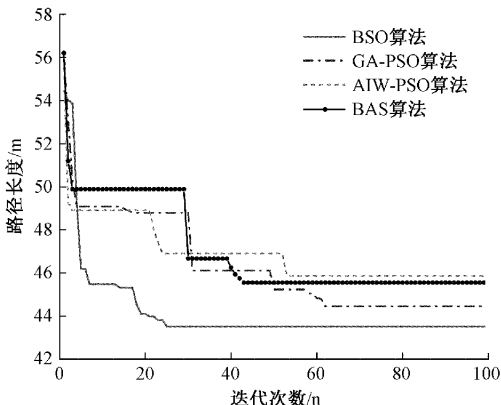


图 13 复杂环境路径长度迭代曲线

通过复杂环境下的收敛曲线可以发现,BSO 算法优化后的路径长度为 43.53 m,达到稳定的迭代次数为 25 次,GA-PSO 算法的路径长度为 44.46 m,达到稳定的迭代次数为 62 次,AIW-PSO 算法的路径长度为 45.87 m,达到稳定的迭代次数为 53 次,BAS 算法的路径长度为 45.55 m,达到稳定的迭代次数为 43 次。通过最优适应度值的比较可以得出,相比与 GA-PSO 算法,路径长度减少了 0.93 m,性能提升 2.1%,迭代次数减少 37 次,性能提升 59.7%,转折点次数少了 1 次,性能提升 33.3%。相比与 AIW-PSO 算法,路径长度减少了 2.34 m,性能提升 5.1%,迭代次数减少 28 次,性能提升 52.8%,转折点次数少了 3 次,性能提升 60%。相比于 BAS 算法,路径长度减少了 2.02 m,性能提

升4.4%,迭代次数减少18次,性能提升41.9%,转折次数减少了2次,性能提升50%。

为了防止实验的偶然性,对该测试进行20次实验,通过多次实验得到路径的最优值、平均值等数据如表2所示。

表2 复杂环境下仿真测试结果

算法名称	最优值/ m	迭代次数	平均值/ m	转折次数
BSO	43.61	34	45.44	2
GA-PSO	44.46	62	45.76	3
AIW-PSO	45.58	83	47.69	5
BAS	45.17	70	47.86	4

通过平均值的比较可以得出,BSO算法相比于GA-PSO算法,路径长度减少了0.32m,性能提升0.7%,BSO算法相比于AIW-PSO算法,路径长度减少了2.25m,性能提升4.7%,相比于BAS算法,路径长度减少了2.42m,性能提升5.1%。

### 5 实验平台搭建与验证

为验证本文融合算法在实际应用中的有效性,将4种不同算法分别进行实验验证。本文搭建了实验平台,其中包括全向移动机器人一台以及断路器装配产线,其中移动机器人包含麦克纳姆轮底盘、上位机、控制器以及仿真环境为运行内存4GB的64位Ubuntu20.04操作系统,实验平台是ROS(noetic),移动机器人结构如图14所示。

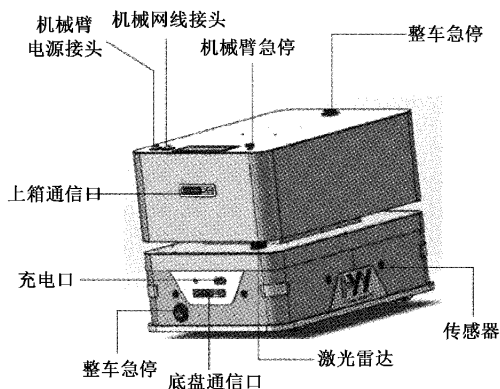


图14 整车结构示意图

底盘主要计算单元为机器人控制器及轮组控制板;机

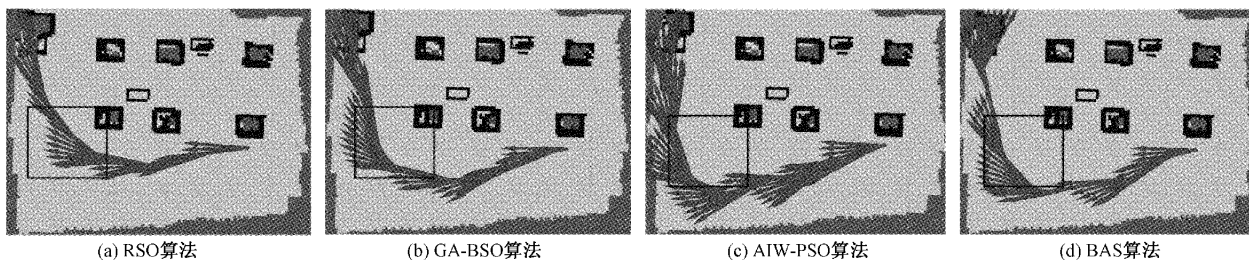


图18 4种算法实际路线图

器人控制器提供Ubuntu环境,能够实现机器人控制、建图导航等功能;轮组控制板实现基本的电机驱动及数据采集。机器人控制器与轮组控制板之间为串口通信连接。底盘上主要配备的传感器为对角放置的两个一线激光雷达、8组超声传感器、正前方配置有realsense d435i相机,其结构如图15所示。

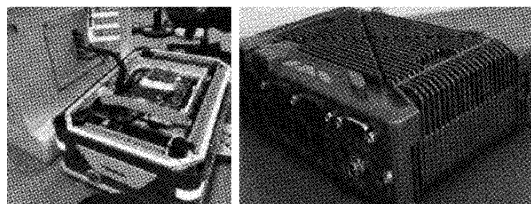


图15 机器人底盘及控制器

实验环境如图16所示,在此条件下进行移动机器人路径规划实验,图中移动机器人处为起点,将装配好的断路器通过机械臂夹取,在运输到标记处的箱子上。

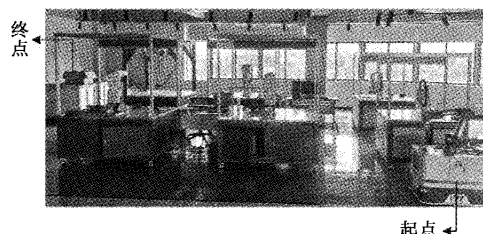


图16 断路器装配产线

通过移动机器人雷达对真实物理环境进行扫描,再通过Slam算法Gmapping进行建图,建好的地图如图17所示,其中白色区域为可行区域,黑色区域为障碍物。

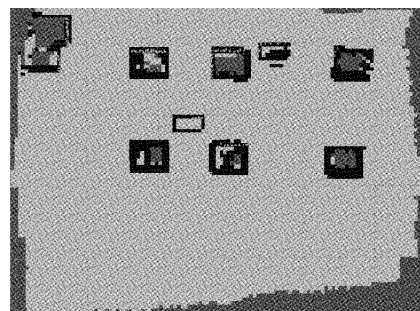


图17 Slam构造地图

在相同的实验环境下,对4种不同的算法进行实验验证,使用Rviz观测实验结果,4种算法的实验结果如图18所示。



通过图中黑框标注的部分可以看出,本文所采用的融合算法相对于其他 3 种算法,从同一起点到终点的过程中,路径更加平滑,最大转折角度更小。

实验数据如表 3 所示,本文所采用的算法相比于融合算法 GA-PSO 算法路径长度减少了 1.18 m,转弯次数减少 1 次,效率分别提升了 7.7% 和 25%。相比于 AIW-PSO 算法,路径长度减少了 2.46 m,转弯次数减少了 4 次,效率分别提升 14.8% 和 57.1%。相比于 BAS 算法,路径长度减少了 2.02 m,转弯次数减少了 2 次,效率分别提升 12.5% 和 40%,从而证明本文所提出融合算法的有效性。

表 3 实际环境下移动机器人路径比较

算法	路径长度/m	转折次数
BSO	14.19	3
GA-PSO	15.37	4
AIW-PSO	16.65	7
BAS	16.21	5

## 6 结 论

本文基于断路器柔性装配提出一种融合算法,该算法能够有效地解决断路器装配中上料、仓储造成的路径冗长、效率低下等问题,能够有效地提高移动机器人在断路器装配产线中的装配效率。

相比于单一粒子群算法容易陷入局部最优的问题,本文所提出的融合算法能够根据天牛群独特的位置更新方式,有效的改善寻优过程中陷入局部最优的问题,提升算法的性能。

### 参考文献

- [1] 焦传佳,江明,徐劲松,等. 基于激光信息的移动机器人定位研究[J]. 电子测量与仪器学报, 2021, 35(9): 1-9.
- [2] 陈晖,周德强. 改进蚁群优化算法的移动机器人路径规划[J]. 电子测量技术, 2020, 43(23): 17-22.
- [3] 曹如月,张振乾,李世超,等. 基于改进 A\* 算法和 Bezier 曲线的多机协同全局路径规划[J]. 农业机械学报, 2021, 52(S1): 548-554.
- [4] 徐骏. 基于超宽带定位的自动导引车辆导航方法及其导航系统的研究[D]. 上海:华东理工大学, 2016.
- [5] 王刚,张方,严大亮,等. 基于改进蚁群算法的机器人三维路径规划[J]. 国外电子测量技术, 2020, 39(11): 1-6.
- [6] 王文焕,郭鹏,詹荣荣,等. 基于最短路径算法的继电保护数据模型结构及搜索优化[J]. 电机与控制学报, 2021, 25(1): 68-78.
- [7] MOHANTY P K, PARH I D R. Controlling the motion of an autonomous mobile robot using various techniques: A review [J]. Journal of Advance Mechanical Engineering, 2013, 1(1): 24-39.
- [8] 于振中,李强,樊启高. 智能仿生算法在移动机器人路径规划优化中的应用综述[J]. 计算机应用研究, 2019, 36(11): 3210-3219.
- [9] 陈秋莲,郑以君,蒋环宇,等. 基于神经网络改进粒子群算法的动态路径规划[J]. 华中科技大学学报(自然科学版), 2021, 49(2): 51-55.
- [10] 徐雪松,曾智,邵红燕,等. 基于个体-协同触发强化学习的多机器人行为决策方法[J]. 仪器仪表学报, 2020, 41(5): 66-75.
- [11] 全秋娟,赵岂,李萌. 基于自适应动态改变的粒子群优化算法[J]. 微电子学与计算机, 2019, 36(2): 6-10,15.
- [12] LIN X, WANG Y, LIU Y, et al. Research on global path planning method of mobile robot based on BAS [C]. Proceedings of 2019 3rd International Conference on Data Mining, Communications and Information Technology(DMCIT), 2019: 119-124.
- [13] 赵辉,郝梦雅,王红君,等. 基于改进 A\* 算法与天牛须搜索算法的农业机器人路径规划方法[J]. 科学技术与工程, 2019, 19(31): 185-190.
- [14] WU Q, CHEN Z, WANG L, et al. Real-time dynamic path planning of mobile robots: A novel hybrid heuristic optimization algorithm[J]. Sensors, 2020, 20(1): 188.
- [15] 胡章芳,冯淳一,罗元. 改进粒子群优化算法的移动机器人路径规划[J]. 计算机应用研究, 2021, 38(10): 3089-3092.
- [16] SUN Y, ZHANG J, LI G, et al. Optimized neural network using beetle antennae search for predicting the unconfined compressive strength of jet grouting coalcretes[J]. International Journal for Numerical and Analytical Methods in Geomechanics, 2019, 43(4): 801-813.
- [17] SONG B, WANG Z, ZOU L. An improved PSO algorithm for smooth path planning of mobile robots using continuous high-degree Bezier curve[J]. Applied Soft Computing, 2021, 100(1): 106960.

### 作者简介

王凌浩,硕士研究生,主要研究方向为电器智能化。

E-mail:wlyhdy123@163.com

舒亮,博士研究生,教授,主要研究方向为机电系统设计与仿真。

E-mail:shuliangalbert@163.com