

基于IALO和BP神经网络的支气管镜机器人系统

张茂杨¹, 王亚刚¹, 白冲², 胡珍丽²

¹上海理工大学, 光电信息与计算机工程学院, 上海

²上海长海医院, 呼吸内科, 上海

收稿日期: 2023年9月26日; 录用日期: 2023年11月23日; 发布日期: 2023年11月30日

摘要

支气管镜机器人在医生诊治传染病患者等需要远程治疗的手术中起到重要作用, 传统支气管镜机器人PID控制效果较差, 难以满足该手术主从同步控制的位置跟踪性能要求, 针对这个问题新设计了支气管镜机器人系统和配套的主从式IALO-BP-PID机器人控制算法。BP-PID控制器通过自适应学习能力在设备响应过程中动态改变PID控制器参数让系统拥有更好的控制性能。但传统BP神经网络性能受网络参数初值影响较大, 针对这个问题提出了基于改进蚁狮算法(IALO)优化权值的BP-PID控制方法, 在传统蚁狮算法(ALO)中引入自适应系数和反调节因子改善算法寻优性能, 在6个基准函数上的测试证明了改进算法的收敛与寻优能力优于原算法与其余对比优化算法。利用改进蚁狮算法(IALO)为BP神经网络选取参数初值配合神经网络的实时在线调整, 实现了对神经网络控制器离线粗调和在线细调, 提高了神经网络的性能。与传统PID控制算法的仿真相比, 该控制算法的系统无超调且调节时间缩短了八分之七, 使得支气管镜机器人的控制具有更佳稳定性和跟随性, 有一定的现实意义。

关键词

支气管镜机器人, 主从控制, 位置跟踪, BP神经网络, PID控制, 蚁狮算法

Bronchoscopy Robot System Based on IALO and BP Neural Network

Maoyang Zhang¹, Yagang Wang¹, Chong Bai², Zhenli Hu²

¹School of Optoelectronic Information and Computer Engineering, University of Shanghai for Science and Technology, Shanghai

²Department of Respiratory Medicine, Shanghai Changhai Hospital, Shanghai

Abstract

Bronchoscopic robots play an important role in doctors' diagnosis and treatment of infectious disease patients and other surgeries that require remote treatment. Traditional bronchoscopic robots have poor PID control performance, making it difficult to meet the position tracking performance requirements of master-slave synchronous control in this surgery. To address this issue, a bronchial robot system and a supporting master-slave IALO-BP-PID robot control algorithm have been designed. BP-PID controller dynamically changes PID controller parameters during equipment response through Adaptive learning ability, so that the system has better control performance. However, the performance of traditional BP neural networks is greatly affected by the initial values of network parameters. In response to this problem, a BP PID control method based on the improved ant lion algorithm (IALO) was proposed to optimize the weights. Adaptive coefficients and inverse adjustment factors were introduced into the traditional ant lion algorithm (ALO) to improve the optimization performance of the algorithm. Tests on six benchmark functions have shown that the improved algorithm has better convergence and optimization capabilities than the original algorithm compared to other optimization algorithms. By using the improved ant lion algorithm (IALO) to select initial parameter values for the BP neural network and real-time online adjustment of the neural network, offline coarse and online fine tuning of the neural network model was achieved, improving the performance of the neural network. Compared with the simulation of traditional PID control algorithm, the system of this control algorithm has no overshoot and the adjustment time has been shortened by seven eighths, making the control of the bronchial robot have better stability and follow-up, which has certain practical significance.

Keywords

Bronchial Robot, Master-Slave Control, Position Tracking, BP Neural Network, PID Control, ALO

Copyright © 2023 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

支气管镜属于内镜的一种，可以有效诊断肺部以及气管内其他的疾病[1]。传统支气管镜诊疗过程中，医生需要站在病患身旁手动将导管经其口或鼻送入人体并控制其运动[2]，而支气管镜机器人可以实现医患分离，满足了传染性疾病患者和异地患者的就医需求，对支气管镜机器人的应用研究有着现实意义。

科技不断发展的今天，机器人越来越广泛的应用于医疗领域，在内镜机器人领域诞生了众多成熟的技术。在国际上，以色列法海医院研发的 Remote navigation system (RNS) 内镜手术机器人系统[3]结构为主从控制式，医生操纵主端控制从端跟随，已在血管介入领域成熟应用。美国 Corindus 公司研发的 Corpath200 系统[4]第一次实现了主从端距离超过 20 公里的远程手术。日本香川大学研发的介入器械机器人[5]系统添加了力反馈环节，可以让医生实时感受操作导管在人体内的受阻情况。国内，北京理工大学研发的导管介入机器人辅助系统[6]通过磁场来实现医生的控制动作并通过磁流变液容器给医生提供力学反馈。不足的是，国内外的研究控制模式单一，多采用传统 PID 控制，其跟踪位置精度不足，而医生在

手术过程中需要频繁细调导管位置，再加上支气管较其他人体腔道更狭小敏感，这就导致了现今众多成熟的内镜机器人技术并不能直接应用到支气管镜机器人领域，该领域目前尚在起步阶段。沈阳理工大学的赵希梅[7]设计出变论域模糊 PID 手术机器人控制系统，将系统的误差和误差变化率作为输入，根据事先制定好的模糊规则表动态调整 k_p 、 k_i 和 k_D ，缺点是模糊规则表制定困难且需要一定的人工经验。Wei Li, Jian Fang 等[8]设计出 BP 神经网络(BPNN) PID 控制器，神经网络通过 Gradient descent 在动态过程中输出自适应控制系数取得了不错的调优结果，但随机初始化的网络初值限制了神经网络的性能。

合理的神经网络初值可以提高神经网络的性能，本文决定利用优化算法的全局寻优能力来为神经网络找寻合适的初值。蚁狮算法(ALO)是 2015 年提出的一种新型元启发式算法[9] [10]，文献[11] [12]中证明了该算法的在优化求解问题上的有效性，基于此设计了一种全新的 IALO (Improve Antlion Optimization Algorithm) BP-PID 控制算法，与传统 PID 控制、随机初始化的 BP-PID 控制以及使用其他优化算法的 BP-PID 控制在 matlab 上的仿真表明，该控制算法的调节时间最低，延迟时间最小，相比较其他控制算法使得系统有更强的位置跟踪性能。

2. 支气管机器人系统

2.1. 机器人系统框架

支气管镜机器人系统为主从式构型，分为医生控制的主端和接受控制信号并执行的从端，主端由显示器与手柄组成，从端由镜体输送部和内镜操作部组成，主从端通过 ADS 通讯实现远程控制。支气管镜机器人控制系统框如图 1 所示。

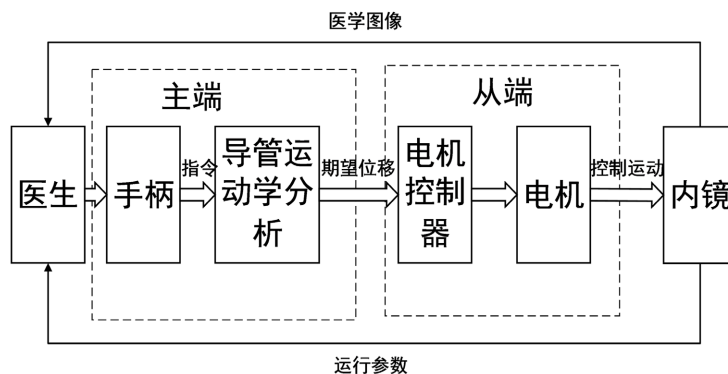


Figure 1. Block diagram of bronchoscopy robot control system

图 1. 支气管镜机器人控制系统框图

机器人系统主端构造如图 2 所示，医生通过手柄操控机器人完成手术诊断。内镜导管的轴向位移运动和径向旋转运动以及导管末端 360 度弯曲运动由从端电机驱动，电机控制按键集成于手柄。导管运动的速率由手柄控制杆倾斜的角度决定，手柄的顶端是控制电机运行与否的使能键，使能键作用是防误触控制杆。

主从端采用 ADS 通讯，选用的软件平台为 Beckhoff TwinCAT。从端部分构造如图 3 所示。镜体输送系统由五自由度机械臂和输送部组成，五自由度机械臂将导管顶部出口的位置与患者口部对齐并负责导管径向旋转运动，镜体输送部内部安有负责导管夹紧以及轴向位移运动的电机，其内置的扭矩传感器可实时向医生反馈诊疗过程中导管所受阻力，为医生提供参考。镜体控制系统由导轨、三自由度机械臂、控制部和支气管镜组成，导轨用于收放支气管镜多余的导管部分，三自由度机械臂用于调整支气管镜方向，控制部操作支气管镜旋钮控制导管末端自由弯曲。

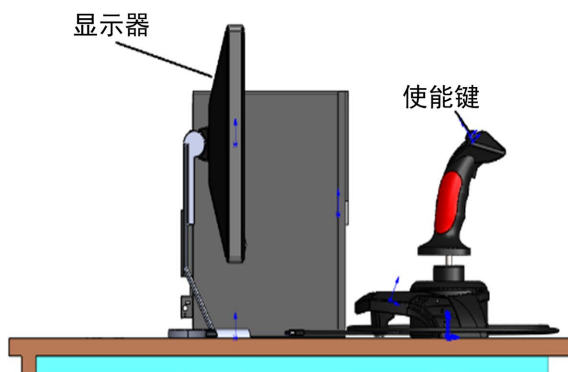


Figure 2. Construction of the main end operation section
图 2. 主端操作部构造

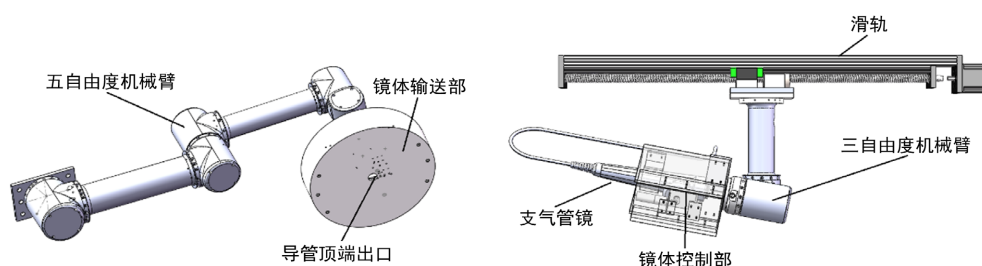


Figure 3. Construction of the slave end operation section
图 3. 从端操作部构造

机器人系统的整体构造为图 4:

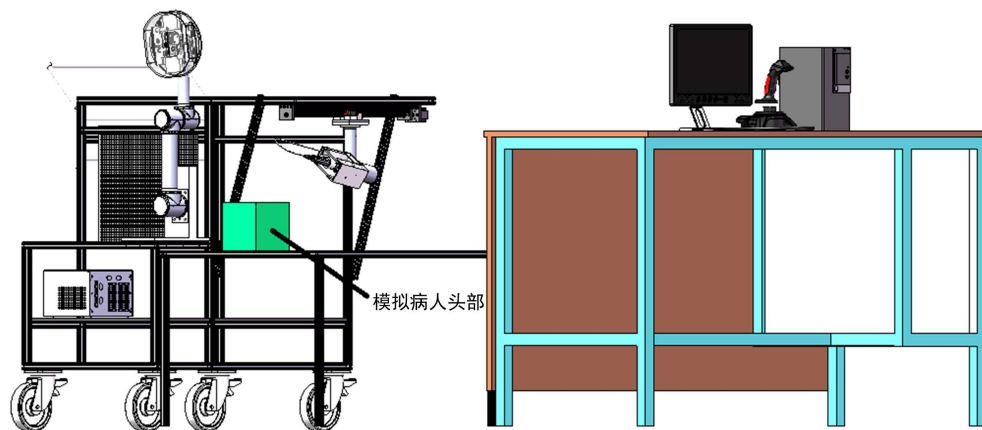


Figure 4. Overall structure
图 4. 整体构造

2.2. 运动学分析

由图 1 可知医生的控制指令转换为期望导管位移需要经过导管运动学分析, 通过安装在导管可控弯曲段的传感器可以得到导管弯曲角度等信息。镜体末端有三个自由度即轴向位移自由度、径向旋转自由度和前端弯曲自由度, 分别用 λ 、 θ 和 α 表示。设 $\{O_0\}$ 是基坐标系, $\{O_6\}$ 为末端坐标系, 如图 5。

根据 DenaviHartenberg (D-H)法建立 $\{O_0\} \sim \{O_6\}$ 两坐标系的齐次变化矩阵, 从基坐标系到末端坐标系的变换步骤如表 1。表 1 中 L1 表示弯曲段长度, L2 表示导管末端长度, 轴向位移为 λ 。

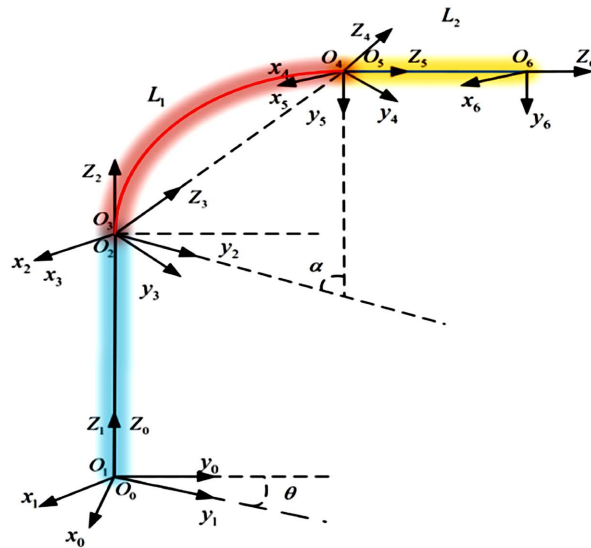


Figure 5. Geometric model of bronchoscopy tube
图 5. 支气管镜管几何模型

Table 1. Coordinate transformation process
表 1. 坐标变换过程

变换步骤	变换操作
1	{O ₀ }绕z ₀ 轴逆时针旋转θ
2	{O ₁ }由z ₁ 轴平移λ
3	{O ₂ }绕x ₂ 轴逆时针旋转α/2
4	{O ₃ }由z ₃ 轴平移L ₁
5	{O ₄ }绕x ₄ 轴逆时针旋转α/2
6	{O ₅ }由z ₅ 轴平移L ₂

s 表示 \sin , c 表示 \cos , 由表 1 变换过程可得基坐标系 $\{O_0\}$ 到末端坐标系 $\{O_6\}$ 的坐标表示为:

$$\begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = \begin{bmatrix} L_2 s \theta s \alpha + L_1 s \theta (1 - c \alpha) / \alpha \\ L_2 c \theta s \alpha + L_1 c \theta (1 - c \alpha) / \alpha \\ L_2 c \alpha + L_1 s \alpha / \alpha + \lambda \end{bmatrix} \quad (1)$$

$$\begin{bmatrix} dx_c \\ dy_c \\ dz_c \end{bmatrix} = \mathbf{J} \begin{bmatrix} d\lambda \\ d\theta \\ d\alpha \end{bmatrix} \quad (2)$$

由式 2 得到导管末端点位置与关节变量的雅可比变换矩阵:

$$\mathbf{J} = \begin{bmatrix} 0 & c\theta(L_2 s \alpha + L_1(1 - c \alpha) / \alpha) & \theta(L_2 c \alpha + L_1 s \alpha / \alpha - L_1(1 - c \alpha) / \alpha^2) \\ 0 & -s\theta(L_2 s \alpha + L_1(1 - c \alpha) / \alpha) & c\theta(L_2 c \alpha + L_1 s \alpha / \alpha - L_1(1 - c \alpha) / \alpha^2) \\ 1 & 0 & -L_2 s \alpha + L_1 c \alpha / \alpha - L_1 s \alpha / \alpha^2 \end{bmatrix} \quad (3)$$

2.3. 动力学模型

本文以设计的机器人轴向位移运动为例构建动力学模型，轴向位移以无刷直流电机驱动，根据牛顿第二定律建立轴向运动的动态模型[13]为：

$$f(t) = m\ddot{x}(t) + c\dot{x}(t) + kx(t) \quad (4)$$

式中 $f(t)$ 为电机输送的轴向位移动力， $x(t)$ 表示导管位移； $\dot{x}(t)$ 表示速度； $\ddot{x}(t)$ 表示加速度； m 为运动部件质量； c 为黏性阻尼系数； k 是弹簧系数。该方程说明了电机驱动力和输出位移的关系，令 $x_1(t) = x(t)$ ， $x_2(t) = \dot{x}(t)$ ，则轴向运动的状态方程为：

$$\begin{cases} \dot{x}(t) = AX(t) + Bf(t) \\ y(t) = CX(t) \end{cases} \quad (5)$$

$$\text{式中 } X(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}; \quad A = \begin{bmatrix} 0 & 1 \\ -k/m & -c/m \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 1/m \end{bmatrix}; \quad C = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}。$$

从式 5 推出轴向位移运动的传递函数为：

$$H(s) = m / (ms^2 + cs + k) \quad (6)$$

本文为研究方便三个参数取理想值：取 $m = 1$ ， $c = 0.04$ ， $k = 2$ ，调用 matlab 的 tf() 和 c2d() 函数将式 (6) 转换为差分方程形式方便仿真 m 文件的编写，差分方程形式如下：

$$y(k) = 1.999y(k-1) - 0.9996y(k-2) + 0.00005u(k-1) + 0.00005u(k-2) \quad (7)$$

式中 $y(k)$ 为被控系统当前时刻的输出； $y(k-1)$ 为被控系统上一时刻的输出； $y(k-2)$ 为被控系统上上时刻的输出； $u(k-1)$ 为控制器上一时刻的输出 $u(k-2)$ 为控制器上上时刻的输出。

3. IALO-BP-PID 控制器

3.1. 蚁狮算法(ALO)

蚁狮算法(Antlion Optimizer, ALO)是 2015 年 Mirjalili 提出的一种新型元启发式算法[14]，其算法内核是模拟自然界蚁狮捕食蚂蚁的过程。算法内数据分为蚂蚁和蚁狮两种，通过蚂蚁的随机游走实现全局数据搜索，通过轮盘赌和精英化策略提高种群的丰富度和收敛寻优性能，蚁狮捕食蚂蚁代表对适应度更佳的蚂蚁位置的更新和保存。蚂蚁的的随机游走由以下等式确定：

$$X(t) = [0, L, \text{cumsum}(2r(t_n) - 1)] \quad (8)$$

式中， $X(t)$ 为蚂蚁随机游走的步数集； cumsum 代表为计算累加和； t 为游走步长； n 为最大迭代次数； $r(t)$ 为随机函数，由式(9)确定：

$$r(t) = \begin{cases} 1, \text{rand}(\ast) > 0.5 \\ 0, \text{rand}(\ast) \leq 0.5 \end{cases} \quad (9)$$

式中 $\text{rand}(\ast) \in [0, 1]$ 。

由于可行域存在边界，不能直接用式(8)更新蚂蚁的位置，为确保蚂蚁在可行域内，需要用式(10)对式(8)进行归一化处理：

$$X_i^t = \frac{(X_i^t - a_i) * (d_i^t - c_i^t)}{b_i - a_i} + c_i^t \quad (10)$$

式中, b_i 和 a_i 是第 i 个变量游走上下限; d_i^t 和 c_i^t 是第 i 个变量在第 t 维的最大值和最小值。

蚁狮对蚂蚁影响:

$$\begin{cases} c_j^t = \text{Al}_j^t + c^t \\ d_j^t = \text{Al}_j^t + d^t \end{cases} \quad (11)$$

式中, d^t 和 c^t 表示所有变量在第 t 代的最大值和最小值; d_j^t 和 c_j^t 是第 j 只蚂蚁在第 t 代的最大值和最小值; Al_j^t 表示第 j 只蚂蚁通过轮盘赌选择的蚁狮在第 t 代的位置。

蚂蚁一旦落入蚁狮的陷阱, 蚁狮会向陷阱外抛沙防止蚂蚁逃脱, 此时蚂蚁随机游走范围将急剧缩小, 模拟为:

$$c^t = \frac{c^t}{I}, d^t = \frac{d^t}{I} \quad (12)$$

$$I = \begin{cases} 1, & t < 0.1T \\ 10^w * \frac{t}{T}, & t > 0.1T \end{cases} \quad (13)$$

式中, I 为比例系数, w 定义如下:

$$w = \begin{cases} 2, \text{when}(t > 0.1T) \\ 3, \text{when}(t > 0.5T) \\ 4, \text{when}(t > 0.75T) \\ 5, \text{when}(t > 0.9T) \\ 6, \text{when}(t > 0.95T) \end{cases} \quad (14)$$

式中, t 为当前迭代次数, T 为最大迭代次数

当蚁狮捕获蚂蚁时, 若蚂蚁位置优于蚁狮则蚁狮根据蚂蚁的位置进行更新, 如下:

$$\text{Al}_j^t = \text{Ant}_i^t, \text{if } f(\text{Al}_j^t) < f(\text{Ant}_i^t) \quad (15)$$

第 i 只蚂蚁在第 $t+1$ 代的位置由下式确定:

$$\text{Ant}_i^t = (R_A^t + R_E^t) / 2 \quad (16)$$

式中, R_A^t 表示在第 t 代迭代时蚂蚁围绕轮盘赌选择的蚁狮的随机游走; R_E^t 表示在第 t 代迭代时蚂蚁围绕精英蚁狮的随机游走, 每一代蚁狮中适应度最低的是精英蚁狮。

3.2. 改进蚁狮算法(IALO)

3.2.1. 自适应系数

由式(13)和式(14)可知传统蚁狮算法在迭代过程中描述陷阱坍塌的速度仅跟其迭代次数呈正相关, 该种 I 间断性改变的做法会导致搜索区域不均匀而易使算法遗漏包含最优解的区域。针对此问题引入自适应系数 f_{stage} 与 J_{trigger} , f_{stage} 由下式定义:

$$f_{\text{stage}} = \frac{\text{fitness}(t)}{\text{fitness}(t-1)} \quad (17)$$

式中, $\text{fitness}(t)$ 和 $\text{fitness}(t-1)$ 当前时刻和上一时刻精英蚁狮的适应度。 $f_{\text{stage}} \in (0, 1]$ 。 f_{stage} 趋向于 1 时

代表算法进化较慢陷入停滞，可能陷入局部最优解，此时适宜让其余蚁狮扩大搜索范围，协助精英蚁狮跳出停滞； f_{stage} 趋向 0 时代表精英蚁狮适应度下降较快，此时采用原算法收敛方式即可。 $J_{trigger}$ 由下式定义：

$$J_{trigger} = \begin{cases} 10^{-2}, & f_{stage} \in (0.9, 1] \\ 10^{-1.5}, & f_{stage} \in (0.7, 0.9] \\ 10^{-1}, & f_{stage} \in (0.5, 0.7] \\ 1, & \text{others} \end{cases} \quad (18)$$

修改式(13)为：

$$I = \begin{cases} 1, & t \leq 0.1T \\ 10^W * \frac{t}{T} * J_{trigger}, & t > 0.1T \end{cases} \quad (19)$$

f_{stage} 与 $J_{trigger}$ 的引入使得算法可以在精英蚁狮适应度下降缓慢时动态调整的值以让算法在一个动态的解空间里寻找最优值帮助精英蚁狮跳出可能的局部最优解，一定程度上缓和了 I 的间断性改变所导致的搜索区域不均匀变化。

3.2.2. 反调节因子

元启发式算法的性能体现于其全局探索和局部开发能力的均衡[15]，通过对轮盘赌算法的研究发现该算法选中精英蚁狮的概率最大，matlab 代码为：

```
accumulation = cumsum(weights);
p = rand() * accumulation(end)
```

其中，wights 为包含所有蚁狮适应度值倒数的矩阵。在前期过多的依赖精英蚁狮会造成算法过早收敛从而导致种群多样性降低[16]，如式(20)所示：

$$Ant_i^t = (R_A^t + R_E^t) / 2 = R_E^t \quad (20)$$

改进思路为设计加入反调节因子 δ ，其式如下：

$$\delta = \varepsilon - (t/T)^2 \quad (21)$$

修改上述代码中概率的部分为：

```
p = *rand() * accumulation(end)
```

经多次实验 ε 取 1.3，蚁狮作用于轮盘赌算法，随迭代次数自适应调节选中精英蚁狮的概率，让精英蚁狮在算法迭代过程中重要性动态变化。 p 在迭代前期较大，选中普通蚁狮概率较高，算法可以更全面的遍历解空间，增强全局探索能力；而在后期，选中精英蚁狮概率较高，着重发挥精英蚁狮的领导作用，引导算法着重于精英蚁狮邻域的开发，增强局部开发能力。

3.3. 改进算法的性能测试

为测试 IALO 算法的性能，选用表 2 所示的 2 个单峰基准函数和 4 个多峰基准函数进行仿真测试，全面的考察算法的局部开发能力、全局探索能力和处理复杂问题的能力。实验仿真软件为 matlab2018a，对比算法为原蚁群算法(ALO)、粒子群算法(PSO)和萤火虫算法(FA)。针对上述算法，种群规模统一设置为 30，最大迭代次数为 100，记录下每种算法单独运行 20 次结果的平均值，标准差，表 3 记录下仿真结果。

Table 2. Standard test function
表 2. 标准测试函数

函数	维度	搜索区间
$f_1 = \sum_{i=1}^n x_i^2$	30	[-100, 100]
$f_2 = \sum_{i=1}^n \left(\sum_{j=1}^i x_j \right)^2$	30	[-100, 100]
$f_3 = -20 \exp \left(-0.2 \sqrt{\frac{\sum_{i=1}^n x_i^2}{n}} \right) - \exp \left(\frac{\sum_{i=1}^n \cos(2\pi x_i)}{n} \right) + 20 + e$	30	[-32, 32]
$f_4 = \frac{\pi \left[10 \sin(\pi y_i) + \sum_{i=1}^n (y_i - 1)^2 [1 + 10 \sin^2(\pi y_i + 1)] + (y_n + 1)^2 \right]}{n} + \sum_{i=1}^n u(x_i, 10, 100, 4)$ $y_i = 1 + \frac{x_i + 1}{4}, u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m, & x_i > a \\ 0, & -a < x_i < a \\ k(-x_i - 1)^m, & x_i < -a \end{cases}$	30	[-100, 100]
$f_5 = \sum_{i=1}^n x_i^2 - \left(\prod_{i=1}^n \cos(x_i \sqrt{i}) + 1 \right)$	30	[-100, 100]
$f_6 = \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i) + 10]$	30	[-5.1, 5.1]

Table 3. Benchmark function simulation results
表 3. 基准函数仿真结果

测试函数	算法	平均值	标准差
f_1	IALO	2.29e-7	2.08e-6
	ALO	1.52e-5	4e-5
	PSO	2.41e+3	1.41e+3
	FA	5.54e+3	2.27e+3
f_2	IALO	1.64	25.61
	ALO	141.56	175.45
	PSO	2.13e+3	1.86e+3
	FA	7.97e+3	2.77e+3
f_3	IALO	7.22e-4	1.14
	ALO	0.65	0.78
	PSO	16.18	3.11
	FA	17.64	2.34

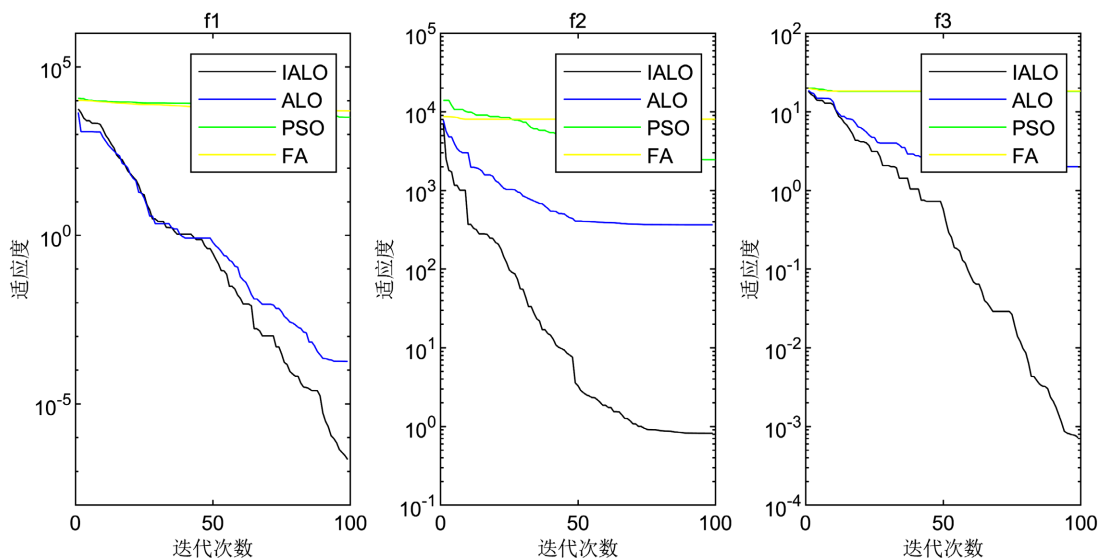
Continued

f_4	IALO	1.84	2.93
	ALO	4.52	2.70
	PSO	1.73e+6	1.93e+6
	FA	9.3e+6	1.36e+7
f_5	IALO	3.46e-2	0.09
	ALO	0.18	0.13
	PSO	2.66	0.66
	FA	2.73	0.75
f_6	IALO	2.98	9.01
	ALO	17.46	9.83
	PSO	75.76	20.29
	FA	63.04	13.82

表3记录了IALO算法、ALO算法、PSO算法和FA算法在相同测试条件下对选定的2个单峰基准函数和4个多峰基准函数的优化结果。从表3可以看出,IALO算法较其他算法取得了更佳的收敛结果和标准差,在单峰函数和中改进算法较原算法均提升了两个数量级的收敛精度,在多峰函数提升了一到四个数量级的收敛精度,表明改进策略优化了算法平衡全局探索和局部开发的能力,提高了算法性能。图6为六个基准函数收敛曲线对比图。

3.4. IALO-BP-PID 控制器

BP-PID控制器可以克服常规PID的缺陷,利用BP神经网络的自适应能力在动态过程中自动调节PID参数构成一个具有自适应能力的控制器,使用IALO算法获得BP神经网络参数初值以提高系统控制性能。其控制器结构图如图7所示。



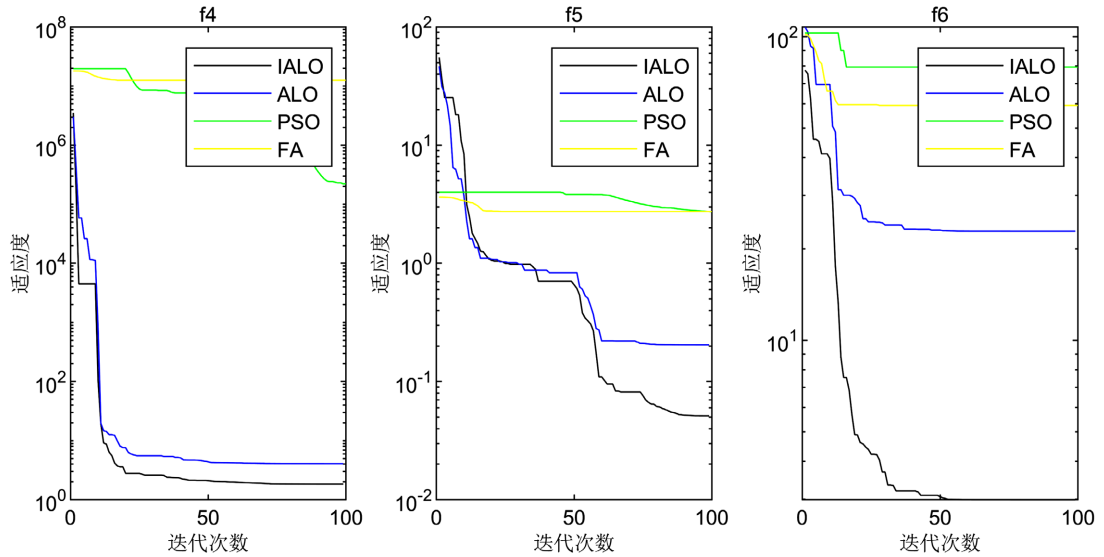


Figure 6. Convergence curve comparison chart
图 6. 收敛曲线对比图

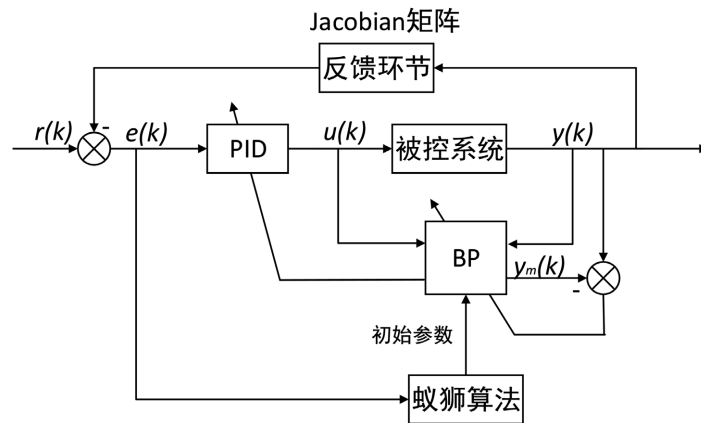


Figure 7. Structure of IALO-BP-PID controller
图 7. IALO-BP-PID 控制器的结构

增量式 PID 控制器根据给定值 $r(k)$ 与被控对象的输出值 $y(k)$ 的差值 $e(k)$ 进行控制[17]，其控制算法表达式为：

$$\Delta u(k) = k_p [e(k) - e(k-1)] + k_i e(k) + k_d [e(k) + e(k-2) - 2e(k-1)] \quad (22)$$

BP 神经网络误差反向计算的方式较好的解决了神经网络权重的调整问题，是当前应用最广泛的神经网络模型之一。其输入层 - 隐含层 - 输出层结构图如图 8 所示。

式中， $X = [x_1, x_2, \dots, x_n]^T$ 为神经网络输入向量，在本文中神经网络输入向量为 $X = [-1, e(k), e(k) - e(k-1), e(k) - 2e(k-1) + e(k-2)]^T$ ； $W = [w_{1j}, w_{2j}, \dots, w_{mj}]^T, j = 1, 2, \dots, m$ 和 $V = [w_{1k}, w_{2k}, \dots, w_{mk}]^T, k = 1, 2, 3$ 分别是输入层到隐含层第 j 个神经元和隐含层到输出层第 k 个神经元的网络权重。隐含层数据计算为：

$$\begin{aligned} net_j &= \sum_{i=1}^n w_{ij} x_i, j = 1, 2, \dots, m \\ y_j &= f(net_j), j = 1, 2, \dots, m \end{aligned} \quad (23)$$

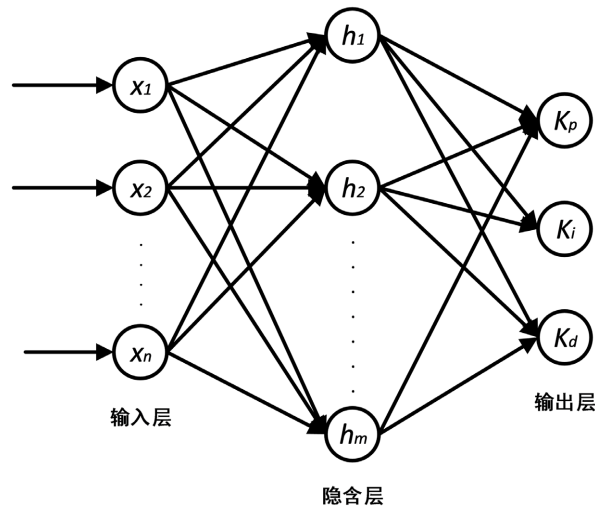


Figure 8. Neural network structure diagram
图 8. 神经网络结构图

输出层的计算和网络代价函数为:

$$net_k = \sum_{j=1}^m w_{jk} y_j, k=1,2,3 \quad (24)$$

$$t_k = g(net_k), k=1,2,3$$

$$E(k) = \frac{(y(k) - y_m(k))^2}{2} \quad (25)$$

式中, $y(k)$ 为 BP 神经网络期望输出值; $y_m(k)$ 为神经网络实际输出; 隐含层和输出层的激活函数分别为 Sigmoid 和非负 Sigmoid 函数:

$$f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (26)$$

$$g(x) = \frac{e^x}{e^x + e^{-x}} \quad (27)$$

BP 神经网络通过对式(25)进行梯度下降法调整网络权值, 输出层和隐含层的权重调整为:

$$\Delta w_{jk}(t+1) = \eta e(t+1) \operatorname{sgn}\left(\frac{\partial y(t+1)}{\partial u(k)}\right) \frac{\partial u(k)}{\partial t_k} g'(net_k) y_j \alpha \Delta w_{jk}(t), k=1,2,3 \quad (28)$$

$$\Delta w_{ij}(t+1) = \eta f'(net_j) \sum_{k=1}^n \left[e(t+1) \cdot \operatorname{sgn}\left(\frac{\partial y(t+1)}{\partial u(k)}\right) \frac{\partial u(k)}{\partial t_k} g'(net_k) w_{ij} \right] X(i) + \alpha \Delta w_{ij}(t), \quad (29)$$

$$i=1,2,\dots,n$$

式中, $\eta \in (0,1)$ 为学习速率, 速率过大系统难以收敛, 过小系统学习缓慢; $\alpha \in (0,1)$ 为动量因子。

根据上文对 BP 神经网络的介绍, 我们知道神经网络的性能与输入层到隐含层和隐含层到输出层网络权重的初值选取相关, 我们可以将网络权重映射为蚂蚁以及蚁狮个体的位置矢量, 设定蚁狮算法的 ITAE 适应度函数为:

$$f = \int_0^N t \cdot |y(t) - y_m(t)| dt \quad (30)$$

式中, N 为蚁狮算法迭代步数; t 为时间。

综上, 我们可以得到 IALO-BP-PID 控制器的控制算法流程图如图 9 所示。

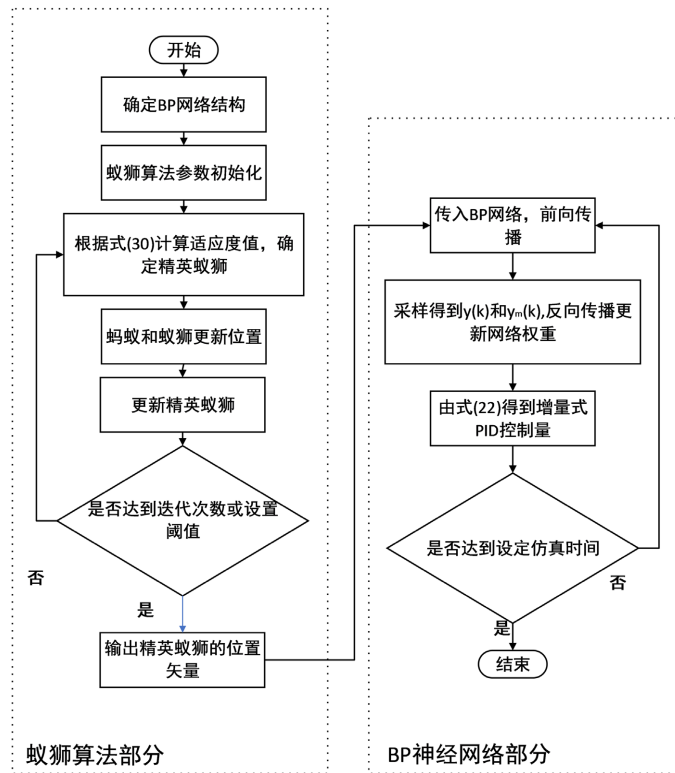


Figure 9. IALO-BP-PID control algorithm flowchart

图 9. IALO-BP-PID 控制算法流程图

4. 系统仿真实验

神经网络初值的选取对网络性能影响很大[18], 仅依靠梯度下降法使得神经网络每次训练结果不一, 图 10 展示了 5 次随机初始值的 BP 神经网络的训练结果, 黑线为期望位移线。

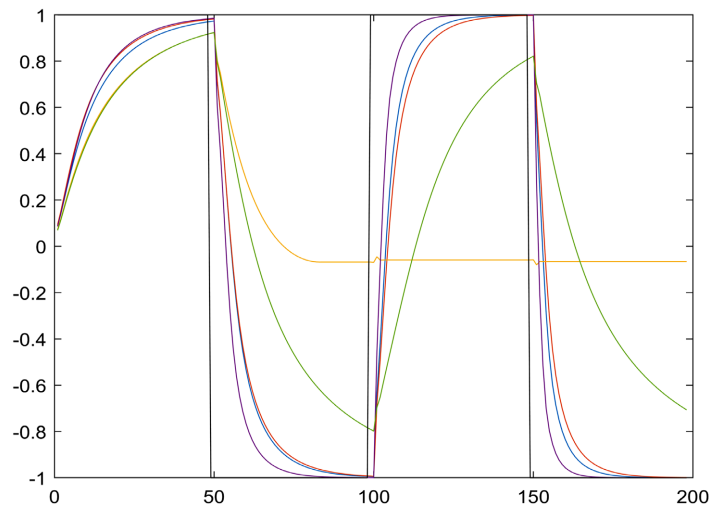


Figure 10. BP neural network PID5 random initialization training results

图 10. BP 神经网络 PID5 次随机初始化训练结果

由图 9 可知随机初始化的初值影响着神经网络性能，原因在于：由式(28)~(29)可知，权值向着使网络代价函数下降最快的方向变化，但受限于学习速率和动量因子，权值的迭代步伐大小收到限制，其调整能力有限只适合作为精调使用。某次初始化生成的权值可能会出现即使大幅度变化亦无法抵消不合理的 PID 参数而导致的积分部分影响，就会出现图 10 中鲁棒性差的情况出现。由此需要使用优化算法来为神经网络确定初值。神经网络和蚁狮算法仿真设置如下：

神经网络：运用 2.3 小节建立的差分方程为被控系统输出，系统参考输入为式(31)，采样时间 0.01 s，仿真时间 2 s；BP 神经网络结构为 4 * 5 * 3；学习速率为 0.25，动量因子为 0.05；PID 的初始参数采用 Ziegler-Nichols 整定[19]，整定的初始值为 $k_p = 0.03$ ， $k_i = 0.25$ ， $k_d = 0.3$ 。蚁狮算法：蚁狮种群数 100，蚁狮位置区间(0, 50)，蚁狮维度 35，迭代次数 100；适应度函数式(30)。在仿真阶段，本文为模拟实际手术过程中医生控制导管位移的操作采用下述信号作为期望路径，期望信号为：

$$r(t) = 1.0 * \text{sign}(\sin(2\pi * t * t_s)) \quad (31)$$

式中， t_s 为采样时间； t 为仿真时间。得到的实验结果如图 11 所示，分别展示了 PID 控制、BP-PID 控制[20]、PSO-BP-PID 控制[21]、FA-BP-PID 控制、GWO-BP-PID 控制、ALO-BP-PID 控制[22]和 IALO-BP-PID 控制的仿真结果图，在该期望信号上的性能测试同时展示了控制算法的鲁棒性能。

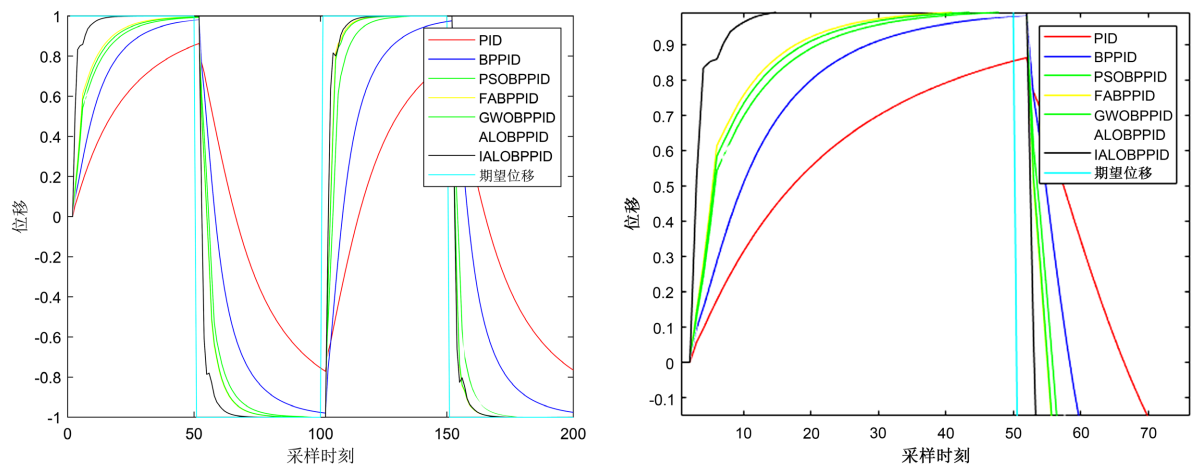


Figure 11. Simulation diagram and local enlarged diagram of each control algorithm

图 11. 各控制算法仿真图及局部放大图

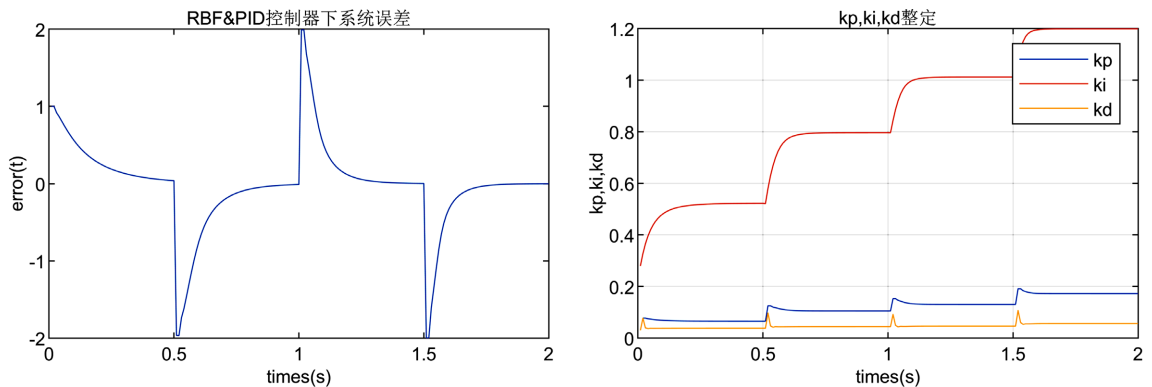


Figure 12. Error and PID parameter variation trend in the IALOBPPID process

图 12. IALOBPPID 过程中误差和 PID 参数变化趋势

图 11、图 12 中有期望位移线作为参考, 采用该信号作为期望路径的优点在于: 一、模拟现实中医生的实际操作; 二、同时也考察了各控制算法在遇到外来干扰下的鲁棒能力。由图 11 可以看出采用了神经网络的 PID 控制器要优于纯 PID 控制器, 从调节时间这个性能指标上看, 所需时间缩短了八分之七, 在该图上亦可以观察到结合了优化算法后神经网络性能有较大提升, 证明了本文的思路是可行的。在参数整定合理的情况下各控制算法皆可以无超调, 但对比之下 IALO-BP-PID 控制的调节时间要优于其他算法, 为 0.2 s。这意味着用此算法控制的从端可以更及时安全的跟随医生主端操作, 有助于提高医生对支气管机器人的掌控度和手术过程中的安全性。各控制算法的调节时间等参数见表 4。

Table 4. Adjusting time of each control algorithm

表 4. 各控制算法调节时间

控制算法	调节时间/s	延迟时间/s
PID	1.6	0.2
BPPID	0.6	0.1
GWO-BP-PID	0.4	0.06
PSO-BP-PID	0.4	0.06
FA-BP-PID	0.45	0.05
ALO-BP-PID	0.5	0.06
IALO-BP-PID	0.2	0.025

5. 结论

本文源于与某医院的校企合作, 通过对支气管机器人系统的分析建立起被控系统的模型, 设计出了一套适用于远程手术治疗的主从同步控制式机器人系统和与之配套的 IALO-BP-PID 控制算法, 新的结构设计较市面上其他的支气管机器人而言更经济实惠降低临床成本, 综合优化算法与 BP 神经网络的 PID 控制器有效的解决了传统 PID 控制器控制精度和响应速度问题, 为医生降低了由医疗设备所带来的意外风险, 提高了设备响应精度, 促进了支气管机器人高精度化和智能化发展。本文的主要贡献为:

1) 设计出了一套有一定现实意义的气管镜机器人系统。2) 改进了蚁狮算法, 引入自适应系数扩大蚁狮算法搜索有效范围帮助算法前中期快速跳出可能陷入的局部最优解, 提高其全局开发能力; 加入反调节因子动态调整精英蚁狮在优化过程的地位, 优化其对全局探索和局部开发能力的平衡, 加快了算法后期的收敛速度。仿真结果表明, 改进后的蚁狮算法具有更好寻优精度和收敛速度。3) 将蚁狮算法与 BP 神经网络相结合, 改善神经网络控制器的性能并将其应用于机器人运动控制。

从仿真结果可以看出 IALO-BP-PID 控制算法使得系统有较好的控制性能, 相较于传统 PID 控制提升了一定的稳定性和跟随性。但本文也存在不足, 如忽视了呼吸气流等非线性因素的影响, 对手术机器人的被控模型为方便仿真而取理想值, 实际情况中应考虑到公式(4)中 c 、 k 的动态变化。

基金项目

国家科技部重点研发计划课题(2020YFC2007502)。

参考文献

[1] 倪彭智, 俞豪杰, 汤杰, 梁瑞雪, 吕望, 胡坚. 机器人支气管镜系统应用的研究进展及其与人工智能结合的展望[J].

- 中国胸心血管外科临床杂志, 2021, 28(10): 1167-1171.
- [2] 赵鸣武. 纤维支气管镜(可弯曲支气管镜)临床应用指南(草案)[J]. 中华结核和呼吸杂志, 2000, 23(3): 5-6.
- [3] Beyar, R., Gruberg, L., Deleanu, D., Roguin, A., Almagor, Y., Cohen, S., Kumar, G. and Wenderow, T. (2006) Remote-Control Percutaneous Coronary Interventions: Concept, Validation, and First-In-Humans Pilot Clinical Trial. *Journal of the American College of Cardiology*, **47**, 296-300. <https://doi.org/10.1016/j.jacc.2005.09.024>
- [4] 奉振球, 边桂彬, 谢晓亮, 侯增广. PCI 手术机器人研究进展[J]. 机器人技术与应用, 2013(5): 26-32.
- [5] Guo, J., Guo, S.X., Xiao, N., Ma, X., Yoshida, S., Takashi, T. and Masahiko, K. (2012) A Novel Robotic Catheter System with Force and Visual Feedback for Vascular Interventional Surgery. *International Journal of Mechatronics and Automation*, **2**, 15-24. <https://doi.org/10.1504/IJMA.2012.046583>
- [6] Yang, X., Wang, H.B., Sun, L. and Yu, H.N. (2015) Operation and Force Analysis of the Guide Wire in a Minimally Invasive Vascular Interventional Surgery Robot System. *Chinese Journal of Mechanical Engineering*, **28**, 249-257. <https://doi.org/10.3901/CJME.2014.1229.181>
- [7] Zhao, X.M., Jiang, M.M., Li, H.Y. and Liu, H. (2013) Adaptive Fuzzy PID Controller of a Master-Slave Robotic Catheter System in Minimally Invasive Surgery. *Applied Mechanics and Materials*, **419**, 706-712. <https://doi.org/10.4028/www.scientific.net/AMM.419.706>
- [8] Wei, L. and Jian, F. (2014) BP Neural Network for Mobile Robot Self-Tuning PID Controller Design. *Advanced Materials Research*, **898**, 755-758. <https://doi.org/10.4028/www.scientific.net/AMR.898.755>
- [9] 朱娜娜. 蚁狮算法的改进及其应用研究[D]: [硕士学位论文]. 南宁: 广西民族大学, 2023.
- [10] Mirjalili, S., Jangir, P. and Saremi, S. (2016) Multi-Objective Ant Lion Optimizer: A Multi-Objective Optimization Algorithm for Solving Engineering Problems. *Applied Intelligence*, **46**, 79-95.
- [11] Petrovi, M., Miljkovi, Z. and Vukovi, N. (2016) Optimization of Flexible Process Planning by Using Ant Lion Optimization Algorithm.
- [12] Talatahari, S. (2016) Optimum Design of Skeletal Structures Using Ant Lion Optimizer. *Iran University of Science & Technology*, **6**, 13-25.
- [13] 赵希梅, 游健康, 刘浩, 李洪谊. 微创介入手术导管机器人系统自适应模糊 PID 控制[J]. 机床与液压, 2015, 43(9): 9-12.
- [14] Mirjalili, S. (2015) The Ant Lion Optimizer. *Advances in Engineering Software*, **83**, 80-98. <https://doi.org/10.1016/j.advengsoft.2015.01.010>
- [15] Yang, X. and Gong, W.Y. (2021) Opposition-Based JAYA with Population Reduction for Parameter Estimation of Photovoltaic Solar Cells and Modules. *Applied Soft Computing Journal*, **104**, Article ID: 107218. <https://doi.org/10.1016/j.asoc.2021.107218>
- [16] 徐钦帅, 何庆, 魏康园. 改进蚁狮算法的无线传感器网络覆盖优化[J]. 传感技术学报, 2019, 32(2): 266-275.
- [17] Wang, Z.H. and Zhang, J. (2022) Incremental PID Controller-Based Learning Rate Scheduler for Stochastic Gradient Descent. *IEEE Transactions on Neural Networks and Learning Systems*, 1-12. <https://doi.org/10.1109/TNNLS.2022.3213677>
- [18] Zhang, M.L., Zhang, Y.J., He, X.L. and Gao, Z.J. (2021) Adaptive PID Control and Its Application Based on a Double-Layer BP Neural Network. *Processes*, **9**, Article 1475. <https://doi.org/10.3390/pr9081475>
- [19] 陈梅, 王健. 基于 MATLAB GUI 的 Ziegler-Nichols PID 参数整定仿真系统[J]. 实验室研究与探索, 2020, 39(6): 98-101, 122.
- [20] Li, J.L., Peng, T., Zhang, S. and Liu, C. (2023) Improved PID Controller Based on BP Neural Network. *Journal of Physics: Conference Series*, **2479**, Article ID: 012062. <https://doi.org/10.1088/1742-6596/2479/1/012062>
- [21] Meng, Z.H., Zhang, L.X., Wang, H., Ma, X., Li, H. and Zhu, F.L. (2022) Research and Design of Precision Fertilizer Application Control System Based on PSO-BP-PID Algorithm. *Agriculture*, **12**, Article 1395. <https://doi.org/10.3390/agriculture12091395>
- [22] 李群, 贾家乐, 潘阳洋, 陈文笛. 基于 ALO-BP 神经网络和机械零件加工特征的工艺方案决策[J]. 成组技术与生产现代化, 2022, 39(3): 28-33, 62.