

基于改进蛇优化算法的轮式机器人路径规划

冯志强, 李磊, 魏铭毅

(国电电力双维内蒙古上海庙能源有限公司, 内蒙古鄂尔多斯 016200)

摘要:为解决轮式机器人路径规划中效率低、寻优速度慢等问题,提出一种改进的蛇优化算法(improved snake optimizer, ISO)。在初始阶段引入正弦混沌映射扩大算法寻优空间,提升解的质量。同时设计了一种双向搜索策略,在最佳和最差个体引导的两个方向上逼近全局最优值,使收敛速度更快。并在算法中增加改进的进化种群动力机制,替换质量较差的个体从而提高种群质量。另外利用精英对立学习策略来提高算法的局部开发性能。仿真结果表明,ISO 算法在轮式机器人路径规划过程中,相比其它对比算法各项指标更优,寻优效率更高,可以有效帮助轮式机器人完成规划任务。

关键词:蛇优化算法;正弦混沌映射;双向搜索;精英对立学习;轮式机器人;路径规划

中图分类号:TP242;TP18

文章编号:1000-0682(2024)03-0072-05

文献标识码:A

DOI:10.19950/j.cnki.CN61-1121/TH.2024.03.013

Wheeled robot path planning based on improved snake optimizer

FENG Zhiqiang, LI Lei, WEI Mingyi

(Guodian Electric Shuangwei Inner Mongolia Shanghai Temple Energy Co., Ltd., Inner Mongolia Ordos 016200, China)

Abstract:To solve the issues of inadequate productivity and sluggish optimization velocity in wheeled robot path planning, an improved snake optimizer (ISO) was proposed. In the initial stage, sine chaotic mapping expansion algorithm is introduced to optimize the space and improve the quality of the solution. A bidirectional search strategy is devised to approximate the global optimal value simultaneously in the two directions led by the best and the worst individual, which makes the convergence speed faster. The improved evolutionary population dynamic mechanism is added to the algorithm to replace the poor quality individuals so as to improve the population quality. In addition, utilizing the elite opposition-based learning strategy is used to improve the local development ability of the algorithm. The simulation results show that the ISO algorithm performs better in various indicators and has higher optimization efficiency compared to other comparative algorithms in wheeled robot path planning process and can effectively help the wheeled robot to complete the planning task.

Keywords:snake optimizer; sine chaotic mapping; bidirectional search; elite opposition-based learning; wheeled robot; path planning

0 引言

轮式机器人是一种能够自主完成工作的机械装置,在极端环境下可以代替人类执行特定的任务。

随着科技的进步,各领域内轮式机器人得到了广泛的应用^[1-5]。在复杂未知环境中完成路径规划任务,需要机器人具备自主学习能力和对环境的探索能力。

路径规划是机器人导航领域的热点课题之一,它的主要功能是在环境中寻得一条起点和目标点之间的最佳无碰路径。为此研究人员开发出了许多路径规划算法。一般来说,路径规划算法可以分为传统方法和启发式方法。传统方法有人工势场法^[6],

收稿日期:2024-01-31

基金项目:陕西省重点研发计划(2020ZDLGR07-06)

第一作者:冯志强(1979—),男,工程师,主要研究方向为机器人技术。E-mail:zhi64136@163.com

A*算法^[7]和RRT算法^[8-9]、强化学习^[10]等。但这些方法有时很难进一步提高路径搜索和路径优化的效率,导致利用率逐渐下降。启发式算法主要有粒子群算法^[11],蚁群算法^[12],哈里斯鹰算法^[13]和灰狼算法^[14]等。启发式算法有着较强的寻优能力,但也同时存在着易陷入局部最优、搜索效率低等问题。

以蛇的交配行为为灵感,Hashim等人于2022年提出了蛇优化算法(snake optimizer, SO)^[14]。它具有计算简单、效率高的优点,然而与大多数优化算法一样,SO同样也存在着收敛速度慢、容易陷入局部最优等问题。因此,该文在SO算法基础之上引入了多种策略进行改进并提出了ISO算法。首先,通过引入正弦混沌映射改进初始种群的多样性。其次,在探索阶段加入双向搜索策略,来加速算法的收敛。然后,通过将变异操作与进化群体动力学相结合提出了种群进化动力机制,从而提升解的质量。最后,利用精英对立学习的引导使得精英个体的邻域空间得到充分利用,扩大了蛇优化算法的搜索范围,避免陷入局部最优。

1 蛇优化算法

SO算法受蛇交配行为启发而提出。SO的数学模型描述如下: y_i 指代维数为 d 、大小为 N 的种群中第 i 个个体的坐标向量。初始化过程表示为:

$$y_i = lb + r \times (ub - lb) \quad (1)$$

式中: lb 和 ub 为下限和上限; r 为 $[0,1]$ 中的随机常值。种群由雌性和雄性组成,假设雌性和雄性各占50%。在迭代之前,将大小为 N 的种群分成两个子群具体如下:

$$N_m \approx N/2, N_f = N - N_m \quad (2)$$

式中: N_m 为雄性子群; N_f 为雌性子群。

食物质量决定了SO是进入探索阶段还是开发阶段,是否进行交配行为由温度决定。食物质量 Q 和温度 T 定义如下:

$$T = e^{-\frac{g}{c}} \quad (3)$$

$$Q = c_1 \times e^{\frac{g-G}{c}} \quad (4)$$

式中: g 和 G 分别为当前和最大迭代次数; c_1 为常数,该文取 $c_1 = 0.5$ 。

当蛇处于不适宜的环境(没有食物)时,它们只会在周围寻找食物。因此,当食物质量 Q 小于一定范围时,SO进入搜索阶段,每个个体随机更新搜索区域内的位置。对于雄性群体,位置更新表示如下:

$$y_{i,m}(g+1) = y_{r,m}(g) \pm c_2 \times A_m \times ((ub - lb) \times rand + lb) \quad (5)$$

式中: $y_{r,m}$ 为随机雄性; $rand$ 为在 $[0,1]$ 中随机生成的数; c_2 为常数,该文取 $c_2 = 0.5$ 。 A_m 为雄性蛇交配能力,其表达式为:

$$A_m = e^{-\frac{F_{r,m}}{F_{i,m}}}$$

式中: $F(\cdot)$ 为适应度值。

对于雌性子群,探索阶段的位置更新与雄性类似如下:

$$y_{i,f}(g+1) = y_{r,f}(g) \pm c_2 \times A_f \times ((ub - lb) \times rand + lb) \quad (6)$$

$$A_f = e^{-\frac{F_{r,f}}{F_{i,f}}} \quad (7)$$

式中: A_f 为第 i 条雌蛇交配的能力。

2 改进策略

2.1 正弦混沌映射

传统SO算法随机生成种群个体,容易出现种群个体质量低和分布不均的问题。该文引入正弦混沌映射提高种群的整体质量,使种群分布均匀多样,提高了算法的收敛速度和精度。具体表达式如下:

$$x_{k+1} = \frac{\alpha}{4} \times \sin(\pi x_k) \quad (8)$$

式中: k 为迭代次数; A 为分岔参数; α 为常数,范围为 $(0, 4]$ 。设置初始值 x_0 ,正弦混沌序列 x 随 α 变化的分布如图1所示。

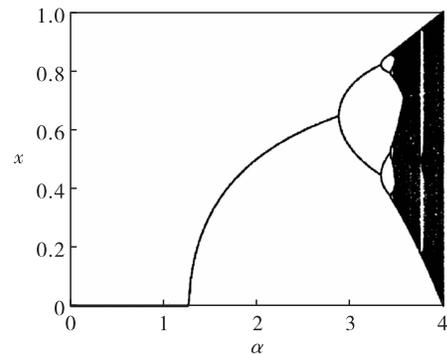


图1 不同分叉参数下的种群序列分布

如图1所示,当 α 在 $(0, 3.436]$ 之间时,序列 x 呈现有限分布,并随着 α 的增加,序列 x 逐渐呈现准发散状态。当 α 在 $(3.436, 4]$ 之间时,出现混沌现象。当 $\alpha = 4$ 时,序列具有最佳的分布和最高的种群随机性。因此,选择 $\alpha = 4$ 处的正弦混沌序列来初始化种群。

式(8)用于将正弦混沌后生成的变量值映射到种群内的单个蛇,从而对蛇群进行初始化,其具体表

达式如下:

$$y_i = lb + x_{k+1} \cdot (ub - lb) \quad (9)$$

2.2 双向搜索策略

双向搜索策略是一种利用最佳和最差个体在搜索区域内进行探索的策略。它的设计目的是让算法远离最差的个体,同时接近最好的个体。具体表示式为:

$$y_{i,m}(g+1) = y_{i,m}(g) + r_1 \times (y_{best,m} - y_{i,m}(g)) - r_2 \times (y_{worst,m} - y_{i,m}(g)) \quad (10)$$

$$y_{i,f}(g+1) = y_{i,f}(g) + r_1 \times (y_{best,f} - y_{i,f}(g)) - r_2 \times (y_{worst,f} - y_{i,f}(g)) \quad (11)$$

式中: $y_{best,f}$ 和 $y_{worst,f}$ 分别为最好和最差的雌性; $y_{best,m}$ 和 $y_{worst,m}$ 为最好和最差的雄性; r_1 和 r_2 为均匀产生的随机数。

传统 SO 算法通过随机搜索寻优区域,该区域具有低随机性和高不确定性,并且搜索范围很窄。BDS 使算法在充分利用搜索区域的同时接近最优值,加快了收敛速度。

2.3 种群进化动力机制

种群进化动力机制旨在通过消除劣势个体来改善解的质量。为增加种群多样性,在种群进化动力机制中引入了突变操作,具体表达式如下:

$$O_i(g+1) = \begin{cases} y_{best}(g) + \text{sign}(r - 0.5) \times \\ (ub - lb \times r + lb), \text{ if } \text{rand} < 0.5 \\ y_i(g) + \text{sign}(r - 0.5) \times \\ (ub - lb \times r + lb), \text{ else} \end{cases} \quad (12)$$

$$y_i(g+1) = \begin{cases} O_i(g+1), \\ \text{if } F(O_i(g+1)) < F(y_i(g+1)) \\ ub - lb \times r + lb, \text{ else} \end{cases} \quad (13)$$

式中: $i = 1, 2, \dots, N/2$, O_i 为在历史个体周围随机搜索的新位置。

种群进化动力机制中的突变操作将在质量更好的个体上进行。除了替换最差的个体外,对前 50% 的个体应用变异算子产生变异个体并进行贪心操作,具体如下:

$$My_i(g) = y_{p_1}(g) + F \cdot (y_{p_2}(g) - y_{p_3}(g)) \quad (14)$$

式中: $My_i(g)$ 为贪心操作后种群个体; y_{p_i} ($i = 1, 2, 3$)为原种群内个体。 F 为比例因子,具体表示如下:

$$F = \frac{1}{2}(\sin(2\pi \times f_{req} \times g) \times (g + G) + 1) \quad (15)$$

式中: f_{req} 为正弦函数的振动频率。

$$y_i(g+1) = \begin{cases} My_i(g), \text{ if } F(My_i(g)) < F(y_i(g+1)) \\ y_i(g+1), \text{ else} \end{cases} \quad (16)$$

围绕最佳个体进行搜索可以改善种群规模的中位数。当新解的质量不如原解时,将在搜索区域内重新搜索。

2.4 精英对立学习策略

精英对立学习策略本质上是一个贪婪的操作。首先利用精英个体计算出对立解,然后根据适应度值选择当前解及其精英对立解进行下一次迭代。

该文中精英个体 Ey_n 是指种群中最高 EN 的个体,其中 EN 是精英解的数量。那么对于当前解 y_i ,其精英对立解计算如下:

$$e\bar{y}_{i,j}(g) = k \cdot (EA_j(g) + EB_j(g)) - y_{i,j}(g) \quad (17)$$

$$EA_j(g) = \min(Ey_j(g)),$$

$$EB_j(g) = \max(Ey_j(g)) \quad (18)$$

$$e\bar{y}_{i,j}(g) = \text{rand} \cdot (EB_j(g) - EA_j(g)) + EA_j(g), \text{ if } e\bar{y}_{i,j} < lb_j \parallel e\bar{y}_{i,j} > ub_j \quad (19)$$

式中: $n = 1, 2, \dots, EN$, EN 取 $0.1 * N$ 。作为动态边界, $EA_j(g)$ 、 $EB_j(g)$ 分别为精英种群中第 j 维的最小值和最大值。

以上几种策略是对 SO 算法的改进。探索中引入双向搜索是为了加速收敛。在最好的和最坏的个体的合作指导下,使种群接近最优解。当算法触发交配模式时,将执行种群进化动力机制。首先,式(12)(13)用于计算新位置以替换较差的一半个体。此操作可以有效利用搜索区域并提高种群质量。其次,根据方程式(14)~(16),更好的一半发生突变,从而增加种群多样性,使算法以更大的概率接近最优值。此外,精英对立学习可以获得当前可行解的对立解,这不仅有利于利用精英个体的邻域空间,也增强了算法局部开发能力。该文算法实现的大致流程如下:

(1) 建立环境设置起终点,初始化参数,同时利用正弦混沌生成种群;

(2) 计算食物质量及当前环境温度值,判断进入哪个阶段;

(3) 若 Q 、 T 小于设置的条件,则算法进入搜索阶段,否则进入局部寻优阶段;

(4) 当前迭代次数 t 与最大迭代次数 T_{max} 作比较,若 $i > T_{max}$ 则退出迭代,若 $i < T_{max}$ 则返回继续迭代;

(5)算法结束,输出最优结果。

3 仿真及实验

3.1 仿真测试

在 MATLAB2020a 平台进行仿真实验,将蛇群大小设为 30,最大迭代次数为 500。为检验 ISO 算法在不同场景下的路径规划性能,分别在 $20\text{ m} \times 20\text{ m}$ 和 $50\text{ m} \times 50\text{ m}$ 两个不同大小场景下进行仿真,同时与 SO 算法、文献[13]算法进行对比。结果如图 2(a)及图 2(b)所示,性能指标对比如表 1 所示。

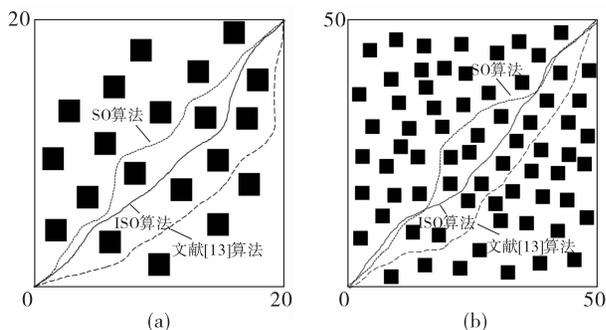


图 2 仿真结果

表 1 指标对比

环境	指标	ISO 算法	SO 算法	文献[13]算法
20×20	路径长度/m	30.26	31.45	31.07
	规划时间/s	1.17	1.36	1.24
50×50	路径长度/m	73.25	76.33	75.19
	规划时间/s	2.25	2.68	2.43

根据图 2,3 种算法均可以在 2 种尺度规模的场景下找到无碰路径。根据表 4 的指标对比可以看出,在 $20\text{ m} \times 20\text{ m}$ 场景下,ISO 算法路径长度为 30.26 m,运算耗时为 1.17 s,相较于其他 2 种算法,路径长度最高缩短 3.78%,最低缩短 2.61%;规划时间最高节省 13.97%,最低节省 5.65%。在 $50\text{ m} \times 50\text{ m}$ 场景下,ISO 算法路径长度为 73.25 m,运算耗时为 2.25 s,路径长度最高缩短 4.04%,最低缩短 2.58%;规划时间最高节省 16.04%,最低节省 7.41%。因此从结果来看,ISO 算法具备优越的算法性能,可以提升路径规划效率。

3.2 实物实验

为测试该文算法可行性,在实际环境中进行实车验证。实物主机为搭载 XR-Lidar S1 雷达的 Jetbot 轮式机器人,从机为笔记本电脑,处理器为英特尔酷睿 i5-6300,运行内存为 16GB。在 Ubuntu 的 ROS Kinetic 系统中,通过机器人自身配置的激光雷

达、惯性、深度相机等设备收集周围环境信息,进行二维地图构建和路径规划。该文选择一个 $6\text{ m} \times 8\text{ m}$ 的室内环境进行验证,并在地面上随机设置了几个纸盒作为障碍物。具体路径规划过程如图 3、4 所示。

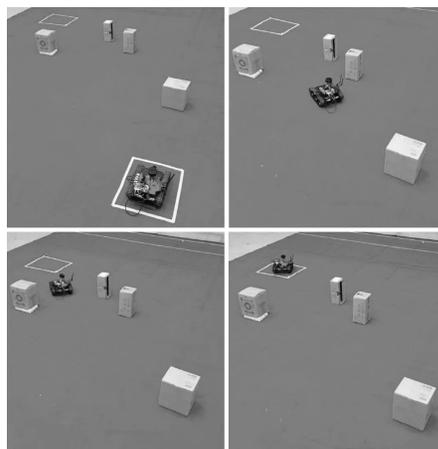


图 3 实际场景 1 规划过程

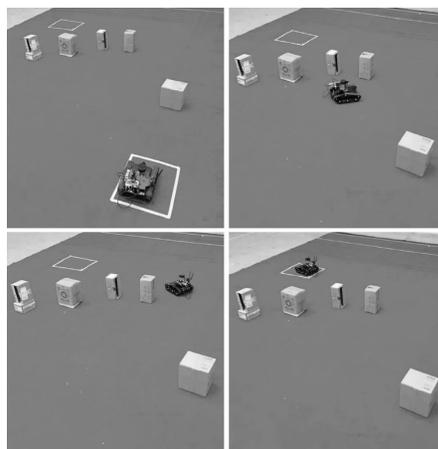


图 4 实际场景 2 规划过程

从环境地图中的路径规划可以看出,改进的算法成功地规划出了一条安全无碰的可行路径,验证了该文改进蛇优化算法的可行性,表明在实际场景中该文算法可以完成轮式机器人路径规划作业。

4 结束语

该文通过将混沌映射、双向搜索、种群进化动力机制和精英对立学习等策略与 SO 算法相结合,提出了一种 ISO 算法,并将其应用在轮式机器人路径规划问题上。仿真结果表明,在 $20 \times 20\text{ m}$ 场景下,ISO 算法路径长度缩短 2.61%~3.78%,规划时间减少 5.65%~13.97%。在 $50 \times 50\text{ m}$ 场景下,ISO 算法路径长度缩短 2.58%~4.04%,规划时间减少 7.41%~16.04%。最后基于实车平台,验证了该文算法的可行性。

