

包装生产线的操作臂轨迹规划的一种优化方法

景会成, 李嘉琪, 王福斌, 曾凯

(华北理工大学 电气工程学院, 河北 唐山 063210)

摘要: **目的** 为提高仓库存储、包装生产领域操作臂运动效率, 提出一种基于混合优化的鲸鱼算法(MAIWOA)来求解操作臂始末2点运动过程中时间最优值。**方法** 首先是通过混沌算法优化初始化种群; 其次, 提出一种对勾函数和包围收缩学习机制来帮助算法跳出局部收敛; 然后利用改进的反向树形拓扑结构来提高种群探索的多样性; 采取改进收敛因子和自适应权重机制来平衡前中期的全局探索能力和后期的收敛性。**结果** 该算法在收敛速度和收敛精度上均有可观的提升。将该算法应用到五次多项式插值关节运动时间最优求解上取得了很好的效果, 关节1、2、3的运动时间由20 s分别缩减到了9.642 5、9.251 5、10.787 s, 效率分别提高了51.79%、53.74%、46.07%。**结论** 将MAIWOA应用到一般性操作臂轨迹规划上, 可提高生产线上操作臂的执行效率。

关键词: 惯性权重; 收敛因子; 反向学习机制; 优化的树形拓扑结构; 时间最优轨迹规划

中图分类号: TP241.2 **文献标识码:** A **文章编号:** 1001-3563(2023)07-0211-11

DOI: 10.19554/j.cnki.1001-3563.2023.07.024

An Optimization Method for Trajectory Planning of Manipulator in Packaging Production Line

JING Hui-cheng, LI Jia-qi, WANG Fu-bin, ZENG Kai

(School of Electrical Engineering, North China University of Technology, Hebei Tangshan 063210, China)

ABSTRACT: The work aims to propose a whale algorithm based on hybrid optimization (MAIWOA) to solve the time optimal value of the manipulator in the process of two-point movement, so as to improve the movement efficiency of the manipulator in warehouse storage, packaging and production. Firstly, the initial population was optimized by chaos algorithm; Secondly, a learning mechanism of tick function and bounding contraction was proposed to help the algorithm jump out of local convergence; Then an improved reverse tree topology was used to improve the diversity of population exploration; The improved convergence factor and adaptive weight mechanism were adopted to balance the global exploration ability in the early and middle stages and the convergence in the later stage. Results showed that the convergence speed and accuracy of this algorithm were improved significantly. The algorithm was applied to the optimal solution of joint motion time of quintic polynomial interpolation, and good results were obtained. The motion time of joints 1, 2 and 3 was reduced from 20 s to 9.642 5, 9.251 5 and 10.787 s respectively, and the efficiency was increased by 51.79%, 53.74% and 46.07% respectively. It is concluded that applying MAIWOA to the trajectory planning of general manipulator can improve the execution efficiency of manipulator in production line.

KEY WORDS: inertia weight; convergence factor; reverse learning mechanism; optimized tree topology; time optimal trajectory planning

收稿日期: 2022-10-26

基金项目: 河北省自然科学基金(E2019209492)

作者简介: 景会成(1966—), 男, 硕士, 副教授, 硕导, 主要研究方向为电子信息、智能控制、新能源应用方面。

随着自动化产业的蓬勃发展,仓库存储的流水线生产调度,离不开机器人的轨迹运动,经过良好优化的轨迹运动路线,可使操作臂执行更加高效。例如,自动化药房系统是近几年自动化产业发展新兴的热门研究领域,带机械臂的堆垛机在对多种药物存取调度控制过程中,其操作臂时间运行效率就显得尤为重要。

轨迹规划是机器人末端执行器运动必不可少的前提规划。目前,国内外对操作臂轨迹规划的研究有很多。例如,赵志强等^[1]提出改进的可达性分析算法的时间最优轨迹规划,该算法考虑了更为复杂的多种约束条件下的最优时间规划,但算法计算较为复杂不实用;Yu 等^[2]提出使用三次均匀 B 样条插值算法导出运动曲线和利用遗传算法求得最优运动时间;杨星涛等^[3]提出 3 次多项式插值结合改进遗传算法求解时间最优轨迹规划问题,将随机排序策略应用于中间参数的调整,使其更适应于对目标函数的探索;李虹等^[4]提出的在非均匀 B 样条曲线融入自适应粒子群优化算法,可使各关节平稳地到达目标点;郭鑫鑫等^[5]提出改进萤火虫算法在轨迹规划上求解时间最优,一定程度上取得不错的应用效果。但以上使用的智能优化算法都存在收敛速度较慢,收敛精度不足的问题等,因此,求解轨迹规划的时间最优方法尚有开发空间。

文中研究旨在提出一个更好的应用方法,来求解轨迹规划时间最优。将对原鲸鱼优化算法进行深入分析改进,提出的 MAIWOA 算法从多角度进行改进升级,使得算法寻优能力更强、更快,避免陷入局部最优。将改进算法应用于五次多项式插值的轨迹规划时间最优求解上,进一步检验算法性能。已知关节角度、角速度、角加速度的 3 种限制条件,算法以求解各关节始末 2 点运动时间段最短为目标进行优化收敛。

1 问题描述

1.1 定义目标函数

以六轴串联操作臂为例,定义目标函数为某一关节运动由始到末两间插值点的时间段之和见式(1)。

$$\min f = \sum_{i=1}^{n-1} h_i \quad (1)$$

1.2 五次多项式插值轨迹规划

以五次多项式函数对操作臂运动轨迹进行插值。关节角度位移函数表达式如下:

$$\theta(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 + a_4 t^4 + a_5 t^5 \quad (2)$$

$$\begin{cases} \dot{\theta}(t) = a_1 + 2a_2 t + 3a_3 t^2 + 4a_4 t^3 + 5a_5 t^4 \\ \ddot{\theta}(t) = 2a_2 + 6a_3 t + 12a_4 t^2 + 20a_5 t^3 \end{cases} \quad (3)$$

$$\begin{cases} \theta(0) = a_0 \\ \theta(t_f) = a_0 + a_1 t_f + a_2 t_f^2 + a_3 t_f^3 + a_4 t_f^4 + a_5 t_f^5 \\ \dot{\theta}(0) = a_1 \\ \dot{\theta}(t_f) = a_1 + 2a_2 t_f + 3a_3 t_f^2 + 4a_4 t_f^3 + 5a_5 t_f^4 \\ \ddot{\theta}(0) = 2a_2 \\ \ddot{\theta}(t_f) = 2a_2 + 6a_3 t_f + 12a_4 t_f^2 + 20a_5 t_f^3 \end{cases} \quad (4)$$

式(4)为函数系数满足始末位置、速度、加速度 6 个约束条件。

对式(4)等式变换可求得 6 个系数值,将系数值带入式(2)、式(3),求得各关节角度、角速度、角加速度 3 种曲线函数。

2 混合策略改进鲸鱼优化算法

2.1 标准的鲸鱼优化算法

鲸鱼优化算法(WOA)是在 2016 年,以 Mirjalili 等^[6]为首的著名学者提出的一种仿生学算法。座头鲸狩猎行动分为如下 3 个阶段。

1) 包围猎物与泡泡网狩猎。鲸鱼围捕猎物时,需要以 50% 的概率来划分收缩包围和螺旋泡泡网狩猎 2 种执行模式,见式(5)。

$$\bar{X}(t+1) = \begin{cases} \bar{X}^*(t) - \bar{A} \cdot \bar{D} & \text{if } p < 0.5 \\ \bar{D} \cdot \exp(bl) \cdot \cos(2\pi l) + \bar{X}^*(t) & \text{if } p \geq 0.5 \end{cases} \quad (5)$$

式中: $\bar{D}$ 为当前种群中随机成员与最佳成员之间的浮动距离; t 为迭代次数; $p \in [0,1]$; $\bar{X}^*$ 为当前种群最优成员位置; $\bar{D}$ 为种群最佳成员和随机成员之间的距离, $\bar{D} = |\bar{X}^*(t) - \bar{X}(t)|$; b 为螺旋形状参数; l 为[0,1]的随机数。

系数向量 $\bar{A}$ 、 $\bar{C}$ 计算式见式(6)。

$$\begin{cases} \bar{A} = 2 \times \bar{a} \times \bar{r}_1 - \bar{a} \\ \bar{C} = 2 \times \bar{r}_2 \end{cases} \quad (6)$$

式中: $\bar{a} = 2 - 2 \cdot \frac{t}{t_{\max}}$, $\bar{a}$ 从 2 线性衰减到 0; r_1 、 r_2 为[0,1]中的随机实数。

2) 随机搜索猎物。座头鲸种群在围捕进攻猎物的同时,需要在搜索域中进行全局探索,以此来寻找更加优越的目标猎物。算法中以 $|\bar{A}|$ 值大小来进行选择,当 $|\bar{A}| \geq 1$ 时鲸鱼在全局空间内进行随机探索,该策略的数学模型表示为:

$$\begin{cases} \bar{D} = |\bar{C} \cdot \bar{X}_{\text{rand}} - \bar{X}(t)| \\ \bar{X}(t+1) = \bar{X}_{\text{rand}} - \bar{A} \cdot \bar{D} \end{cases} \quad (7)$$

式中: $\bar{X}_{\text{rand}}$ 为当前种群内随机鲸鱼成员矢量位置; $\bar{X}(t+1)$ 为第 $t+1$ 次迭代的鲸鱼成员矢量位置。

2.2 Tent 混沌初始化择优机制

根据研究表明, 一个初始化集群如果具有良好的品质, 则对算法的收敛速度和收敛精度具有很好的帮助^[7]。原 WOA 算法的初始化种群是依赖于 rand 函数, 使初始种群具有无规律的随机性, 不能均匀分布在可行域空间中, 致使初始化种群分布丧失多样性。因此, 文中决定利用 Tent 混沌映射原理来使初始化种群均匀的溅射到各个位置, 丰富种群的多样性^[8-10]。例如, 选取 100 头鲸鱼进行种群初始化, 比较其适应度值, 挑选前 30 头优质鲸鱼种群作为迭代鲸鱼种群, 使初始化不仅具有多样性, 同时还进一步提高其质量。

Tent 混沌映射表达式:

$$x(i+1) = \begin{cases} 2 \cdot x(i) & \text{if } 0 \leq x < 0.5 \\ 2 \cdot [1 - x(i)] & \text{if } 0.5 < x < 1 \end{cases} \quad (8)$$

2.3 对勾函数和包围收缩学习机制

当前解决陷入局部极值问题主要有 2 种方案: 一是在当前最佳位置基础上向其他区域进行搜寻, 二是以放弃当前最佳位置为条件, 转向其他区域进行搜寻^[11]。

为更好地使算法具有跳出局部最优能力, 文中选择第 1 种方案, 采取 2 种机制来配合增强算法的寻优能力。在多峰函数中存在多个局部极值, 单纯依赖本身的包围收缩机制来跳出局部极值是不能实现的。然而跳出局部极值并不是需要鲸鱼位置随便溅射, 通常只是需要在位置矢量附近邻域内溅射, 使其越过当前极值范围并继续收敛下去直到寻找到最优。

根据初等数学函数中的对勾函数思想, 便可将当前迭代停滞的最佳位置矢量 $\vec{X}^*(t)$ 中每一维度位置在其邻域范围通过函数表达式溅射到一个新位置。

对勾函数表达式:

$$f(x) = a \cdot x + \frac{b}{x} \quad a \cdot b > 0 \quad (9)$$

根据测试经验选取 $a=b=0.1$ 为适宜。该算法策略步骤如下:

If 最佳鲸鱼位置适应度值不为零 And 前后适应度比值为 1

While(1)

For i=1:dim

利用公式 (5) 中包围收缩和公式 (9) 分别对当前最佳鲸鱼位置进行扰动更新。

End

If 当更新后的适应度值不等于优化前的适应度值

选择求得满足条件的适应度值最低的位置为最佳位置矢量。

End

End

End

2.4 收敛因子优化

根据原 WOA 算法中描述, 算法的探索与收敛能力取决于收敛因子 $\vec{a}$ 的变化情况。原 WOA 算法中参数 $\vec{a}$ 的值呈线性衰减至 0, 不利于算法探索功能在中后期发挥, 容易落入局部陷阱^[12-14], 因此, 应尽可能在前中期使收敛因子保持较高的数值以及较低的下降速率。本文使收敛因子呈非线性衰减, 使全局探索功能在算法运行前中期得到良好发挥, 并且使后期快速地收敛到最优解, 使算法具有更好的收敛速度与精度。总结得出公式见式 (10)。

$$a = \left| \sqrt{-\frac{2}{250} \cdot t + 4} + 1.5 \cdot \exp \left[-\frac{(t-450)^2}{2000} \right] \cdot \sin \frac{\pi \cdot t}{t_{\max}} \right| \quad (10)$$

2.5 惯性权重系数

惯性权重作为仿生学算法中重要的参数调节指标, 其对系统的探索能力有显著提升^[15]。根据原 WOA 算法的描述, 算法在迭代后期收敛时鲸鱼个体容易停留在理论位置附近, 无法进一步收敛, 因此, 本文引用惯性权重系数对算法的迭代公式进行更新调整。权重系数 W 随时间的变化曲线以初始值 1 开始, 前中期缓慢衰减, 后期快速衰减的方式, 使得算法在前中期保持较强的全局探索能力, 算法后期能够进一步收敛到全局最优解, 避免陷入理论局部极值。结合卡方分布的逆累积分布函数总结得出式 (11)。

$$\begin{cases} F(W) = \int_0^W t^{\frac{1}{2}} \cdot \exp(-\frac{t}{2}) dt \\ W = F^{-1}(0.9 - \frac{0.9 \cdot t}{t_{\max}}) \end{cases} \quad (11)$$

2.6 优化的树形拓扑结构

原 WOA 算法采用围绕一个鲸鱼核心进行探索, 种群探索模式单一化, 缺乏种群探索多样性。当陷入局部最优时, 很难使整个种群快速跳出^[7]。本文对树形拓扑结构进行了修改, 主要思想是将初始化种群 (大树) 划分为多个小种群并定义为子树。原树形结构由大树将信息传递给各子树枝干直到叶节点。现将树形结构将信息传递顺序进行颠倒, 在某子树中所有子节点进行信息比较, 并将最佳子节点信息传递给父节点, 其他子节点都向父节点看齐, 如图 1a 所示。

为了综合样本空间越大探索能力越强和多个小样本分别探索多样性能力这两者的优点, 本文将种群数量进行如下划分: 假定选取 30 只鲸鱼个体、分为 3 个子树种群, 则每个子树种群中包含 10 只鲸鱼个体, 即 10 个叶节点, 如图 1b 所示。

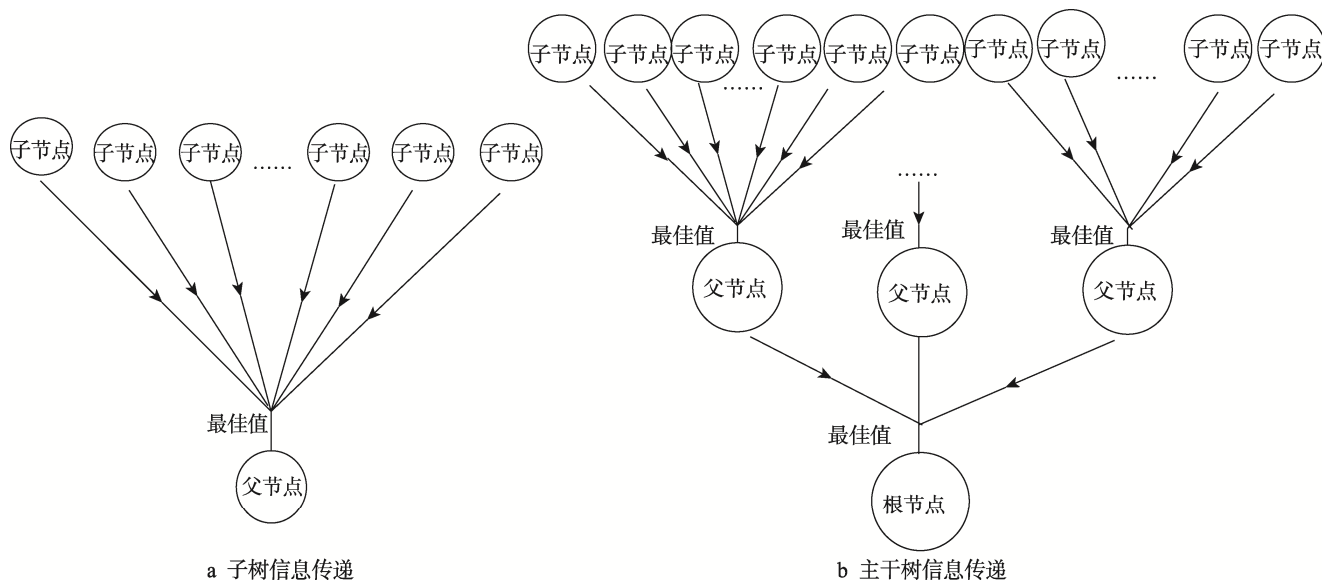


图 1 树形结构信息传递
Fig.1 Tree structure information transmission

算法运行第 1 阶段：划分为 3 个子树分别进行迭代运算，推选出当前种群中最佳鲸鱼领导者并将其推选为父节点，同时其余叶节点在迭代过程中都向所属树系中父节点逼近靠拢。以此类推，可获得 3 个父节点。

该阶段使得各子树独立进行探索，有利于算法探索结果多样性，使得鲸鱼种群有更强的全局探索能力。

算法运行第 2 阶段：将得到的 3 个父节点视为新的一支树，称其为主干树，并对其进行迭代运算推选出最佳鲸鱼代理，将其视为树形结构最终的根，即根代表着鲸鱼算法迭代的全局最优值。同时各父节点将要向根节点进行逼近靠拢，保证了根节点对各子节点的全局领导力，有利于拉动种群整体朝全局最优值迭代收敛。

由于 WOA 算法种群迭代是互相依赖迭代收敛，这就导致了迭代速度是有限的。为了最大化收敛速度，不使迭代资源浪费，同时为平衡算法全局探索能力和收敛速度，需要加快子树整体向其根茎迭代逼近的收敛速度，这样有利于提高种群的收敛精度。采取螺旋收缩和收缩包围机制于主干树迭代，保证全局快速收敛。综上，优化后的树形拓扑结构更新公式如下。

1) 各子树的迭代按照原 WOA 算法并融入改进后的参数，根据式 (5) 和式 (7) 进行包围收缩、螺旋更新和随机学习，将公式中 $\bar{X}^*$ 替换为 $\bar{X}_{father_node}(t)$ 。

螺旋攻击与包围收缩公式见式 (12)。

$$\bar{X}(t+1) = \begin{cases} W \cdot \bar{X}^*(t) - \bar{A} \cdot \bar{D} & \text{if } p < 0.5, \\ \bar{D} \cdot b \cdot l + W \cdot \bar{X}^*(t) & \text{if } p \geq 0.5, \end{cases} \quad (12)$$

随即学习公式见式 (13)。

$$\bar{X}(t+1) = W \cdot \bar{X}_{rand} - \bar{A} \cdot \bar{D} \quad (13)$$

2) 根据阿基米德螺旋更新原理^[16]将该阶段螺旋更新公式设定为：

$$\begin{cases} \bar{X}_{new_father_node}(i) = ones(i, d_{im}) \cdot \\ \left(\bar{D} \cdot b \cdot l + W \cdot \bar{X}_{root_node} \right) \\ \bar{D} = \left| \bar{X}_{root_node}(j) - \bar{X}_{father_node}(i, j) \right| \end{cases} \quad (14)$$

$b \cdot l$ 的变化方式控制着鲸鱼种群螺旋更新的变化速度，将其控制在 (1,3) 内，其值越小变化速度越快，可使得阶段 2 父节点向根茎收缩整体速度变快。同时根据“若算法中只含有螺旋更新机制，则不能依概率收敛到最优解”的原理^[17]，需要增加收缩包围机制以帮助主干树迭代过程中具有跳出局部最优值的能力。主干树收缩包围机制公式设定见式 (15)。

$$\begin{cases} \bar{X}_{n_father_node}(i) = W \cdot \bar{X}_{root_node} - \bar{A} \cdot \bar{D}^* \\ \bar{D}^* = \left| \bar{X}_{root_node} - \bar{X}_{new_father_node}(i) \right| \\ \bar{A} = 2 \times \bar{a} \times \bar{r} - \bar{a} \end{cases} \quad (15)$$

每次循环迭代的过程中对主干树进行一次螺旋更新与收缩包围，通过贪婪策略推选出最优解值。综上，既保证了优化树形拓扑结构的全局探索种群多样性能力，又维持了算法整体的收敛速度，保证了最终解值为全局最优解。

2.7 MAIWOA 算法步骤

总结给出 MAIWOA 鲸鱼优化算法的执行步骤如下：

1) 对鲸鱼算法各参数进行初始化定义，种群规模设定为 N ，设定鲸鱼维数为 d_{im} 螺旋形状控制参数

为 b , 最大迭代次数为 $t_{\max}$ 。

2) 使用 Tent 混沌映射对鲸鱼成员进行位置初始化, 分别对子树 1、2、3 初始化 50 头鲸鱼个体, 选取其适应度值最高的前 10 头鲸鱼个体作为其子树种群, 对子树进行编码、鲸鱼种群划分, 并推选出父节点、根节点。

3) 按照式 (10)、式 (11) 和式 (6) 分别更新收敛因子 a 、惯性权重 W 、参数 A 和 C , 更新概率 $p \in [0, 1]$ 。

4) 枝叶围绕子树父节点按照改进后的 WOA 算法进行位置更新。将 $\overline{X}^*(t)$ 替换为 $\overline{X}_{\text{father_node}}(t)$, 根据式 (12)、式 (13) 进行包围收缩、螺旋更新和随即学习, 并更新最优位置和适应度值。

5) 主干树迭代先推选出最优父节点为根节点, 然后利用式 (14) 和式 (15) 更新主干树成员下一时刻的值, 通过贪婪策略比较前后适应度值选择最优解值, 并竞选出新一轮的全局最佳根节点位置值和适应度值。

6) 每轮子树迭代结束后, 当本轮父节点适应度值与上轮父节点适应度值相同时, 对父节点进行对勾函数机制和包围收缩机制学习, 通过式 (5) 包围收缩和式 (9) 来帮助父节点跳出局部极值。通过对比学习前后的函数值, 更新最佳位置和最佳函数值。当学习后和学习前的适应度值不同时则跳出, 否则循环学习。

7) 检测算法是否抵达停止的条件, 若达到则终止跳出, 返回全局中的当前最佳解值 (根节点) 及适应度值, 反之继续运行步骤 3—6。

2.8 基准函数测试与结果分析

1) 算法选择。文中里选用鲸鱼优化算法 (WOA)、灰狼优化算法 (GWO)、粒子群优化算法 (PSO)、加入 Tent 映射初始化策略和树形优化拓扑的鲸鱼优化算法 (WOA-1)、加入收敛因子和惯性权重的鲸鱼优化算法 (WOA-2) 和加入跳出局部最优的学习机制鲸鱼优化算法 (WOA-3)。对以上所有算法在表 1 中 9 种基准测试函数下重复进行 20 次实验的结果进行对比。对比结果如图 2 所示。

2) 基准函数的选择。这 9 个测试函数的最佳值均设定为 0。前 5 个为单峰函数, 衡量算法的收敛程度; 后 4 个为多峰函数, 衡量算法跳出局部极值的能力。测试函数见表 1。

3) 测试结果分析。在 Matlab2020a 软件、Windows10 系统、8GB 内存等计算机测试环境中, 使每个算法在每个测试函数中分别独立运算 20 次, 从结果中总结出各算法的最优值 (Best)、最差值 (Worst)、平均值 (Mean) 和标准差 (Std), 见表 2。

表 1 9 个基准测试函数
Tab.1 Nine benchmark functions

编号	函数名	维度	搜索区域	理论值
F1	Sphere	30	$[-100, 100]$	0
F2	Schwefel2.22	30	$[-10, 10]$	0
F3	Schwefel1.2	30	$[-100, 100]$	0
F4	Schwefel2.21	30	$[-100, 100]$	0
F5	Rosenbrock	30	$[-30, 30]$	0
F6	Quartic	30	$[-1.28, 1.28]$	0
F7	Rastrigin	30	$[-5.12, 5.12]$	0
F8	Penalized1	30	$[-50, 50]$	0
F9	Penalized2	30	$[-50, 50]$	0

从表 2 数据统计可知, MAIWOA 算法在所有测试函数的评价指标中取得的结果都是最佳的, 更加接近理论最佳值, 尤其是在 F1—F4、F7 函数中取得了理论最佳值 0。

图 2 中, 纵坐标是对数坐标轴以 $\lg$ 为底数, 横坐标为 500 次迭代次数。从收敛曲线可知, 在 F1—F4 的单峰函数中可以看出, 惯性权重和收敛因子在算法寻找最优解时, 起到很好的收敛效果; 在 F5—F9 的多峰函数寻优中可以看出, 对勾函数和包围收缩机制使算法可以更好地跳出局部极值, 同时惯性权重对跳出局部极值能力也起到了很好的作用。由图 2、表 2 可知, MAIWOA 混合算法在提高收敛速度、增强收敛精度和提高跳出局部极值能力三方面间都取得了很好的效果。

3 优化与仿真

通过 Matlab 仿真, 以六自由度的操作臂 PUMA560 为例进行讨论, 如图 3 所示。图 3 中对 6 关节建立了坐标系, 其所有关节均为旋转关节, 其中 a_i 表示两相邻关节的 z 轴距离, d_i 表示两相邻关节的 x 轴距离, θ_i 表示两相邻关节 x 轴的夹角。本文针对其末端执行器给出了 6 个路径点, 在运动学约束条件下对各个关节求解最佳时间值。表 3 为各关约束条件, 各关节中间点轨迹运动信息如表 4 所示。现设定各关节轨迹运动总时间为 20 s, 平均每段时间间隔 4 s。

为更好地模拟实际效果, 将中间过渡点的速度值设定规则: 若两相邻中间点线段的斜率符号为同号, 则速度为两斜率的平均值; 若两线段斜率符号不同, 则设置为 0。

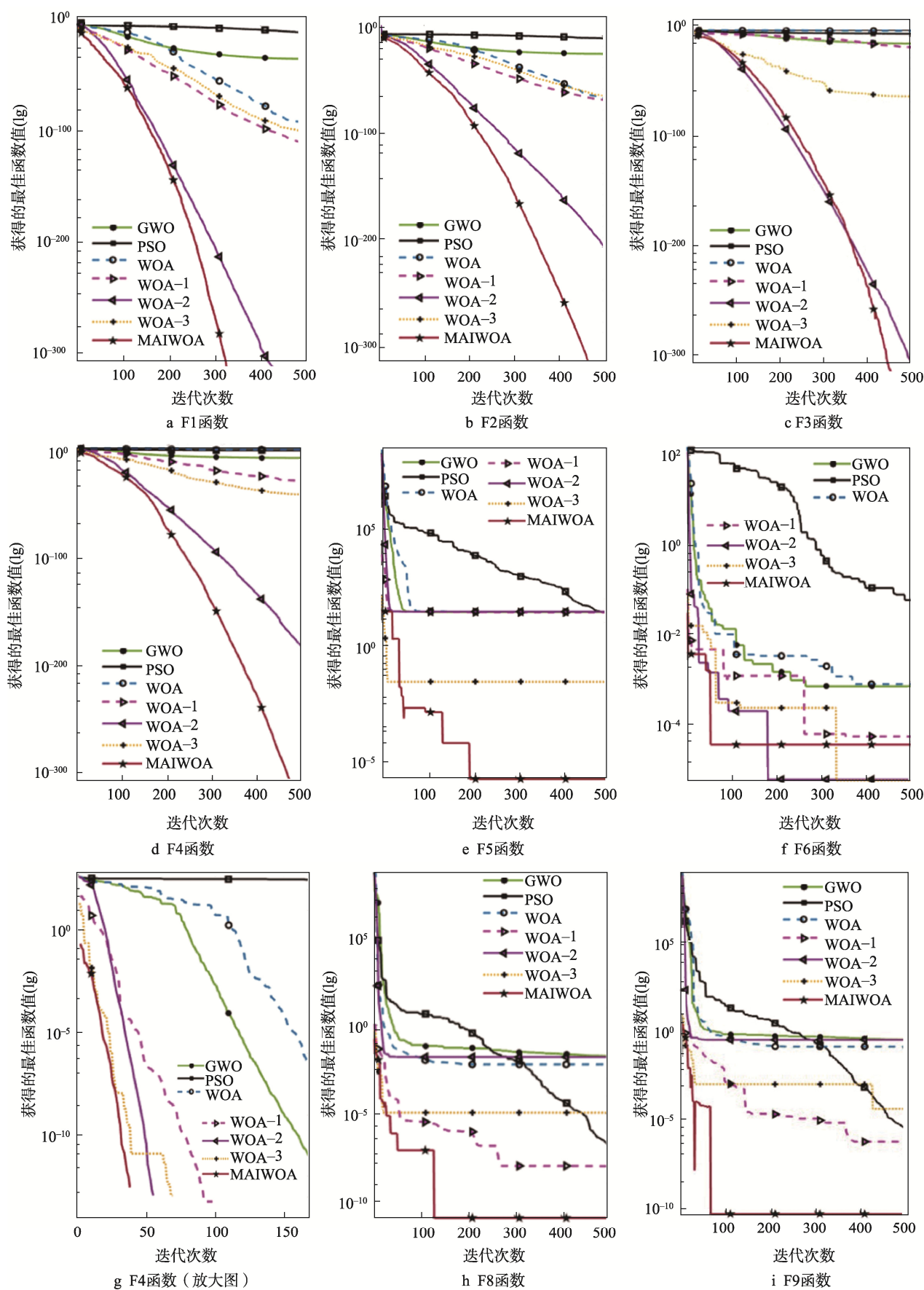


图2 9个基准函数7种算法收敛对比结果

Fig.2 Convergence comparison results of 7 algorithms for 9 benchmark functions

表 2 基准测试函数性能对比
Tab.2 Comparison of benchmark function performance

Function	Algorithm	Best	Worst	Mean	Std
F1	GWO	1.31×10^{-28}	1.71×10^{-27}	5.63×10^{-28}	5.42×10^{-28}
	PSO	3.20×10^{-5}	0.001 979	3.08×10^{-4}	5.96×10^{-4}
	WOA	2.84×10^{-86}	9.48×10^{-74}	1.01×10^{-74}	2.98×10^{-74}
	WOA-1	4.99×10^{-104}	2.92×10^{-96}	3.17×10^{-97}	9.17×10^{-97}
	WOA-2	0.00	0.00	0.00	0.00
	WOA-3	1.04×10^{-94}	5.59×10^{-84}	5.60×10^{-85}	1.77×10^{-84}
	MAIWOA	0.00	0.00	0.00	0.00
F2	GWO	1.62×10^{-17}	1.40×10^{-16}	7.59×10^{-17}	3.79×10^{-17}
	PSO	0.006 889	30.143 90	8.040 700	10.358 50
	WOA	2.91×10^{-59}	6.60×10^{-52}	1.15×10^{-52}	2.09×10^{-52}
	WOA-1	6.98×10^{-61}	2.00×10^{-55}	3.18×10^{-56}	6.66×10^{-56}
	WOA-2	4.27×10^{-200}	5.36×10^{-189}	5.75×10^{-190}	0.00
	WOA-3	4.24×10^{-57}	3.74×10^{-49}	3.78×10^{-50}	1.18×10^{-49}
	MAIWOA	0.00	0.00	0.00	0.00
F3	GWO	6.14×10^{-08}	0.000 594	7.94×10^{-5}	0.000 183
	PSO	51.633 80	141.224 7	85.830 80	27.537 40
	WOA	26 437.32	59 432.51	41 427.38	10 684.61
	WOA-1	2.81×10^{-11}	0.666 300	0.108 100	0.221 170
	WOA-2	3.39×10^{-302}	3.23×10^{-290}	4.27×10^{-291}	0.00
	WOA-3	4.86×10^{-56}	7.21×10^{-29}	7.21×10^{-30}	2.28×10^{29}
	MAIWOA	0.00	0.00	0.00	0.00
F4	GWO	8.91×10^{-08}	1.55×10^{-6}	4.87×10^{-7}	4.24×10^{-7}
	PSO	0.873 120	1.462 575	1.256 251	0.200 089
	WOA	6.071 600	83.644 32	53.622 60	27.729 46
	WOA-1	1.26×10^{-28}	2.68×10^{-15}	2.68×10^{-16}	8.46×10^{-16}
	WOA-2	1.40×10^{-182}	2.81×10^{-172}	2.82×10^{-173}	0.00
	WOA-3	1.50×10^{-41}	2.79×10^{-33}	2.79×10^{-34}	8.82×10^{-34}
	MAIWOA	0.00	0.00	0.00	0.00
F5	GWO	27.084 1	27.978 20	27.376 50	0.405 700
	PSO	26.076 5	99.153 60	69.980 00	22.620 30
	WOA	27.495 4	28.769 70	28.009 40	0.494 590
	WOA-1	26.328 0	27.164 20	26.737 00	0.262 540
	WOA-2	28.400 0	28.787 10	28.692 60	0.139 280
	WOA-3	0.0273 1	28.854 00	20.720 70	12.958 10
	MAIWOA	1.84×10^{-6}	0.003 300	0.000 450	0.001 020
F6	GWO	0.000 687	0.004 340	0.001 870	0.001 330
	PSO	0.057 843	24.242 60	4.418 000	7.391 300
	WOA	0.000 773	0.009 620	0.004 250	0.002 730
	WOA-1	5.25×10^{-5}	0.003 300	0.001 040	0.001 220
	WOA-2	5.75×10^{-6}	0.000 200	0.000 110	6.56×10^{-5}
	WOA-3	5.46×10^{-6}	0.000 200	7.96×10^{-5}	7.94×10^{-5}
	MAIWOA	3.45×10^{-5}	0.000 340	0.000 130	9.89×10^{-5}
F7	GWO	5.68×10^{-14}	7.056 300	1.501 000	2.698 500
	PSO	40.049 90	142.392 0	95.323 70	27.775 80
	WOA	0.00	134.683 0	13.468 30	42.590 50
	WOA-1	0.00	5.68×10^{-14}	5.68×10^{-15}	1.80×10^{-14}
	WOA-2	0.00	0.00	0.00	0.00
	WOA-3	0.00	0.00	0.00	0.00
	MAIWOA	0.00	0.00	0.00	0.00
F8	GWO	0.018 304	0.089 208	0.048 616	0.023 209
	PSO	1.36×10^{-7}	7.56×10^{-6}	1.42×10^{-6}	2.21×10^{-6}
	WOA	0.006 091	0.060 640	0.021 853	0.017 823
	WOA-1	1.01×10^{-8}	4.83×10^{-7}	1.82×10^{-7}	1.35×10^{-7}
	WOA-2	0.015 996	0.085 541	0.041 434	0.022 210
	WOA-3	1.09×10^{-5}	0.007 197	0.001 049	0.002 188
	MAIWOA	1.13×10^{-11}	2.47×10^{-7}	3.63×10^{-8}	7.51×10^{-8}
F9	GWO	0.314 906	1.232 074	0.817 029	0.276 649
	PSO	3.06×10^{-6}	0.054 793	0.008 831	0.016 953
	WOA	0.125 104	0.973 138	0.616 296	0.270 638
	WOA-1	5.16×10^{-7}	0.011 780	0.002 349	0.004 938
	WOA-2	0.303 468	0.805 238	0.463 416	0.138 443
	WOA-3	3.73×10^{-5}	0.179 453	0.038 712	0.062 170
	MAIWOA	4.13×10^{-11}	2.02×10^{-7}	5.24×10^{-8}	7.49×10^{-8}

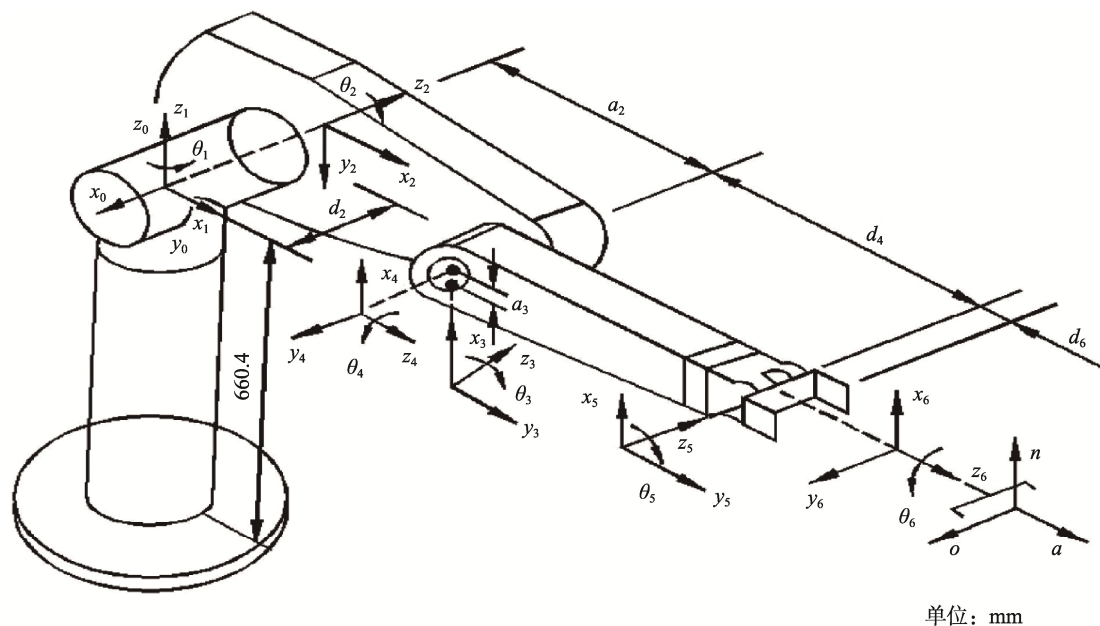


图 3 操作臂 PUMA560 模型结构
Fig.3 Structure diagram of PUMA560 model of manipulator

表 3 关节运动约束条件
Tab.3 Constraint conditions for joint motion

关节编号	角速度/ $[(^{\circ})\cdot s^{-1}]$	角加速度/ $[(^{\circ})\cdot s^{-2}]$
Joint 1	100	45
Joint 2	95	40
Joint 3	100	75
Joint 4	150	70
Joint 5	130	90
Joint 6	110	80

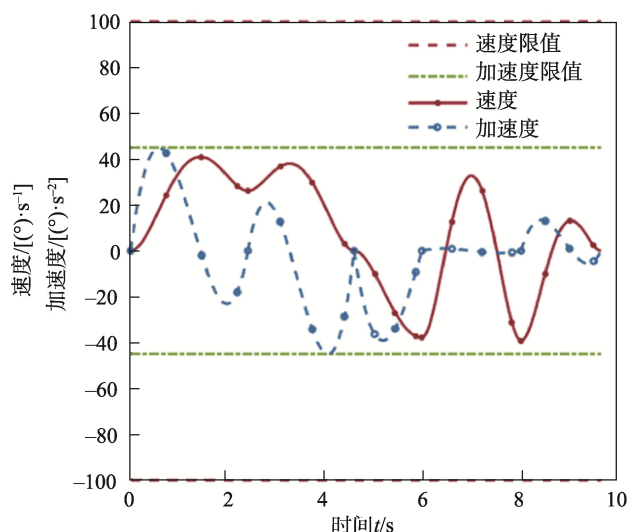
按照改进算法 MAIWOA, 对关节 1、2、3 在约束条件下求解最优时间, 优化后的关节运动轨迹及其

算法收敛曲线如图 4 所示。

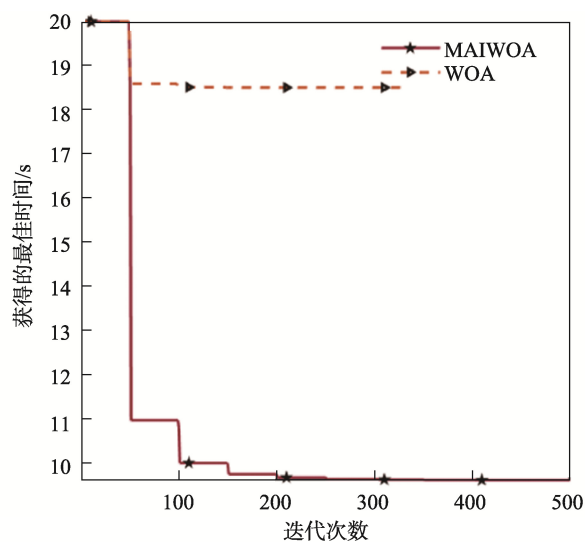
由图 4 可以看出, 在 MAIWOA 算法优化后, 关节 1 运动时间由 20 s 缩短到 9.642 5 s, 减少了 51.79%, 比原鲸鱼算法的收敛时间 (10.188 1 s) 减少约 0.55 s; 关节 2 运动时间由 20 s 缩短到 9.251 5 s, 减少了 53.74%, 比原算法的收敛时间 (9.576 8 s) 减少约 0.33 s; 关节 3 运动时间由 20 s 缩短到 10.787 s, 减少了 46.07%, 比原算法的收敛时间 (10.895 s) 减少近 0.11 s。并且在符合约束的条件下, 操作臂各关节得到的运动曲线光滑, 角速度、角加速度变化都较为平稳, 没有突变。证明改进后的算法可有效地对操作臂轨迹规划进行时间优化, 相较于 GWO、PSO、WOA 算法, MAIWOA 算法在求解目标函数最优问题有更优越的效果。

表 4 各关节中间点角度值
Tab.4 Angle value of middle point of each joint

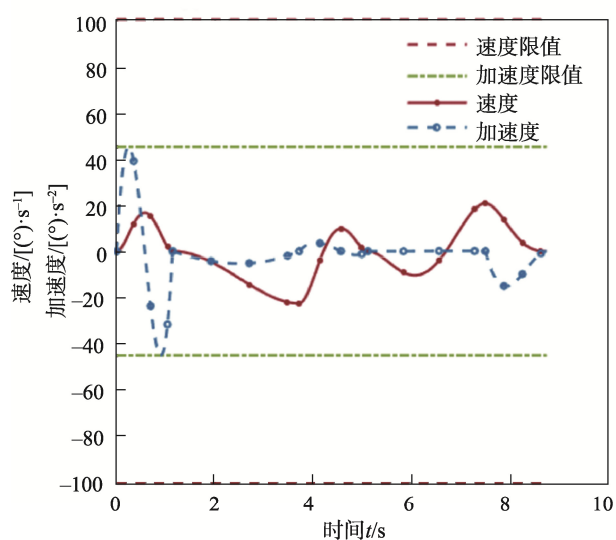
关节编号	Point 1/ $(^{\circ})$	Point 2/ $(^{\circ})$	Point 3/ $(^{\circ})$	Point 4/ $(^{\circ})$	Point 5/ $(^{\circ})$	Point 6/ $(^{\circ})$
Joint 1	10	75	130	100	-10	-50
Joint 2	15	25	-45	-70	-10	10
Joint 3	45	120	15	-10	100	50
Joint 4	5	60	110	20	60	-100
Joint 5	10	30	-60	10	50	-40
Joint 6	6	40	80	-10	30	20



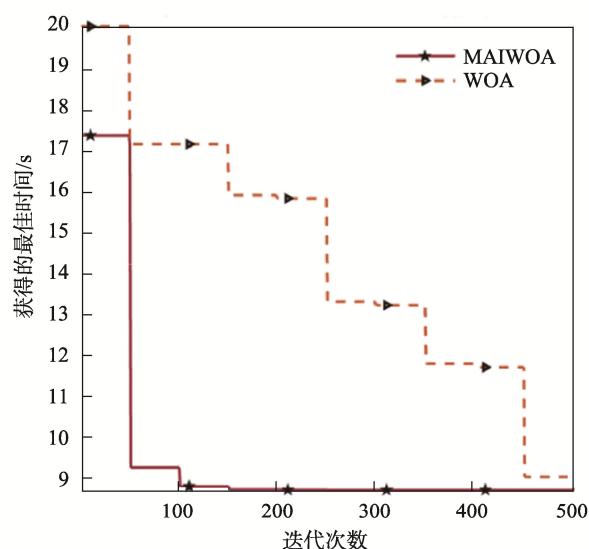
a 关节1的运动曲线



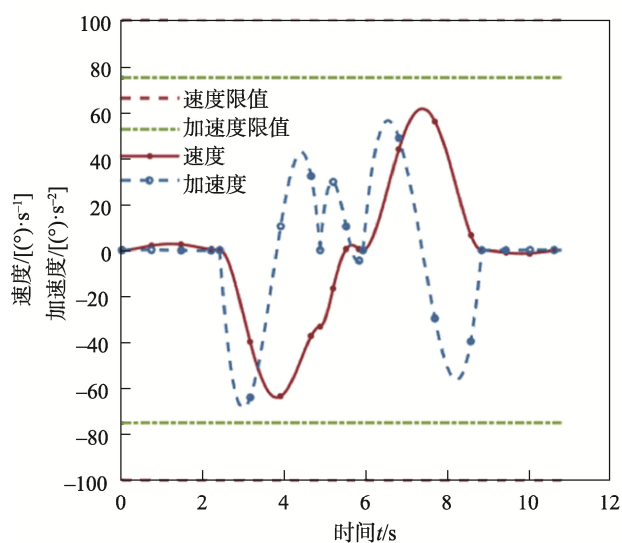
b 关节1不同算法的迭代收敛曲线



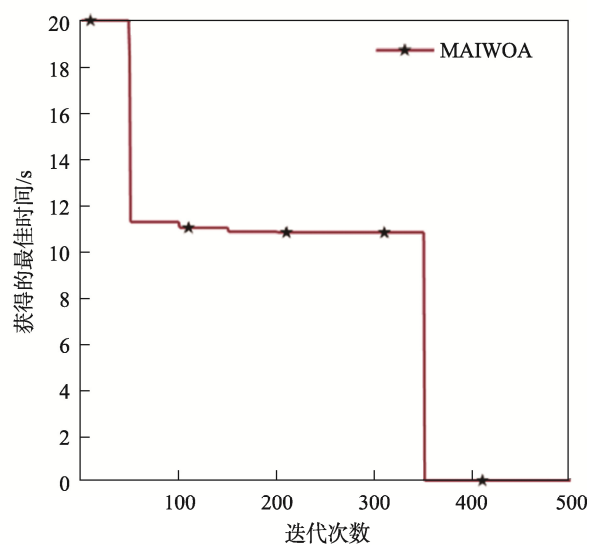
c 关节2的运动曲线



d 关节2不同算法的迭代收敛曲线



e 关节3的运动曲线



f 关节3采用MAIWOA算法的迭代收敛曲线

图4 关节运动速度、加速度曲线和算法收敛曲线
Fig.4 Joint velocity, acceleration curve and algorithm convergence curve

4 结语

文中提出了一种应用在仓库存储、包装生产流水线领域的操作臂末端执行器轨迹运动规划的一种方法。以实现操作臂始末两点间轨迹运动时间最短为目标,建立了目标优化函数。

为更好地解决 WOA 算法的不足,提出了优化的 MAIWOA 混合算法,从多方位角度对原算法进行优化。经实验仿真结果看出,改进算法不论全局收敛性、全局探索性和种群探索多样性都取得了很好的提升。在基准函数测试下,经多种算法的比较,MAIWOA 算法取得的结果都是最优的。将改进算法应用于求解轨迹运动时间最优上,能够快速地收敛到最佳时间,验证了算法在实际应用中的有效性。

参考文献:

- [1] 赵志强,朱志红,周星,等.全动力学约束的机器人高效时间最优轨迹规划[J].信息与控制,2021,50(6):701-708.
ZHAO Zhi-qiang, ZHU Zhi-hong, ZHOU Xing, et al. High-Efficiency Time-Optimal Trajectory Planning for Robots under Complete Dynamic Constraints[J]. Information and Control, 2021, 50(6): 701-708, 721.
- [2] YU Xiu-li, DONG Ming-shuai, YIN Wei-min. Time-Optimal Trajectory Planning of Manipulator with Simultaneously Searching the Optimal Path[J]. Computer Communications, 2021, 181: 446-453.
- [3] 杨星涛,库祥臣,赵欢乐,等.基于改进遗传算法的时间最优轨迹规划[J].制造技术与机床,2022(3):74-79.
YANG Xing-tao, KU Xiang-chen, ZHAO huan-le, et al. Time Optimal Trajectory Planning Based on Improved Genetic Algorithm[J]. Manufacturing Technology and Machine Tools, 2022(3): 74-79.
- [4] 李虹,刘松,孙志毅,等.基于非均匀 B 样条曲线的挖掘机最优时间轨迹规划[J].中国工程机械学报,2021,19(4):302-306.
LI Hong, LIU Song, SUN Zhi-yi, et al. Optimal Time Trajectory Planning of Excavator Based on Non-uniform B-spline Curve[J]. Chinese Journal of construction machinery, 2021, 19(4): 302-306.
- [5] 郭鑫鑫,薄瑞峰,贾竣臣,等.基于改进萤火虫算法的机械臂时间最优轨迹规划[J].机械设计与研究,2021,37(3):55-59.
GUO Xin-xin, BO Rui-feng, JIA Jun-chen, et al. Time Optimal Trajectory Planning of Manipulator Based on Improved Firefly Algorithm [J]. Mechanical design and research, 2021, 37(3): 55-59.
- [6] MIRJALILI S, LEWIS A. The Whale Optimization Algorithm[J]. Advances in Engineering Software, 2016, 95: 51-67.
- [7] 郝晓弘,宋吉祥,周强,等.混合策略改进的鲸鱼优化算法[J].计算机应用研究,2020,37(12):3622-3626.
HAO Xiao-hong, SONG Ji-xiang, ZHOU Qiang, et al. Improved Whale Optimization Algorithm Based on Hybrid Strategy[J]. Application Research of Computers, 2020, 37(12): 3622-3626.
- [8] 滕志军,吕金玲,郭力文,等.一种基于 Tent 映射的混合灰狼优化的改进算法[J].哈尔滨工业大学学报,2018,50(11):40-49.
TENG Zhi-jun, LYU Jin-ling, GUO Li-wen, et al. An Improved Hybrid Grey Wolf Optimization Algorithm Based on Tent Mapping[J]. Journal of Harbin Institute of Technology, 2018, 50(11): 40-49.
- [9] HUA Zhong-yun, ZHU Zhi-hua, YI Shuang, et al. Cross-plane Colour Image Encryption Using A Two-dimensional Logistic Tent Modular Map[J]. Information Sciences, 2021, 546: 1063-1083.
- [10] LI Mao-dong, XU Guang-hui, LAI Qiang, et al. A Chaotic Strategy-Based Quadratic Opposition-Based Learning Adaptive Variable-speed Whale Optimization Algorithm[J]. Mathematics and Computers in Simulation, 2022, 193: 71-99.
- [11] 徐航,张达敏,王依柔,等.混合策略改进鲸鱼优化算法[J].计算机工程与设计,2020,41(12):3397-3404.
XU Hang, ZHANG Da-min, WANG Yi-rou, et al. Hybrid Strategy to Improve Whale Optimization Algorithm[J]. Computer Engineering and Design, 2020, 41(12): 3397-3404.
- [12] 秋兴国,王瑞知,张卫国,等.基于混合策略改进的鲸鱼优化算法[J].计算机工程与应用,2022,58(1):70-78.
QIU Xing-guo, WANG Rui-zhi, ZHANG Wei-guo, et al. Improved Whale Optimizer Algorithm Based on Hybrid Strategy[J]. Computer Engineering and Applications, 2022, 58(1): 70-78.
- [13] 徐航,张达敏,王依柔,等.基于高斯映射和小孔成像学习策略的鲸鱼优化算法[J].计算机应用研究,2020,37(11):3271-3275.
XU Hang, ZHANG Da-min, WANG Yi-rou, et al. Whale

- Optimization Algorithm Based on Gauss Map and Small Hole Imaging Learning Strategy[J]. Application Research of Computers, 2020, 37(11): 3271-3275.
- [14] YANG Wen-biao, XIA Ke-wen, FAN Shu-rui, et al. A Multi-Strategy Whale Optimization Algorithm and Its Application[J]. Engineering Applications of Artificial Intelligence, 2022, 108: 1-28.
- [15] 褚鼎立, 陈红, 王旭光. 基于自适应权重和模拟退火的鲸鱼优化算法[J]. 电子学报, 2019, 47(5): 992-999.
- CHU Ding-li, CHEN Hong, WANG Xu-guang. Whale Optimization Algorithm Based on Adaptive Weight and Simulated Annealing[J]. Acta Electronica Sinica, 2019, 47(5): 992-999.
- [16] 尤中桐. 基于阿基米德螺线的小线段压缩处理及其插补算法研究[D]. 天津: 天津大学, 2018.
- YOU Zhong-tong. Research on Compression Processing and interpolation Algorithm of Small Segment Based on Archimedean Spiral[D]. Tianjin: Tianjin University, 2018.
- [17] 冯文涛, 邓兵. 鲸鱼优化算法的全局收敛性分析及参数选择研究[J]. 控制理论与应用, 2021, 38(5): 641-651.
- FENG Wen-tao, DENG Bing. Global Convergence Analysis and Research on Parameter Selection of Whale Optimization Algorithm[J]. Control Theory & Applications, 2021, 38(5): 641-651.

责任编辑: 曾钰婵