

基于多策略融合改进粒子群算法的路径规划研究*

陈旭东,杨光永,徐天奇,樊康生
(云南民族大学电气信息工程学院,昆明 650000)

摘要:针对传统粒子群算法(particle swarm optimization, PSO)在路径规划中易陷入局部最优使得规划路径较长以及搜索后期由于种群多样性降低容易陷入停滞等问题,提出一种多策略融合粒子群算法(multi-strategy fusion particle swarm optimization, MFPSO)并将其应用于路径规划中。首先,利用中垂线算法(midperpendicular algorithm)的粒子位置更新方法提升粒子的收敛速度;其次,在最优粒子附近采用生成爆炸粒子的策略使算法跳出局部最优;然后,引入线性动态惯性权重调整方法,增加算法的搜索能力;最后,在路径规划应用中采用全局最优解局部搜索策略,在算法后期得出的最优路径再进行局部搜索得出更优的路径,增加机器人路径规划能力。仿真结果表明,多策略融合粒子群算法在路径规划中具有更高的路径搜索能力。

关键词:路径规划;中垂线算法;爆炸粒子;全局最优解局部搜索

中图分类号:TH166;TG659

文献标识码:A

Path Planning Based on Multi-Strategy Fusion Particle Swarm Optimization Algorithm

CHEN Xudong, YANG Guangyong, XU Tianqi, FAN Kangsheng

(School of Electrical Information Engineering, Yunnan Minzu University, Kunming 650000, China)

Abstract: In order to solve the problem that the traditional particle swarm optimization algorithm is easy to fall into the local optimal in the path planning, which makes the planned path long and easy to stall due to the decrease of population diversity in the late search period, a multi-strategy fusion particle swarm optimization algorithm is proposed and applied to the path planning. Firstly, the particle position updating method of the midvertical algorithm is used to improve the convergence rate of particles. Secondly, the strategy of generating explosive particles near the optimal particles is used to make the algorithm jump out of the local optimal. Then the linear dynamic inertia weight adjustment method is introduced to increase the searching ability of the algorithm. Finally, the local search strategy of the global optimal solution is adopted in the path planning application. The optimal path obtained at the later stage of the algorithm is then locally searched to obtain a better path, which increases the path planning ability of the robot. The simulation results show that the multi-strategy fusion particle swarm optimization algorithm has higher path search capability in path planning.

Key words: path planning; midvertical algorithm; explosive particle; global optimal solution local search

0 引言

机器人的路径规划问题在机器人导航领域的研究一直都是热点,而路径规划是指机器人根据地图信息寻找起点到终点的一条无碰撞且代价最小的路径。

针对路径规划问题,许多学者提出了众多智能算法应用在路径规划研究中,如蚁群算法、遗传算法、神经网络算法、粒子群算法等智能算法。杨北辰等^[1]针对传统蚁群算法存在收敛速度慢、容易陷入局部最优

等问题将一种自适应归档更新方法应用在蚁群算法中,根据路径信息建立了多目标性能评估模型,对路径进行多指标优化,提高算法的收敛速度。张毅等^[2]将初始化成功的无碰撞路径转化为一连串的栅格地图中栅格序号,再将路径序号转化成路径进行优化,但是该算法中参数选择人需凭借经验选取,且对环境的依赖性强。魏冠伟等^[3]利用人工神经网络优化路径,但是由于人工神经网络的局限性,该方法存在收敛速度慢、

收稿日期:2023-04-21;修回日期:2023-06-19

* 基金项目:国家自然科学基金项目(61761049, 61261022)

作者简介:陈旭东(1998—),男,硕士研究生,研究方向为机器人运动控制, (E-mail) 1938989359@qq.com;通信作者:杨光永(1970—),男,副教授,硕士生导师,博士,研究方向为算法设计、信号处理、智能控制, (E-mail) guangyong_yang@126.com。

引用本文:陈旭东,杨光永,徐天奇,等. 基于多策略融合改进粒子群算法的路径规划研究[J]. 组合机床与自动化加工技术, 2024(2): 44-50.

CHEN Xudong, YANG Guangyong, XU Tianqi, et al. Path planning based on multi-strategy fusion particle swarm optimization algorithm[J].

Modular Machine Tool & Automatic Manufacturing Technique, 2024(2): 44-50.

全局搜索能力差、易陷入局部最优等缺点。

而对于粒子群算法 (particle swarm optimization) 在路径规划中的应用,其具有收敛速度快,模型简单,待调参数少等特点。但是算法在后期由于种群多样性减少容易陷入局部最优值,使得优化效果并不理想。

国内外许多学者为提升粒子群算法的性能提出了许多改进方法。LIU 等^[4]在粒子群算法寻优过程中引入非线性惯性权重调整提高了算法性能。花赞昊等^[5]通过在粒子群算法寻优过程中同步调整学习因子与惯性权重,提高了算法收敛速度以及收敛精度,并且为增强算法的全局寻优能力引入了变异机制扩大粒子搜索空间。朱任等^[6]借鉴了小生境思想,基于粒子群算法基础上提出利用顺序聚类算法将不同的粒子划分到不同的小生境,并根据小生境的迭代数制定不同的搜索策略一次提升算法总体性能。杨妹兰等^[7]采用比粒子自身适应值更好的邻近粒子为学习对象,并且分两个阶段用不同更新速度公式,提高了粒子间的信息交流,同时平衡了算法的局部开发性能与全局寻优能力。廖玮霖等^[8]利用三黑洞系统捕获策略以及多维随机干扰策略,使算法提高了局部搜索能力同时还兼顾了全局开拓能力,并且通过协调因子从全局寻优转变维向局部搜索,进而提高收敛速度。

上述各改进方法均综合提高了粒子群算法的性能,然而目前改进的粒子群算法大多着重于平衡算法的收敛精度和收敛速度。虽具有一定的改善效果,但是由于两者的矛盾性,提升效果依旧具有局限性。针对所述问题,本文提出一种多策略融合改进粒子群算法应用在路径规划中,首先建立机器人路径规划所需的栅格地图模型,并对传统的粒子群算法作进一步的改进,利用中垂线算法的收敛策略加快算法的收敛速度;以最优粒子爆炸策略避免陷入局部最优解;采用自适应惯性权重更新方法进一步提升算法的性能;最后将全局最优解继续进行局部搜索,提高全局最优解适应度值。

1 多策略融合粒子群算法

1.1 粒子群算法

粒子群算法的核心是利用群体中的个体对信息的共享使整个群体的运动在问题求解空间中产生问题的最优解。其数学模型为:

$$v_i^{d+1} = \omega v_i^d + c_1 \text{rand}(P_{\text{best}} - x_i^d) + c_2 \text{rand}(G_{\text{best}} - x_i^d) \quad (1)$$

$$x_i^{d+1} = x_i^d + v_i^{d+1} \quad (2)$$

式中: d 为搜索空间, v_i^d 为第 i 次迭代第 d 维的粒子速度, x_i^d 为第 i 次迭代第 d 维的粒子位置, c_1 和 c_2 为学习因子, ω 为惯性权重, rand 为 $0 \sim 1$ 之间的随机数。其中,每个粒子的状态都受种群中个体最优解 P_{best} 和全局最优解 G_{best} 的影响。

1.2 多策略融合粒子群算法

1.2.1 中垂线算法

中垂线算法的核心思想是:任意线段的中垂线上任意一点到该线段的两端点的距离相等。示意图如图

1 所示,假设在一个足够小的区域内(矩形区域)存在一个最优点 P 和两个随机点 M 和 N ,连接 MN 并作线段 MN 的中垂线 L 。由图 1 可知 $|PM| < |PN|$,因中垂线 L 上任意点到 M 点与 N 点的距离相等,则 P 点必定位于 L 左侧 M 点的区域。在中垂线算法中,以最优点到该点的距离作为适应度,即在图 1 中 M 点适应度比 N 点适应度更好。

以图 1 为例,中垂线算法的收敛过程为:判定线段 $|PM|$ 是否小于 $|PN|$,若小于,则判定最优点 P 必存在于矩形左侧网格区域;舍弃原 N 点,在矩形左侧网格区域随机生成 N' 点;作 M 和 N' 构成线段的中垂线 L' ,此时可知, N' 点的适应度必定比 M 点的适应度更佳,由此可知最优点所在区域必存在于 L' 右侧区域(图 1 中阴影网格区域)。重复以上步骤可不断缩小最优点 P 的所处区域。

图 1 算法原理图

1.2.2 融合中垂线算法的粒子变化策略

若在图 1 中右侧区域随机加入粒子 G ,如图 2 所示。采用传统粒子群更新策略必然需要经多次更新才能使 N 点、 G 点的位置到达中垂线左侧区域。

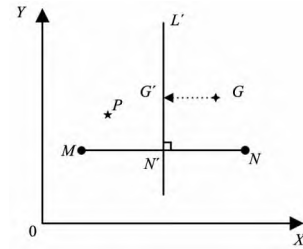


图 2 粒子变化原理图

针对传统粒子群更新速度慢的问题首先将粒子 N 和 G 的位置先移动到 N' 和 G' 的位置,由于传统粒子群的更新策略会使粒子回到中垂线左侧区域,因此本文采用改进粒子群更新策略更新粒子速度和位置。

粒子移动的实现方式为:首先判断粒子是否存在于最优点所在区域,设:

$$\begin{cases} A = (a_1, a_2, \dots, a_d) \\ B = (b_1, b_2, \dots, b_d) \end{cases} \quad (3)$$

式中: d 表示粒子维度,假设某粒子存在于最优点所在区域,则 x 满足:

$$\begin{cases} \lambda_1 x_i^j + \lambda_2 x_i^{j+1} + \lambda_3 < 0, j < d \\ \lambda_1 x_i^j + \lambda_2 x_i^1 + \lambda_3 = 0, j = d \end{cases} \quad (4)$$

式中: j 表示 x 的第 j 维,

$$\begin{cases} \lambda_1 = 2(b_j - a_j) \\ \lambda_2 = 2(b_{j+1} - a_{j+1}) \\ \lambda_3 = (a_j)^2 - (b_j)^2 + (a_{j+1})^2 - (b_{j+1})^2 \end{cases}, j \in [1, D-1]$$

$$\begin{cases} \lambda_1 = 2(b_j - a_j) \\ \lambda_2 = 2(b_1 - a_1) \\ \lambda_3 = (a_j)^2 - (b_j)^2 + (a_{j+1})^2 - (b_{j+1})^2 \end{cases}, j = D \quad (5)$$

为减小算法的时间复杂度和增强算法全局寻优能力,判定仅需任意二维元素满足式(3)则位于中垂线左侧区域。

若该粒子位于中垂线右侧区域,则 N 点更新公式为:

$$\begin{cases} x'_{i,j} = \frac{\lambda_2^2 x_{i,j} - \lambda_1 \lambda_2 x_{i,j+1} - \lambda_1 \lambda_3}{\lambda_1^2 + \lambda_2^2} \\ x'_{i,j+1} = \frac{\lambda_1^2 x_{i,j+1} - \lambda_1 \lambda_2 x_{i,j} - \lambda_2 \lambda_3}{\lambda_1^2 + \lambda_2^2} \end{cases}, j < D \quad (6)$$

$$x'_{i,j} = \frac{\lambda_2^2 x_{i,j} - \lambda_1 \lambda_2 x_{i,1} - \lambda_1 \lambda_3}{\lambda_1^2 + \lambda_2^2}, j = D \quad (7)$$

为验证中垂线算法的性能,本文选取 Rosenbrock 函数作为实验函数,验证其全局寻优能力。图3和图4为融合中垂线算法的改进 PSO 算法寻找 Rosenbrock 函数最小值时粒子的变化。公式为:

$$f(x) = \sum_{i=1}^n (100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2) \quad (8)$$

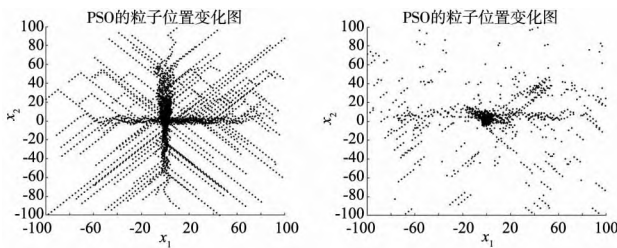


图3 PSO 算法粒子变化图 图4 改进 PSO 算法粒子变化图

从图3和图4的粒子位置变化对比可看出,改进 PSO 算法能够更快收敛到最优值,而传统粒子群算法粒子位置变化太过分散。因此验证了中垂线算法具有很好的收敛能力。

1.2.3 最优粒子爆炸策略

在传统粒子群算法中,全局最佳粒子 G_{best} 按常规更新公式更新时,下一代种群的粒子位置一般会比前一代的粒子位置更差,最优粒子的作用并没有显现出来。因此本文在最优粒子 G_{best} 附近生成一定数量的爆炸粒子,生成方式为:

$$ex_{nj} = G_{best_j} + \frac{rand(-1,1)(G_{best_j} - b_j)}{4t} \quad (9)$$

由式(9)生成的第 n 个爆炸粒子 ex_n 的 j 维变量,这些粒子生成于最优粒子的周围。

采用生成爆炸粒子的策略是为了算法更好的跳出局部最优,该策略依旧选取 Rosenbrock 函数作为实验函数,验证其该策略的可行性。由图5可知,传统 PSO 算法会陷入局部最优值,而采用最优爆炸粒子策略能够使算法更好的跳出局部最优值。

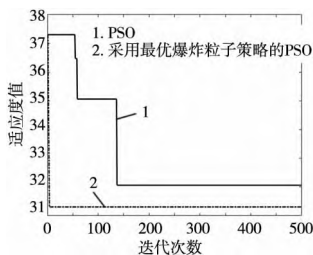


图5 适应度收敛对比

1.2.4 改进粒子速度更新

利用中垂线算法改进后,若依旧利用传统粒子群算法的粒子历史位置信息反而会导致粒子远离最优

点。故粒子速度更新策略变为:

$$v_i^{d+1} = \omega v_i^d + c_2 rand(G_{best} - x_i^d) \quad (10)$$

由于经过最优粒子爆炸策略后,若种群粒子总数为 m ,在最佳粒子周围生成了 n 个爆炸粒子,那么 $m-n$ 个传统粒子按照式(10)更新,新生成的爆炸粒子按照式(11)更新。

$$ev_i^{d+1} = \omega v_i^d + c_2 rand(ex_{nj} - x_i^d) \quad (11)$$

1.2.5 线性动态惯性权重调整方法

在传统粒子群算法中,惯性权值是一个定值。如果取值太大虽然加快了搜索速度但是在算法后期不利于粒子后期小范围的搜索,若取值太小,虽然增加了局部搜索能力但是则不利于算法速度会锐减。针对以上问题,本文采用线性递减规律改变惯性权重。权重 ω 的变化式为:

$$\omega = \omega_{max} - \frac{(\omega_{max} - \omega_{min})t}{T_{max}} \quad (12)$$

式中: ω_{max} 是最大权值,通常设为 0.8; ω_{min} 是最小权值,通常设为 0.5; T_{max} 为最大迭代, t 为粒子当前迭代。引入线性动态惯性权重调整方法是为了增加算法的全局搜索能力以及局部探索能力,本文采用 Rosenbrock 函数作为实验函数进行验证。由图6可看出引入线性动态惯性权重能够很好的提升算法的收敛速度以及搜索精度。

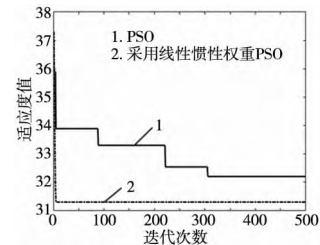


图6 适应度收敛对比

1.3 多策略融合粒子群算法求解流程

根据上述所提策略并将其融合而提出的多策略融合改进粒子群算法的具体步骤为:

- 步骤1:随机初始化各个参数,如学习因子,初始粒子速度位置,爆炸粒子数目,种群学习率;
- 步骤2:通过适应度函数计算适应度值;
- 步骤3:适应度最佳的粒子作为全局最优值 M ,适应度次优的粒子作为 N 点;
- 步骤4:利用式(12)求出粒子惯性权重;
- 步骤5:利用式(9)生成爆炸粒子;
- 步骤6:利用式(10)和式(11)更新粒子速度;再由式(2)更新粒子位置;
- 步骤7:计算所有更新后粒子的适应度值;
- 步骤8:更新 G_{best} (M 点)和 N 点的值;
- 步骤9:结束条件达到则停止搜索,否则回到步骤2继续执行。

1.4 算法时间复杂度分析

假设粒子数为 m ,最大迭代次数为 T ,粒子维度为 D ,爆炸粒子数为 e ,粒子群算法的时间复杂度主

要是粒子速度和位置更新,可知标准粒子群算法的时间复杂度为 $O(mDT)$ 。由本文算法的步骤可知,在中垂线算法策略中,粒子每更新一次,本文的粒子位置更新策略增加的时间有判断粒子是否处于正确位置以及将粒子移动的时间都是 $O(mD)$,而确定两个最优粒子的位置,增加的时间为 $O(m)$,舍弃个体历史最优位置则减少的时间为 $O(m)$;最优粒子爆炸策略的时间为 $O(eD)$;而线性惯性权重调整是减小了算法的运行时间,因此该策略的增加的时间为 $O(0)$,则本文算法总的时间复杂度为 $O(2mDT + eDT)$,略去低阶项后可知本文算法时间复杂度比标准粒子群算法大致相同。

2 算法性能测试

为了验证多策略融合粒子群算法的可行性和优越性,将本文改进后的算法(MFPSO)与均值粒子群优化算法(MeanPSO)、线性递减惯性权重粒子群优化算法(LDWPSO)、基于终端交叉和转向扰动的粒子群优化算法(TCSPSO)以及文献[8]的多策略融合的粒子群优化算法(MSPSO)^[8]在5个基准测试函数下进行对比实验。设置迭代次数为1000,种群数量为80,各算法其他参数设置如表1所示,5个测试函数的取值范围、最小值等信息如表2所示,其中F1~F3为单峰基准测试函数,F4、F5为多峰基准测试函数。本文实验平台为:MATLAB 2022(Intel Core i5,2.11 GHz CPU and 16 GB RAM)。

表1 各算法参数设置

算法	参数
MFPSO	$\omega_{\max} = 0.9, \omega_{\min} = 0.4, c_2 = 2$
MeanPSO	$\omega = 0.9, c_1 = c_2 = 2$
LDWPSO	$\omega_{\max} = 0.9, \omega_{\min} = 0.4, c_1 = c_2 = 2$
TCSPSO	$\omega_{\max} = 0.9, \omega_{\min} = 0.4, c_1 = c_2 = 2$
MSPSO	$\omega_{\max} = 0.9, \omega_{\min} = 0.4, c_1 = c_2 = 2, r = 0.1, \psi = 0.05, p = 0.35, p_1 = 0.8, p_2 = 0.2$

表2 基准测试函数

编号	函数描述	维度	变量区间	最小值
F1	$f(x) = \sum_{i=1}^n x_i^2$	30	$[-100, 100]$	0
F2	$f(x) = \sum_{i=1}^n (\sum_{j=1}^i x_j)^2$	30	$[-100, 100]$	0
F3	$f(x) = \sum_{i=1}^n (ix_i)^2 + \text{random}(0, 1)$	30	$[-1.30, 1.30]$	0
F4	$f(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	30	$[-600, 600]$	0
F5	$f(x) = \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i) + 10]$	30	$[-5.12, 5.12]$	0

为了验证本文算法的性能,本文使用两组基准函数进行测试并且从收敛速度和收敛精度两方面来对比,为了降低由于偶然性引起的实验误差,本文分别在5个测试函数中独立进行50次实验。测试函数的收敛曲线对比如图7所示,实验结果如表3所示。其中横坐标代表迭代次数,纵坐标代表适应度的对数值。

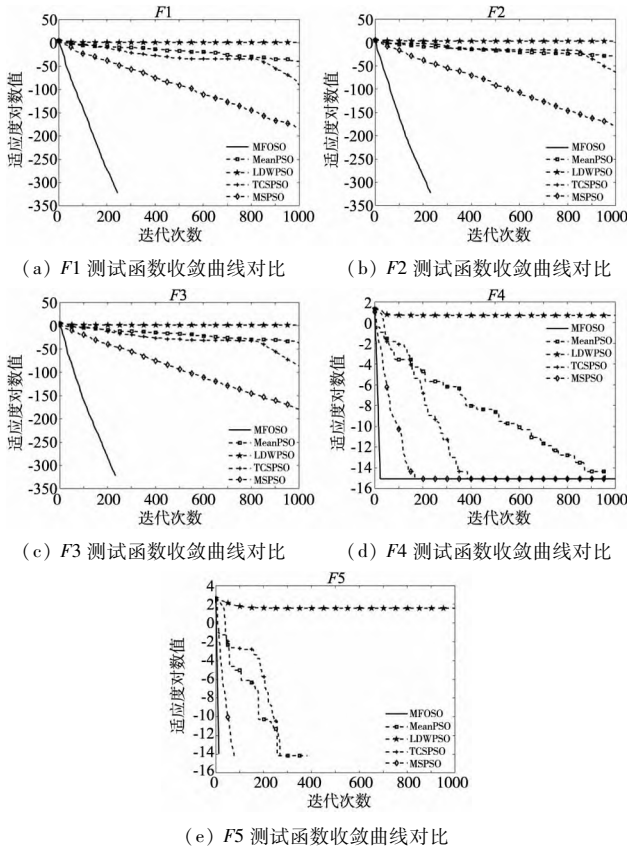


图7 不同测试函数收敛曲线对比

表3 实验数据

函数	数据	MFPSO	MeanPSO	LDWPSO	TCSPSO	MSPSO
F1	最小值	0	1.03E-40	4.27E+00	6.84E-91	2.07E-184
	平均值	1.9763E-323	1.54E-28	1.50E+01	2.58E-82	1.38E-173
F2	最小值	0	2.12E-37	1.05E+02	5.38E-69	4.62E-183
	平均值	0	6.54E-28	4.27E+02	1.91E-64	1.60E-170
F3	最小值	0	6.34E-39	5.97E+00	2.73E-89	7.65E-185
	平均值	1.1364E-322	1.41E-28	2.69E+01	7.39E-80	2.60E-174
F4	最小值	8.8818E-16	8.88E-16	3.31E+00	8.88E-16	8.88E-16
	平均值	1.5987E-15	2.10E-15	4.87E+00	8.88E-16	8.88E-16
F5	最小值	0	0	1.24E+01	0	0
	平均值	0	0	2.93E+01	0	0

从图7a~图7e可以看出,本文算法相较于其它4种算法,在收敛速度以及收敛精度方面具有很大的提升。本文算法不仅寻优精度明显优于其他5种算法,同时在算法初期由于引入了中垂线算法增加了粒子的游离速度,使得算法前期收敛速度明显加快。并且能够在较少的迭代次数就能达到较好的寻优精度,这是因为在粒子速度更新阶段引入了线性递减规律惯性权重,使得算法能够更好的平衡全局搜索能力以及局部拓展能力,具有更好的搜索范围,而其余算法不同程度的出现了停滞过程,寻优精度较低等问题。由表3可以看出,测试函数为F1~F3和F5时无论是单峰或多峰基准测试函数,本文算法在独立实验结果的最小值和均值都是最小,而当测试函数为F4时,最小值虽与MeanPSO、TCSPSO、MSPSO相等,但是平均值相较于其余4种算法是最低的。综上所述,本文改进的MFPSO

算法整体性能优于其他4种改进算法。

3 多策略融合粒子群算法在路径规划中的应用

3.1 全局最优解局部搜索策略

传统 PSO 算法的初始化策略通常为随机策略,在算法末期,可能得到的全局最优解并不是最优的解,针对此问题,本文在算法末期采取把全局最优解(最优路径)继续进行局部搜索,如果新的适应度值小于原先适应度值,则将新路径作为最优路径。效果对比图(以 20×20 为例)如图 8 ~ 图 10 所示。

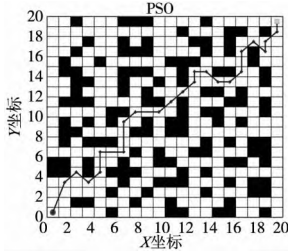


图 8 传统 PSO 路径规划

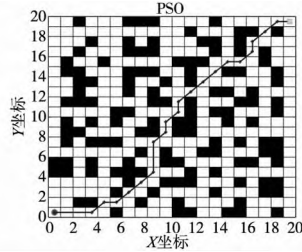


图 9 全局最优解局部搜索

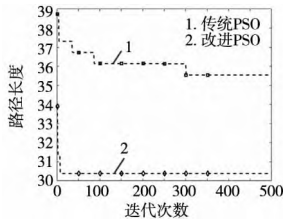


图 10 路径长度对比

从图 8 ~ 图 10 可以看出,采取全局最优解局部搜索策略的粒子群路径规划能够将传统粒子群产生的最优路径继续搜索寻优得到更佳的规划路径。

3.2 适应度函数建立

机器人路径规划是根据地图信息寻找起点到终点的一条无碰撞且代价最小的路径。其代价函数是检验算法优化程度的重要指标。本文代价函数分成两部分,第一部分用来判断路径的长短,第二部分用来判断平滑程度。路径长度的计算公式为:

$$fit_1 = \sum_{i=1}^d \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2} \quad (13)$$

式中: fit_1 为相邻点的距离的和, d 表示维度。

平滑度的计算公式为:

$$d = \sum_{i=1}^d (\sqrt{(x_{i+2} - x_{i+1})^2 + (y_{i+2} - y_{i+1})^2} + \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2}) / 2 \quad (14)$$

路径中机器人行进的角度有 4 种情况,分别为直线、钝角、直角和锐角,其中直线平滑度最好,钝角、直角次之。由于机器人行进中锐角是不允许的,所以对除直线外的另外 3 种情况给予惩罚,可以根据实际情况更改惩罚值大小。

$$fit_2 = 1 / (fit_2 + c) \quad (15)$$

适应度函数的两部分分别取一个权重值,则适应度函数公式如下:

$$f = a \cdot fit_1 + b \cdot fit_2 \quad (16)$$

式中: a 、 b 分别为长度因子、平滑度因子,可根据路径长度和平滑度之间的权重选择参数 a 和 b 。

3.3 路径规划问题

为进一步验证本文提出的融合多策略改进的粒子

群算法的实用性与可行性,本文采用传统栅格地图来模拟机器人的工作环境。黑色区域为障碍物,把白色区域为可行区域,其中深灰色点标志(左下角)为机器人的起始点,浅灰色点标志(右上角)为机器人的结束点;地图的边界是整个路径规划的最外围区域,当成障碍对待。

本文采用在 20×20 与 30×30 规模的简单栅格环境和 20×20 与 30×30 规模的复杂栅格环境,将本文改进粒子群算法用于栅格地图路径规划问题中,并与文献[8]算法(MSPSO)、传统粒子群算法(PSO)、麻雀搜索算法(SSA)、进行对比并记录平均路径长度、最优路径长度。本文粒子群初始参数设置为:迭代次数为 500;种群数量设置为 100;权重系数最大值 ω_{\max} 为 0.9;权重系数最小值 ω_{\min} 为 0.4;社会因子 c_2 设为 2。

3.3.1 简单环境

各算法在 20×20 栅格地图简单模型的最短规划路径对比如图 11 所示,在 30×30 栅格地图简单模型的最短规划路径对比如图 12 所示,简单环境的最优路径收敛曲线对比如图 13 所示,各算法实验数据统计如表 4 所示。

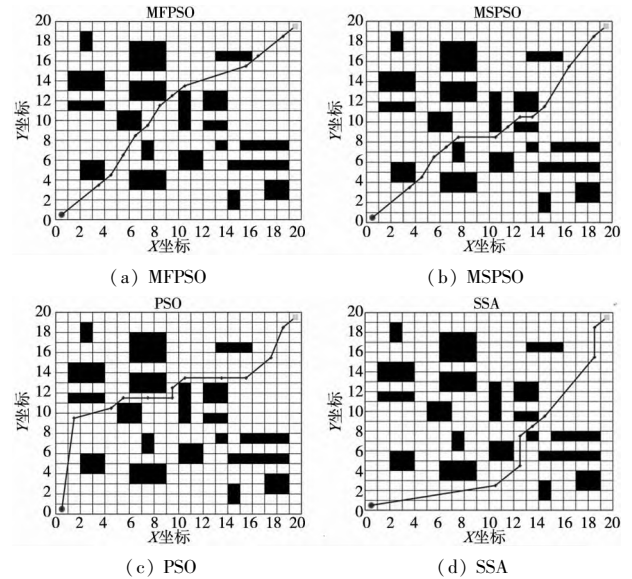


图 11 20×20 简单栅格环境下各算法的最优路径对比图

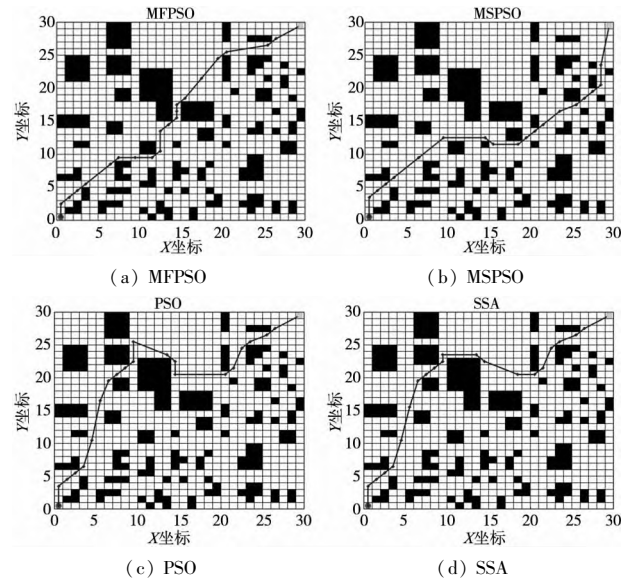
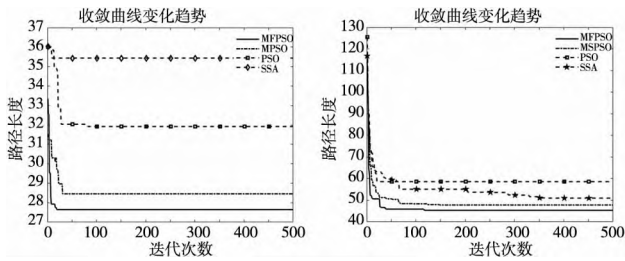


图 12 30×30 简单栅格环境下各算法的最优路径对比图



(a) 20 × 20 简单路径收敛曲线 (b) 30 × 30 简单路径收敛曲线

图 13 简单环境最优路径对比图

表 4 简单环境各算法路径统计表

地图	指标	MFPSO	MSPSO	PSO	SSA
20 × 20	最优路径	27.649 7	28.455 9	31.918 5	35.433 9
	平均路径	27.698 5	28.555 8	32.093 4	35.443 3
30 × 30	最优路径	45.300 4	47.774 7	53.648 4	50.963 4
	平均路径	46.280 3	48.978 6	54.900 8	54.564 6

由图 11 和图 12 可得,本文算法在 20 × 20,30 × 30 的简单环境下找到的最优路径均优于其余 3 种算法;从表 4 可以看出在 20 × 20 栅格地图得到最优路径是 27.649 2,比 MSPSO 减少了 0.806 2,并且在 30 × 30 栅格地图下,得到的最优路径比 MSPSO 减少了 2.474 3,由此可以说明 MSPSO 在较复杂的栅格地图环境的搜索精度会降低;从图 13 可以看出 MSPSO 虽然在算法初期能够快速收敛到最优值,但是正在后期存在搜索能力不足,搜索精度不够。

3.3.2 复杂环境

各算法在 20 × 20 栅格地图复杂模型的最短规划路径对比如图 14 所示,在 30 × 30 栅格地图复杂模型的最短规划路径对比如图 15 所示,复杂环境的最优路径收敛曲线对比如图 16 所示,各算法实验数据统计如表 5 所示。

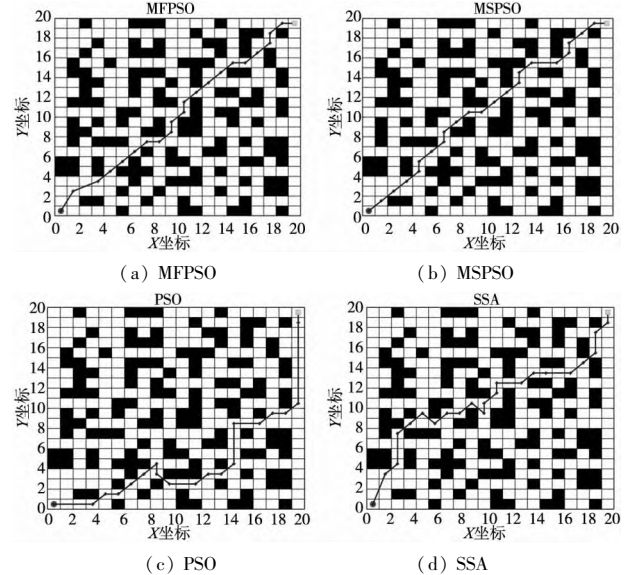
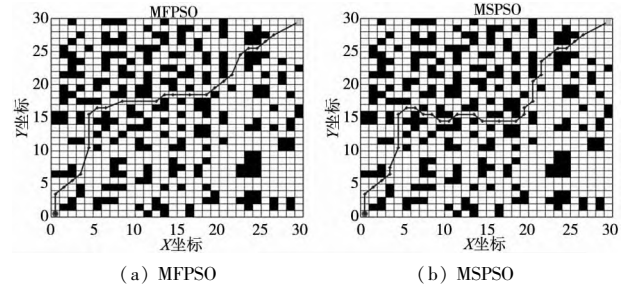
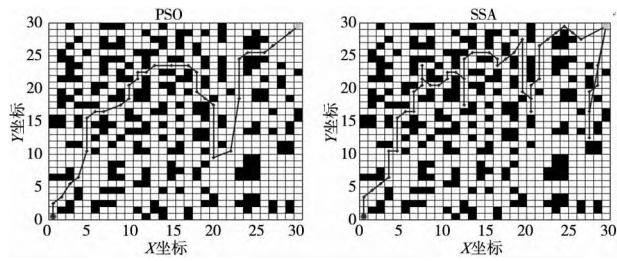


图 14 20 × 20 复杂栅格环境下各算法的最优路径对比图

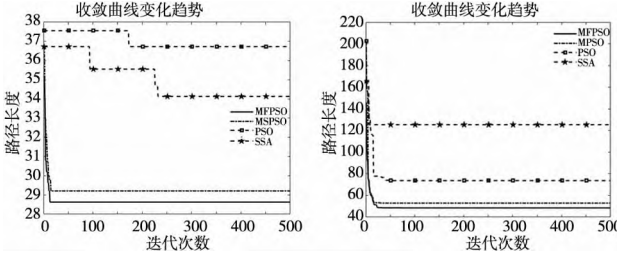


(a) MFPSO (b) MSPSO



(c) PSO (d) SSA

图 15 30 × 30 复杂栅格环境下各算法的最优路径对比图



(a) 20 × 20 复杂环境 (b) 30 × 30 复杂环境
最优路径收敛曲线 最优路径收敛曲线

图 16 复杂环境最优路径对比图

表 5 复杂环境各算法路径统计表

地图	指标	MFPSO	MSPSO	PSO	SSA
20 × 20	最优路径	28.627 4	29.213 2	36.727 9	34.132 8
	平均路径	28.687 3	29.285 7	37.012 9	34.998 5
30 × 30	最优路径	48.477 2	52.981 0	73.692 6	125.361 6
	平均路径	49.356 2	54.155 7	75.794 0	125.842 8

由图 14 和图 15 最优路径对比图可知,算法都能够实现到达终点且寻找到了最优路径的目的。但是除了 MFPSO 和 MSPSO,其它算法路径的波动性明显,并且路径会经过障碍点,生成的路径平滑度差、距离长、算法规划出的路径存在绕弯路甚至于规划出有障碍的路径等情况。而本文算法相对 MSPSO,本文提出的改进粒子群算法在路径长度和路径平滑度上都优于 MSPSO,能够迅速的找到路径最短且不经障碍物障碍物的最优路径,且路径平滑度也适于机器人实际运动轨迹。

由图 16 最优路径收敛曲线反应的情况可知由于 MFPSO 能迅速收敛至最小适应度值,这是由于通过中垂线算法(MA)优化粒子群算法的粒子位置变化,搜索前期加快了粒子群的收敛能力;搜索后期采用最有粒子爆炸策略与全局最优解局部搜索策略使算法不易陷入局部最优解,增强了全局搜索能力。综上所述,本文提出的多策略融合粒子群算法无论在收敛速度还是寻优能力都优于其它 4 种算法,因此将本文算法应用在机器人路径规划中能有效的减小机器人规划路径的长度。

由表 5 各算法的路径规划统计数据可知,可以清楚的看出 MFPSO 具有更好以及更稳定的最优路径规划能力。两种不同规格地图环境下 MFPSO 得到的最短路径长度分别为 28.627 4 和 48.472 2,均小于其它算法得到的最短路径,且 MFPSO 的平均值也最小,由于最优路径长度反映了算法的有效性和优越性;而平

均值则反映算法进行路径规划的稳定性。因此由结果可知把多策略融合改进的粒子群算法应用在移动机器人路径规划中具有一定的优势。

4 结论

针对传统的粒子群算法在移动机器人路径规划中存在惯性权值不适、迭代后期粒子群体多样性下降等问题,提出一种融合多策略的改进粒子群搜索算法(MFPSO)。改进后的粒子群算法具有更快的收敛速度和寻优能力,同时也加强了跳出局部最优解的能力。在算法初期,利用中垂线算法的收敛策略加快算法的收敛速度;采用自适应惯性权重更新方法进一步提升算法的性能,在算法后期以最优粒子爆炸策略使算法跳出局部最优值;再将全局最优解继续进行局部搜索提高机器人路径的规划能力。仿真结果表明,本文所提出的多策略融合改进粒子群算法在机器人路径规划中展现出更好的路径寻优能力。

[参考文献]

- [1] 杨北辰,余粟.改进蚁群算法在路径规划中的应用[J].计算机应用研究,2022,39(11):3292-3297,3314.
- [2] 张毅,代恩灿,罗元.基于改进遗传算法的移动机器人路径规划[J].计算机测量与控制,2016,24(1):313-316.
- [3] 魏冠伟,付梦印.基于神经网络的机器人路径规划算法[J].计算机仿真,2010,27(7):112-116.
- [4] LIU H,ZHANG X W,TU L P. A modified particle swarm optimization using adaptive strategy[J]. Expert Systems with Applications,2020,152:113353.
- [5] 花赞昊,朱武,靳一奇,等.基于自适应变异粒子群算法的光伏 MPPT 控制研究[J].太阳能学报,2022,43(4):219-225.
- [6] 朱任,庄铭杰.小生境粒子群算法在多信号 DOA 估计中的应用[J].电讯技术,2022,62(4):473-481.
- [7] 杨妹兰,刘衍民,张倩,等.基于平均位置学习的改进粒子群算法研究[J].重庆工商大学学报(自然科学版),2022,39(3):9-19.
- [8] 廖玮霖,程杉,尚冬冬,等.多策略融合的粒子群优化算法[J].计算机工程与应用,2021,57(1):69-76.
- [9] 孙冬,赵开新.改进粒子群算法在移动机器人路径规划中的应用研究[J].河南工学院学报,2020,28(6):19-23.
- [10] 陈嘉林,魏国亮,田昕.改进粒子群算法的移动机器人平滑路径规划[J].小型微型计算机系统,2019,40(12):2550-2555.
- [11] 吴大飞,杨光永,樊康生,等.多策略融合的改进粒子群优化算法[J].计算机应用研究,2022,39(11):3358-3364.
- [12] 韩仪洒,张莉,谭海燕,等.改进粒子群算法的移动机器人路径规划[J].西安工程大学学报,2019,33(5):517-523.
- [13] 刘艳红,陈田田,张方方.基于改进粒子群算法的移动机器人路径规划[J].郑州大学学报(理学版),2020,52(1):114-119.
- [14] 郭世凯,孙鑫.基于改进粒子群算法的移动机器人路径规划[J].电子测量技术,2019,42(3):54-58.
- [15] 陈跃跃.基于粒子群优化算法的移动机器人运动规划[D].杭州:杭州电子科技大学,2018.
- [16] 吴帅,魏文红,张宇辉,等.基于改进蚁群算法的移动机器人路径规划[J].东莞理工学院学报,2023,30(1):24-34.
- [17] 唐旭晖,辛绍杰.改进蚁群算法的移动机器人路径规划[J].计算机工程与应用,2022,58(5):287-295.
- [18] 欧阳城添,唐风,朱东林.融合禁忌搜索的 SSA 算法及其路径规划的应用[J].电子测量技术,2022,45(22):32-40.

(编辑 赵 蓉)

(上接第 43 页)

- [3] 沈雪瑾,孙浩洋,李雄,等.十字万向节凸度滚针摆动疲劳寿命对比试验[J].汽车技术,2009(7):44-46,61.
- [4] 万诺,史宏斌,屈转利,等.柔性接头疲劳摆动寿命分析研究[J].推进技术,2022,43(5):269-277.
- [5] 李科军,邓旻涯,黄文静,等.混凝土湿喷机摆动系统工作特性研究[J].工程设计学报,2022,29(4):519-526.
- [6] 周坤.基于加速退化试验的 RV 减速器可靠性分析[D].天津:河北工业大学,2022.
- [7] 周坤,叶楠,吴锦辉,等. RV 减速器高应力加速退化试验及可靠性分析[J].哈尔滨工业大学学报,2022,54(7):37-44.
- [8] 张跃明,冀永虎,纪姝婷,等. RV 减速器的寿命计算与加速寿命试验[J].机械设计与制造,2020(8):270-274.
- [9] 黄健,李朝阳,陈兵奎. RV 减速器用新型交错滚子主轴承承载能力分析 with 试验研究[J].机械工程学报,2021,57(7):68-77.
- [10] 刘艳梨,朱永梅.基于 Lagrange 方法的平面双摆机构多体动力学研究[J].机械设计与制造,2009(5):75-77.
- [11] 崔敏其. SCARA 机器人的拉格朗日动力学建模[J].机械设计与制造,2013(12):76-78.
- [12] 李小彭,尚东阳,李凡杰,等.输电线巡检机器人动力学建模与 DME 评价[J].东北大学学报(自然科学版),2020,41(9):1280-1284.
- [13] 黄忠明,陈爱文,黄凤良.基于正弦加减速控制算法的工业机器人位姿轨迹规划研究[J].制造业自动化,2019,41(2):31-34.
- [14] 陈多多,李丽宏,原钢.面向工业机器人的位姿同步加减速算法[J].重庆理工大学学报(自然科学版),2021,35(4):165-173.
- [15] 岳晴晴,林明,林永才.基于 NURBS 算法的工业机器人轨迹规划研究[J].机床与液压,2019,47(9):28-32,71.
- [16] 陈伟华,张铁,崔敏其.基于五次多项式过渡的机器人轨迹规划的研究[J].煤矿机械,2011,32(12):49-50.
- [17] 王斌锐,王涛,李正刚,等.多路径平滑过渡的自适应前瞻位姿插补算法[J].控制与决策,2019,34(6):1211-1218.
- [18] LEI X G. Dynamic simulation analysis of a central swing device[C]//2019 International Conference on Robots & Intelligent System (ICRIS),2019:505-508.
- [19] 黄嫦娥,高中庸,王亚洲,等.基于 ADAMS 的摩擦摆试验装置仿真分析[J].机械设计,2016,33(7):78-81.
- [20] 程曼,邱城,李金峰,等.机器人减速器疲劳寿命测试装置开发与研究[J].机械传动,2019,43(5):156-160.

(编辑 祝智铭)