

一种改进量子行为粒子群优化算法的移动机器人路径规划

刘 洁 赵海芳 周德廉

(宿迁学院机电工程学院 宿迁 223800)

摘 要 为实现移动机器人最优路径规划,提出了一种改进量子行为粒子群的优化算法(LTQPSO)。针对粒子群算法存在过早收敛的问题,利用个体粒子进化速度与群体离散度来动态调整惯性权重,使惯性权重具有自适应性与控制性,从而避免过早收敛;同时将自然选择方法引入传统位置更新公式中,以保持种群的多样性,加强 LTQPSO 算法的全局搜索能力,加快算法的收敛速度;将改进后的 LTQPSO 算法应用于移动机器人路径规划中;最后通过理论仿真与移动机器人平台实验验证了该方法的有效性与可行性。

关键词 路径规划,混合量子行为粒子群优化算法,移动机器人,优化算法

中图分类号 TP615 **文献标识码** A

Improved Quantum Behaved Particle Swarm Optimization Algorithm for Mobile Robot Path Planning

LIU Jie ZHAO Hai-fang ZHOU De-lian

(School of Mechanical and Electrical Engineering, Suqian College, Suqian 223800, China)

Abstract In order to realize the optimal path planning of mobile robot, an improved quantum behaved particle swarm optimization (LTQPSO) algorithm was proposed. Aiming at that the particle swarm algorithm has the problem of premature convergence, the individual particle evolution speed and the group dispersion are used to dynamically adjust the inertia weight, which makes the inertia weight adaptive and controllable, and avoids premature convergence. Meanwhile, the natural selection method was introduced into the traditional location update formula in order to maintain the population diversity, strengthen the search ability of the global QPSO algorithm, and improve the convergence speed of the algorithm. The improved QPSO algorithm was applied to the path planning of mobile robot. Finally, the effectiveness and feasibility of the proposed method was verified by theoretical simulation and experimental results of a mobile robot platform.

Keywords Path planning, Hybrid quantumbehaved particle swarm optimization, Mobile robots, Optimization algorithm

1 前言

寻找一条无碰撞、可行的路径是进行移动机器人路径规划的最终目标。近年来人工神经网络、蚁群算法、进化计算和粒子群优化(PSO)算法等被广泛地应用到路径规划中,对有关移动机器人路径规划的理论研究也逐渐向群体优化算法的方向转变,其中 PSO 算法作为一种新的群体智能优化算法,对目标函数的性质没有要求,易于实现并且拥有良好的优化性能,逐渐成为了当前群体智能优化方法研究的一个热点。基于 PSO 算法的移动机器人路径规划建模简单,算法计算难度低,但是存在搜索性能对参数的依赖性过大、容易陷入局部极小值等问题。针对如何提高量子粒群的全局收敛性以及精确度,许多学者对粒子群算法进行了改进。文献[1]对 PSO 算法进行改进,在多维混合空间下分析粒子的运动轨迹,建立了具有量子运动粒子群的势阱模型,该算法可以在整个可行解空间内进行全局搜索,提高了 PSO 算法的稳定性与收敛性。文献[2]提出一种改进的 CPSO 算法,该算法在 PSO 算法更新公式的过程中采用一种新颖的学习策略和辅助的搜索机制,但此算法的规划结果易受模型以及次优路径的影响,规

划的线路多为折线,对移动机器人的控制十分不利。文献[3]在 PSO 算法的搜索策略中使用 Fletcher-Reeves 方法的局部最优搜索方法来对当前位置的粒子进行更新,此方法可以对不同区域进行解空间的有效搜索,以获得高质量的解。但多群协同 PSO 优化算法存在过早停滞现象,从而致使求解效率严重下降。

粒子群算法中的信息传递主要通过全局极值向全体粒子单向流通,其相对于其他算法更快收敛于最优解,但在过程中存在过早收敛的现象。针对此缺陷,本文提出 LTQPSO 算法,该算法采用个体粒子进化速度与群体离散度来动态调整惯性权重,使惯性权重具有自适应性与控制性,拥有可以有效地提升算法的收敛速度、保持种群多样性、加强算法的全局搜索能力等优点。最后将本文提出的算法应用到移动机器人轨迹规划中,并与 QPSO 常规算法进行对比。实验结果表明该算法的最优路径规划能力比原算法更强。

2 混合改进的量子行为粒子群优化算法

2.1 QPSO 算法

根据粒子群的基本收敛性质,受量子物理基本理论的启

本文受宿迁学院科技计划项目(2016KY26, 2014KY08),宿迁市科技计划项目(H201421),江苏高校品牌专业建设工程资助项目(PPZY 2015C252)资助。

刘 洁(1981—),女,硕士,实验师,主要研究方向为人工智能自动控制, E-mail: 2565923389@qq.com; 赵海芳(1982—),硕士,实验师,主要研究方向为机电仿真分析; 周德廉(1958—),男,硕士,教授,主要研究方向为人工智能。

发, Sun 等人提出了 QPSO (Quantum Behaved Particle Swarm Optimization) 算法。该算法对整个 PSO 算法的搜索策略进行了改进, 其进化方程不需要速度向量且形式简单, 参数更少, 更容易控制。QPSO 算法在搜索能力上优于所有开发的 PSO 算法^[4]。

对 PSO 算法进行转化, 得到的方程表达式如下:

$$X_{i,j}(t+1) = p_{i,j}(t) \pm \alpha \cdot |M_j(t) - X_{i,j}(t)| \cdot \ln\left(\frac{1}{u}\right) \quad (1)$$

$$P_{i,j} = \varphi \cdot Pbest_{i,j}(t) + (1-\varphi) \cdot Gbest_j(t), i=1, 2, \dots, N, \\ j=1, 2, \dots, M \quad (2)$$

$$mbest = (M_1(t), M_2(t), \dots, M_D(t)) \\ = \left\{ \frac{1}{N} \sum_{i=1}^N Pbest_{i,1}(t), \frac{1}{N} \sum_{i=1}^N Pbest_{i,2}(t), \dots, \right. \\ \left. \frac{1}{N} \sum_{i=1}^N Pbest_{i,N}(t) \right\} \quad (3)$$

其中, 第 i 个粒子的 t 时刻位置 $X_i = \{X_{i,1}(t), X_{i,2}(t), \dots, X_{i,D}(t)\}$, 个体的最好位置为 $Pbest_i = \{Pbest_{i,1}(t), Pbest_{i,2}(t), \dots, Pbest_{i,D}(t)\}$, 群体的最优位置为 $Gbest = \{Gbest_1(t), Gbest_2(t), \dots, Gbest_D(t)\}$, 平均最好位置为 $mbest = \{M_1(t), M_2(t), \dots, M_D(t)\}$, 粒子的维度、粒子数和迭代次数分别为 D, N 和 M , $\varphi = \frac{c_1 r_1}{c_1 r_1 + c_2 r_2}$, α 为扩张-压缩因子, c_1 和 c_2 代表学习因子, r_1 和 r_2 代表均匀分布值。

将应用式(2)的粒子群算法统称为量子行为粒子群优化算法。

2.2 混合改进的量子行为粒子群优化算法的原理

在进化算法中, 合理的算法控制参数对算法性能的影响较大, 为了提高 QPSO 算法的收敛性, 对其进化表达式(1)–(3)进行了研究, 根据粒子的吸引点 $P_i(t+1)$, 式(2)可转化为^[5]:

$$P_{i,j}(t+1) = Gbest_{i,j}(t) + \varphi(Pbest_{i,j}(t) - Gbest_{i,j}(t)) \quad (4)$$

由式(1)与式(4)可知, 更新粒子的吸引点 $P_i(t+1)$ 与全局群体最好位置 $Gbest(t)$ 和当前粒子最好位置 $pbest(t)$ 的差存在一定关系, $X_i(t+1)$ 与当前粒子的平均位置 $mbest$ 和自身位置 $X(t)$ 的差也存在关系。目前许多研究人员研究了整个群体平均位置 $mbest$ 在搜索过程中对粒子群体搜索行为的影响, 针对于整个群体平均位置 $mbest$, 提出了许多控制参数的方法。惯性权重作为 QPSO 可调整参数中重要的一环, 惯性权重变大时可以提高算法的全局搜索性能, 惯性权重变小时可以增强算法的局部搜索性能^[6]。但 QPSO 算法的惯性权重值的大小将随着进化代数的变大而线性变小, 若使用这种方法来描述实际非线性、复杂的搜索过程, 算法较容易过早陷入局部优化极值, 从而导致算法的收敛速度也变慢。

本文在算法参数选择上采用个体粒子进化速度和群体离散度来动态调整惯性权重, 使收惯性权重具有自适应性, 从而避免陷入局部最优; 同时将自然选择方法引入最优位置后处理中, 以保持种群多样性与稳定性, 加强 QPSO 算法的全局搜索能力, 提升算法的收敛速度^[7]。

定义 1(个体粒子的进化速度) 假设 $Fitness(Gbest(t))$ 与 $Fitness(P_i(t))$ 分别代表全局最优位置的适应值和当前最优位置的适应值, 那么定义个体粒子的进化速度为:

$$ip_i(t) = \frac{Fitness(Gbest(t))}{Fitness(Pbest_i(t))} \quad (5)$$

在 $0 < ip \leq 1$ 范围内, ip 的值越小, 进化速度将越快。粒子位置与其历史最优位置保持一致, 比较当前适应值与变化

的粒子适应值, 当 ip 的值保持为 1 时, 可判定算法得到了粒子的最优解。

定义 2(群体离散度) 设在粒子的维度中粒子最好的位置标准差为 $\partial_P(t) = \{\partial_{P1}(Pbest_{i,1}(t)), \partial_{P2}(Pbest_{i,2}(t)), \dots, \partial_{PD}(Pbest_{i,D}(t))\}$, 与此同时, 进化过程中的粒子群体离散度为:

$$gs_i(t) = \{gs_{i,1}(t), gs_{i,2}(t), \dots, gs_{i,D}(t)\} \\ = \left\{ \frac{\partial_{P1}(Pbest_{i,1}(t))}{\partial_{X1}(X_{i,1}(t))}, \frac{\partial_{P2}(Pbest_{i,2}(t))}{\partial_{X2}(X_{i,2}(t))}, \dots, \right. \\ \left. \frac{\partial_{PD}(Pbest_{i,D}(t))}{\partial_{XD}(X_{i,D}(t))} \right\} \quad (6)$$

由式(6)可知, gs 可以描述粒子的离散化程度以及种群的多样性。若 gs 的值变大, 则粒子的离散化程度变大, 但粒子种群的多样性将变少。当 $gs=1$ 时, 最好位置 $Pbest$ 与当前位置 X 完全相同, 但是 gs 值的大小仍将继续改变。

使用标准函数对提出的惯性权重进行测试, 选取的测试函数为两个非线性的单峰函数 Sphere 与两个非线性的多峰函数 Rastrigrin, 测试粒子数取 20, 最大迭代次数取 100, 粒子维度为 4; 在 QPSO 算法中, $C_1=1, C_2=2.1$, 扩张-收缩系数 α 依次从 1.0 递减到 0.5。个体粒子进化速度的变化曲线如图 1 所示。

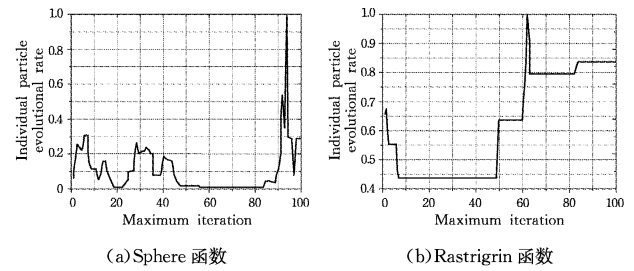


图 1 个体粒子进化速度的变化曲线

从图 1 可以看出, 在最大迭代次数中, 个体粒子进化速度的浮动区间为 $0 \sim 1$, 群体离散化程度在不断震荡后逼近 1, 由此可知惯性权重可以维持种群的稳定性与多样性。选用上述的基于惯性权重对 QPSO 算法进行改进的变换方程, 在个体粒子进化速度 ip 的表达式中增加 $Pbest-Gbest$ 项, 群体离散度 gs 增加 $mbest-X$ 项, 得到改进 QPSO 算法的方程为:

$$P_{i,j}(t+1) = Gbest_{i,j}(t) + \varphi \cdot ip_i(t) \cdot (Pbest_{i,j}(t) - Gbest_{i,j}(t)) \quad (7)$$

$$mbest = (M_1(t), M_2(t), \dots, M_D(t)) \\ = \left\{ \frac{1}{N} \sum_{i=1}^N Pbest_{i,1}(t), \frac{1}{N} \sum_{i=1}^N Pbest_{i,2}(t), \dots, \right. \\ \left. \frac{1}{N} \sum_{i=1}^N Pbest_{i,N}(t) \right\} \quad (8)$$

$$x_{i,j}(t+1) = p_{i,j}(t) \pm \alpha(t) \cdot |mbest - gs_{i,j}(t) \cdot x_{i,j}(t)| \cdot \ln\left[\frac{1}{u(t)}\right] \quad (9)$$

由式(7)和式(9)可知, 个体粒子的速度参数对粒子最好位置和全局群体最优位置的变化影响较大, 在群体离散度调整粒子位置更新过程中, 这种参数为动态控制参数。

为了提高本文算法的准确性与稳定性, 对采用自然选择的算法选择粒子的位置, 在迭代的过程中依次按照函数值从好到差排列粒子, 根据选择规则选择最好的粒子的状态量取代最差的粒子, 最后与变化的粒子群体进行排序, 式(9)对之前的粒子群体进行排序, 将原来每个个体的历史最优解进行保存^[8]。图 2 给出了 LTQPSO 算法的基本流程。

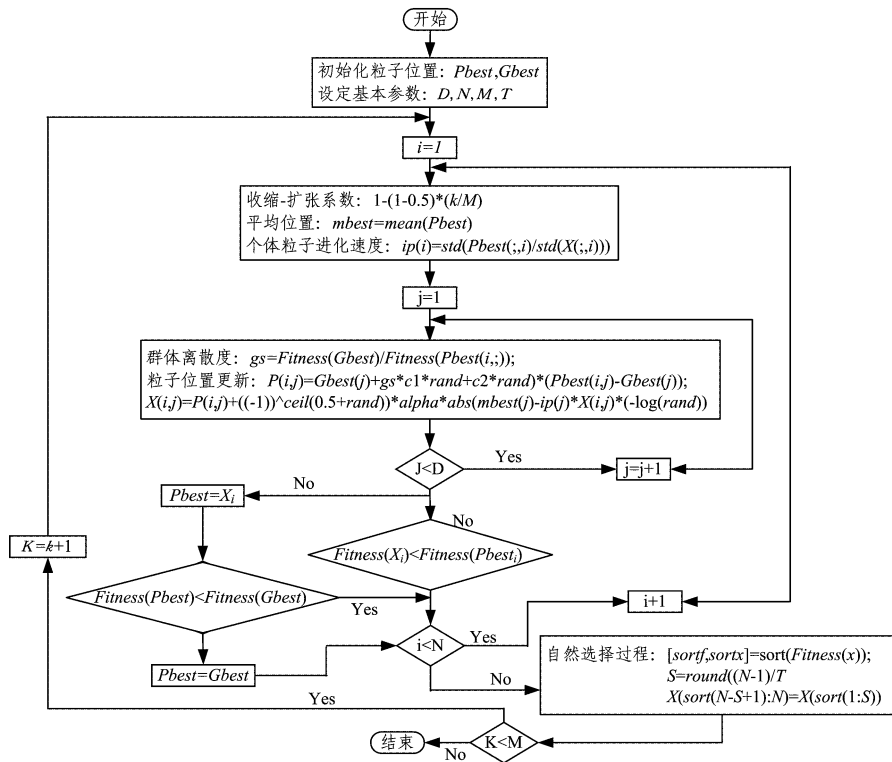


图 2 LTQPSO 算法的流程图

3 LTQPSO 算法的基本参数估计方法研究

对于粒子群算法的路径规划,通常需要进行反复模拟实验,在进化算法中,粒子数、粒子维度以及最大迭代次数是至关重要的 3 个参数^[9]。进行路径规划时,首先确定基本参数,再考虑移动机器人的大小、尺寸与工作环境,建立参数之间关系的估计方法可以实现连接路径规划。本文通过无障碍环境下基于 LTQPSO 算法的收敛可行的参数仿真实验,得到了基本参数的一元线性回归方程。

为了进一步分析粒子维度的变化粒子数与迭代次数之间的关系,使用 Matlab2014a 进行仿真,实验选取的工作环境空间大小为 $10\text{m} \times 10\text{m}$,起始点为 $(0,0)$,终点为 $(10,10)$,粒子维度为 $D(5 \leq D \leq 20)$,学习因子 $c_1=2, c_2=2.1$,扩张-收缩系数 α 依次从 1.0 递减到 0.5。采用 c_1 和 c_2 值的大小固定不变、 α 依次递减的方法,移动机器人在无障碍的环境下利用极坐标与直角坐标的最优路径为 14.14m,移动机器人路径规划方法利用极坐标和直角坐标,因为初始分布对收敛速度有影响,所以将初始分布设定为均匀分布和正态分布 $N(30^\circ, 45^\circ)$ 。

仿真实验的具体方法为:根据粒子维度的排序,依次增加粒子数与迭代数反复进行实验,进行 100 次实验得出最优解的均值,在标准差小于 0.05 时记录最大和最小的粒子数与迭代次数。可行收敛区间可用最大值与最小值表示,对于本

文算法的参数,分别进行两种初始分布的回归分析。图 3 给出了 LTQPSO 在粒子维度变化下平均初始分布得到的 100 次实验结果的最大与最小的粒子数和迭代次数。

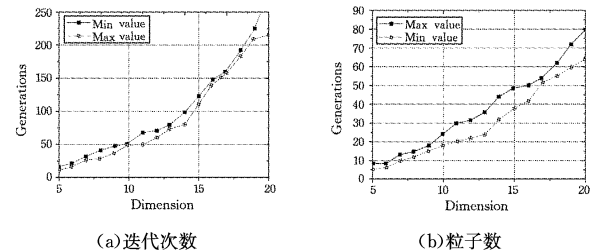


图 3 均匀分布时的仿真实验结果

从图 3 可以看出粒子数与迭代次数的变化特点。其参数变化与指数函数的变化规律相似,因此可以用一元线性回归方程来表示路径规划的基本参数。表 1 列出了利用 Matlab 回归分析“Regress”命令得到的基本参数的回归分析结果。

由表 1 的参数回归分析实验结果可知,LTQPSO 算法的基本参数的最大值的回归变化差优于最小值,基本参数的一元线性回归方程为:

$$\begin{cases} N = \text{round}(0.4321 \cdot D^{1.7292}) \\ M = \text{round}(0.2855 \cdot D^{2.218}) \end{cases} \quad (10)$$

可以根据 LTQPSO 算法的粒子位数变化,由式(10)计算平均初始分布中收敛的粒子数与迭代次数。

表 1 均匀分布时的参数回归分析结果

参数		回归系数		系数区		统计量			
		b_0	b_1	b_0	b_1	R^2	F	P	$\hat{\sigma}$
N	Min	-1.4499	1.8720	(-1.70, -1.19)	(1.76, 1.97)	1	1493.9	0	0
	Max	-0.8392	1.7291	(-1.08, -0.59)	(1.62, 1.82)	1	1366.3	0	0
M	Min	-0.5832	2.0079	(-0.92, -0.24)	(2.06, 2.50)	1	1002.6	0	0
	Max	-1.2537	2.2080	(-1.57, -0.93)	(2.08, 2.33)	1	1374.8	0	0

按照上述方法,使用 LTQPSO 算法对粒子维度在正态分布 $N(30^\circ, 45^\circ)$ 下进行 100 次实验,得到的最大粒子数、最小粒子数与迭代次数如图 4 所示。

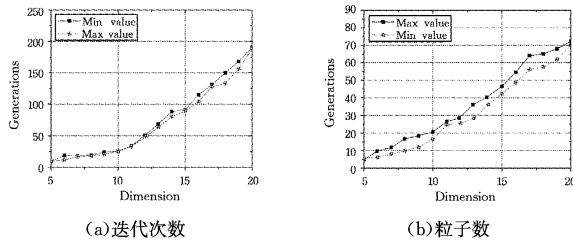


图 4 正态分布 $N(30^\circ, 45^\circ)$ 时的仿真实验结果

表 2 正态分布 $N(30^\circ, 45^\circ)$ 时的参数回归分析结果

参数		回归系数		系数区		统计量			
		b_0	b_1	b_0	b_1	R^2	F	P	$\frac{\Delta}{\sigma}$
N	Min	-1.7085	1.9905	(-2.08, -1.32)	(1.83, 2.14)	0.9823	776.26	0	0.013
	Max	-1.1492	1.8375	(-1.44, -0.85)	(1.72, 1.95)	1	1129.40	0	0
M	Min	-1.3845	2.1799	(-2.02, -0.74)	(1.92, 2.43)	0.9592	329.31	0	0.039
	Max	-1.8788	2.3354	(-2.40, -1.35)	(2.12, 2.54)	0.9757	561.43	0	0.026

综合上述,在本文所提 LQPSO 算法的移动机器人路径规划中,LTQPSO 算法可使用式(10)和式(11)来确定基本参数,因此此回归方程可被用于移动机器人的路径规划。

4 路径规划问题描述与建模

粒子群算法的路径规划中的每个粒子代表一条合理的路径,设有 N 条可行路径,粒子维度 D 代表起始点到终点的连线条数。路径规划的过程可认为是每个路径分割点到最优角度值的收敛过程。文献[10]证明了 QPSO 算法的全局收敛性,基于 QPSO 算法的路径规划过程实质就是马尔科夫链的过程,每条实时路径状态与前一时刻的路径选择有关,而与其他状态无关。

(1)依据移动机器人的工作环境,本文采用栅格法进行环境建模。栅格法采用尺寸相同的栅格对机器人的二维工作空间进行划分,栅格大小以机器人能在其中自由运动为限。若某一栅格的尺寸范围内不含任何障碍物,则称此栅格为自由栅格;反之,称其为障碍栅格。自由空间和障碍物均可表示成栅格块的集合。栅格粒度是根据地图中的障碍物疏密程度来决定的,在得到结果后考虑机器人自身的尺寸,从而得到最终的栅格粒度。

采用栅格法描述的环境条件可用极坐标和直角坐标表示,极坐标的参数是长度和角度,直角坐标的参数是长度,两个坐标结合表示路径规划的数值。基于极坐标与直角坐标关系的探测方法如图 5 所示。极坐标的长度表示从出发点 to 目标点的距离,角度表示探测范围。

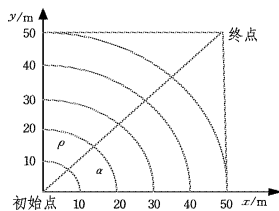


图 5 基于极坐标和直角坐标的路径探测原理

(2)设置相关参数:粒子维度为 D ,最大迭代次数为 M ,粒子数为 N ,学习因子为 C_1 与 C_2 ,扩张-收缩系数为 α ,则粒子群维度为:

从图 4 可以看出,粒子数与迭代次数的变化特点。参数变化与指数函数的变化规律也相似,因此可以用一元线性回归方程来表示路径规划的基本参数。表 2 列出了利用 Matlab 回归分析“Regress”命令得到的基本参数的回归分析结果。

由表 2 的参数回归分析实验结果可知,LTQPSO 算法的基本参数的最大值的回归变化差优于最小值,基本参数的一元线性回归方程为:

$$\begin{cases} N = \text{round}(0.3161 \cdot D^{1.8376}) \\ M = \text{round}(0.1525 \cdot D^{2.3358}) \end{cases} \quad (11)$$

$$D \approx \frac{\text{distance}(\text{path})}{\text{length}_{\text{robot}}} \quad (12)$$

其中, $\text{distance}(\text{path})$ 表示起始点到终点的距离, $\text{length}_{\text{robot}}$ 表示移动机器人的长度。

(3)极坐标的长度取决于初始点与终点的距离,探测角度的区间为 $[0, \pi/2]$,表达式如下:

$$\begin{cases} \alpha_{\max} = \alpha_{\text{top}}, \alpha_{\min} = \alpha_{\text{down}}, \rho L \leq L_{\text{target}} \\ \alpha_{\max} = \alpha_{\text{top}}, \alpha_{\min} = \alpha_{\text{down}}, \rho L > L_{\text{target}} \end{cases} \quad (13)$$

其中, $\beta = \arccos(L_{\text{target}} / \rho)$, α_{top} 和 α_{down} 分别代表探测的最大与最小角度范围, L_{target} 为目标在直角坐标的目标点位置, ρ_L 为极坐标的长度,且满足:

$$\rho_L = \frac{\sqrt{(x_j - x_0)^2 + (y_j - y_0)^2}}{D}, j = 1, 2, \dots, D \quad (14)$$

(4)确定粒子群初始化的分布为均匀分布,粒子的位置与速度在环境模型的有效区域 $\alpha_{i,j}$ 为:

$$\alpha_{i,j} = \text{rand} * (\alpha_{\text{top}} - \alpha_{\text{down}}) + \alpha_{\text{down}} \quad (15)$$

(5)根据直角坐标与极坐标的变换来确定路径坐标值的大小,按照粒子约束条件判断粒子是否有效,若无效则再一次进行初始化,直到确保全部粒子有效为止,同时确定粒子与障碍的位置。

(6)通过适应度函数计算每个粒子的适应度值,将粒子个体的历史最优值与当前粒子适应值进行对比,并选择较小的值,适应度函数如下:

$$\text{Fitness}(\text{path}) = f_1 L_D + f_2 L_s \quad (16)$$

$$L_D = \sum \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2} \quad (17)$$

$$L_s = \sum \sqrt{(\rho_{i+1}^2 + \rho_i^2 - 2\rho_{i+1}\rho_i \cos(\alpha_{i+1} - \alpha_i))} \quad (18)$$

其中, f_1 和 f_2 分别代表路径长度和平滑度函数中的加权函数, L_D 表示路径长度, x_i 与 y_i 表示路径的初始点与终点, L_s 表示粒子平滑度函数, ρ_i 为极坐标长度, α 代表粒子角度。

(7)将粒子历史最优适应值序列中最小的值与当前的全局历史最优值进行比较,通过计算迭代得到的 $P_g(t)$ 与全局最好位置 $G(t-1)$ 的适应度值来获得粒子群的最好位置,表达式如下:

$$c_j(t) = \frac{1}{M} \sum_{i=1}^M P_{i,j}(t) \quad (19)$$

(8)对粒子种群的位置信息进行更新,当粒子搜索位置超

出设定区域时,选取设定的区域作为位置信息,粒子更新的方程为:

$$P_{i,j} = \varphi_j(t) \cdot P_{i,j}(t) + [1 - \varphi_j(t)] \cdot G_j(t) \quad (20)$$

$$\varphi_j(t) \sim U(0,1) \quad (21)$$

$$\varphi_j(t) = \frac{c_1 \cdot rand_1(t)}{c_1 \cdot rand_1(t) + c_2 \cdot rand_2(t)} \quad (22)$$

粒子的位置更新表达式为:

$$X_{i,j}(t+1) = P_{i,j}(t) \pm \alpha |C_j(t) - X_{i,j}(t)| \ln \left[\frac{1}{u_{i,j}(t)} \right] \quad (23)$$

(9)将得到的最新迭代粒子最优值与历史连续迭代结果进行比较,若未发生变化则认为变量已经达到最优值,停止循环;否则,在最大迭代数内反复计算直到达到最大迭代数,采用 Shuman 滤波法进行平滑处理,对粒子路径进行优化,从而得到计算结果与最优路径。

5 移动机器人路径规划仿真与实验验证

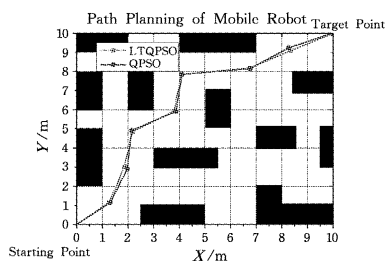
5.1 路径规划仿真

将本文提出的 LTQPSO 算法运用于移动机器人的路径规划,编程环境为 Matlab R2014a,仿真实验的实验空间为 $10\text{m} \times 10\text{m}$,移动机器人的大小为 $0.3\text{m} \times 0.4\text{m}$,初始点设为 $(0,0)$,正态分布为 $N(30^\circ, 45^\circ)$,粒子维度设为 $D=8, N=30, M=50$,由式(9)和式(10)决定 LTQPSO 算法的粒子数和迭代次数。最大线速度 $V=0.24\text{m/s}$,角速度 $W=0.2\text{rand/s}$,控制周期为 0.4s 。

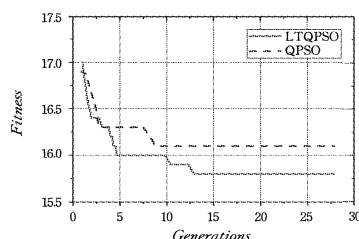
为了比较 QPSO 算与 LTQPSO 算法,本文设置了同样的基本参数,将每种算法在每个初始分布条件下运行 50 次,分别记录每次实验得到的最优路径的代价值,然后求出这 50 个最优路径代价值的平均代价、标准差和平均耗时。均分布和正态分布 $N(45^\circ, 30^\circ)$ 时的路径规划仿真结果分别如表 3、表 4 和图 6、图 7 所示。

表 3 均匀分布时的路径规划仿真结果

D	优化算法	平均代价/m	标准差/m	平均耗时/s
8	QPSO	16.4604	3.2653	4.2507
	LTQPSO	15.7631	1.0213	2.5283



(a) 路径规划结果

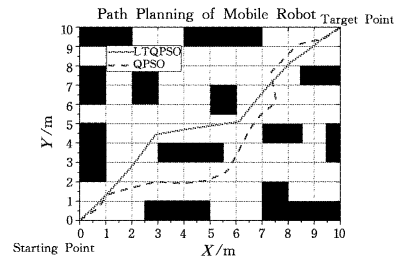


(b) 最优路径代价值收敛曲线

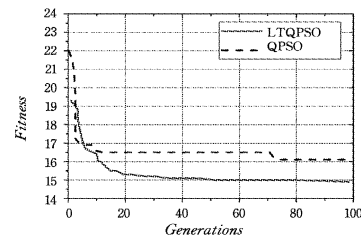
图 6 均匀分布($D=8$)时两种优化算法的路径规划仿真结果

表 4 正态分布 $N(45^\circ, 30^\circ)$ 的路径规划仿真结果

D	优化算法	平均代价/m	标准差/m	平均耗时/s
16	QPSO	16.0554	5.6474	2.8812
	LTQPSO	15.8156	1.1982	1.6082



(a) 路径规划结果



(b) 最优路径代价值收敛曲线

图 7 正态分布 $N(45^\circ, 30^\circ)$ ($D=16$) 时两种优化算法的路径规划仿真结果

从表 4、图 6 以及图 7 可以看出,与 QPSO 算法相比,LTQPSO 算法在正态分布 $N(45^\circ, 30^\circ)$ 时两种粒子维度的优势依旧较大。对比表 3、图 6 可知,在均匀分布 $D=8$ 时两种算法生成的路径相似,但在正态分布 $N(45^\circ, 30^\circ)$ $D=16$ 时生成的路径各不相同。相比而言,QPSO 算法不能求得最优路径解,计算得到的 QPSO 算法的参数值比 LTQPSO 算法小 $1.0771 \sim 1.963$ 倍($N=83, M=418$),QPSO 算法不能够收敛到最优路径解。LTQPSO 算法的最优路径的平均代价值、标准差和平均计算时间均小于 QPSO 算法,因此 LTQPSO 算法具有较好的稳定性和收敛速度。

仿真结果表明,基于 LTQPSO 算法的机器人路径规划方法具有更强的搜索能力、更快的收敛速度以及更好的稳定性,而且确定了 LTQPSO 算法基本参数的评估方程能够有效地解决路径规划问题。

5.2 移动机器人平台的实验

为了进一步验证本文提出的路径规划算法的有效性,选取两轮独立三轮差速的移动机器人进行实验。移动机器人机构如图 8 所示。差速驱动轮式移动机器人平台的机构特性值如表 5 所列,该平台的两个主动轮在前,从动轮在后。采用该平台进行 100 次实验,移动机器人轨迹规划和轨迹跟踪的实验结果如图 9 所示。

表 5 差速驱动轮式移动机器人平台的机构特性值

项目	参数
机器人的质量/kg	2.3
机器人一个驱动轮的质量/kg	0.28
机器人质心到驱动轮几何中心的距离/m	0.065
机器人转到惯量/ $\text{kg} \cdot \text{m}^2$	0.1
驱动轮转动惯量/ $\text{kg} \cdot \text{m}^2$	0.0022
驱动轮间距/m	0.3
驱动轮半径/m	0.068
电机周转脉冲数	16
控制芯片倍频	2
编码器脉冲计数周期/s	0.005
减速器减速比	131
机器人通过串口向上位机发送脉冲编码的时间间隔/s	0.3

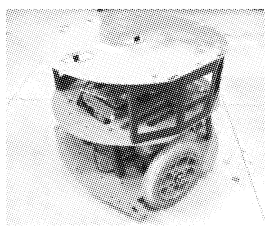


图8 差速驱动移动机器人平台的机构

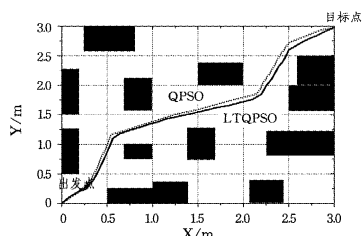


图9 LTPQSO算法与PQSO算法的最优路径对比

(1) 最优路径长度的比较

从图9可以看出,LTPQSO算法搜索的路径优于PQSO算法。LTPQSO的路径长度为4.0252m,而PQSO规划的路径长度为4.3812m。PQSO算法在搜索的过程中陷入局部最优,而LTPQSO算法规划选择的路径是全局最优的。

(2) 收敛速度的比较

从图10、图11可以看出,PQSO算法与LTPQSO算法的收敛速度曲线均在149次左右就达到了局部的最优值,在接下来的迭代次数中LTPQSO算法大概在200次时跳出局部最优值,又经过50次左右的迭代搜索到了全局的最优值,而PQSO算法仍然陷入局部的最优值。显而易见,LTPQSO算法跳出局部最优解的能力强于PQSO算法。

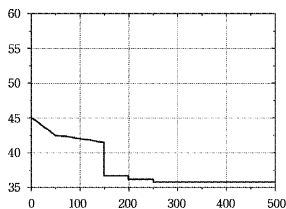


图10 LTPQSO的收敛过程

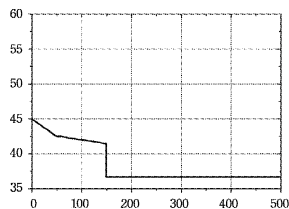


图11 PQSO的收敛过程

结束语 本文提出了一种改进的LTQPSO算法。将LTQPSO算法应用于移动机器人的路径规划中,初始分布取均匀分布和正态分布时,随着粒子维度的变化,分析了LTQPSO算法的粒子数与迭代次数的关系,提出了一元线性回归基本参数评估方程。在不同粒子维度环境下比较了LTQPSO算法与QPSO算法的性能。仿真实验结果表明,与QPSO算法的路径规划方法相比,基于LTQPSO算法和基本参数评估方程的路径规划方法具有更强的搜索能力、更快的收敛速度以及更好的稳定性。

参考文献

- [1] CLERC M, KENNEDY J. The Particle Swarm-Explosion, Stability, and Convergence in a Multidimensional Complex Space[J]. IEEE Transaction on Evolutionary Computation, 2002, 6(1): 58-73.
- [2] TAN G Z, BAO K, RICHARD M R. A composite particle swarm algorithm for global optimization of multimodal functions[J]. Journal of Central South University, 2014, 21(5): 1871-1880.
- [3] LUNG X L, LI W F, ZHANG Y, et al. An adaptive particle swarm optimization method based on clustering[J]. Soft Comput, 2014, 14: 1262-1280.
- [4] SHIKHA A, SANJAY S. FRPSO: Fletcher-Reeves based particle swarm optimization for multimodal function optimization[J]. Soft Computing, 2014, 18: 2227-2243.
- [5] 周游. 基于粒子群优化的动态优化研究[D]. 杭州: 浙江大学, 2014.
- [6] 徐小俊, 雷秀娟, 郭玲. 基于SWGPSO算法的多序列比对[J]. 计算机工程, 2011(6): 184-186.
- [7] 黄仁全, 靳聪, 等. 自适应局部增强微分进化改进算法[J]. 空军工程大学学报(自然科学版), 2011(3): 84-89.
- [8] 武鲁晓. 改进粒子群算法在电力系统无功优化中的应用[D]. 济南: 山东大学, 2012.
- [9] 李仁府, 独孤明哲, 胡麟. 基于PSO算法的路径规划收敛性与参数分析[J]. 华中科技大学学报(自然科学版), 2013(1): 271-275.
- [10] 方伟, 孙俊, 谢振平, 等. 量子粒子群优化算法的收敛性分析与控制参数研究[J]. 物理学报, 2010, 59(6): 3686-3694.
- [11] (上接第105页)
- [9] 桂诗春. 潜语义分析的理论及其应用[J]. 现代外语, 2003(1): 76-84.
- [10] HOFMANN, THOMAS. Unsupervised Learning by Probabilistic Latent Semantic Analysis[J]. Machine Learning, 2001, 42(1/2): 177-196.
- [11] BLEI D M, NG A Y, JORDAN M I. Latent Dirichlet Allocation[J]. The Journal of Machine Learning Research, 2003, 3: 993-1022.
- [12] TEH Y W, JORDAN M I, BEAL M J, et al. Hierarchical Dirichlet Processes[J]. Journal of the American Statistical Association, 2006, 101(476): 1566-1581.
- [13] MEI Q, ZHAI C. A note on EM algorithm for probabilistic latent semantic analysis[C]// Proceedings of the International Conference on Information and Knowledge Management, CIKM, 2001.
- [14] PORTEOUS I, NEWMAN D, IHLER A, et al. Fast collapsed gibbs sampling for latent dirichlet allocation[C]// Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2008.
- [15] KE X H, LUO H J. Using LSA and PLSA for Text Quality Analysis[C]// International Conference on Electronic Science and Automation Control. Atlantis Press, 2015.
- [16] 曹娟, 张勇东, 李锦涛, 等. 一种基于密度的自适应最优LDA模型选择方法[J]. 计算机学报, 2008, 31(10).
- [17] 哈工大社会计算与信息检索研究中心-同义词词林[EB/OL]. <http://www.ltp-cloud.com/download>.
- [18] SHERMIS M D, BURSTEIN J C. Automated essay scoring: A cross-disciplinary perspective[M]. Routledge, 2003.