

改进 PSO-PH-RRT* 算法在智能车 路径规划中的应用

蒋启龙, 许健

(西南交通大学 电气工程学院, 四川 成都 611756)

摘要: 在机器人控制、智能车自主导航等应用场景中, 路径规划需要考虑到环境中的障碍物、地形等因素. 针对路径规划中快速拓展随机树(RRT)算法拓展目标方向盲目、效率较低的问题, 提出了基于粒子群算法优化的均匀概率快速拓展随机树(PSO-PH-RRT*)算法. 该算法在基于均匀概率的快速拓展随机树(PH-RRT*)算法的基础上, 利用粒子群算法更新方向概率作为随机树节点的速度方向, 从而改善了节点的位置更新策略, 并将节点到目标向量的距离和轨迹平滑度作为粒子群算法的适应度函数. 最后在多种障碍环境下进行仿真. 结果表明, PSO-PH-RRT*算法能大大减少迭代时间成本, 同时改善路径长度和平滑度.

关键词: 路径规划; RRT算法; 改进粒子群优化算法; 目标向量; 代价函数; 适应度函数

中图分类号: TP 242.6 文献标志码: A 文章编号: 1005-3026(2025)03-0012-08

Application of Improved PSO-PH-RRT* Algorithm in Intelligent Vehicle Path Planning

JIANG Qi-long, XU Jian

(School of Electrical Engineering, Southwest Jiaotong University, Chengdu 611756, China. Corresponding author: JIANG Qi-long, E-mail: double_long@126.com)

Abstract: In application scenarios like robot control and autonomous navigation of intelligent vehicle, path planning needs to account for factors including obstacles and terrain. To address the issues of directionless expansion target and low efficiency in rapidly-exploring random tree (RRT) algorithm in path planning, a particle swarm optimization for probabilistically homogeneous rapidly-exploring random tree (PSO-PH-RRT*) algorithm is proposed. This algorithm base on the probabilistically homogeneous rapidly-exploring random tree (PH-RRT*) algorithm by using the particle swarm optimization algorithm to update the probability of direction as the velocity direction for random tree nodes, thereby improving the node position update strategy. It also uses the distance between the node and the target vector, along with trajectory smoothness, as the fitness function in the particle swarm optimization algorithm. Finally, simulations across various scenarios demonstrate that the PSO-PH-RRT* algorithm can significantly reduce iteration time costs while improving path length and smoothness.

Key words: path planning; RRT algorithm; improved particle swarm optimization algorithm; target vector; cost function; fitness function

机器人的普及应用加快了工业生产的自动化水平, 随之而来的诸多技术问题掀起了研究热潮. 移动机器人最基本的一项工作任务——路径规划, 旨在为机器人规划一个无障碍的从起始点到目标点的可行路径, 目前已有诸多成熟算法可

用于解决此类问题. 例如: Dijkstra算法能在栅格图中找到一个经过所有目标点的最短路径^[1]; 改进生物神经网络等智能方法实现了对复杂空间的最短路径规划^[2]; 基于粒子群算法或遗传算法的运动时间最优算法^[3]. 单独的启发式算法显著

的缺点是需要生成较大的节点样本来获得最佳路径,因此迭代时产生节点的随机性以及迭代次数的增加都会降低算法的效率,对计算机内存有较高要求。

在复杂动力学环境下高质量运动轨迹的研究中,快速拓展随机树(RRT)算法因搜索效率和低显存占有度得到了广泛研究.RRT是由La等^[4]提出的新型避障算法,这种随机搜索算法收敛速度相对较慢.为了减少规划路径的弯曲程度,文献[5]提出了一种试探性的RRT搜索偏置策略,该方法虽然改善了路径的平滑度但成倍增加了搜索代价.为了解决RRT算法的目标导向问题,文献[6]提出了一种变步长的树干快速搜索随机树算法,通过转化RRT算法可搜索空间中的随机点并能根据目标位置调整搜索步长.为了解决RRT随机生成子节点的盲目性和低效率的问题,产生了诸如双向快速扩展随机树(RRT-connect)^[7]、改进快速扩展随机树(RRT*)^[8]、多向快速扩展随机树(Bi-RRT*)^[9]、智能快速扩展随机树(RRT*-smart)^[10]、同步递归快速拓展随机树(SRRT*)^[11]等多种变式算法.这些方法主要解决RRT算法新生成节点较为随机的问題,提高了收敛速率,同时也存在着一些问题.例如:RRT*算法虽然通过重新布线能优化路径长度,但所带来的时间成本成倍增加;Bi-RRT*算法虽然能够双向搜索产生新节点,在运行时间和搜索效率上优于经典RRT算法,但存在路径不平滑、转向角过大、稳定性差等问题。

粒子群(PSO)算法是一种基于种群的进化算法,粒子活动范围由社会行为模拟而来,且每个候选解都与速度相关联.文献[12]提出基于PSO算法的平滑路径规划,将图搜索(A*)算法得到的路径作为PSO算法的先验启发信息.针对传统PSO算法搜索效率较低的问题,文献[13]提出了一种基于目标粒子群算法的智能车路径规划方法.首先由智能车进行环境建模,然后引入相应准则,使算法靠近目标并达到较快的收敛速度.为了提高机器人路径规划系统的灵活性和适应性,文献[14]提出了混合粒子群-模拟退火(PSO-SA)算法对智能车系统进行路径优化,虽然此方法减少了陷入局部最优的可能,但是其适用范围局限于规则分格障碍.为增强PSO算法规划路径的适应能力,Krell等^[15]提出了基于PSO算法的无障碍环境下自主导航方法.这些方案仍难以解决此类启发式算法迭代中变量利用率较低

的问题。

针对以上问题,本文在原有目标均匀概率的快速拓展随机树(PH-RRT*)算法^[16-17]基础上通过随机树生成避障路径引入粒子群算法进行迭代寻找最优结果,主要目的是克服单次RRT算法搜索结果的随机性和偶然性,同时解决PSO算法容易陷入局部最优的问题.初次迭代会利用PH-RRT*算法初步生成一条可行轨迹,在后续迭代过程中,用粒子代替随机树中的节点,用轨迹平滑度作为PSO算法的适应度函数,将均匀概率参数值 P 作为粒子速度方向更新值对初次迭代产生的所有粒子进行迭代和位置重构,并判断寻找得到的最优轨迹,适应度函数值不再发生明显变化时停止迭代,认为此时算法产生了最优解.最后在多种二维空间障碍下进行仿真,通过比较迭代时间、路径长度等指标,将PSO-PH-RRT*算法与RRT算法的结果进行对比,同时对算法的鲁棒性和稳定程度进行了研究。

1 RRT路径规划算法

1.1 问题描述

本节将给出智能车路径规划所需的相关符号定义和物理概念.令 $M \in \mathbb{R}^N$ 代表整个规划问题的布局空间, d 为布局空间的维度($d \geq 2$), M_o 为布局空间的障碍区域, M_f 为布局空间中除了障碍以外的其他区域, M_s 为起始点, $M_s \in M_f$, M_g 为在非障碍区域的目标点, $M_g \in M_f$.在起始点和目标点之间通过生成一个随机树 $T(T \in M_f)$ 进行连接,路径规划的目标是在避开障碍的情况下尽可能使生成的路径长度、搜索时间和平滑度的相对最优,即保证PSO-PH-RRT*算法的适应度函数 F 最小。

$$F = \operatorname{argmin} \left\{ \sum_{k=1}^{S(T)} f(M_k) \mid T(1) = M_s, T(g) = M_g, \forall k \in [1, g], T(k) \in M_f \right\}. \quad (1)$$

式中: $S(T)$ 为随机树包含的节点数; $f(M_k)$ 为随机树的适应度函数; M_k 为布局空间的第 k 个节点; $T(1)$ 为随机树起始点; $T(g)$ 为随机树的目标点; g 为目标点在随机树所有节点的序号; $T(k)$ 为随机树的第 k 个节点。

1.2 RRT算法介绍

RRT算法广泛应用于智能车的路径规划,其基于全局统一策略,不需要构建机器人环境模

型,将传统的障碍物信息映射步骤省略,可以节约计算成本。

RRT算法中更新子节点的单位步长 s 为

$$s = \frac{\rho}{\|M_n - M_r\|_2}. \quad (2)$$

式中: ρ 为随机树增加步长;一般取3~5; M_n 为临近点; M_r 为随机点。

新节点 M_c 为

$$M_c = M_n + s(M_r - M_n). \quad (3)$$

每个随机树上的节点信息除了包含节点坐标外,还需要用代价函数衡量其本身的优越程度,其定义方式为用 D_i 记录包括新生成节点在内的目前已生成的所有节点到其父节点的距离,用以表示新生成节点所需要的距离和时间成本。

$$D_i = \|M_c - M_n\|_2 + \sum D_j, j = 1, 2, \dots, i-1. \quad (4)$$

式中, i 为随机树节点的序号。

在RRT算法中,采样点与临近点相连得到最终路径,这样既无法保证路径最短也有很大随机性.因此如图1所示的RRT*算法在临近点加入随

机树以后,在代价函数小于1个补偿范围 r 内的节点中判断是否有代价函数更小的父节点,这样更新父节点的操作会使整个路径的代价函数保持相对最优。

$$M_n = T_i(x_1, y_1, 0, \min C, p_i), \|T_i - M_c\| < r. \quad (5)$$

式中: T_i 为随机树上的第 i 个节点; x_1 为新采样点的横坐标; y_1 为新采样点的纵坐标; $\min C$ 为当前最小代价函数; p_i 为该节点的父节点的序号。

RRT*拓展随机树的流程如图2所示.父节点重布线的流程如图3所示。

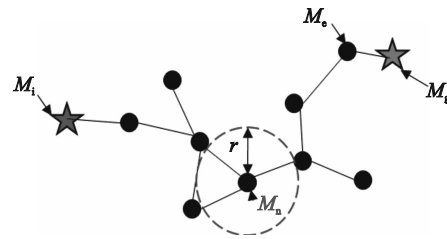


图1 RRT*算法节点拓展示意图

Fig. 1 Node expansion schematic diagram of RRT* algorithm

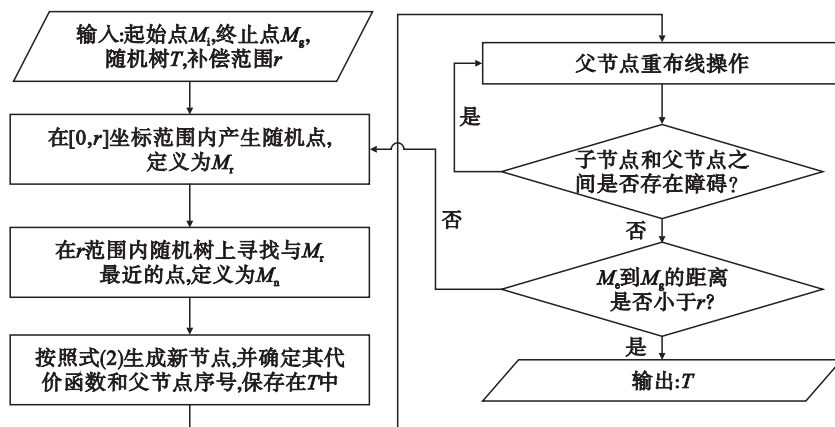


图2 RRT*算法拓展随机树流程图

Fig. 2 Expanded random tree flowchart of RRT* algorithm

针对RRT*算法中新节点拓展过于盲目的问题,提出了基于均匀概率的RRT*(PH-RRT*)算法,新节点 M_c 的生成方式为

$$M_c = \begin{cases} M_n + \frac{(M_r - M_n)\rho}{\|M_r - M_n\|_2}, n_t \leq P; \\ M_n + \frac{(M_g - M_n)\rho}{\|M_g - M_n\|_2}, P < n_t. \end{cases} \quad (6)$$

式中: P 为均匀概率参数值,取值范围是 $[0, 1]$; n_t 为判断值。

如图4所示,根据判断值 n_t 与 t 的大小决定新节点拓展方法.若 $n_t \in [0, P]$ 则拓展策略与式(2)一致,否则将考虑沿着目标方向向量拓展。

2 PSO-PH-RRT*算法改进

2.1 参数定义

考虑到RRT算法生成节点的随机性,这与粒子群算法生成随机粒子再寻找最优适应度相似,因此考虑用PSO算法更新RRT算法的节点位置.在二维空间中,第 u 个粒子为 x_u ,粒子更新速度方向记为 P_u ,即PH-RRT*算法中的均匀概率参数值.初始迭代为RRT*算法在二维环境下生成的随机点(包括连接起始点和目标点的随机树);在之后的迭代中这些节点(粒子)通过跟踪适应度

函数,即 RRT*算法中的代价函数极值来更新 P 的初始值,其规则为

$$P_h = w_h P_{h-1} + c_h (g_b - x_h). \quad (7)$$

$$w_h = \begin{cases} w_n + (w_a - w_n) \frac{f(x_h) - f_n}{f_a - f_n}, & f(x_h) \leq f_a; \\ w_a, & f_a < f(x_h). \end{cases} \quad (8)$$

式中: w_h 为第 h 代的惯性因子; w_n 为最小惯性因子; w_a 为最大惯性因子; 每一维粒子的均匀概率都会被限制在 $[0.2, 0.9]$ 范围内; c_h 为第 h 代的学习因子; g_b 为历次迭代中最优适应度函数情况; $f(x_h)$ 为第 h 代的适应度函数; f_n 为最小适应度函数; f_a 为最大适应度函数。

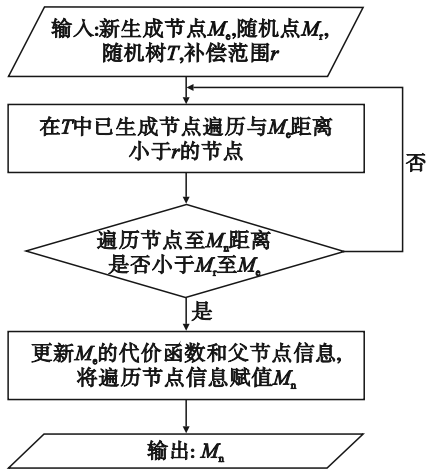


图 3 RRT*算法重布线流程图

Fig. 3 Rewire flowchart of RRT* algorithm

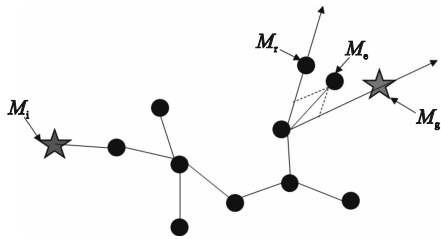


图 4 PH-RRT*算法节点拓展示意图

Fig. 4 Node expansion schematic diagram of PH-RRT* algorithm

在每次迭代过程中,考虑到搜索目标和范围的变化情况,方向概率初始值 P_0 的变化规律为

$$P_0 = \begin{cases} P_0 - \frac{0.05 \times a(T, 1)}{80}, & 0.2 < P_0; \\ P_0, & P_0 \leq 0.2. \end{cases} \quad (9)$$

式中: $a(T, 1)$ 为本次迭代新生成的粒子数。

2.2 适应度函数定义

适应度函数即 RRT 算法的代价函数,本文考虑最短距离和平滑度 2 种情况. 本文 RRT 算法原

代价函数在物理意义上表征累计迭代次数,并不能表示新生成节点距离目标方向的偏离程度. 若用 $\overline{M_i M_g}$ 表示随机树拓展的方向向量,代价函数 E 的定义方法如图 5 所示.

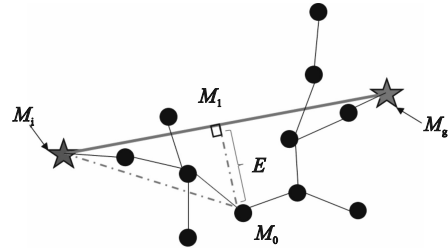


图 5 代价函数定义

Fig. 5 Definition of cost function

$\overline{M_i M_0}$ 向量在目标向量上投影的单位长度 m 为

$$m = \frac{\overline{M_i M_g} \cdot \overline{M_i M_0}}{\|M_i - M_g\|_2}. \quad (10)$$

式中, M_0 为任意选取的一个节点。

以 M_i 为起点, $\|M_i - M_g\|_2$ 个单位长度得到新拓展节点 M_1 为

$$M_1 = M_i + m(M_g - M_i). \quad (11)$$

新定义的代价函数 E' 为

$$E' = \|M_1 - M_0\|_2. \quad (12)$$

每个新节点的代价函数包括节点本身和其父节点的代价函数之和为

$$E_i = E' + E_{i-1}. \quad (13)$$

2.3 新生成粒子的约束条件定义

若按照 PH-RRT*算法的节点更新规则更新粒子的位置,在实验仿真中易出现新节点方向与目标点方向相反的情况,极大影响了粒子的利用率. 因此本文考虑加入方向约束角 γ 作为约束条件,其定义如图 6 所示.

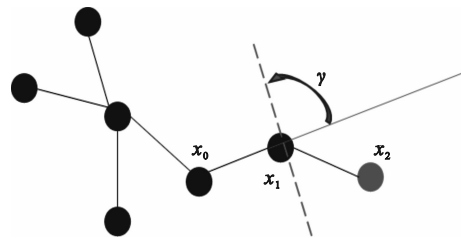


图 6 方向约束角 γ 定义

Fig. 6 Definition of direction constraint angle γ

假设随机树中相互连接的两个节点和新节点分别为 x_0, x_1, x_2 ,若最大限度保证新生成节点是有效节点则需满足式(14).

$$\arccos \frac{\|x_0-x_1\|_2^2 + \|x_1-x_2\|_2^2 - \|x_0-x_2\|_2^2}{2 \times \|x_0-x_1\|_2 \times \|x_1-x_2\|_2} \leq 90^\circ. \quad (14)$$

粒子生成方式更新为 RRT 拓展随机节点的模式, 将粒子速度更新为 P 的大小, 目的是在保证步长一定的情况下自适应调整粒子拓展方向. 算法流程如图 7 所示.

2.4 具体流程及实施算例分析

PSO-PH-RRT*算法在 PSO 算法的基础上将

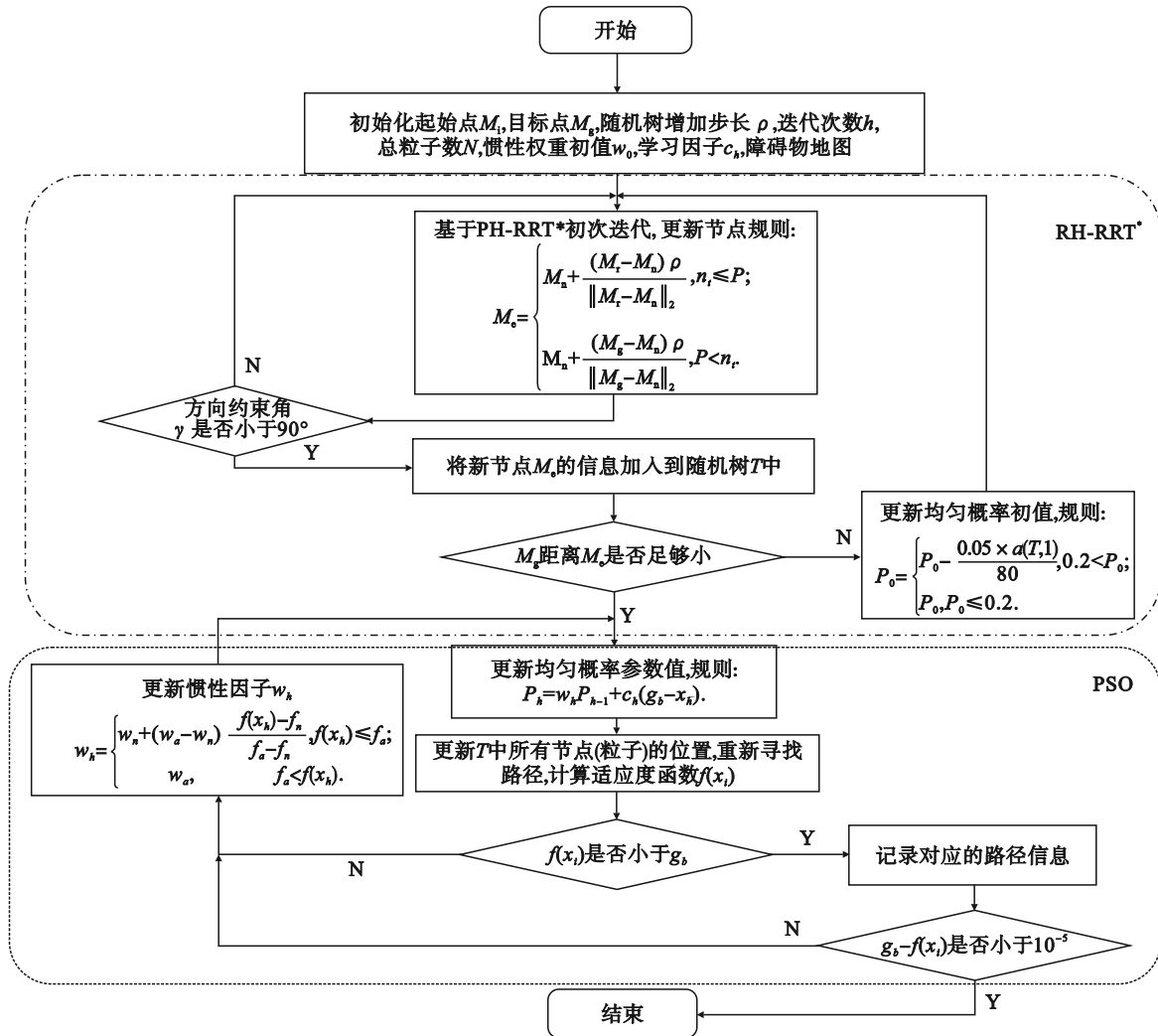


图7 PSO-PH-RRT*算法流程

Fig. 7 PSO-PH-RRT* algorithm flowchart

为验证其合理性, 在 500×500 且拥有 5 个随机障碍点的环境中进行测试. 测试实验中, 以平滑度指标作为适应度函数, 迭代过程中适应度函数值变化情况如图 8 所示.

3 实验及分析

为了验证上述改进策略的效果和效率, 每种算法都在典型环境中进行了实验, 其内容为任务主体需要在障碍物地图智能采样区域中规划从起始点到目标点的可行路径, 并记录搜索时间、平滑度等指标的数据.

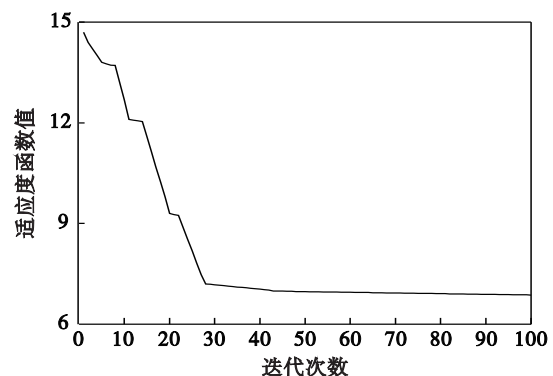


图8 PSO-PH-RRT*算法适应度函数值变化曲线
Fig. 8 PSO-PH-RRT* algorithm fitness function value change curve

3.1 实验条件

为验证本文提出方法的鲁棒性和有效性,在硬件环境为 Intel Core i78550U CPU, 8 GB, Windows 11, 64 位操作系统、软件为 MATLAB R2023a 进行仿真实验. 主要针对原始 RRT*算法在代价函数(PSO 算法的适应度函数)方面优化, 产生了考虑平滑度和节点至目标向量距离 2 种策略(RRT*(IC), RRT*(E)); 在迭代算法搜索过程中考虑是否加入粒子群算法, 产生了 PH-RRT*,

PSO-PH-RRT*2 种策略, 并根据采用代价函数的不同分成 4 种策略.

各种实验方法的名称如表 1 所示, 实验在迷宫、单路径和不同形状的随机障碍 5 种环境中开展, 采用简化后的全局栅格地图, 图片像素即环境规模为 500×500 , 起始点坐标为 (450, 100), 终止点坐标为 (100, 450), 验证所提方法的通用性. 根据算法实验中的收敛情况, 粒子群算法的迭代次数选取为 50 次, 其他参数的设置如表 2 所示.

表 1 优化算法与代价函数对比实验方案
Table 1 Optimization algorithm and cost function comparative experiment plans

算法	步长代价函数	距离代价函数
原始 RRT*算法	RRT*(IC)	RRT*(E)
均匀概率改进的 RRT*算法	PH-RRT*(IC)	PH-RRT*(E)
粒子群优化的均匀概率 RRT*算法	PSO-PH-RRT*(IC)	PSO-PH-RRT*(E)

表 2 程序中的主要参数
Table 2 Main parameters in the program

参数	设定值
随机树增加步长 ρ	5
均匀概率参数值 P	0.9
补偿范围 r	5
迭代次数 h	50
学习因子 c_h	1
惯性因子初值 w_0	0.5

3.2 实验结果对比分析

每种方法都在 1 个典型的环境中重复 30 次实验, 并用平均值表征该方法的实验结果. 评价指标是算法迭代时间、规划得到的最短路径长度和路径平滑度.

实验 1 验证改进算法的稳定性. 在单路径、随机障碍环境中重复实验, 统计结果如图 9 所示. 通过对比可知, PSO-PH-RRT*(E) 的数据波动范围较小, 且没有较为明显的离群点, 说明改进代价函数提高了稳定性.

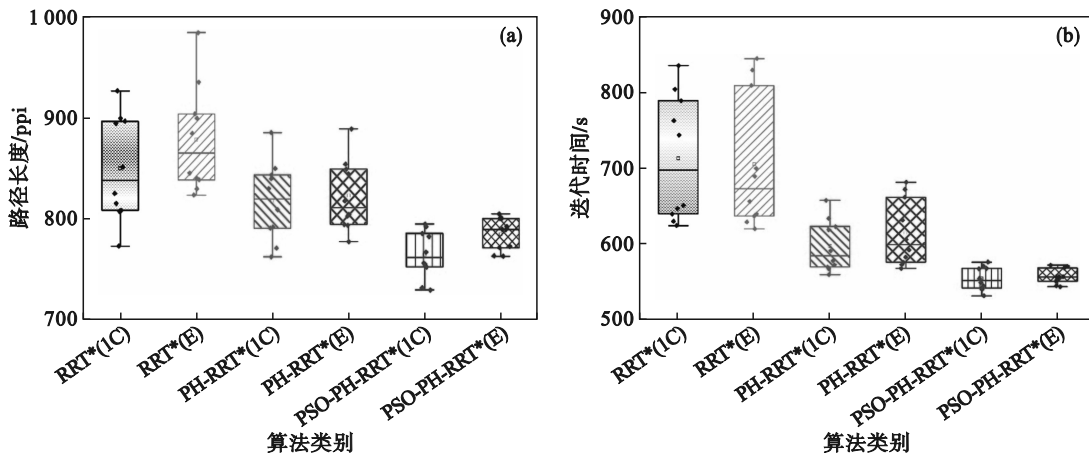


图 9 随机障碍环境的重复实验箱形图

Fig. 9 Repeated experimental box plot in random obstacle environment

(a) 单通路障碍环境下的路径长度; (b) 随机障碍环境下的迭代时间.

实验 2 验证改进算法的搜索速度和结果的路径. 图 10a、图 10b 为 5 种障碍环境中原始方法与改进之后方法的比较. 在 5 种环境中, 在采用平滑度适应度函数情况下搜索时间的平均值分别

减少了 64.58%, 114.70%, 82.65%, 12.43% 和 94.64%. RRT*算法在 5 种环境下所得路径的平均长度为 750.04 ppi, 而 PSO-PH-RRT*算法为 617.56 ppi, 缩短了 132.48 ppi. 结果证明, 在障碍

零星分布的情况下,改进后的自适应概率权重方法路径优化效果更好.相较于平滑度适应度函数,改进后的距离适应度函数路径长度的平均值减少了 22.13%, 9.26%, 28.29%, 10.60% 和 16.71%, 且对于 RRT*和 PH-RRT*算法,改进适应度函数对于规划后的路径指标都有所改善.

针对平滑度作适应度函数的结果数据,如图 10c 所示.可以得出同种拓展方式的情况下,采用平滑度作适应度函数得到的平滑度相对较高, PSO-PH-RRT*算法得到的平滑度均优于原算法.根据如图 11 所示的路径规划结果,可得 PSO-PH-RRT*算法的路径结果在不进行平滑处理的情况下仍较为平滑.

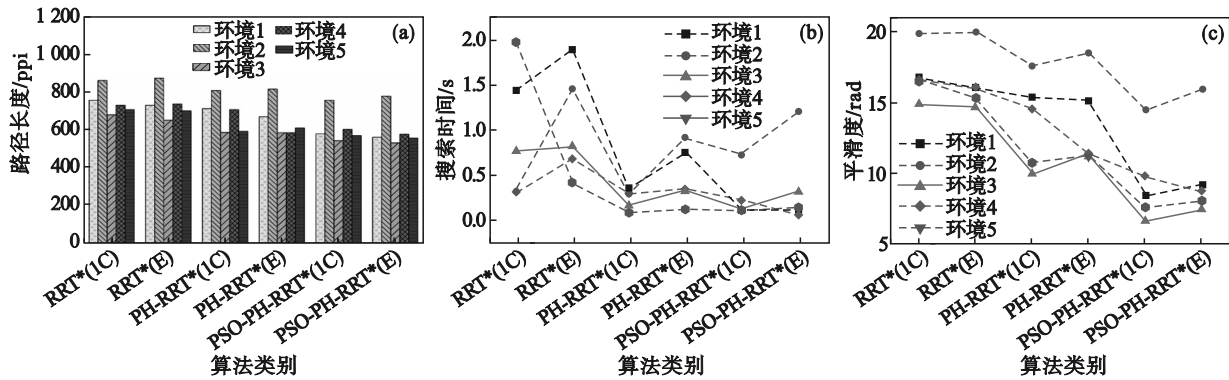


图 10 实验结果统计图

Fig. 10 Experiment results statistical chart

(a)—路径长度; (b)—搜索时间; (c)—平滑度.

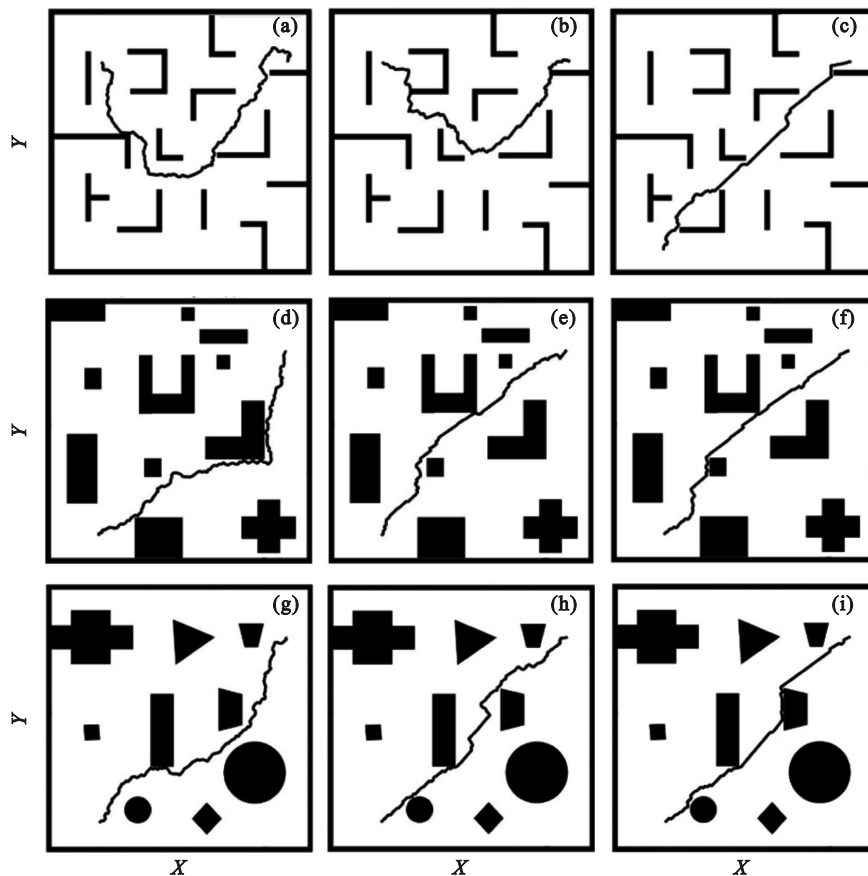


图 11 规划路径结果

Fig. 11 Planning paths consequence

(a)—RRT*(E); (b)— PH-RRT*(E); (c)—PSO-PH-RRT*(E); (d)—RRT*(E); (e)—PH-RRT*(E); (f)—PSO-PH-RRT*(E); (g)—RRT*(E); (h)—PH-RRT*(E); (i)—PSO-PH-RRT*(E).

4 结 论

1) 提出 PSO-PH-RRT*算法,利用 PSO 算法搜索更新粒子位置的迭代思想更新方向概率初始值,减小了搜索后期的迭代时间,搜索得到的路径在不经平滑处理的情况下平滑度较好。

2) 改进 RRT 随机树算法每个节点的代价函数为表征节点到目标向量的距离,并作为 PSO 算法的适应度函数.通过实验验证了改进后算法的优化效果。

参考文献:

- [1] 车建涛,高方玉,解玉文,等.基于 Dijkstra 算法的水下机器人路径规划[J].机械设计与研究,2020,36(1):44-48. (Che Jian-tao, Gao Fang-yu, Xie Yu-wen, et al. Path planning of underwater robot based on Dijkstra algorithm [J]. *Machine Design & Research*, 2020, 36(1):44-48.)
- [2] Tang F. Coverage path planning of unmanned surface vehicle based on improved biological inspired neural network[J]. *Ocean Engineering*, 2023, 278:114354.
- [3] Madridano Á, Kaff A, Martín D, et al. Trajectory planning for multi-robot systems: methods and applications [J]. *Expert Systems with Applications*, 2021, 173:114660.
- [4] La S M, Kuffner J J Jr. Randomized kinodynamic planning [J]. *The International Journal of Robotics Research*, 2001, 20(5):378-400.
- [5] Urmson C, Simmons R. Approaches for heuristically biasing RRT growth [C]//RSJ International Conference on Intelligent Robots and Systems. Las Vegas, 2003: 1178-1183.
- [6] Li Z, Ma H B, Zhang X F, et al. Path planning of the dual-arm robot based on VT-RRT algorithm [C]//Chinese Control Conference. Guangzhou: 2019:4359-4364.
- [7] Kuffner J J, LaValle S M. RRT-connect: an efficient approach to single-query path planning [C]//IEEE International Conference on Robotics and Automation. San Francisco, 2000:995-1001.
- [8] Karaman S, Frazzoli E. Sampling-based algorithms for optimal motion planning [J]. *The International Journal of Robotics Research*, 2011, 30(7):846-894.
- [9] Wang B Y, Liu Z, Li Q B, et al. Mobile robot path planning in dynamic environments through globally guided reinforcement learning [J]. *IEEE Robotics and Automation Letters*, 2020, 5(4): 6932-6939.
- [10] Noreen I, Khan A, Habib Z. A comparison of RRT, RRT* and RRT*-smart path planning algorithms [J]. *International Journal of Computer Science and Network Security*, 2016, 16(10):20-27.
- [11] Hess R, Jerg R, Lindeholz T, et al. SRRT*-a probabilistic optimal trajectory planner for problematic area structures [J]. *IFAC-PapersOnLine*, 2016, 49(30):331-336.
- [12] 白晓兰,周文全,张振朋,等.基于启发式粒子群算法的机器人平滑路径规划[J].组合机床与自动化加工技术,2022(8):44-47,52. (Bai Xiao-lan, Zhou Wen-quan, Zhang Zhen-peng, et al. Smooth path planning of wheeled robot based on heuristic particle swarm optimization algorithm [J]. *Modular Machine Tool & Automatic Manufacturing Technique*, 2022 (8):44-47,52.)
- [13] 屈新怀,单笛,孟冠军.基于靠近目标粒子群算法的 AGV 路径规划[J].合肥工业大学学报(自然科学版),2022,45(1):1-6. (Qu Xin-huai, Shan Di, Meng Guan-jun. AGV path planning based on particle swarm optimization approaching the target [J]. *Journal of Hefei University of Technology (Natural Science Edition)*, 2022, 45(1):1-6.)
- [14] Lin S W, Liu A, Wang J G, et al. An intelligence-based hybrid PSO-SA for mobile robot path planning in warehouse [J]. *Journal of Computational Science*, 2023, 67:101938.
- [15] Krell E, Sheta A, Balasubramanian A P R, et al. Collision-free autonomous robot navigation in unknown environments utilizing PSO for path planning [J]. *Journal of Artificial Intelligence and Soft Computing Research*, 2019, 9(4): 267-282.
- [16] 彭君.改进 RRT 算法在移动机器人路径规划中的应用研究[D].南京:南京邮电大学,2022. (Peng Jun. Application of improved RRT algorithm in mobile robot path planning [D]. Nanjing: Nanjing University of Posts and Telecommunications, 2022.)
- [17] 左国玉,陈国栋,刘月雷,等.基于均匀概率的目标启发式 RRT 机械臂路径规划方法[J].北京工业大学学报,2022,48(8):812-821. (Zuo Guo-yu, Chen Guo-dong, Liu Yue-lei, et al. Target heuristic RRT based on uniform probability for manipulator path planning [J]. *Journal of Beijing University of Technology*, 2022, 48(8):812-821.)