

DOI: 10.13382/j.jemi.B2408082

多样性种群增强遗传算法机器人全局路径规划研究*

刘俊毅 汪 洋

(辽宁工程技术大学机械工程学院 阜新 123000)

摘要:为解决传统遗传算法面临局部最优解和迭代速度慢等问题,从多个方面对遗传算法进行了改进。首先,将传统的8方向搜索扩展为24邻域16方向,以增强全局搜索能力;引入PT(piecewise and tent)混沌映射融合策略,通过Piecewise混沌映射生成的序列作为Tent混沌映射参数,以提升种群多样性;其次,结合莱维(Levy)飞行策略避免局部停滞,并提出新的越界粒子处理策略,以防初始化种群越界;接着,设计了全新配对交换和差分扰动机制,防止优良个体丧失导致陷入局部最优;最后,提出了一种新的压力等级拆分选择机制和自适应交叉变异概率调整算子,通过调整系数解决选择压力过大问题,采用非线性指数函数调整交叉选择概率,以避免早期发散,并通过互补调整变异概率,扩大搜索空间,减少收敛震荡。实验结果表明,所提方法相比传统遗传算法及其他改进算法,显著提高了路径规划性能,路径长度分别减少5.13%和2.06%,验证了其在机器人路径规划中的优越性与实用性。

关键词: 移动机器人;全局规划;路径规划;遗传算法;混沌映射;差分扰动;Levy飞行;交叉变异概率

中图分类号: TP242;TN96 **文献标识码:** A **国家标准学科分类代码:** 510.80

Research on genetic algorithm for robot global path planning with diversity population enhancement

Liu Junyi Wang Yang

(School of Mechanical Engineering, Liaoning Technical University, Fuxin 123000, China)

Abstract: To address the issues of local optima and slow convergence speed encountered by traditional genetic algorithms, several improvements have been made to the genetic algorithm in this paper. Firstly, the conventional 8-direction search is extended to a 24-neighborhood, 16-direction search to enhance the global search ability. Secondly, a Piecewise and Tent (PT) chaotic mapping fusion strategy is introduced, where the sequence generated by the Piecewise chaotic mapping is used as the parameter for the Tent chaotic mapping to improve population diversity. Furthermore, the Levy flight strategy is integrated to avoid local stagnation, and a new strategy for handling out-of-bounds particles is proposed to prevent the initialization population from exceeding boundaries. A novel pairing exchange and differential perturbation mechanism is then designed to prevent the loss of good individuals, which may lead to the algorithm getting stuck in local optima. Lastly, a new pressure level splitting selection mechanism and an adaptive crossover and mutation probability adjustment operator are proposed. Coefficients are adjusted to resolve the issue of excessive selection pressure, and a nonlinear exponential function is used to adjust the crossover selection probability to avoid early divergence. Additionally, complementary adjustments to mutation probabilities are introduced to expand the search space and reduce convergence oscillations. Experimental results show that the proposed method significantly improves path planning performance compared to traditional genetic algorithms and other improved algorithms, with path lengths reduced by 5.13% and 2.06%, respectively. The superiority and practicality of the method in robot path planning are validated.

Keywords: mobile robot; global planning; path planning; genetic algorithm; chaotic mapping; differential perturbation; Levy flight; crossover and mutation probability

收稿日期: 2024-12-31 Received Date: 2024-12-31

* 基金项目:江苏省自然科学基金面上基金(20KJB530008)、国家科学信息技术部研究中心“十四五”课题基金(KXJS71057)、教育部“十四五”教育科研规划课题基金(JXKY24391)项目资助

0 引言

随着机器人技术的快速发展,机器人在复杂环境中的自主导航已成为研究的热点问题,尤其是在狭小和复杂的管道等封闭空间中。机器人在这些环境中执行任务时,除了要确保避障安全外,还需要寻找最优或近似最优路径,以提高作业效率和降低能耗。路径规划作为机器人自主导航中的核心问题之一,遗传算法(genetic algorithm, GA)作为一种模拟自然选择和生物进化过程的全局优化算法,在路径规划问题中得到了广泛应用。它能够通过群体搜索的方式寻找问题的全局最优解,避免陷入局部最优,因此被广泛应用于复杂环境下的路径规划,陈丽芳等^[1]将遗传算法与B样条技术进行融合,并引入自适应交叉变异概率,优化机器人移动路径节点的同时快速找到最短路径,田雅琴等^[2]采用跳点搜索算法与遗传算法进行融合,采用跳点遗传(jump point search genetic, JPSG)算法提高整体搜索性能,加快算法整体收敛趋势,再利用遗传算法改变跳点(jump point search, JPS)算法不能在复杂障碍物状况下解析最优路径的状态,提高算法对动态环境的适应性,魏书鑫等^[3]采用萤火虫算法与遗传算法进行融合,当萤火虫中算法陷入局部最优时采用遗传算法进行二次寻优,证明了新的算法具有良好的精度和性能,徐兴等^[4]在遗传算法中引入灾变策略改进,相比较与传统的遗传算法能够更快的寻找的全局最优解上述算法虽然能够有效的找到最优解,但是相应的也会陷入局部最优的缺陷,灵活性不足。

针对这些问题,本文提出了一种改进的遗传算法,旨在解决传统遗传算法在复杂环境下路径规划中的局限性。首先,为了提高全局搜索能力,本文将传统的8方向搜索扩展为24邻域16方向,从而增加了搜索空间的覆盖度,避免了搜索过程中可能出现的局部最优解。其次,本文引入了PT(piecewise and tent)混沌映射融合策略,并采用Piecewise混沌映射生成的序列作为Tent混沌映射的参数,增强了种群的多样性,有效避免了种群过早收敛。为了克服传统遗传算法在复杂环境中容易发生的局部停滞问题,本文结合了莱维(Levy)飞行策略,并提出了新的越界粒子处理方法,确保了初始化种群的有效性。接着,设计了配对交换和差分扰动机制,以避免优良个体丧失并防止陷入局部最优。最后,通过自适应调整交叉和变异概率的算子,采用非线性指数函数调整以解决早期发散问题,并通过互补思想调整变异概率,扩大搜索空间,减少后期收敛时的震荡。通过对比实验,本文所提改进算法在机器人路径规划任务中的表现明显优于传统遗传算法及其他改进算法。证明了该方法在复杂环境下的优越性和实用性。

1 初始模型建立

1.1 栅格地图模型

机器人路径规划中,障碍物表示和环境建模至关重要。常见地图类型有矢量地图(适合复杂建模但实时性差)、栅格地图和拓扑地图(适合大规模规划但精度较低)^[5]。

栅格地图因其简单、高效,广泛应用于路径规划^[6],通过均匀网格表示障碍物,适合实时规划和动态响应。图1所示为40个栅格的示意图,障碍物随机分布,蓝色栅格为起点和目标点。

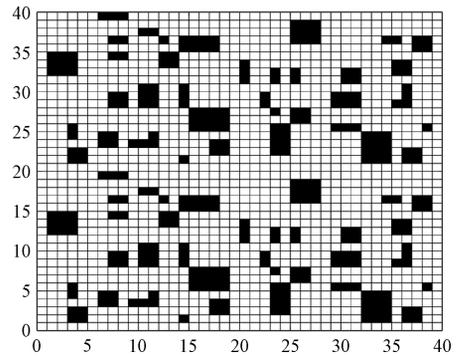


图1 40 栅格地图障碍物分布

Fig. 1 Obstacle distribution in a 40-grid map

假设机器人位于栅格地图的任意位置,规定机器人每次只能移动到相邻的栅格位置^[7],环境地图为 $G=40 \times 40$ 的矩阵,令地图中的任意栅格编号为 m , m 的坐标为 (x, y) ,则 x 和 y 的坐标表达式为:

$$\begin{cases} x = \begin{cases} \text{bmod}(m, 40) - 0.5, & \text{bmod}(m, 40) \neq 0 \\ 39.5, & \text{mod}(m, 40) = 0 \end{cases} \\ y = \begin{cases} 39.5 - \text{floor}(m, 40), & \text{mod}(m, 40) \neq 0 \\ 40.5 - \text{floor}(m, 40), & \text{mod}(m, 40) = 0 \end{cases} \end{cases} \quad (1)$$

式中: $\text{bmod}(m, 40)$ 表示为 m 除以40得到的余数; $\text{floor}(m, 40)$ 表示 m 除以40得到的结果向下取整数。

将地图平均分为 $M \times N$ 个长度相等的小栅格,用 S_{xy} 表示每个小栅格,整个地图用 Γ 表示:

$$\Gamma = S_{xy}, x \in [1, M], y \in [1, N] \quad (2)$$

每个栅格有无障碍物的信息表达式为:

$$S_{xy} = \begin{cases} 0, & \text{无障碍物} \\ 1, & \text{有障碍物} \end{cases} \quad (3)$$

为减少坐标计算的计算量,采用坐标到序号的映射转换^[8],设栅格地图的大小为 $\text{rows} \times \text{cols}$,按照 $1 - \text{rows} \times \text{cols}$ 来表示栅格序号,序号与坐标的映射公式如下:

$$\text{numb} = [\text{rows} - 1] \times \begin{bmatrix} x \\ y \end{bmatrix} \quad (4)$$

进行序号标记的栅格地图如图2所示。

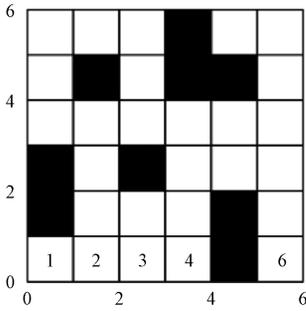


图2 栅格地图序号标记

Fig. 2 Grid map index labeling

在机器人运行中,遗传算法的染色体表示从起始到结束节点的可行路径,常规算法通过二维坐标和网格法对栅格编号并转换^[9],路径转换模型如图3和4所示。

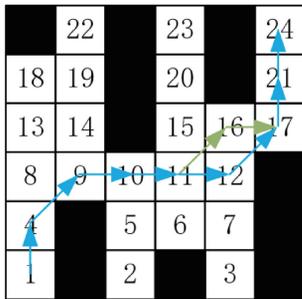


图3 网络标号地图

Fig. 3 Network labeled map

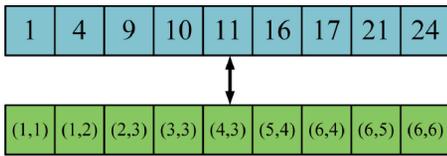


图4 二维坐标与网络标号转换

Fig. 4 Conversion between 2D coordinates and network labels

自由网格按顺序编号,起始节点为1,目标节点为24。网格号与坐标可相互转换,构成遗传算法染色体,生成路径。

1.2 运动目标函数与约束条件

1) 染色体编码与解码

为了满足条件,栅格地图的坐标通过网格序号编码^[10]。首先,根据二维坐标生成初始种群,再通过公式将二维坐标转化为网格序号。

$$GN_i = (x_i - 1) + (y_i - 1) \times R_{grid} \quad (5)$$

其中, (x_i, y_i) 表示栅格地图的二维坐标, R_{grid} 表示网格图的行数, GN_i 表示解码之后的网格号。

2) 目标函数

机器人在运动过程中需要满足以下的基础约束条件。

(1) 路径不可以超出栅格地图^[11],其约束限制的公式为:

$$C_1 = \sum_{i=1}^n [(x_i < 1) \vee (x_i > M) \vee (y_i < 1) \vee (y_i > N)] = 0 \quad (6)$$

式中: n 代表路径上节点的个数; (x_i, y_i) 表示路径上第 i 个节点的二维坐标; M, N 分别表示栅格地图的最大行数和最大列数。

(2) 路径不可以跨越禁区,其约束限制的公式为:

$$C_2 = \sum_{i=1}^n [(x_i, y_i) \in \mathcal{O}] = 0 \quad (7)$$

式中: \mathcal{O} 代表禁区的坐标集合。

(3) 每次只能移动一步,也就是从栅格移动到相邻的栅格位置^[12],其约束限制的公式为:

$$C_3 = \sum_{i=1}^{n-1} [|x_i - x_{i+1}| > 1 \vee |y_i - y_{i+1}| > 1] = 0 \quad (8)$$

以上的约束要求就是每两个相邻节点之间的坐标差都不能超过1,确保每次只能移动一步。

根据以上的基础条件,设计了一种以路径、安全度、平滑度3种因素作为遗传算法的目标函数。其具体表达式如下:

$$Fit = \rho_1 \cdot \frac{1}{L(k)} + \rho_2 \cdot \frac{1}{S(k)} + \rho_3 \cdot \frac{1}{F(k)} \quad (9)$$

式中: $L(k)$ 为距离函数; $S(k)$ 为安全度函数; $F(k)$ 为平滑度函数; $\rho_1 \sim \rho_3$ 分别为3个指标对应的权重系数。

路径的长度是根据邻近节点的欧氏距离求和得出,距离函数为:

$$D = \sum_{i=1}^{n-1} \sqrt{(x_i - x_{i+1})^2 + (y_i - y_{i+1})^2} \quad (10)$$

式中: n 为路径的总节点数量。过程中机器人生成的路径所产生的障碍物用 O_{num} 表示,其具体公式为:

$$O_{num} = \sum_{i=1}^N \sigma_i \quad (11)$$

$$\sigma_i = \begin{cases} 1, & \text{有障碍物} \\ 0, & \text{无障碍物} \end{cases}$$

式中: σ_i 代表是否经过障碍物。根据以上的障碍物公式与欧氏距离可得机器人的路径目标函数为:

$$L(k) = D + O_{num} \cdot R_{grid} \cdot 2 \quad (12)$$

式中: R_{grid} 表示栅格地图生成序列的行数。

路径平滑度通过计算相邻点间的角度变化来衡量^[13-14],本文采用角度差总和作为目标函数,公式如下:

$$F(k) = \sum_{i=2}^{n-1} (\theta_{i-1,i} - \theta_{i,i+1})^2 \quad (13)$$

式中: $\theta_{i-1,i}$ 和 $\theta_{i,i+1}$ 分别为路径上第 $i-1$ 点、第 i 点与第

$i + 1$ 点之间的切线角度; n 是路径中总节点数量。较小的角度变化表示较平滑的路径, 因此对其进行取倒数处理。

安全度函数的目标是避免路径与障碍物碰撞, 通过检查路径上每个点到障碍物的距离来定义, 公式如下:

$$S(k) = \sum_{i=1}^n (\max(0, d_{\min} - d_{\text{obstacle}}(p_i))) \quad (14)$$

式中: p_i 是路径上的第 i 个点; $d_{\text{obstacle}}(p_i)$ 是点 p_i 到最近障碍物之间的距离; d_{\min} 是设定的最小安全距离。

在矩形障碍物中, 假设矩形的左下角标为 (x_1, y_1) , 右上角坐标为 (x_2, y_2) , 路径点到矩形障碍物的最小距离为:

$$d_{\text{obstacle}}(p_i) = \max \left(\begin{array}{l} 0, \max(|x_i - x_1|, |x_i - x_2|) \\ -(x_2 - x_1), \max(|y_i - y_1|, |y_i - y_2|) - (y_2 - y_1) \end{array} \right) \quad (15)$$

安全距离时进行人为设定的, 如果路径点 p_i 离障碍物太近, 即 $d_{\text{obstacle}}(p_i) < d_{\min}$, 则该项参数大于 0, 表明路径不安全, 反之, 该项参数为 0, 表示该点为安全的。

3) 目标约束条件

遗传算法的目标函数考虑路径、安全度和平滑度, 约束条件包括路径长度非负、路径连续性、角度差、平滑度非负以及避免障碍物穿越, 且考虑逆行情况。

最短路径 L_{path} 是路径中所有相邻点之间的距离总和, 路径长度的约束必须确保路径是合理的^[15-16], 且不会产生负的或无效的路径。

$$L_{\text{path}} = \sum_{i=1}^{n-1} d(p_i, p_{i+1}) \geq 0 \quad (16)$$

式中: $d(p_i, p_{i+1})$ 是两点之间的距离。如果路径上的存在无效点或跳跃, $d(p_i, p_{i+1})$ 就会导致负值。

路径中的每一个相邻点 p_i 和 p_{i+1} 必须是连续的, 两个相邻点之间不能超过某个最大阈值。

$$d(p_i, p_{i+1}) \leq d_{\max} \quad (17)$$

式中: d_{\max} 是可接受的最大距离。

相邻两段路径的角度变化应保持在一定的范围内, 防止机器人产生剧烈转向, 假设路径点 p_i 和 p_{i+1} 之间的方向角为 $\Delta\theta_i$, 则两者之间的角度差应满足:

$$\Delta\theta_i \leq \Delta\theta_{\max} \quad (18)$$

式中: $\Delta\theta_{\max}$ 为设定的最大角度变化值, 过大的角度差会导致路径变得不平滑。

由于角度变化的平方和必须为非负, 因此相应的平滑度函数应满足:

$$S_{\text{smooth}} \geq 0 \quad (19)$$

路径上任意一点 $p_i(x_i, y_i)$ 与环境中的障碍物的最短距离 $d_{\text{safe}}(p_i)$ 应大于或等于设定的最小安全距离 $d_{\text{safe}, \min}$, 其表达式如下:

$$d_{\text{safe}}(p_i) \geq d_{\text{safe}, \min} \quad \forall p_i \in \text{path} \quad (20)$$

路径不得与障碍物发生重叠或穿越, 假设路径中每段路径 p_i 和 p_{i+1} 之间的直线段在空间中不与障碍物相交, 路径中的每个线段必须与障碍物的边界保持足够的距离, 并且不与障碍物相交。

$$d_{\text{safe}}(p_i, p_{i+1}) \geq d_{\text{safe}, \min} \quad \forall i \quad (21)$$

这些约束确保遗传算法规划的路径可行、安全、平滑且长度合理。

4) 方向搜索机制

在基于栅格地图的路径规划中, 主要有 4 个方向和 8 个方向两种搜索机制, 如图 5 所示。4 个方向仅包括上下左右 4 个方向的栅格搜索; 由图 5(b) 可见, 8 个方向则包括上下左右及 4 个斜方向。为避免局部最优解, 本文针对管道清洁机器人采用 8 方向或更高方向的搜索机制进行路径优化。

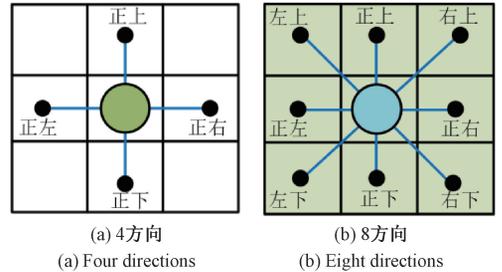


图 5 方向搜索机制示意图

Fig. 5 Schematic diagram of directional search mechanism

图 5(b) 中 8 方向搜索机制具有 45° 角度分辨率, 而 4 方向搜索机制为 90° 。相比之下, 8 方向搜索视野更广, 但遗传算法的活动范围被限制在 45° 和 90° 之间, 导致寻优过程受到限制。

为提高算法活动范围, 减少限制, 本文提出了一种开拓视野的遗传算法, 将视野扩大 1 倍, 栅格地图搜索机制从 8 方向扩展为 24 邻域 16 方向, 如图 6 所示。

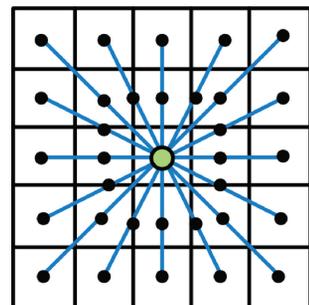


图 6 8 方向改进为 24 邻域 16 方向

Fig. 6 Improvement from 8 directions to 24-neighbor 16 directions

开拓视野后,邻域栅格数量增加,提高了机器人移动的机动性。与8方向搜索机制的45°分辨率相比,视野扩展后粒子角度分辨率为22.5°,极大提高了粒子运动的灵活性。

为展示开拓视野的优越性,将24邻域16方向搜索机制与4方向和8方向搜索机制进行对比,如图7所示,路径对比数据如表1所示。

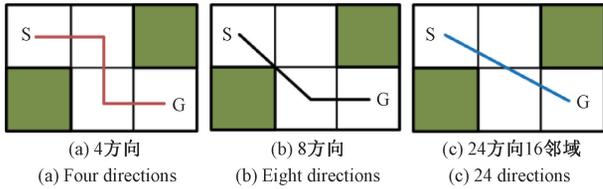


图7 不同方向搜索机制路径对比
Fig. 7 Path comparison of different directional search mechanisms

表1 方向搜索机制路径对比数据
Table 1 Path comparison data of directional search mechanisms

| 搜索机制 | 4方向 | 8方向 | 24邻域16方向 |
|--------|------|------|----------|
| 路径长度/m | 3.12 | 2.75 | 1.33 |
| 冗余点 | 2 | 1 | 0 |

由图7可见,开拓视野后的算法路径拐弯减少,路径平滑,规划次数和最短路径均缩短,提升了规划效率。

2 原始遗传算法

遗传算法模拟生物进化,通过选择、交叉、变异等过程将初始解映射为最优解^[17]。主要包括初始化种群、个体评价、选择、交叉、变异和终止判断6个过程,流程见图8所示。

遗传算法通过选择、交叉和变异操作,探索解空间并生成新一代个体。选择采用轮盘赌机制,交叉方式包括单点和多点交叉。

其中选择操作的具体公式如下:

$$P_i = \frac{f(i)}{\sum_{i=1}^n f(i)} \quad i = 1, 2, 3, \dots, n$$

$$newpop(t+1) = \{pop_j(t) \mid j = 1, 2, 3, \dots, n\} \quad (22)$$

式(22)根据个体适应度与种群总适应度的比例计算选择概率,适应度高的个体更易被选中参与交叉和变异。

传统的自适应交叉和变异因子公式如下:

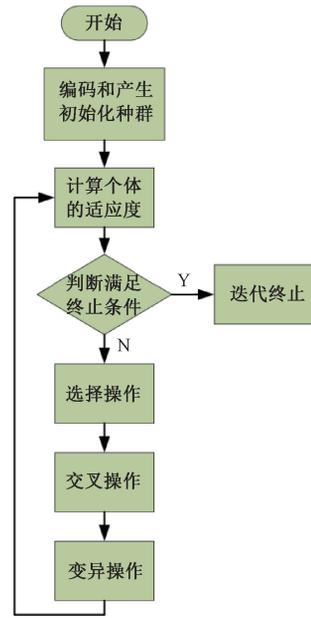


图8 传统遗传算法流程

Fig. 8 Flowchart of traditional genetic algorithm

$$P_c = \begin{cases} k_1 \frac{(f_{\max} - f)}{f_{\max} - f_{\arg}} & f \geq f_{\arg} \\ k_2 & f < f_{\arg} \end{cases}$$

$$P_m = \begin{cases} k_3 \frac{(f_{\max} - \hat{f})}{f_{\max} - f_{\arg}} & \hat{f} \geq f_{\arg} \\ k_4 & \hat{f} < f_{\arg} \end{cases} \quad (23)$$

式中: f_{\max} 表示群体的最大适应值; f_{\arg} 表示群体的平均适应值; k_1, k_2, k_3, k_4 分别为0~1之间常数; f 和 \hat{f} 是被选择参与交叉和变异的个体的适应度。

传统的交叉和变异操作方式多为单点和随机选点交叉,单点交叉为在父代的某个位置随机选择一个点,交换该点之后的所有基因位置片段,形成两个新的子代。

假设父代和母代的染色体片段序列为:

$$P_1 = (p_{1,1}, p_{1,2}, \dots, p_{1,n}) \quad (24)$$

$$P_2 = (p_{2,1}, p_{2,2}, \dots, p_{2,n})$$

经过单点交叉以后的子代染色体片段变为:

$$\text{子代 1} = (p_{1,1}, p_{1,2}, \dots, p_{1,i}, p_{2,i+1}, p_{2,i+2}, \dots, p_{2,n})$$

$$\text{子代 2} = (p_{2,1}, p_{2,2}, \dots, p_{2,i}, p_{1,i+1}, p_{1,i+2}, \dots, p_{1,n}) \quad (25)$$

单点交叉的公式中, i 为随机交叉点,随机点交叉与单点交叉类似,但选择不同位置交换基因,示意图如图9所示。

根据原始算法的叙述,本文将从初始化种群、选择、交叉、变异等^[18]几个方面进行改进,提高算法的精度与收敛性能。

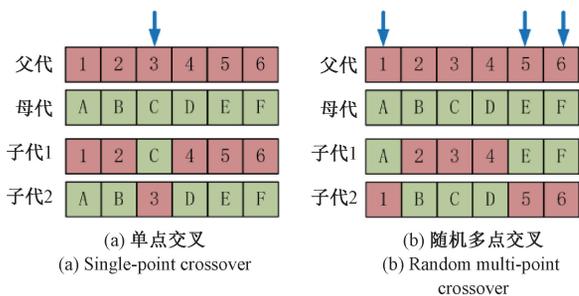


图 9 单点交叉和随机多点交叉示意图

Fig. 9 Schematic diagram of single-point crossover and random multi-point crossover

3 改进遗传算法

3.1 改进融合混沌映射兼 Levy 飞行策略

1) 改进 PT 融合混沌映射

遗传算法依赖初始种群的质量和多样性,随机初始化可能导致分布不均,增加局部最优风险^[19]。为提高精度和全局搜索能力,本文采用 Piecewise 混沌映射生成序列作为帐篷混沌映射的参数,增强映射的混沌性,其两种混沌映射如式 (26) 和 (27) 所示。

$$x_{i+1} = \begin{cases} \frac{x_i}{P}, & 0 \leq x_i < P \\ \frac{x_i - P}{0.5 - P}, & P \leq x_i < 0.5 \\ \frac{1 - P - x_i}{0.5 - P}, & 0.5 \leq x_i < 1 - P \\ \frac{1 - x_i}{P}, & 1 - P \leq x_i < 1 \end{cases} \quad (26)$$

$$x_{i+1} = \begin{cases} \frac{x_i}{a}, & x_i < a \\ \frac{(1 - x_i)}{(1 - a)}, & x_i \geq a \end{cases} \quad (27)$$

$\alpha \in (0, 1)$

式中: x_i 均为当前第 i 个种群个体; P 和 a 分别为两种混沌映射的控制参数,分别取为 0.3 和 0.5。

混沌过程如下:

- 1) 创建 3 个全零数组,分别命名为 a 、 x 和 $Position$, 其中 a 和 x 用于执行 Piecewise 映射和 Tent 映射, $Position$ 数组用于存储生成的混沌初始化种群;
- 2) 随机生成种群 a 和 x 的第 1 维数据;
- 3) 将代数变量 i 初始化为 1,用以表示当前的种群代数;
- 4) 对于种群 x 的第 i 维,通过尺度变换将其映射到预设的 lb 和 ub 范围内。变换后的值赋给 $Position$ 的第 i

维,作为该个体的混沌初始解;

5) 对 a 的第 i 维应用 Piecewise 映射,生成第 $i+1$ 维数据;

6) 对 a 的第 $i+1$ 维执行 Tent 映射,生成更新后的 a 的第 $i+1$ 维数据;

7) 对 i 进行增值处理,检查是否已遍历完所有种群。如果 i 超出种群数量,则结束初始化过程,否则,返回步骤 5),继续进行混沌初始化。

为对比混沌映射在种群初始化中的优越性,选取 100 个种群、50 次迭代进行多样性对比。图 10(a)~(c) 分别为 Tent 混沌映射、改进混沌映射和 Piecewise 混沌映射初始化的种群。

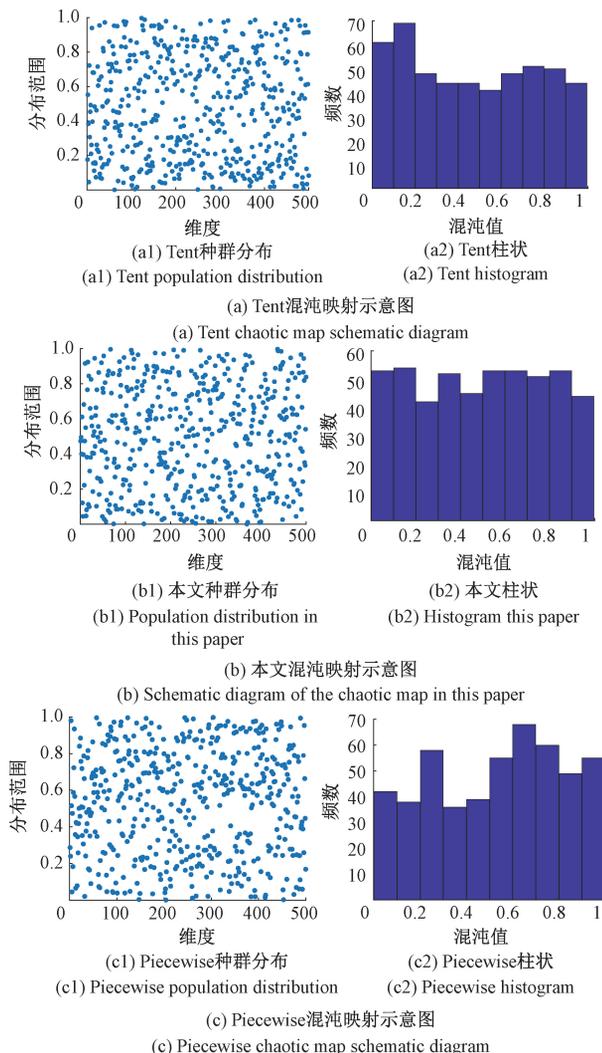


图 10 不同混沌映射的种群分布和柱状图对比

Fig. 10 Comparison of population distribution and histogram for different chaotic mappings

为验证混沌映射的优越性,采用 F23 复合模态函数进行测试,并与 Tent、Circle、Piecewise 和 Chebyshev 混沌

映射进行对比。测试设置为 1 000 次迭代,实验结果如图 11 所示。

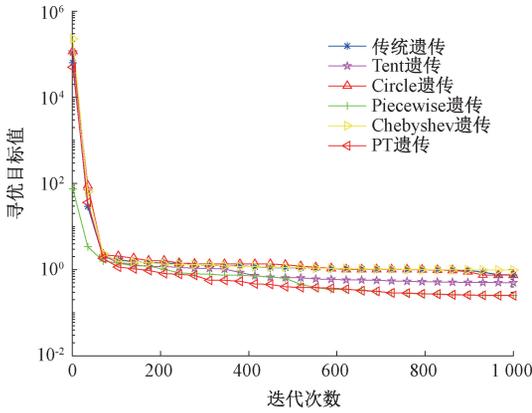


图 11 不同混沌映射的遗传算法性能对比

Fig. 11 Comparison of genetic algorithm performance with different chaotic mappings

由图 11 可知,Tent 混沌映射生成的种群分布不均,存在适应度较低的个体,影响算法性能。Piecewise 混沌映射改善了分布,但仍有种群重叠问题。相比之下,本文改进的混沌映射有效解决了这些缺陷,增强了种群多样性和空间遍历性。

2) 引入 Levy 飞行策略尽管上述改进提升了算法的全局搜索能力和空间遍历性,但未能解决局部最优问题。因此,本文引入 Levy 飞行策略,通过赋予个体更高的自由度,增强探索灵活性,有效避免了算法陷入局部最优,提升了跳出局部最优的能力,公式如下:

$$x'(t) = x(t) + s \oplus \text{levy}(\beta) \quad (28)$$

$$\text{levy}(\beta) \sim \frac{u}{|v|^{-\beta}}$$

式中: $x(t)$ 表示第 t 次迭代的最优个体; $x'(t)$ 表示经过 Levy 飞行的个体; s 代表步长参数,在本文中取为 1; $\text{levy}(\beta)$ 表示莱维飞行。

其中, μ 和 ν 服从正态分布,参数 β 取为 1.5,Levy 飞行公式如下:

$$u \sim N(0, \sigma_u^2), \nu \sim N(0, \sigma_\nu^2)$$

$$\sigma_u = \left[\frac{\Gamma(1 + \beta) \sin(\frac{\pi\beta}{2})}{\Gamma(\frac{1 + \beta}{2})\beta \times 2^{\frac{\beta-1}{2}}} \right]^{\frac{1}{\beta}}, \sigma_\nu = 1 \quad (29)$$

3) 改进越界粒子处理策略

传统 GA 算法在迭代过程中可能出现粒子越界,导致种群丢失并陷入局部最优。常见的边界处理方法有“隐形边界”、“反射边界”和“吸收边界”,但这些方法可能降低种群多样性,影响算法收敛性。本文在传统吸收边界法基础上改进,提出了一种新型越界粒子处理策

略,公式如下:

$$Lb > 0 \Rightarrow x_i = \begin{cases} Ub - (Ub - x_i) \cdot rand, x_i > Ub \\ Lb + (Lb - x_i) \cdot rand, x_i < Lb \end{cases}$$

$$Ub < 0 \Rightarrow x_i = \begin{cases} Ub + (Ub - x_i) \cdot rand, x_i > Ub \\ Lb - (Lb - x_i) \cdot rand, x_i < Lb \end{cases} \quad (30)$$

式中: $Ub - x_i$ 和 $Lb - x_i$ 表示越界种群到边界的距离; $rand$ 表示 0 到 1 之间的随机数。

该改进策略基于种群边界值,通过分析越界种群与边界值的关系来确定处理方法。引入随机数增加了越界种群在处理后期的多样性,有效避免了粒子在边界处聚集的问题。

3.2 改进配对交换机制兼差分扰动机制

1) 改进配对交换机制

传统遗传算法中的交叉与变异操作可能导致局部最优解。为解决染色体结构变化问题,本研究提出了一种新型配对交换策略,优化配对与交叉,减少不良变异,提升种群稳定性和搜索效率。配对流程如下:

- (1) 分别随机选取一个染色体作为父本,一个染色体作为母本;
- (2) 随机选取部分位置进行分割,还原多染色体状态;
- (3) 计算分割之后的个染色体对应的适应度大小,选取父本中最大的染色体和母本中最小的染色体进行相互配对;
- (4) 在配对过程中对中间部分的编码区作为交换点,在交换点出进行分割;
- (5) 根据分割情况交换配对染色体的编码,生成新的染色体;
- (6) 将其中的染色体重新组合,并进行修复。将每个子代中的第一重复位置进行交换;
- (7) 最后生成符合结果的最优序列,保留无重复性个体。

配对示意图如图 12 所示。

可以清晰的看出改进的过程,经过以上的配对过程能够有效的避免种群过早收敛,保持种群的多样性,避免因随机交换可能带来的不良基因重组,减少陷入局部最优的缺陷。

2) 引入差分扰动机制

通过上述配对交换机制,生成全新的子代个体。然而,在配对过程中可能出现基因重叠。为提高子代个体的随机性并消除重叠影响,本文采用差分进化(difference equation, DE)算法,通过对 3 个子代个体 x_1, x_2, x_3 进行差分计算,增强基因的随机性,得到差分向量 D_v 表示为:

$$D_v = \gamma \cdot (x_2 - x_3) \quad (31)$$

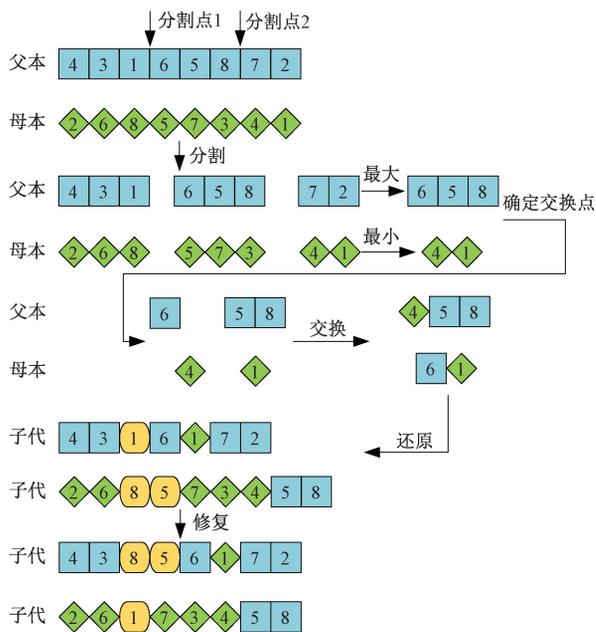


图 12 交换配对示意图

Fig. 12 Schematic diagram of pairing exchange

式中： γ 为缩放因子，用于控制差异大小。将第 1 个个体 x_1 与差分向量结合，生成全新的个体 x_4 ，公式如下：

$$x_4 = x_1 + D_v \quad (32)$$

通过差分进化策略，有效增加了基因的多样性，从而提升了算法的随机性和搜索效率。

3.3 压力等级拆分选择机制

传统的轮盘赌选择机制通过适应度比例进行选择，导致高适应度个体被过度选择，选择压力过大，并在局部最优时导致算法过早收敛。为解决这一问题，本文提出了一种压力等级拆分选择机制，平衡勘探和开发，避免选择压力过大。

首先，将个体按适应度值排序，并将种群分为两等份，假设种群大小为偶数。

上半部分是考虑其 1 到 $K/2$ 个个体，其具体公式为：

$$\frac{K}{4} \left(\frac{K}{2} + 1 \right) \quad (33)$$

后半部分是考虑其 $k/2+1$ 到 k 个个体，其具体公式为：

$$\frac{K^2}{4} + \frac{K}{4} \left(\frac{K}{2} + 1 \right) \quad (34)$$

根据以上的个体分配，得到的压力等级差分选择机制公式为：

$$p(i) = \begin{cases} \lambda^- \left(\frac{8i}{K(K+2)} \right), & i \leq \frac{K}{2} \\ \lambda^+ \left(\frac{8i}{K(3K+2)} \right), & i > \frac{K}{2} \end{cases} \quad (35)$$

其中， λ^- 和 λ^+ 控制压力大小，且 $\lambda^- + \lambda^+ = 1$ ， i 为算法迭代次数，通过控制 λ^+ 来抑制选择压力，为了保持开发和勘探的平衡，进行选取合适的调整参数，采用 150 个个体进行选择，分别对参数调整为 0.6、0.7、0.75、0.8 进行选择，其对比如图 13 所示。

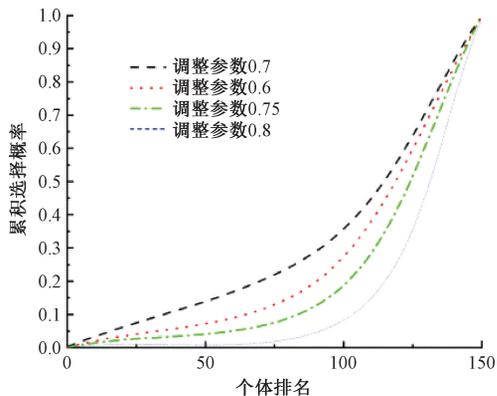


图 13 调整参数对比测试

Fig. 13 Comparison test of parameter adjustments

由图 13 可以看出，调整参数为 0.7 时具有最好的性能，因此本文调整参数 λ^+ 为 0.7，最终的选择机制公式为：

$$p(i) = \begin{cases} \frac{12i}{5K(K+2)}, & i \leq \frac{K}{2} \\ \frac{28i}{5K(3K+2)}, & i > \frac{K}{2} \end{cases} \quad (36)$$

3.4 改进自适应交叉变异概率

传统的自适应交叉变异算子可能导致算法前期发散或后期振荡，使得算法无法按预期进行交叉和变异，从而提前停止或陷入局部最优解。

为解决这一问题，本文提出了一种新型交叉变异算子：通过非线性指数函数调整交叉和变异概率，解决早期发散；基于互补思想调整变异概率，降低种群阶数，增加搜索空间，避免后期震荡。

采用的指数调节函数是标准指数函数的变分，其中 α 和 β 是调节因子，用于调节函数的衰减率， G 为遗传算法的迭代次数，公式如下：

$$h(G) = 1 / (1 + e^{\frac{G - \alpha^* \beta}{\alpha}}) \quad (37)$$

在初始阶段， $h(G)$ 几乎不变，随着迭代次数的增加， $h(G)$ 逐渐减少，其中， α 控制 $h(G)$ 的扩展和收缩， α 越大，其下降越快， β 控制 $h(G)$ 的位置移动， $h(G)$ 随着的 β 增大而右移，示意图如图 14 所示。

引入以上的指数调节函数形成的新的交叉概率可以表示为：

$$P_c = \begin{cases} \frac{1}{1 + \exp((G - \alpha * \beta) / \alpha)} \cdot [k_0 + \frac{(k_1 - k_0) * (f_{\max}(G) - f_{\text{avg}}(G - 1))}{f_{\max}(G) - f_{\text{avg}}(G)}], & f_{\text{avg}}(G) \geq f_{\text{avg}}(G - 1) \\ k_2, & f_{\text{avg}}(G) < f_{\text{avg}}(G - 1) \end{cases} \quad (38)$$

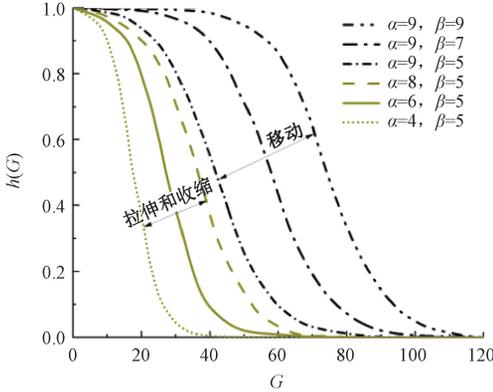


图 14 不同参数调节示意图

Fig. 14 Schematic diagram of different parameter adjustment

式中: $f_{\max}(G)$ 和 $f_{\text{avg}}(G)$ 个体的平均适应度和最大适应度; k_0, k_1, k_2 分别控制自适应交叉概率的范围。本文将 P_c 的范围定为 $[k_0, k_1]$, 其中, $k_0 = 0.5, k_1 = k_2 = 0.8$, 该调整函数作为尺度因子, 增加了遗传算法在迭代过程中的协调性, 自适应的降低交叉和变异概率, 降低了进化后期出现收敛振荡问题的概率。

传统的自适应变异算子基于固定数量的个体, 但实际的进化准则是随机的^[20], 因此本文为更好的模拟实际情况, 采用互补思想进行自适应调整突变概率。具体步骤如下。

1) 采用精英保存策略确定个体 $nXoverKids$, 将当前一代中最优秀的个体保存到下一代。

2) 根据上述的交叉概率确定交叉个体数目, 公式如下:

$$nXoverKids = [P_c \cdot \text{populationsize}] \quad (39)$$

3) 采用互补思想确定突变个体数目, 公式如下:

$$nMutateKids = \text{populationsize} - nXoverKids - nEliteKids \quad (40)$$

式中: populationsize 代表当前一代的种群大小。

根据以上计算可得基于互补思想的变异概率为:

$$P_m = \frac{nMutateKids}{\text{populationsize}} = 1 - P_c - \frac{nEliteKids}{\text{populationsize}} \quad (41)$$

通过改进, 变异概率随交叉概率减少而增大, 扩展搜索空间; 交叉概率增加时, 变异概率减小, 提高收敛速度和稳定性, 尤其在算法后期, 能有效避免低质量个体, 保证良好的稳定性。

3.5 算法流程

根据以上的改进策略, 其具体流程如图 15 所示。

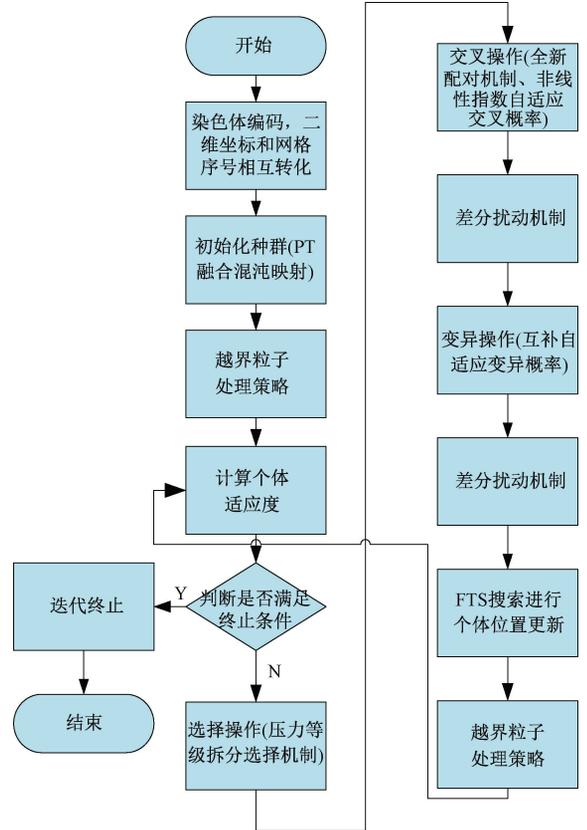


图 15 改进遗传算法示意图

Fig. 15 Schematic diagram of the improved genetic algorithm

4 算法仿真

在 MATLAB2023b 中进行仿真实验, 计算机的操作系统为 Windows 11 64 bit, 内存为 128 GB, CPU 为 13th Gen Intel (R) Core(TM) i9-13900HX@2.20 GHz。

根据以上的改进策略, 形成的全新的多种群策略增强遗传算法 (multi-group strategy enhances genetic algorithm, MGEGA) 算法, 其具体参数如表 2 所示。

4.1 函数测试对比

将本文算法与蜣螂算法 (dung beetle optimization, DBO)、粒子群算法 (particle swarm optimization, PSO)、GA、麻雀算法 (sparrow search algorithm, SSA) 4 种优化算法进行对比, 采用 CEC2022 的单模态、多模态和复合基准函数进行测试, 选取了 $F1 \sim F12$ 函数作为测试函数。迭代次数设置为 1 000 次, 每个算法重复计算 30 次。对比结果通过箱线图和适应度迭代图如图 16、17 所示。

表 2 改进遗传算法的具体参数

Table 2 Specific parameters of the improved genetic algorithm

| 参数 | 设置值 |
|-----------------|-------------|
| 种群数量 | 100 |
| 迭代次数 | 100 |
| 混沌映射 α | 0.3 |
| 混沌映射 p | 0.5 |
| Levy 飞行 β | 1.5 |
| γ | 0.01 |
| P_c | [0.5, 0.8] |
| P_m | [0.01, 0.1] |
| 调节因子 α | 9 |
| 调节因子 β | 7 |
| k_0 | 0.5 |
| k_1 | 0.8 |
| k_2 | 0.8 |
| λ^+ | 0.7 |

的表现,尤其在 $F1$ 、 $F2$ 、 $F3$ 和 $F7$ 函数中,表现出较强的稳定性。这表明 MGEGA 能够在多次运行中保持一致的性能,显著减少解的波动性。此外,在 $F4$ 、 $F5$ 和 $F6$ 函数中,MGEGA 相比其他算法在精度上也有显著提升,最优解的提升幅度为 0.1~2.65。与此同时,MGEGA 在多个测试函数中的中值和平均值表现同样出色,能够在多次实验中维持稳定的性能,避免出现极端值。

4.2 多栅格地图仿真

为了验证 MGEGA 算法在路径规划中的有效性,本文在 MATLAB 中构建了 20×20 、 30×30 、 40×40 、 50×50 四种障碍物地图模型,用于模拟不同复杂度的移动机器人行驶环境。实验中,种群数目设为 100,最大迭代次数为 100,参数配置见表 1。在较简单的 20×20 栅格地图中,起点和终点设置为 $(0.5, 0.5)$ 、 $(19.5, 19.5)$;在 30×30 、 40×40 、 50×50 等较复杂的栅格地图中,分别设置不同的起点和终点位置。栅格地图的路径规划轨迹图像和迭代图像分别如图 18 所示,数据如表 3 所示。

MGEGA 算法在多个函数测试中展现了接近最优值

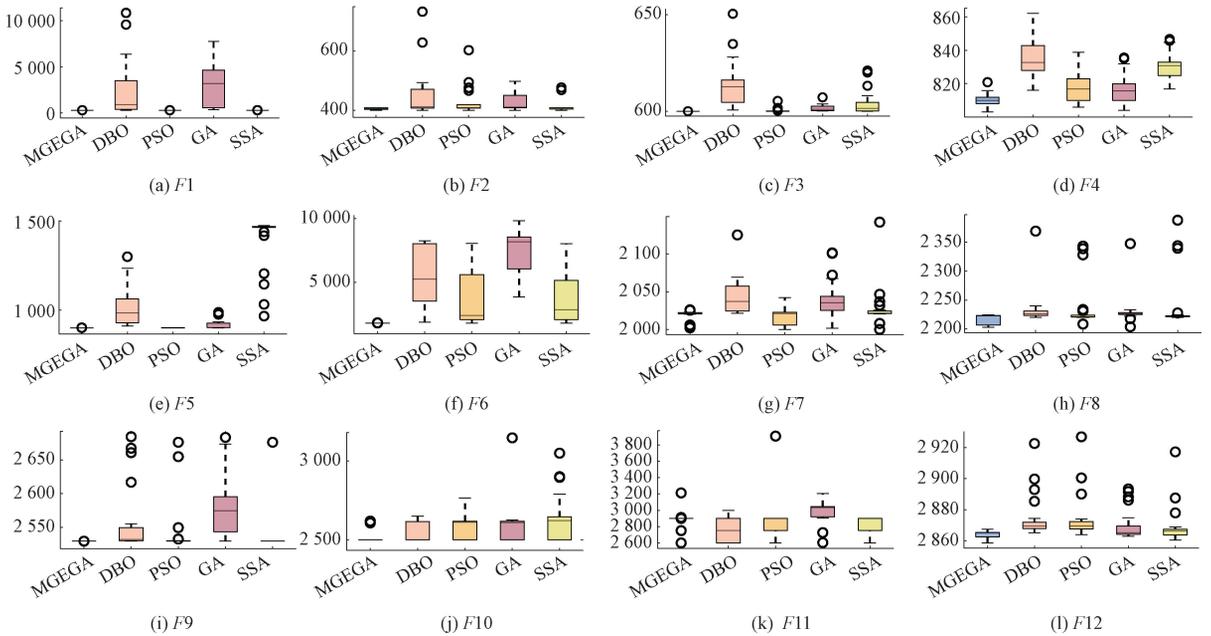


图 16 不同测试函数的箱线图对比

Fig. 16 Comparison of boxplots for different test functions

根据栅格地图和迭代曲线分析,改进后的 MGEGA 算法在障碍物避障和最短路径规划方面表现优异。通过对 20×20 、 30×30 、 40×40 三种复杂度的栅格地图实验验证,MGEGA 算法能够有效应对复杂路径规划问题,在充满障碍物的环境中快速找到最优路径。

实验结果表明,MGEGA 算法通过多策略优化,不仅实现了避障功能,还有效缩短了路径长度,减少了机器人行驶时间和能量消耗。在 20×20 地图中,算法能够快速

找到最短路径,展现了较高的计算效率。在更复杂的 30×30 、 40×40 、 50×50 地图中,尽管障碍物分布复杂,算法依然能够在较少迭代次数内找到平滑且可行的避障路径。

迭代曲线分析显示,随着迭代次数增加,算法逐步收敛,路径质量不断提升,最终接近最优解。即使在障碍物密集的复杂地图中,MGEGA 算法仍展现出强大的全局优化和局部搜索能力,成功实现避障和最短路径目标。本

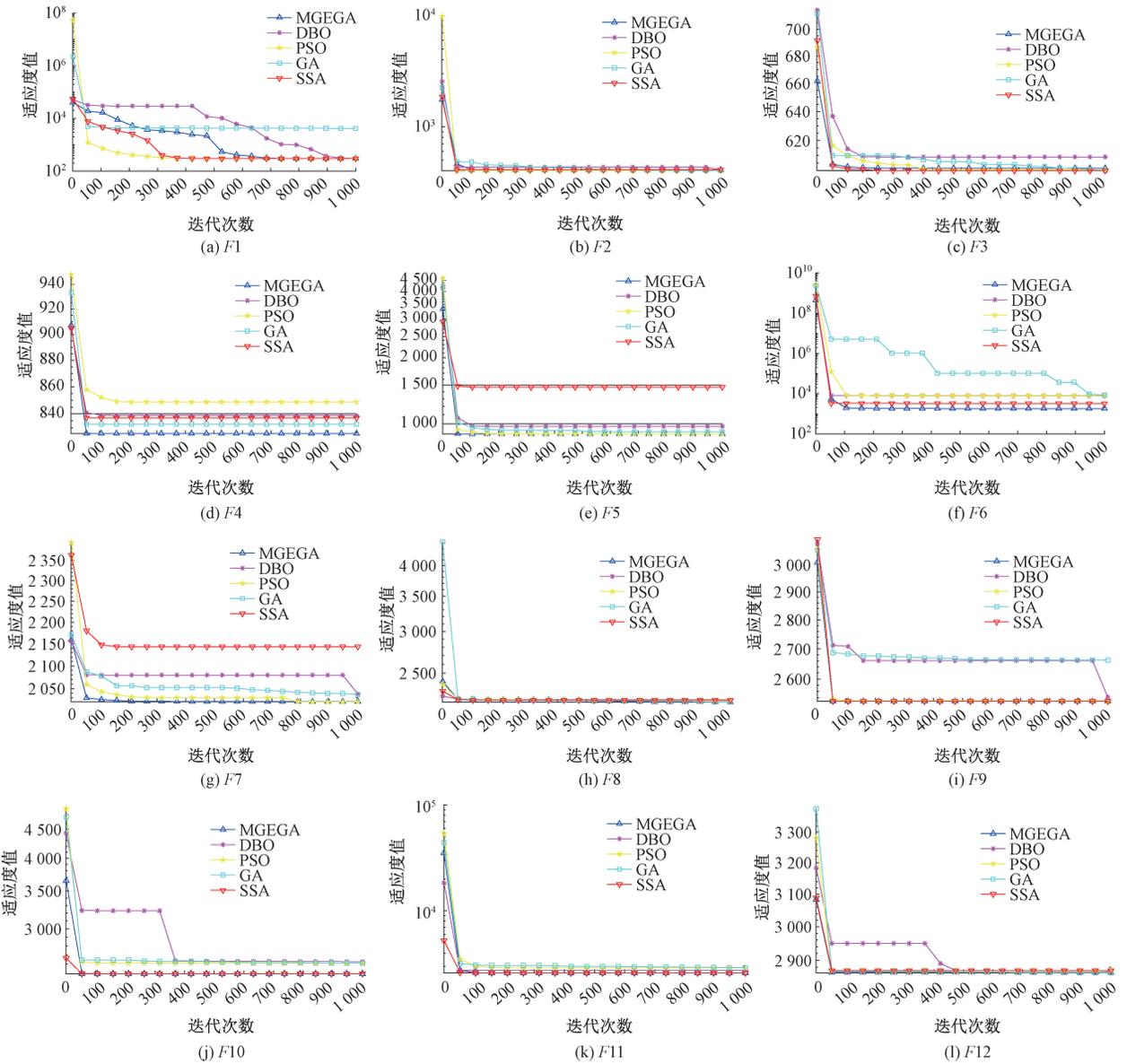


图 17 不同测试函数的算法性能对比

Fig. 17 Comparison of algorithm performance for different test functions

表 3 不同栅格地图的路径实验数据

Table 3 Experimental path data for different grid maps

| 地图类型 | 20×20 | 30×30 | 40×40 | 50×50 |
|--------|----------|----------|----------|----------|
| 实验数据 | 28.665 4 | 42.876 3 | 57.645 2 | 75.149 7 |
| | 29.020 3 | 42.987 6 | 58.736 4 | 74.198 4 |
| | 28.873 5 | 42.864 4 | 58.674 5 | 75.173 7 |
| | 28.646 4 | 42.183 6 | 59.274 6 | 74.780 9 |
| | 29.058 7 | 43.814 6 | 58.191 8 | 74.214 9 |
| | 28.903 4 | 43.122 7 | 57.918 3 | 75.094 1 |
| 最优值/m | 28.646 7 | 42.874 6 | 58.193 7 | 74.614 6 |
| | 28.646 4 | 42.183 6 | 57.645 2 | 74.198 4 |
| 平均时间/s | 2.076 | 12.334 | 56.788 | 123.065 |
| 转弯次数 | 9 | 11 | 9 | 13 |

文设计了一个栅格地图,模拟机器人全自动路径寻优,起点为(0.5,0.5),终点为(29.5,29.5),种群数目为100,迭代次数为200次。蓝色、红色和紫色图线分别代表改进的灰狼算法、MGEGA算法和改进的IGA算法。收敛系数为2,交叉概率为0.6,变异概率为0.3,结果如图19和表4所示。

仿真分析表明,本文算法在最短路径问题上表现优异,最优值较改进的灰狼算法和遗传算法分别减少了3.4%和13.6%,显示出更高的精度和效率。

4.3 传统算法对比仿真

通过本文算法与传统的SSA、DBO、GA、PSO算法进

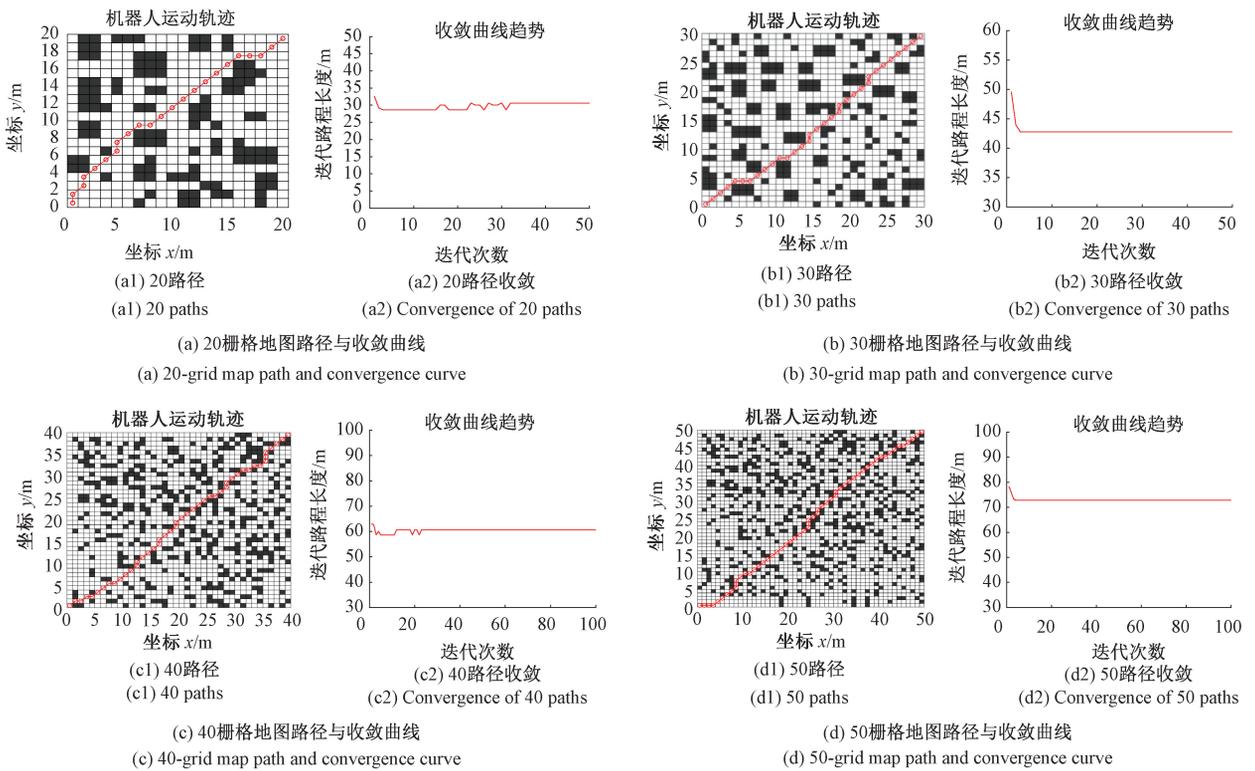


图 18 不同栅格地图的路径和收敛曲线

Fig. 18 Paths and convergence curves for different grid maps

表 4 实际工况模拟下的路径数据

Table 4 Path data under actual operating conditions simulation

| 算法类别 | IGA | IGWO | MGEGA |
|--------|-------|-------|-------|
| 实验数据 | 47.03 | 43.03 | 42.64 |
| | 48.77 | 42.87 | 41.33 |
| | 48.65 | 42.77 | 41.87 |
| | 47.76 | 42.08 | 40.65 |
| | 47.35 | 42.13 | 41.23 |
| | 48.02 | 42.33 | 41.55 |
| 最优值 | 47.03 | 42.08 | 40.65 |
| 平均时间/s | 5.341 | 5.233 | 4.871 |
| 转弯次数 | 11 | 10 | 8 |

表 5 不同地图环境的传统算法路径数据对比

Table 5 Comparison of path data for traditional algorithms in different map environments

| 地图类型 | MGEGA | SSA | DBO | GA |
|------------|----------|----------|----------|----------|
| 20×20 栅格地图 | 28.123 1 | 28.786 6 | 29.657 8 | 28.554 3 |
| | 27.998 6 | 29.043 5 | 29.645 3 | 28.656 4 |
| | 28.167 3 | 28.656 4 | 29.645 6 | 28.346 7 |
| | 27.938 8 | 28.778 9 | 29.345 2 | 28.377 8 |
| 40×40 栅格地图 | 58.546 3 | 66.324 5 | 62.897 6 | 63.197 4 |
| | 58.632 5 | 65.901 3 | 61.923 4 | 62.147 4 |
| | 59.147 7 | 66.279 5 | 61.174 7 | 63.147 8 |
| | 58.324 6 | 65.128 4 | 61.347 9 | 62.087 1 |
| 平均时间/s | 4.231 | 4.332 | 4.021 | 4.665 |
| 转弯次数 | 5 | 6 | 6 | 8 |

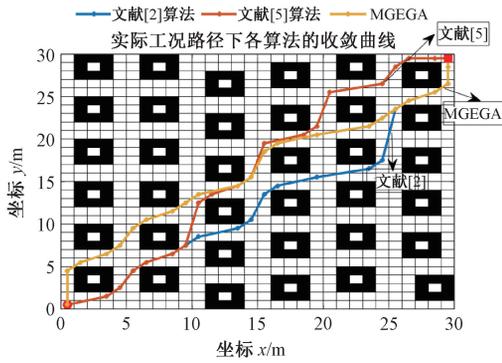
行对比,体现本文算法在路径规划中的有效性,根据环境,设置最大迭代次数和种群数目以及起点和终点坐标,栅格地图采用 20×20 和 40×40 两种不同的地图,采用的适应度函数设计为路径最短,其中,PSO 算法的学习因子 C_1 和 C_2 分别设置为 2 和 3,GA 算法的交叉概率设置为 0.3,变异概率设置为 0.7,DBO 算法的步长因子、学习因子、收敛因子分别设置为 0.8、2、0.8,SSA 算法跳跃因子、探索因子、开发因子分别设置为 1.0、1.2、0.8,对每个算法在以上的栅格地图环境中分别运行 4 次,其结果和数据具体如图 20、21 和表 5 所示。

整理表格中的数据,计算其最优值、平均值和标准差并进行对比,其结果如表 6 所示。柱状图如图 22 所示。

实验结果显示,本文算法在路径寻优上优于其他 4 种算法,显著减少最短路径,快速找到最优解,且在多个场景中表现出更强的搜索能力和优化效率。

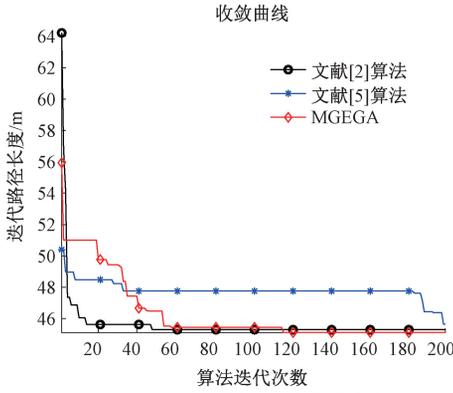
4.4 与其他算法对比仿真

为进一步验证 MGEGA 算法的先进性与优越性,本文进行了文献对比实验。实验中,采用了 MGEGA、文献[2]的 JPSGA 算法、文献[5]的改进遗传算法,分别在



(a) 实际工况模拟下算法路径曲线

(a) Algorithm path curve under actual working condition simulation

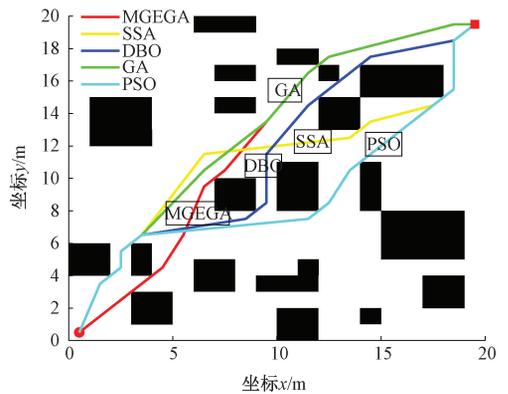


(b) 实际工况模拟下路径收敛曲线

(b) Path convergence curve for actual working condition simulation

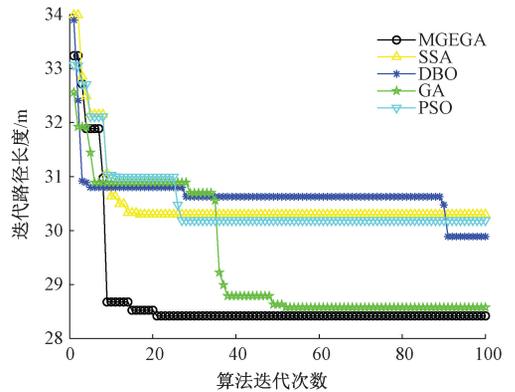
图 19 实际工况模拟下的路径和收敛曲线

Fig. 19 Paths and convergence curves under actual operating conditions simulation



(a) 简单地图传统算法路径

(a) Traditional algorithm path on a simple map



(b) 简单地图传统算法收敛曲线

(b) Traditional algorithm convergence curve on a simple map

图 20 简单地图传统算法路径和收敛曲线

Fig. 20 Paths and convergence curves of traditional algorithms on simple map

表 6 整理之后的数据

Table 6 Processed data

| 地图类别 | 算法 | 最优值/m | 平均值/m | 标准差 |
|------------|-------|----------|----------|---------|
| 20×20 栅格地图 | MGEGA | 27.938 8 | 28.057 0 | 0.092 1 |
| | SSA | 28.656 4 | 28.816 4 | 0.140 9 |
| | DBO | 29.345 2 | 29.573 5 | 0.131 9 |
| | GA | 28.346 7 | 28.483 8 | 0.127 3 |
| | PSO | 30.066 5 | 30.270 6 | 0.230 7 |
| 40×40 栅格地图 | MGEGA | 58.324 6 | 58.662 8 | 0.302 |
| | SSA | 65.128 4 | 65.908 4 | 0.479 3 |
| | DBO | 61.174 7 | 61.835 9 | 0.673 7 |
| | GA | 62.087 1 | 62.644 9 | 0.528 |
| | PSO | 59.900 8 | 60.529 8 | 0.491 8 |

40×40 和 50×50 两种不同尺寸和复杂度的栅格地图中进行验证。每种栅格地图的边长均设定为 1,且障碍物分布在不同实验中有所变化,以模拟不同的复杂环境。

实验中,所有算法的种群数为 100,最大迭代次数为 30 次,记录每次实验的最短路径和运行时间。起点为(0.5, 0.5),终点为右上角目标点,黑色栅格为障碍

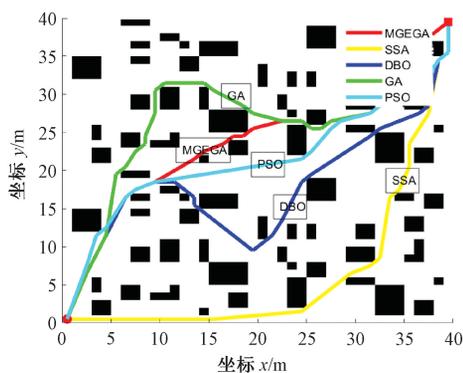
物,白色栅格为可行区域。文献算法参数与原配置一致,本文 MGEGA 算法参数如表 1 所示,路径规划和迭代曲线如图 23 和 24 所示,蓝色为文献[2]算法路径,红色为文献[5]算法路径,橙色为本文算法路径。

由于实验地图中的算法路径轨迹存在重合型,因此本文列出了实验中个算法的路径数据,其中 40×40 和 50×50 的栅格地图分别进行运行 3 次进行统计,如表 7 所示,柱状图如图 25 所示。

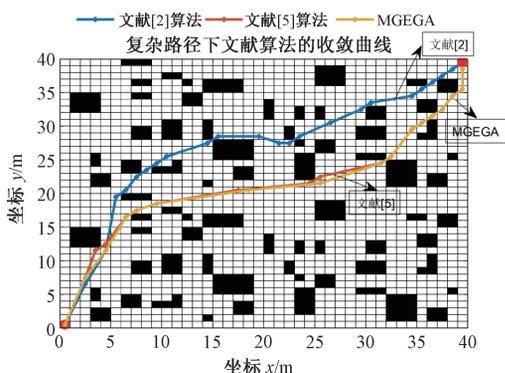
MGEGA 算法在 40×40 和 50×50 栅格地图中分别比文献[2,5]算法减少了 5.13%、2.06%和 0.91%、2.02%的最优值,证明其在不同环境下有效优化路径,提升质量,具有良好适应性。

5 结 论

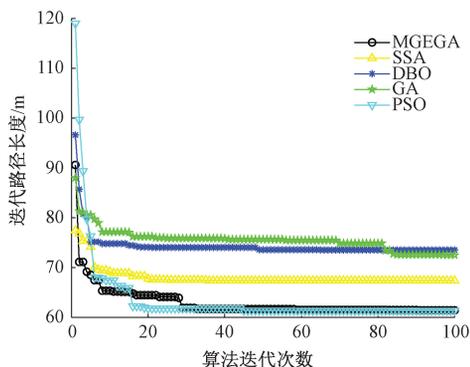
本文提出了一种改进的遗传算法,用于解决机器人全局避障路径规划问题。通过扩展搜索方向、引入混沌



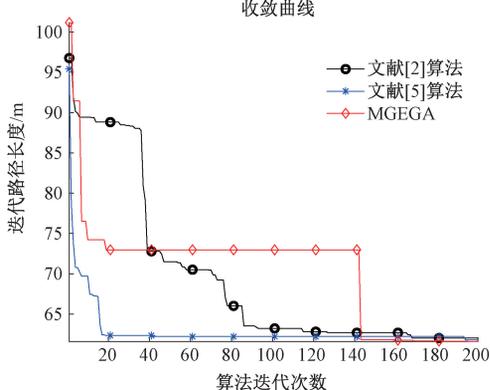
(a) 复杂地图传统算法路径
(a) Traditional algorithm path on a complex map



(a) 40 栅格地图文献算法路径
(a) Literature algorithm path on a 40-grid map



(b) 复杂地图传统算法收敛曲线
(b) Traditional algorithm convergence curve on a complex map



(b) 40 栅格地图文献算法收敛曲线
(b) Literature algorithm convergence curve on a 40-grid map

图 21 复杂地图传统算法路径和收敛曲线
Fig. 21 Paths and convergence curves of traditional algorithms on complex map

图 23 40 栅格地图文献算法对比
Fig. 23 Comparison of literature algorithms on 40-grid map

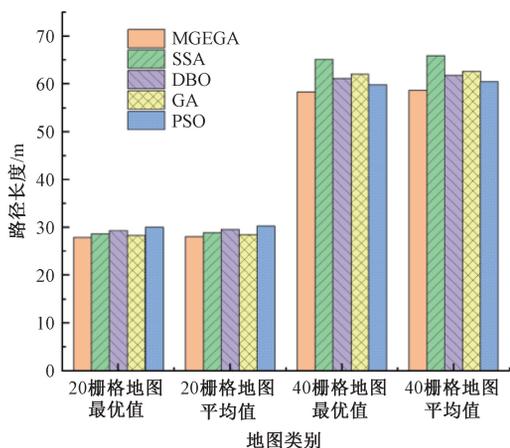


图 22 整理之后的数据柱状图
Fig. 22 Bar chart of processed data

验证了其优越性和实用性。

表 7 不同栅格地图的文献算法对比数据
Table 7 Comparison data of literature algorithms on different grid maps

| 地图类型 | 文献[2]算法 | 文献[5]算法 | 本文算法 |
|------------|----------|----------|----------|
| 40×40 栅格地图 | 66.635 | 63.646 | 60.476 |
| | 65.239 | 62.146 | 61.983 |
| | 63.747 | 63.186 | 60.648 |
| | 65.127 | 61.747 | 60.897 |
| 50×50 栅格地图 | 87.758 3 | 88.389 5 | 83.983 5 |
| | 86.908 8 | 87.902 8 | 84.591 4 |
| | 87.735 9 | 88.930 5 | 82.030 2 |
| | 87.957 0 | 88.145 7 | 82.122 4 |
| 平均时间/s | 4.456 | 4.211 | 4.189 |
| 转弯次数 | 7 | 8 | 6 |

该算法有效克服了传统遗传算法在复杂环境中易陷入局部最优和收敛慢的问题,通过增强全局搜索能力和提高种群多样性,提供了更高效、稳定的路径规划方案。然而,在极端复杂或动态环境下,算法的搜索效率可能受到影响,特别是在障碍物密集或环境变化频繁时,收敛速

映射,结合 Levy 飞行策略,设计新的配对交换与差分扰动机制,并自适应调整交叉和变异概率,显著提升了路径规划性能。实验表明,与传统遗传算法及其他改进算法相比,所提出的方法在复杂环境中路径长度减少约 5%,

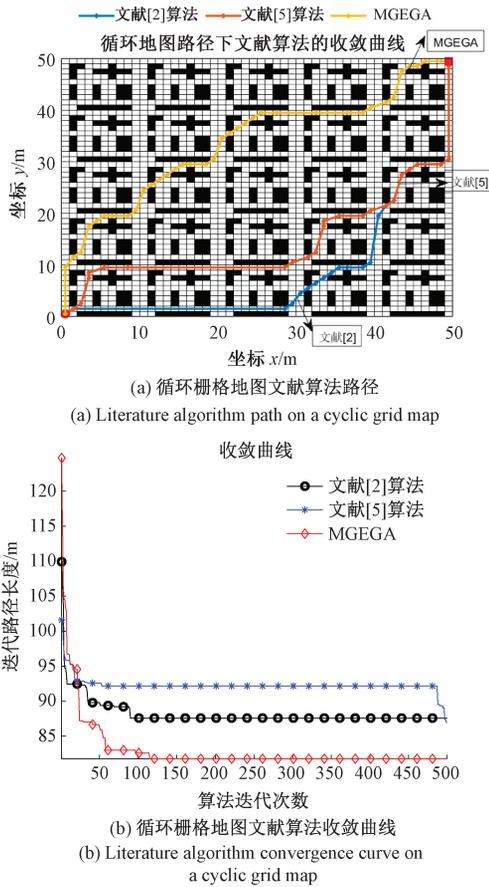


图 24 循环栅格地图文献算法对比

Fig. 24 Comparison of literature algorithms on circular grid map

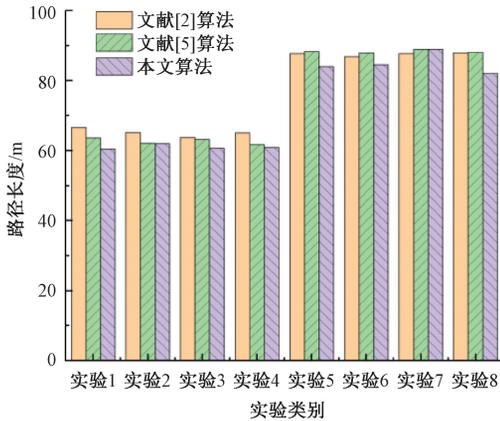


图 25 数据柱状图

Fig. 25 Data bar chart

度较慢,计算资源需求较高。此外,尽管自适应调整交叉和变异概率可减少早期发散,但在高维解空间中,仍可能出现过早收敛,无法完全避免局部最优解的产生。

参考文献

[1] 陈丽芳, 杨火根, 陈智超, 等. B 样条技术与遗传算

法融合的全局路径规划 [J]. 浙江大学学报 (工学版), 2024, 58 (12): 2520-2530.

CHEN L F, YANG H G, CHEN ZH CH, et al. Global path planning with integration of B-spline technique and genetic algorithm [J]. Journal of Zhejiang University (Engineering Science), 2024, 58 (12): 2520-2530.

[2] 田雅琴, 胡梦辉, 刘文涛, 等. 基于跳点搜索-遗传算法的自主移动机器人路径规划 [J]. 工程设计学报, 2023, 30 (06): 697-706.

TIAN Y Q, HU M H, LIU W T, et al. Path planning of autonomous mobile robot based on jump point search-genetic algorithm [J]. Path Planning of Autonomous Mobile Robot Based on Jump Point Search-Genetic Algorithm, 2023, 30 (6): 697-706.

[3] 魏书鑫, 王群京, 李国丽, 等. 萤火虫算法结合遗传算法的移动机器人路径规划 [J]. 制造业自动化, 2024, 46 (10): 69-82.

WEI SH X, WANG Q J, LI G L, et al. Firefly algorithm combined with genetic algorithm for mobile robot path planning [J]. Manufacturing Automation, 2024, 46 (10): 69-82.

[4] 徐兴, 俞旭阳, 赵芸, 等. 基于改进遗传算法的移动机器人全局路径规划 [J]. 计算机集成制造系统, 2022, 28 (6): 1659-1672.

XU X, YU X Y, ZHAO Y, et al. Global path planning of mobile robot on improved genetic algorithm [J]. Computer Integrated Manufacturing Systems, 2022, 28 (6): 1659-1672.

[5] 王雷, 王艺璇, 李东东, 等. 基于改进遗传算法的移动机器人路径规划研究 [J]. 华中科技大学学报 (自然科学版), 2024, 52 (5): 158-164.

WANG L, WANG Y X, LI D D, et al. Research on path planning of mobile robot based on improved genetic algorithm [J]. Journal of Huazhong University of Science and Technology (Natural Science Edition), 2024, 52 (5): 158-164.

[6] 周伟, 潘金宝, 王林琳, 等. 基于改进鲸鱼算法和 A* 算法的地面放线机器人路径规划 [J]. 现代制造工程, 2023 (12): 68-75, 83.

ZHOU W, PAN J B, WANG L L, et al. Improved whale algorithm and A* algorithm for path planning of ground paying-off robot [J]. Modern Manufacturing Engineering, 2023 (12): 68-75, 83.

[7] 周龙港, 刘婷, 卢劲竹. 基于 Floyd 和改进遗传算法的丘陵地区农田遍历路径规划 [J]. 智慧农业 (中英文), 2023, 5 (4): 45-57.

ZHOU L G, LIU T, LU J ZH. Traversal path planning for farmland in hilly areas based on floyd and improved

- genetic algorithm[J]. *Smart Agriculture*, 2023, 5(4): 45-57.
- [8] NI X, HU W, FAN Q, et al. A Q-learning based multi-strategy integrated artificial bee colony algorithm with application in unmanned vehicle path planning [J]. *Expert Systems with Applications*, 2024, 236: 121303.
- [9] 伞红军, 杨晓园, 陈久朋, 等. 基于拟水流算法在移动机器人路径规划中的应用[J]. *仪器仪表学报*, 2024, 45(7): 263-278.
- SAN H J, YANG X Y, CHEN J P, et al. Research on path planning of mobile robot based on the stream algorithm[J]. *Chinese Journal of Scientific Instrument*, 2024, 45(7): 263-278.
- [10] 李章萍, 徐鑫. 改进遗传算法应用于地震场景下无人机路径规划研究[J]. *安全与环境学报*, 2025, 25(1): 237-249.
- LI ZH P, XU X. Research on drone path planning in earthquake scenarios using an improved genetic algorithm[J]. *Journal of Safety and Environment*, 2025, 25(1): 237-249.
- [11] 于逸然, 赖惠成, 高古学, 等. 基于遗传算法和 A* 算法的多农机协同作业优化方法[J]. *系统仿真学报*, 2025, 37(9): 2397-2408.
- YU Y R, LAI H CH, GAO G X, et al. Optimization method for multi agricultural machinery collaborative operation based on genetic algorithm and A* algorithm[J]. *Journal of System Simulation*, 2025, 37(9): 2397-2408.
- [12] 吴越安, 杜昌平, 杨睿, 等. 基于改进遗传算法的倾转旋翼无人机区域覆盖路径规划[J]. *浙江大学学报(工学版)*, 2024, 58(10): 2031-2039.
- WU Y AN, DU CH P, YANG R, et al. Area coverage path planning for tilt-rotor unmanned aerial vehicle based on enhanced genetic algorithm[J]. *Journal of Zhejiang University (Engineering Science)*, 2024, 58(10): 2031-2039.
- [13] 章文俊, 林保瞳, 孟祥坤, 等. 基于改进灰狼算法的北极冰区船舶路径规划研究[J]. *中国航海*, 2024, 47(4): 11-18.
- ZHANG W J, LIN Y T, MENG X K, et al. Research on ship path planning in Arctic ice area based on improved Gray Wolf Algorithm[J]. *Navigation of China*, 2024, 47(4): 11-18.
- [14] 瞿少康, 胡伟, 柴凯凯, 等. 基于改进斑翠鸟优化算法的无人机组网路径规划[J]. *探测与控制学报*, 2025, 47(5): 51-59.
- ZI SH K, HU W, CHAI K K, et al. Path planning for unmanned aerial vehicle network based on improved pied kingfisher optimizer[J]. *Journal of Detection & Control*, 2025, 47(5): 51-59.
- [15] 唐宏伟, 罗佳强, 邓嘉鑫, 等. 基于学习机制蚁群算法的移动机器人路径规划[J]. *现代制造工程*, 2024(12): 48-53, 129.
- TANG H W, LUO J Q, DENG J X, et al. Path planning of mobile robot based on learning mechanism ant colony algorithm [J]. *Modern Manufacturing Engineering*, 2024(12): 48-53, 129.
- [16] LAMINI C, BENHLIMA S, ELBEKRI A. Genetic algorithm based approach for autonomous mobile robot path planning [J]. *Procedia Computer Science*, 2018, 127: 180-189.
- [17] AB WAHAB M N, NAZIR A, KHALIL A, et al. Improved genetic algorithm for mobile robot path planning in static environments [J]. *Expert Systems with Applications*, 2024, 249: 123762.
- [18] YAN F, CHU J, HU J, et al. Cooperative task allocation with simultaneous arrival and resource constraint for multi-UAV using a genetic algorithm[J]. *Expert Systems with Applications*, 2024, 245: 123023.
- [19] SHI K, WU Z, JIANG B, et al. Dynamic path planning of mobile robot based on improved simulated annealing algorithm[J]. *Journal of the Franklin Institute*, 2023, 360(6): 4378-4398.
- [20] ZHOU C, LIU G, LIAO S. Probing dominant flow paths in enhanced geothermal systems with a genetic algorithm inversion model [J]. *Applied Energy*, 2024, 360: 122841.

作者简介



刘俊毅(通信作者), 2023 年于临沂大学获得学士学位, 现为辽宁工程技术大学硕士生兼科研助理, 主要研究方向为机器人设计及路径规划。

E-mail: 1312529551@qq.com

Liu Junyi (Corresponding author)

received his B. Sc. degree from Linyi University in 2023. He is now a M. Sc. candidate and research assistant at Liaoning Technical University. His main research interests include robot design and path planning.



汪洋, 2023 年于苏州科技大学获得学士学位, 现为辽宁工程技术大学硕士, 主要研究方向为: 电机设计及优化。

E-mail: 2356384131@qq.com

Wang Yang received his B. Sc. degree

from Suzhou University of Science and Technology in 2023. He is now a M. Sc. candidate at Liaoning Technical University. His main research interests include motor design and optimization.