

基于改进 NURBS 和 Delta 并联机器人的食品分拣轨迹优化

陈广东¹ 邵 健² 王菲妮³ 赵学成⁴

(1. 江苏省涟水开放大学, 江苏 淮安 223401; 2. 杭州电子科技大学, 浙江 杭州 310018;
3. 江苏理工学院, 江苏 常州 213001; 4. 江苏科技大学, 江苏 镇江 212100)

摘要: [目的]降低 Delta 机器人在食品分拣轨迹中的运行时间和冲击,提高 Delta 机器人运行效率和稳定性。[方法]在分析食品分拣系统(应用 Delta 机器人)的基础上,提出了一种结合 NURBS 算法、麻雀搜索算法和蝴蝶优化算法的 Delta 机器人食品分拣轨迹优化方法。以运行时间和冲击最小化为目标对 NURBS 曲线规划的 Delta 机器人运行轨迹进行优化,结合麻雀搜索算法和蝴蝶优化算法求解模型,实现 Delta 机器人分拣轨迹优化。通过试验对 Delta 机器人分拣轨迹的运行时间和冲击进行分析。[结果]所提 Delta 机器人分拣轨迹优化方法可以兼顾运行时间和冲击,运行时间降低 6.00% 以上,运行冲击降低 80.00% 以上,系统的动态抓取成功率在 99.00% 以上,可满足食品分选要求。[结论]通过运行时间和冲击综合最优对 NURBS 曲线进行优化,可以有效提高 Delta 机器人食品分拣性能。

关键词: 食品分拣; 规划优化; Delta 机器人; 麻雀搜索算法; NURBS 算法; 蝴蝶优化算法

Food sorting trajectory optimization based on improved NURBS and Delta parallel robots

CHEN Guangdong¹ SHAO Jian² WANG Feini³ ZHAO Xuecheng⁴

(1. Jiangsu Lianshui Open University, Huai'an, Jiangsu 223401, China; 2. Hangzhou Dianzi University, Hangzhou, Zhejiang 310018, China; 3. Jiangsu University of Technology, Changzhou, Jiangsu 213001, China; 4. Jiangsu University of Science and Technology, Zhenjiang, Jiangsu 212100, China)

Abstract: [Objective] Reduce the running time and impact of Delta robot in food sorting trajectory, improve the efficiency and stability of Delta robot operation. [Methods] Based on the analysis of the food sorting system (using Delta robots), a Delta robot food sorting trajectory optimization method was proposed, which combined NURBS algorithm, sparrow search algorithm and butterfly optimization algorithm. Optimizing the running trajectory of Delta robot for NURBS curve planning with the goal of minimizing running time and impact, combining sparrow search algorithm and butterfly optimization algorithm to solve the model, which realized the optimization of Delta robot sorting trajectory. [Results] The proposed Delta robot sorting trajectory optimization method could balance running time and impact, reduce running time by more than 6.00% and running impact by more than 80.00%. The dynamic grasping success rate of the system was over 99.00%, which could meet the requirements of food sorting. [Conclusion] Optimizing NURBS curves through a combination of runtime and impact optimization can effectively improve the food sorting performance of Delta robots.

Keywords: food sorting; planning optimization; Delta robot; butterfly optimization algorithm; NURBS algorithm; sparrow search algorithm

在工业 5.0 的浪潮下,Delta 机器人作为一种自由度少、速度快、精度高的工业机器人,在食品、医药等轻工行业的应用越来越广泛^[1-2]。好的轨迹规划方法可以极大地提高 Delta 机器人在食品分选作业时的运行效率和稳定性,如何在快速运动的同时降低运动过程的冲击,成为目前研究重点^[3-4]。

基金项目:江苏省科技攻关项目(编号:23JS34981302);江苏省教育厅教学改革研究与实践项目(编号:zc76)

通信作者:陈广东(1975—),男,江苏省涟水开放大学高级讲师,学士。E-mail:walnmeind@sohu.com

收稿日期:2024-07-30 改回日期:2024-10-07

目前,有较多学者对 Delta 机器人轨迹规划方法进行研究,并取得了一些显著的成果。章鸿^[5]提出了一种轨迹规划方法,以时间和抖动最优为目标,结合三次样条函数和遗传算法进行优化。结果表明,该方法能有效提高 Delta 机器人的食品分拣性能,提高了末端执行器的稳定性。姚学峰等^[6]提出了一种轨迹规划方法,以时间最优为目标,结合 NURBS 算法和改进遗传算法进行优化。结果表明,该方法可以有效降低运行时间,规划曲线也更加平滑。封燕芳等^[7]提出了一种改进门字形轨迹与改进蚁群算法相结合的 Delta 机器人食品分拣轨迹规划方法,以时间最优为目标建立模型。结果表明,该方法具有较优的分拣精度和效率。郭俊等^[8]提出了结合改进门字型轨迹与改进蚁群算法的 Delta 机器人食品分拣轨迹规划方法,以时间最优为目标建立模型。结果表明,该方法有效提高了 Delta 机器人食品分拣的运行稳定性,降低了执行器的运行抖动。但上述方法的运行时间和冲击还有待进一步提高。

基于此,试验拟提出一种以运行时间和冲击最小化为目标对 NURBS 算法规划的 Delta 机器人运行轨迹进行优化,结合麻雀搜索算法和蝴蝶优化算法求解模型。通过试验对其性能进行分析,旨在为食品生产自动化提供助力。

1 食品分拣系统结构

食品分拣系统结构如图 1 所示,主要由视觉系统、上位机 PC 机、Delta 机器人 3 部分组成^[9-11]。视觉系统是 Delta 机器人食品分拣系统的核心部分,对带式输送机上的食品进行识别、定位,并将其信息传递给控制系统,使 Delta 机器人能够准确抓取和分拣。在上位机 PC 机中运行的控制系统软件,负责将视觉系统提供的目标位置和特征信息转换为 Delta 机器人运动控制命令。另外,PC 机也负责监视整个分拣过程,确保系统的稳定运行。Delta 机器人的主要功能是分拣食品。通过三自由度的运动,可以快速准确地到达指定位置,分拣目标食品,并将其放在正确的位置或容器中。

Delta 机器人结构如图 2 所示,Delta 机器人通过动平台在空间内部的平移运动,完成食品分拣。主要由动平台、静平台、主动杆、从动杆等组成^[12-13]。动平台直接连接 Delta 机器人的抓取机构或执行结构。静平台作为 Delta 机器人的固定基座,安装在外部支架上。主动杆是连接静平台和动平台的连杆,由电机驱动,负责将旋转运动转化为动平台的平移运动。从动杆采用平行四边形结构,用于保持动平台的姿态稳定。

2 Delta 机器人轨迹优化方法

对 Delta 机器人的轨迹规划在关节空间进行试验测试,利用 NURBS 曲线规划 Delta 机器人运行轨迹,以运行

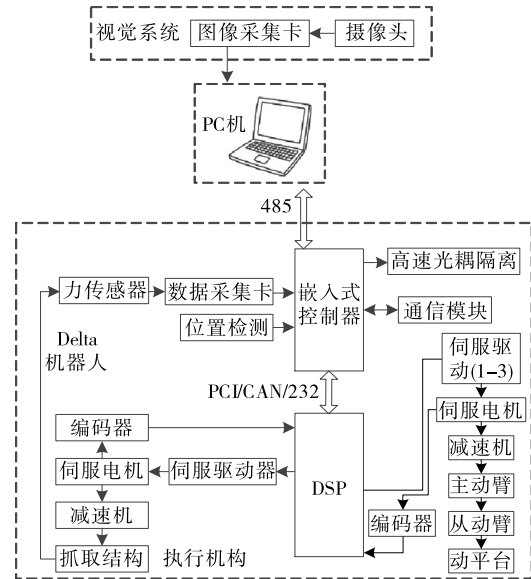


图 1 Delta 机器人食品分拣系统结构

Figure 1 Structure of Delta robot food sorting system

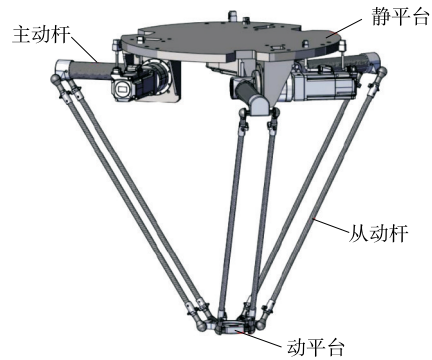


图 2 Delta 机器人结构

Figure 2 Delta robot structure

时间和冲击综合最优为目标建立轨迹优化模型,结合麻雀搜索算法和蝴蝶优化算法求解模型。

2.1 NURBS 轨迹规划

NURBS 曲线通过调整控制点的位置以及控制点对应的权值大小来改变曲线曲面的形状,适用于有复杂连续任务要求的运动^[14-15]。NURBS 曲线可以表示为参数 u 的分段有理多项式矢函数,一条 k 次 NURBS 曲线如式(1)所示。

$$c(u) = \frac{\sum_{i=0}^n \omega_i d_i N_{i,k}(u)}{\sum_{i=0}^n \omega_i N_{i,k}(u)}, 0 \leq u \leq 1, \quad (1)$$

式中:

- u ——曲线控制变量;
- k ——插值基函数次幂;
- $c(u)$ ——关键点;

- n ——插值点数;
- d_i ——曲线控制点;
- ω_i ——对应控制点权重因子;
- $N_{i,k}(u)$ ——一组B样条基函数。

为描述 k 次B样条曲线的多项式,首先设定其节点矢量 U ,如式(2)所示。

$$N_{i,0}(u) = \begin{cases} 1, u_i \leq u \leq u_{i+1} \\ 0, \text{others} \end{cases}$$

$$N_{i,k}(u) = \frac{u - u_i}{u_{i+k} - u_i} N_{i,k-1}(u) + \frac{u_{i+k-1} - u}{u_{i+k-1} - u_{i+1}} N_{i+1,k-1}(u)$$

插补的过程求取 Δu 的过程,通过式(1)和式(4)求出所有插补点 $C(u_i)$ 。

$$u_{i+1} = u_i + \Delta u_0 \quad (4)$$

通过5次NURBS曲线插值得到Delta机器人门型分拣轨迹如图3所示。

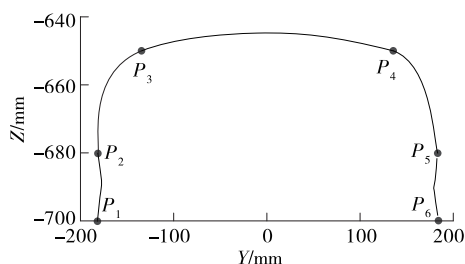


图3 Delta机器人门型分拣轨迹
Figure 3 Delta robot door type sorting trajectory

2.2 轨迹优化模型

试验综合考虑时间最优和冲击最小,时间最优:在满足任务要求和机器人动力学限制的基础上,寻找使总运行时间最小化的轨迹。冲击最小:在保证运动合理的同时,找到平稳的最佳轨迹。目标函数如式(5)所示。

$$\begin{cases} C_1 = \sum_{i=1}^{n-1} t_i \\ C_2 = \sum_{i=1}^n \sqrt{\frac{1}{T} \int_0^T J_i^2 dt} \end{cases} \quad (5)$$

式中:

- C_1 ——运行时间,s;
- C_2 ——冲击衡量指标;
- T ——机器人分拣一次时间,s;
- t_i ——第 i 个路径相邻节点时间间隔,s;
- J_i ——机器人在第 i 个路径点脉动冲击量, $(^\circ)/s^3$ 。

Delta机器人为确保运行中的性能和稳定性,必须考虑一系列的限制条件和约束^[16-17]。试验选取关节位移、速度、加速度、加加速度、力矩和载荷作为轨迹优化模型约束,如式(6)所示。

$$U = [u_0, u_1, \dots, u_k, u_{k+1}, \dots, u_n, u_{n+1}, \dots, u_{n+k+1}] \quad (2)$$

U 中各节点的分布不均匀,且非周期的,一般情况下设 $u_0 = u_1 = \dots = u_k = 0, u_n = u_{n+1} = \dots = u_{n+k+1} = 1$,中间 $n-k$ 个向量选择有多种方式,节点序列选择不同,NURBS曲线的形状不同。

k 次B样条曲线的基函数 $N_{i,k}(u)$ 如式(3)所示。

$$\quad \quad \quad (3)$$

$$\begin{cases} |h_j(t)| \leq h_{\max} \\ |h_j^*(t)| \leq v_{\max} \\ |h_j^{**}(t)| \leq a_{\max} \\ |h_j^{***}(t)| \leq J_{\max} \\ |M_j(t)| \leq M_{\max} \\ |F_j(t)| \leq F_{\max} \end{cases} \quad (6)$$

式中:

- h_{\max} ——关节角位移限值, $^\circ$;
- v_{\max} ——速度限值, $(^\circ)/s$;
- a_{\max} ——加速度限值, $(^\circ)/s^2$;
- J_{\max} ——加加速度限值, $(^\circ)/s^3$;
- M_{\max} ——力矩限值, $N \cdot mm$;
- F_{\max} ——载荷限值, N/mm^2 。

试验所提时间最优和冲击最小是一组相互矛盾的目标,为了简化问题,找到满足所有要求的解,试验采用加权目标函数将多目标问题转化为单目标优化问题,如式(7)所示。

$$\min f = a \frac{C_1}{C_{1\max}} + b \frac{C_2}{C_{2\max}} \quad (7)$$

式中:

- a, b ——最小运行时间和最小冲击的权重系数,
- $a + b = 1$;
- $C_{1\max}, C_{2\max}$ ——运行时间和冲击的最大值。

不同的权重分配会直接影响到机器人的运动轨迹规划和控制策略,较大的冲击权重会导致机器人在运动过程中更加平缓地加速或减速,从而减少对机械结构的应力和磨损,但可能会增加完成特定任务所需的时间。相反,较大的时间权重则会促使机器人尽快完成任务,但可能会牺牲一定的平稳性和精度。在实际应用中,会根据具体的应用场景和要求来分配权重系数。试验对平稳性和精确性要求较高,倾向于选择较大的冲击权重,将时间和冲击两个优化目标的权重因子分别设为0.1和0.9。

2.3 轨迹优化

蝴蝶优化算法作为一种新兴的群智能优化方法,在轨迹优化等领域展现出了良好的应用潜力,蝴蝶优化算法与现有的群智能算法相比,算法结构简单、参数少、搜索效率高,适用于模型求解。但蝴蝶优化算法存在局部最优和收敛速度等问题,蝴蝶优化算法后期容易陷入局部最优,影响求解质量。蝴蝶优化算法虽然搜索效率高,但在某些情况下可能会提前收敛,需要通过参数调整或策略改进来避免。文中通过麻雀搜索算法、自适应参数、柯西逐维变异对其进行优化^[18-19]。

2.3.1 麻雀搜索优化 在蝴蝶优化算法中,引入麻雀搜索算法的策略可以增强算法的探索和开发能力。将蝴蝶种群分为两个不同大小的子群体,发现者负责全局觅食,加入者负责局部搜索。蝴蝶种群中的发现者进行大范围的全局搜索,位置更新如式(8)所示。

$$X_i^{t+1} = \begin{cases} X_i^t + r^2 \cdot \left[\sum_{l=i,j,k} \left[g^* - Q \cdot \exp\left(\frac{X_w^t - X_i^t}{l^2}\right) \right] + \sum_{m \neq n} \left[\exp\left(\frac{X_w^t - X_n^t}{m^2}\right) - Q \cdot \exp\left(\frac{X_w^t - X_n^t}{n^2}\right) \right] \right] \cdot f_i, i > n/2 \\ g^* + |X_i^t - g^*| \cdot A \cdot L, \text{others} \end{cases} \quad (9)$$

式中:

X_w^t ——全局最差适应度个体;

A ——随机矩阵($1 \times d$)。

2.3.2 自适应参数 自适应权重系数 ω 可以根据迭代次数和当前的搜索情况进行调整,使得算法在初期可以广泛探索,在后期则更集中于精细开发。其计算如式(10)所示。

$$\omega^t = 1 - \cosh\left[e^{t/N_{gen}} / (e - 1)\right]^d, \quad (10)$$

式中:

d ——幂指数。

2.3.3 柯西逐维变异优化 在蝴蝶优化算法中,特别是在迭代的后期阶段,种群往往会面临多样性减少和陷入局部最优解的问题。为了解决这个问题,可以通过引入柯西逐维变异来增加种群的多样性并提高算法的全局搜索能力,如式(11)所示。

$$g^* = g^* + C_y \times g^*, \quad (11)$$

式中:

C_y ——与 g^* 同维的柯西随机数。

轨迹优化过程(如图4所示):

步骤1:初始化参数,如种群大小、迭代次数、幂指数、预警值和阈值等。

步骤2:求解目标函数。

步骤3:计算个体适应度,更新适应权重系数,根据发现人数、预警值、阈值的大小比较,更新发现者位置。根据加入者的数量,更新加入者位置。根据侦察者数量和边缘个体位置,对种群位置进行调整。

$$X_i^{t+1} = \begin{cases} X_i^t + [r^2 \cdot g^* - X_i^t \cdot \exp\left(\frac{-i}{\alpha \cdot N_{gen}}\right)] \cdot f_i, R_2 < S_T \\ X_i^t + Q \cdot L, R_2 \geq S_T \end{cases} \quad (8)$$

式中:

r, α ——[0,1]间随机数;

X_i^{t+1} ——迭代 $t+1$ 次后蝴蝶 i 的位置;

Q ——正态分布随机数;

L ——矩阵($1 \times d$);

g^* ——当前最优位置;

N_{gen} ——最大迭代次数;

f_i ——当前个体适应度值;

S_T, R_2 ——预警值和阈值。

对于加入者,在发现者带领的局部范围内进行搜索,位置更新如式(9)所示。

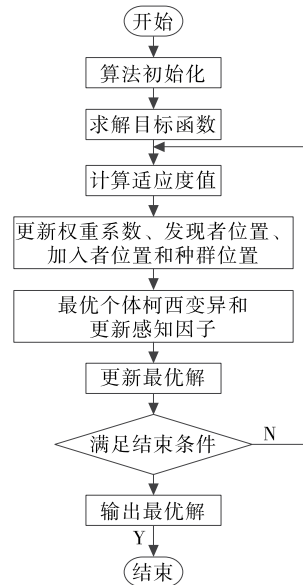


图4 轨迹优化

Figure 4 Trajectory optimization

步骤4:柯西变异和更新种群感知因子。

步骤5:将当前最优解与以前代理解进行比较,优于则替换,否则保持。

步骤6:满足结束条件输出最优解,不满足转到步骤3。

3 轨迹优化试验

3.1 参数设置

为验证试验所提 Delta 机器人食品分拣轨迹规划方

法的有效性,建立了Delta机器人食品分拣试验平台,试验平台组成如表1所示,机器人参数如表2所示。计算机为华为PC笔记本,操作系统为Windows 11 64位旗舰,英特尔酷睿 i7-3700, 5.2 GHz 主频, 32 GB 内存,通过MATLAB2020a环境进行数据分析,利用NURBS曲线在Delta机器人分拣空间规划运行轨迹,以运行时间和冲击综合最优为目标建立轨迹优化模型,结合麻雀搜索算法和蝴蝶优化算法求解模型。预设门型轨迹关键点如表3所示。试验参数如表4所示。

表1 试验平台组成

Table 1 Composition of the experimental platform

| 设备 | 具体参数 | 设备 | 具体参数 |
|------|--------------|----------|------------------------------|
| 试验目标 | 30 g 袋装食品 | 光源 | LED 光源 |
| 机器人 | 三轴 Delta 机器人 | 伺服电机 | MHMF082L1V2M MHMF012L1U2M |
| 相机 | acA2500-14gc | 减速机 | 42MM23A89P |
| 传送带 | 速度可调 | 振动加速度传感器 | ACC385H |

表2 机器人参数

Table 2 Robot parameters

| 动、静平台半径/mm | 主、从动臂长度/mm | 分拣速率/(次·min ⁻¹) | 角速度约束/(°·s ⁻¹) | 角加速度约束/(°·s ⁻²) | 角加加速度约束/(°·s ⁻³) | 角位移约束/(°) | 力矩约束/(N·mm) | 载荷约束/(N·mm ⁻²) |
|-------------|------------|-----------------------------|----------------------------|-----------------------------|------------------------------|-----------|-------------|----------------------------|
| 59.5, 135.0 | 250, 360 | 120 | 600 | 30 000 | 2 000 000 | 100 | 21.6 | 2 |

表3 预设轨迹点

Table 3 Preset trajectory points

| 关键点 | (X,Y,Z)数值/mm | 关键点 | (X,Y,Z)数值/mm |
|-----|------------------|-----|-----------------|
| 1 | -250, -180, -700 | 4 | -250, 138, -650 |
| 2 | -250, -180, -680 | 5 | -250, 180, -680 |
| 3 | -250, -138, -650 | 6 | -250, 180, -700 |

表4 试验参数

Table 4 Experimental parameters

| 参数 | 数值 | 参数 | 数值 |
|--------|------|----------|---------|
| 种群大小 | 50 | 预警值 | [0,1] |
| 最大迭代次数 | 100 | 阈值 | [0.5,1] |
| 转换概率 | 0.8 | <i>a</i> | 0.1 |
| 幂指数 | 0.1 | <i>b</i> | 0.9 |
| 感知因子 | 0.01 | | |

3.2 试验分析

为了验证试验所提方法的优越性,将试验方法与优化前的BOA算法、文献[6]的改进遗传算法、文献[7]的改进蚁群算法进行对比分析,通过上述方法对Delta机器人食品分拣的NURBS轨迹进行时间—冲击最优轨迹优化,不同方法目标函数收敛曲线如图5所示。

由图5可以看出,文献[6]和文献[7]方法收敛精度较高,但收敛速度慢,在约20和28步收敛。BOA算法约36步收敛,但易陷入局部最优,导致后期种群多样性降低。相比之下,试验方法在继承BOA算法优点的基础上,通过麻雀搜索算法、自适应参数、柯西逐维数变异对其进行优化,有效地跳出局部最优,继续寻找全局最优解。这种改进不仅提高了搜索最优解所在空间的概率,且提

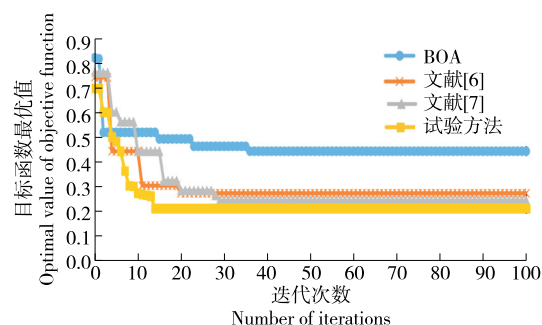


图5 不同方法目标函数收敛曲线

Figure 5 Convergence curves of objective functions for different methods

高了算法跳出局部最优的能力,使得试验方法在轨迹优化的应用中优于其他算法。

Delta机器人在实际作业中,食品分拣拾取频率为120次/min,每个周期为0.50 s,但考虑到每个周期的动作包括拾取和放置,拾取和放置单动作的实际时间应为0.25 s,不同方法对Delta机器人运行时间的优化结果如表5所示。

由表5可以看出,BOA优化Delta机器人的运行时间性能较差,优化程度最低。这可能是搜索最优解的过程

表5 不同方法Delta机器人运行时间优化结果

Table 5 Optimization results of Delta robot running time using different methods

| 方法 | 优化前时间/s | 优化后时间/s |
|-------|---------|---------|
| BOA | 0.250 0 | 0.241 5 |
| 文献[6] | 0.250 0 | 0.238 2 |
| 文献[7] | 0.250 0 | 0.236 7 |
| 试验方法 | 0.250 0 | 0.233 5 |

中易陷入局部最优,找不到更有效的轨迹。文献[6]、文献[7]方法具有一定程度的优化效果,可以在一定程度上缩短运动时间,但效果不如试验方法的显著。试验方法对 Delta 机器人食品分拣过程运行时间优化取得了显著效果,优化后运行时间为 0.233 5 s,比优化前降低 6.60%,提高了机器人的工作效率和整体性能。

不同方法对 Delta 机器人运行冲击的优化结果如表 6 所示。不同方法各关节运行冲击运动曲线如图 6 所示。

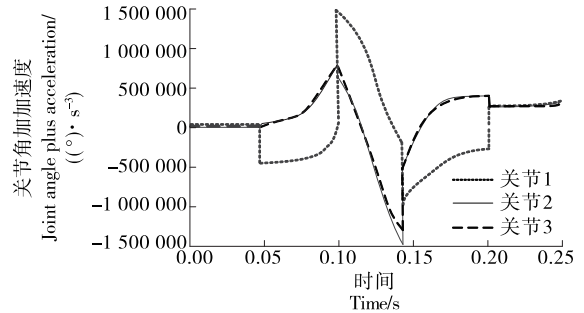
表 6 不同方法 Delta 机器人运行冲击优化结果

Table 6 Optimization results of Delta robot running impact using different methods

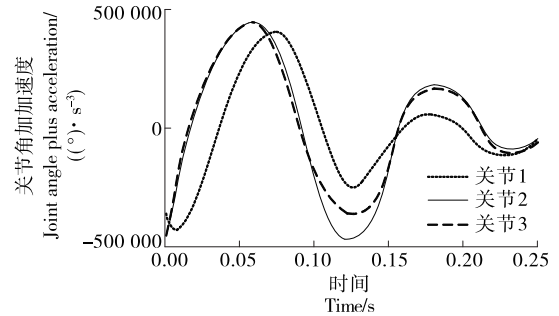
| 方法 | 优化前平均冲击/($^{\circ}$) \cdot s $^{-3}$ | 优化后平均冲击/($^{\circ}$) \cdot s $^{-3}$ |
|-------|--|--|
| BOA | 2.230×10^6 | 0.486×10^6 |
| 文献[6] | 2.230×10^6 | 0.471×10^6 |
| 文献[7] | 2.230×10^6 | 0.468×10^6 |
| 试验方法 | 2.230×10^6 | 0.445×10^6 |

由表 6 和图 6 可以看出,与 BOA 方法、文献[6]方法和文献[7]方法相比,试验方法显著降低了驱动臂各关节的平均冲击。优化前的平均冲击为 2.230×10^6 ($^{\circ}$)s $^{-3}$,优化后下降为 0.445×10^6 ($^{\circ}$)s $^{-3}$,冲击下降 80.00%。减少关节冲击有助于提高机器人的运行效率,同时延长机器人的寿命。

综上所述,试验方法在 Delta 机器人食品分拣过程轨迹优化应用中,不仅在运行时间上取得了显著的优化效果,且在关节冲击的降低上也表现突出,有效地提高了机



(a) 优化前运行曲线



(b) 优化后运行曲线

图 6 不同方法各关节运行冲击运动曲线

Figure 6 Different methods for running impact motion curves of each joint

器人的运行效率和稳定性。

选择某食品厂生产的袋装食品 30 g 进行分拣试验,将试验方法与优化前的 BOA 算法、文献[6]的改进遗传算法、文献[7]的改进蚁群算法进行对比分析。输送食品数为 500 个,随机放置,输送速度为 100,200 mm/s。不同方法分拣结果如表 7 所示。

表 7 不同方法分拣结果

Table 7 Sorting results from different methods

| 方法 | 输送速度 100 mm/s | | | 输送速度 200 mm/s | | |
|-------|---------------|---------|---------|---------------|---------|---------|
| | 输送食品数/个 | 分拣食品数/个 | 分拣成功率/% | 输送食品数/个 | 分拣食品数/个 | 分拣成功率/% |
| BOA | 500 | 466 | 93.20 | 500 | 450 | 90.00 |
| 文献[6] | 500 | 496 | 99.20 | 500 | 481 | 96.20 |
| 文献[7] | 500 | 500 | 100.00 | 500 | 490 | 98.00 |
| 试验方法 | 500 | 500 | 100.00 | 500 | 495 | 99.00 |

由表 7 可以看出,在输送速度为 100 mm/s 时,试验方法和文献[7]分拣成功率为 100.00%,BOA 方法和文献[6]方法的成功率为 93.20%,99.20%。这一差异表明,试验方法和文献[7]在相对较低的速度下具有较高的可靠性和准确性。随着输送速度增加到 200 mm/s,分拣成功率都有所下降,但试验方法由 100% 减小到 99.0%,文献[7]方法由 100.00% 减小到 98.00%。这表明试验方法在高速条件下仍能保持较高的稳定性和准确性,能满足严格的食品

生产需求。

4 结论

试验提出了一种以运行时间和冲击最小化为目标,对 NURBS 算法规划的 Delta 机器人运行轨迹进行优化,并通过改进的蝴蝶优化算法求解模型。结果表明,试验方法能有效降低运行时间和运行冲击,优化后运行时间为 0.233 5 s,与优化前相比下降 6.60%,冲击下降 80.00%,优化后平均冲击为 0.445×10^6 ($^{\circ}$)s $^{-3}$,与优化前相比下降

80.00%。但试验仅对轨迹规划方法进行了优化,对控制方法和Delta机器人软硬件等未进行研究。后续在完善现有方法的基础上,不断优化整个食品分拣系统。

参考文献

- [1] 傅明娣, 李忠, 王倩茹, 等. 基于多目标优化的油茶果分选机器人轨迹规划方法研究[J]. 食品与机械, 2023, 39(10): 105-111.
FU M D, LI Z, WANG Q R, et al. Research on trajectory planning method for *Camellia oleifera* fruit sorting robot based on multi-objective optimization[J]. Food & Machinery, 2023, 39(10): 105-111.
- [2] 刘现伟, 颀潭成, 徐彦伟, 等. 基于叠加摆线运动规律的Delta机器人轨迹规划[J]. 制造业自动化, 2021, 43(6): 88-94.
LIU X W, JIE T C, XU Y W, et al. Delta robot trajectory planning based on superimposed cycloidal motion laws[J]. Manufacturing Automation, 2021, 43(6): 88-94.
- [3] 徐岩. 基于改进引力搜索算法的高速并联机器人轨迹优化[J]. 食品与机械, 2022, 38(5): 82-86.
XU Y. Trajectory optimization of high-speed parallel robots based on improved gravity search algorithm[J]. Food & Machinery, 2022, 38(5): 82-86.
- [4] 梅江平, 孙玉德, 贺莹, 等. 基于能耗最优的4自由度并联机器人轨迹优化[J]. 机械设计, 2018, 35(7): 82-86.
MEI J P, SUN Y D, HE Y, et al. Trajectory optimization of a 4-degree of freedom parallel robot based on optimal energy consumption[J]. Mechanical Design, 2018, 35(7): 82-86.
- [5] 章鸿. Delta快速分拣机器人轨迹优化算法研究[J]. 机械设计与制造, 2021, 12(6): 288-290, 295.
ZHANG H. Research on trajectory optimization algorithm for delta rapid sorting robot[J]. Mechanical Design & Manufacturing, 2021, 12(6): 288-290, 295.
- [6] 姚学峰, 李超. 基于改进NURBS曲线插补算法的食品分拣机器人轨迹规划[J]. 食品与机械, 2022, 38(3): 80-85.
YAO X F, LI C. Trajectory planning of food sorting robot based on improved NURBS curve interpolation algorithm[J]. Food & Machinery, 2022, 38(3): 80-85.
- [7] 封燕芳, 吴必瑞, 赵倩. 基于改进门字形轨迹的番茄生产线Delta分选机器人轨迹规划[J]. 食品与机械, 2023, 39(8): 71-75, 154.
FENG Y F, WU B R, ZHAO Q. Trajectory planning of Delta sorting robot for tomato production line based on improved door shaped trajectory[J]. Food & Machinery, 2023, 39(8): 71-75, 154.
- [8] 郭俊, 王新. 食品拾放的3-PUU并联机器人轨迹规划[J]. 食品工业, 2021, 42(2): 223-226.
GUO J, WANG X. Trajectory planning of 3-PUU parallel robot for food picking and placing[J]. Food Industry, 2021, 42(2): 223-226.
- [9] 郝琳, 张坤平. 基于并联机器人的食品分拣控制系统设计[J]. 食品工业, 2020, 41(4): 209-212.
HAO L, ZHANG K P. Design of food sorting control system based on parallel robots[J]. Food Industry, 2020, 41(4): 209-212.
- [10] 郝大孝, 舒志兵, 孙学. 基于机器视觉的Delta机器人分拣与跟踪系统设计[J]. 机床与液压, 2019, 47(17): 36-42.
HAO D X, SHU Z B, SUN X. Design of delta robot sorting and tracking system based on machine vision[J]. Machine Tool & Hydraulic, 2019, 47(17): 36-42.
- [11] 刘芳, 刘玉坤, 林森, 等. 基于改进型YOLO的复杂环境下番茄果实快速识别方法[J]. 农业机械学报, 2020, 51(6): 229-237.
LIU F, LIU Y K, LIN S, et al. Rapid identification method of tomato fruit in complex environment based on improved Yolo [J]. Acta Agriculturalis Sinica, 2020, 51(6): 229-237.
- [12] 伍经纹, 徐世许, 王鹏, 等. 基于Adams的三自由度Delta机械手的运动学仿真分析[J]. 软件, 2017, 38(6): 108-112.
WU J W, XU S X, WANG P, et al. Kinematics simulation analysis of 3-DOF Delta manipulator based on ADAMS[J]. Software, 2017, 38(6): 108-112.
- [13] 张皓宇, 刘晓伟, 任川, 等. 并联机器人正运动学与NURBS轨迹规划[J]. 机械设计与制造, 2021, 12(4): 282-292.
ZHANG H Y, LIU X W, REN C, et al. Forward kinematics and NURBS trajectory planning of parallel robot[J]. Mechanical Design & Manufacturing, 2021, 12(4): 282-292.
- [14] 赵利平, 吴德刚. 基于小波与模糊相融合的苹果分级算法[J]. 食品与机械, 2020, 36(4): 142-145.
ZHAO L P, WU D G. Apple grading algorithm based on wavelet and fuzzy fusion[J]. Food & Machinery, 2020, 36(4): 142-145.
- [15] AZUMAYA C M, DAYS E L, VINSON P N, et al. Screening for AMPA receptor auxiliary subunit specific modulators[J]. PLoS One, 2017, 12(3): 1 523-1 538.
- [16] 马伟苹, 李文新, 孙晋川, 等. 基于粗一精立体匹配的双目视觉目标定位方法[J]. 计算机应用, 2020, 40(1): 227-232.
MA W P, LI W X, SUN J C, et al. Binocular vision target location method based on coarse fine stereo matching[J]. Computer Applications, 2020, 40(1): 227-232.
- [17] 罗久飞, 邱广, 张毅, 等. 基于自适应双阈值的SURF双目视觉匹配算法研究[J]. 仪器仪表学报, 2020, 41(3): 240-247.
LUO J F, QIU G, ZHANG Y, et al. Research on SURF binocular vision matching algorithm based on adaptive double threshold[J]. Journal of Instrumentation, 2020, 41(3): 240-247.
- [18] 程禹, 王晓华, 王文杰, 等. 基于改进AKAZE算法的图像特征匹配方法[J]. 西安工程大学学报, 2020, 34(8): 51-56.
CHENG Y, WANG X H, WANG W J, et al. Image feature matching method based on improved AKAZE algorithm[J]. Journal of Xi'an Engineering University, 2020, 34(8): 51-56.
- [19] 苏婷婷, 张好剑, 王云宽, 等. 基于费拉里法的Delta机器人动态目标抓取算法[J]. 华中科技大学学报(自然科学版), 2018, 46(6): 128-132.
SU T T, ZHANG H J, WANG Y K, et al. Delta robot dynamic target grasping algorithm based on Ferrari method[J]. Journal of Huazhong University of Science and Technology (Natural Science Edition), 2018, 46(6): 128-132.