

基于两阶段多目标模糊决策的乒乓球机器人击球点选择

苏虎^{1†}, 徐德², 陈国栋¹, 方灶军¹

(1. 中国科学院自动化研究所复杂系统管理与控制国家重点实验室, 北京 100190;

2. 中国科学院自动化研究所精密感知与控制研究中心, 北京 100190)

摘要: 对乒乓球机器人而言, 回球时根据来球状态选择合适的击球点, 可以降低回球难度, 提高回球成功率. 本文提出了一种利用多目标模糊决策确定击球点的方法. 首先, 根据机器人最优运动区间和来球速度确定搜索区域. 然后, 定义搜索区域内的相关轨迹点为决策论域. 在此决策论域内, 根据回球目标参数分别计算期望出球速度, 并根据球拍参数定义了3个能量函数, 分别代表乒乓球被击打前后的能量和动量变化. 以此能量函数为指标函数决策得到优化的击球点. 最后, 在不同的实验情况下, 对比了本文算法与固定高度平面算法, 验证了本文算法的有效性.

关键词: 击球点选择; 模糊决策; 乒乓球机器人; 期望出球速度; 球拍参数

中图分类号: TP273 文献标识码: A

Striking position selection based on two-step multi-purpose fuzzy decision method for robotic table tennis

SU Hu^{1†}, XU De², CHEN Guo-dong¹, FANG Zao-jun¹

(1. State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China;

2. Research Center of Precision Sensing and Control, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China)

Abstract: For robotic table tennis, selecting the optimal striking position in the returning process according to the ball's trajectory, not only reduces the difficulties in returning the coming ball, but also increases the returnable success rate. A two-step multi-purpose fuzzy decision making method is proposed for striking-position selection. Firstly, according to robot's optimum range of motion and the coming ball's velocity, the searching region containing the striking position is determined. Then, the positions of the ball in the searching region are regarded as decision-making universe. In the universe, the desired outgoing velocity is respectively calculated with the target returning parameters. Three energy functions are defined based on the parameters of racket, representing energetic and momentum changes of the ping-pong ball in the striking process. The functions are used as index functions for striking point selection in the searching region. Finally, comparison experiments of the proposed method with conventional ones are well conducted under different experimental conditions, demonstrating effectiveness of the proposed method.

Key words: striking position selection; fuzzy decision; table tennis robot; desired outgoing velocity; racket parameter

1 引言(Introduction)

打乒乓球机器人是研究快速视觉感知和运动控制的良好平台, 自从20世纪80年代Billingsley教授提出举办乒乓球机器人比赛的倡议以来, 各国先后展开了乒乓球机器人方面的研究^[1]. Andersson等^[2-3], Acosta等^[4], Matsushima等^[5-7]等先后设计了不同的机器人系统, 成功实现了人机对打.

实现机器人与人对打乒乓球不仅需要机器视觉和运动控制的相互配合, 更为重要的是策略和智能在机器人中的应用. 对于人而言, 如何选择落点和击

球点以及如何判断旋转并据此调整球拍位姿等竞技性策略是在平时大量的训练中潜移默化得到的. 对于机器人而言, 则需要根据经验数据计算得到. 人打球时, 当观测到乒乓球来球轨迹时, 会预测其反弹后轨迹, 将球拍运动到相应位置, 选择回球落点, 并根据此落点选择一个合适的击球点, 当乒乓球经过此击球点时完成击球任务. 这个过程持续时间非常短, 甚至是人在无意识中完成的, 但却集中体现了乒乓球运动的智能性和策略性.

当选定了回球落点后, 如何选择击球点就显得格

外重要. 一个合适的击球点会降低回球难度, 提高回球成功率. 在现有的竞技性球类机器人中, 人们通常选定球类轨迹与设定平面的交点作为击球点. 如文献[8-9]给出的固定高度平面法. Matsushima等^[5-7]设计的4自由度乒乓球机器人本身具有固定高度, 也属于固定高度平面算法. Acosta等^[4]在设计乒乓球机器人时将来球轨迹与球拍平面的交点作为击球点. Mülling等^[10-11]根据机器人的击打区域设定了一个约 1 m^2 的虚拟平面, 乒乓球轨迹与此平面的交点即为击球点. 张培艳等^[12-13]根据来球速度和速度向量夹角设计模糊规则, 选择合适的击球区域. 这些算法均实现了人机对打, 且简单易实现, 具有一定的合理性. 然而, 这些算法没有充分考虑各个相关因素对回球难度的影响, 如落点、来球速度大小和机器人工作区间等. 由于乒乓球轨迹的多变性, 其选择的击球点无法有效降低回球难度, 且有可能超出机器人的运动范围, 从而造成回球失败.

根据打球经验, 在打球过程中, 影响运动员击球点选择的主要因素如图1所示. 观察对手的击球状态主要是为了判断是否是旋球和旋转类型, 预测乒乓球的飞行状态并得到初始的回球策略. 观测反弹前乒乓球的运动轨迹, 是为了估计乒乓球的来球落点和反弹后轨迹, 将球拍运动到合适的位置, 作好击球准备. 关于自身的击球状态, 对于人而言, 是在长期的击球训练过程中, 自己最熟悉最擅长的击打区域. 对机器人而言, 则必须首先考虑到机器人机械结构和各关节运动范围的限制. 图1中, 回球目标一般包括回球落点和回球后的落球速度. 对于给定的回球落点和落球速度, 在不同的击球点, 回球难度是不同的. 回球难度是一个广泛的概念, 其主要与击球点的位置, 回球时的球拍速度和位姿变化有关. 机器人打球和人打球一样, 都有初始位置和初始姿态. 在完成一个击球动作之后都有一个复位过程, 即球拍从击球位置和击球姿态回复到初始位置和初始姿态. 笔者认为, 与球拍的初始位置和姿态相比, 在某击球点回球时所需的球拍速度越大, 球拍姿态变化越大, 则在此点的回球难度也越大.

在本文中, 暂不考虑如何判断对手的击球状态, 球拍识别和模式分类等内容. 本文主要根据乒乓球运动轨迹和回球目标选择合适的击球点. 在击球过程中, 机器人和人打球时的不同之处主要体现在, 人回球时球拍的轨迹和击球点灵活多变, 受空间位置和运动方式的限制较少, 而机器人的运动区域和运动速度则相对有限. 在不同区域, 机器人回球效果相差很大. 人打球时可以根据相关因素综合决策, 选择合适的击球点, 而机器人则必须优先考虑到运动区域和运动速度的限制, 尽可能地选择机器人有效工作空间内的击球点. 如若不然, 即使选择的击球点符

合人为决策, 机器人也无法成功回球.

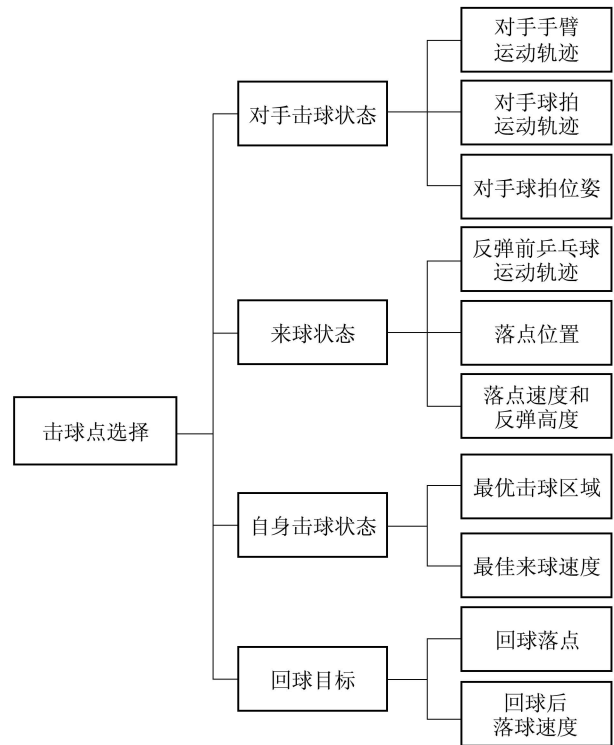


图1 影响击球点选择的因素

Fig. 1 The factors that affect striking point selection

击球点选择是一个多目标决策问题. 本文基于多目标模糊决策算法, 提出了决策击球点的两阶段搜索算法. 首先, 把乒乓球反弹后轨迹分为若干个离散区域, 以最优击球区域和来球速度为决策指标, 选择一个包含击球点的搜索区间, 以使选择的击球点尽可能地处于机器人的最优工作区间内, 保证机器人能够相对容易地到达击球点, 完成击球任务. 然后, 针对机器人的3个回球参数, 分别定义了能量函数, 衡量机器人在不同位置定点定速回球的难度. 最后, 以定义的能量函数为决策指标, 利用决策算法在搜索区间中决策最优击球点.

2 乒乓球机器人系统(Table tennis player robot system)

本文中的乒乓球机器人系统如图2所示. 智能相机A、B和视觉计算机组成了机器人的双目视觉系统^[8]. 视觉系统坐标系 $\{V\}$ 原点建立在人一侧的乒乓球台上, X_V 轴与乒乓球台的长轴平行且指向机器人方向, Z_V 轴与乒乓球桌面垂直, 竖直向上为其正方向. 视觉系统捕捉乒乓球的运动轨迹, 并将击球点信息发送至控制计算机, 由控制计算机规划机器人运动并控制机器人击球^[14-15]. 建立机器人控制坐标系 $\{M\}$. 选择乒乓球机器人所在球台侧, 球桌后边沿的中心作为坐标系 $\{M\}$ 的原点, 其 X_M 轴平行于机器人的 X 关节导轨, 以水平向右为正方向, 其 Y_M 轴平行于 Y 关节导轨, 以向前为正方向. Z_M 轴方向

与视觉坐标系 Z_V 轴相同. 乒乓球机器人具有5个自由度, 分别为沿 X_M, Y_M, Z_M 轴的平移和绕 X_M, Z_M 轴的旋转.

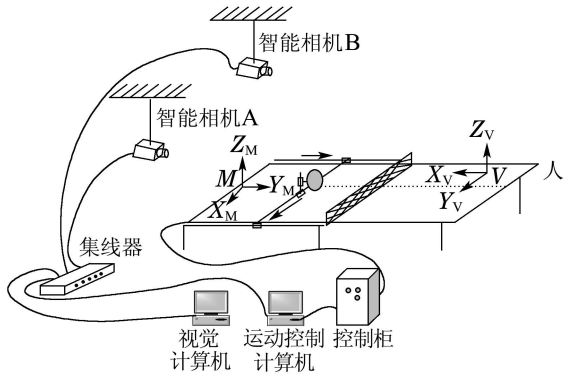


图2 乒乓球机器人系统

Fig. 2 Table tennis player robot system

3 多目标模糊决策(Multi-purpose fuzzy decision method)

3.1 多目标模糊决策(Multi-purpose fuzzy decision method)

假设 $U = \{u_1, u_2, \dots, u_m\}$ 为决策论域, $V = \{f_1, f_2, \dots, f_n\}$ 为决策指标, 则各方案的因素指标值向量为

$$u_j = (f_{1j}, f_{2j}, \dots, f_{nj})^T. \quad (1)$$

把第 j 个方案的第 i 个指标记为 f_{ij} , 得到 m 个方案的因素指标函数矩阵:

$$F = \begin{pmatrix} f_{11} & f_{12} & \dots & f_{1m} \\ f_{21} & f_{22} & \dots & f_{2m} \\ \vdots & \vdots & & \vdots \\ f_{n1} & f_{n2} & \dots & f_{nm} \end{pmatrix}. \quad (2)$$

令

$$\delta_{ij} = \frac{f_{ij} - f_{i \min}}{f_{i \max} - f_{i \min}}, \quad (3)$$

其中:

$$f_{i \max} = \max(f_{i1}, f_{i2}, \dots, f_{im}),$$

$$f_{i \min} = \min(f_{i1}, f_{i2}, \dots, f_{im}).$$

于是, $n \times m$ 个相对偏差值 δ_{ij} 构成了一个模糊矩阵

$$\Delta = \begin{pmatrix} \delta_{11} & \delta_{12} & \dots & \delta_{1m} \\ \delta_{21} & \delta_{22} & \dots & \delta_{2m} \\ \vdots & \vdots & & \vdots \\ \delta_{n1} & \delta_{n2} & \dots & \delta_{nm} \end{pmatrix}. \quad (4)$$

给出相关因素重要程度模糊子集

$$A = (a_1, a_2, \dots, a_n). \quad (5)$$

计算各方案因素指标向量 u_j 与 m 个方案的指标

标准值向量 $f^0 = (f_{1 \min}, f_{2 \min}, \dots, f_{n \min})$ 之间的加权相对偏差距离:

$$d_j = d_j(u_j, f^0) = \frac{1}{a} \sqrt{\sum_{i=1}^n (a_i \delta_{ij})^2}, \quad (6)$$

其中 $a = (\sum_{i=1}^n a_i) / n$ 为 n 项指标权值的平均值.

本文把 m 个方案中的 n 个因素指标的标准值向量 $f^0 = (f_{1 \min}, f_{2 \min}, \dots, f_{n \min})$ 构成的方案设定为最理想的方案. 因此, m 个评价方案中与最理想方案之间加权相对偏差距离 d_j 最小者所对应的方案被选为最优方案, 即当

$$d_{j^*} = d_{j^*}(u_{j^*}, f^0) = \min_{1 \leq j \leq m} (d_j) \quad (7)$$

时, 方案 u_{j^*} 为最优方案.

3.2 击球点决策流程(Flow chart of the striking position selection)

本文的击球点决策分为两个阶段, 如图3所示. 第1阶段, 根据机器人的最优击球区域和来球速度确定搜索区域, 确保机器人能够较容易地实现决策结果; 第2阶段, 在此搜索区域中, 根据定义的能量函数, 确定最优击球点, 使乒乓球被击打前后能量变化最小, 以降低回球难度.

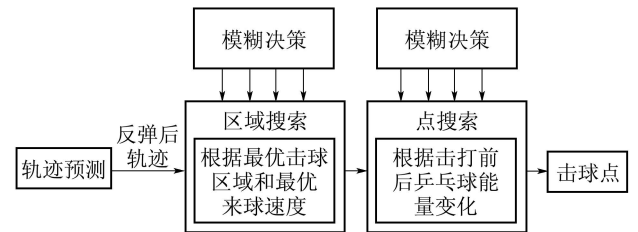


图3 击球点决策流程

Fig. 3 Flow chart of the striking position selection algorithm

4 决策搜索区间(Searching region determination)

4.1 机器人最优击球参数(Robot's optimum range of motion)

根据击球经验, 选定乒乓球机器人的最佳击球区域($[P_{x \min}, P_{x \max}], [P_{y \min}, P_{y \max}], [P_{z \min}, P_{z \max}]$), 其中心位置为 $(P_{x \text{avg}}, P_{y \text{avg}}, P_{z \text{avg}}) = [(P_{x \min}, P_{y \min}, P_{z \min}) + (P_{x \max}, P_{y \max}, P_{z \max})] / 2$. 最佳来球速度区域为($[V_{x \min}, V_{x \max}], [V_{y \min}, V_{y \max}], [V_{z \min}, V_{z \max}]$), 最佳来球速度平均值为 $(V_{x \text{avg}}, V_{y \text{avg}}, V_{z \text{avg}})^T = \{(V_{x \min}, V_{y \min}, V_{z \min})^T + (V_{x \max}, V_{y \max}, V_{z \max})^T\} / 2$.

4.2 决策搜索区域(Searching region determination)

在乒乓球的反弹后轨迹上选择 m 个平均分布的离散点作为决策论域 $U1 = \{p_1, p_2, \dots, p_m\}$. 根据乒乓球反弹后位置坐标和速度与最优击球区域和来

球速度的欧氏距离定义指标函数. 由于击球点位置和来球速度均有3个分量, 故共有6个目标函数:

$$V = \{d_1, d_2, d_3, d_4, d_5, d_6\}, \quad (8)$$

其中: d_1, d_2, d_3 是与轨迹点位置相关的指标函数, d_4, d_5, d_6 是与轨迹点速度相关的指标函数, 其值通过下式确定:

$$d_i = \begin{cases} e^{|p_i - D_{i \min}|}, & d_i < D_{i \min}, \\ 0, & d_i \in [D_{i \min}, D_{i \max}], \\ e^{|p_i - D_{i \max}|}, & d_i > D_{i \max}, \end{cases} \quad (9)$$

其中 $[D_{i \min}, D_{i \max}]$ 为第 i 指标所定义的最优区域.

定义相关因素重要程度模糊子集

$$A_1 = \left(\frac{k_1}{P_{x \max} - P_{x \min}}, \frac{k_2}{P_{y \max} - P_{y \min}}, \frac{k_3}{P_{z \max} - P_{z \min}}, \frac{k_4}{V_{x \max} - V_{x \min}}, \frac{k_5}{V_{y \max} - V_{y \min}}, \frac{k_6}{V_{z \max} - V_{z \min}} \right), \quad (10)$$

其中 $k_i (i = 1, 2, \dots, 6)$ 是位置和速度相关分量的权重系数, 由实验确定其值.

由决策算法选定最优方案 p_j , 并将区间 $[p_{j-1}, p_{j+1}]$ 设为击球点的搜索区域.

5 决策击球点(Determination of striking position)

5.1 求解期望回球速度(Computation of desired outgoing velocity of the ball)

给定击球点 $p_h = (x_h, y_h, z_h)$, 期望落球点 $p_L = (x_L, y_L, z_r)$, z_r 表示乒乓球的半径. 在 p_3 处, 乒乓球在 X - Y 平面上的期望合速度为 V_{xyd} .

考虑到空气阻力的影响, 用二阶多项式拟合乒乓球轨迹. 如下式所示:

$$\begin{pmatrix} x(t) \\ y(t) \\ z(t) \end{pmatrix} = \begin{pmatrix} a_x t^2 + b_x t + c_x \\ a_y t^2 + b_y t + c_y \\ a_z t^2 + b_z t + c_z \end{pmatrix}, \quad (11)$$

其中 $a_x, a_y, a_z, b_x, b_y, b_z, c_x, c_y, c_z$ 是未知参数.

当 $t = 0$, 乒乓球经过击球点 p_h ,

$$\begin{pmatrix} x(t) \\ y(t) \\ z(t) \end{pmatrix} = \begin{pmatrix} a_x t^2 + b_x t + x_h \\ a_y t^2 + b_y t + y_h \\ a_z t^2 + b_z t + z_h \end{pmatrix}, \quad (12)$$

则球的飞行速度为

$$\begin{pmatrix} v_x(t) \\ v_y(t) \\ v_z(t) \end{pmatrix} = \begin{pmatrix} 2a_x t + b_x \\ 2a_y t + b_y \\ 2a_z t + b_z \end{pmatrix}. \quad (13)$$

假设球在空气中的飞行时间为 t_f , 则 $t = t_f$ 时, 乒乓球经过落点 p_L ,

$$\begin{pmatrix} a_x t_f^2 + b_x t_f + c_x \\ a_y t_f^2 + b_y t_f + c_y \\ a_z t_f^2 + b_z t_f + c_z \end{pmatrix} = \begin{pmatrix} x_L \\ y_L \\ z_r \end{pmatrix}. \quad (14)$$

忽略Magnus力的影响, 只考虑空气阻力和重力的影响, 得到球的加速度约束如下:

$$\begin{pmatrix} 2a_x \\ 2a_y \\ 2a_z \end{pmatrix} = \begin{pmatrix} -k_m \|\vec{v}\| (2a_x t_f + b_x) \\ -k_m \|\vec{v}\| (2a_y t_f + b_y) \\ -k_m \|\vec{v}\| (2a_z t_f + b_z) - g \end{pmatrix}, \quad (15)$$

其中 g 为重力加速度. 利用落点速度计算加速度, 则 $\|\vec{v}\| = \sqrt{v_{xyd}^2 + (2a_z t_f + b_z)^2}$.

由期望落球速度, 可得

$$(2a_x t_f + b_x)^2 + (2a_y t_f + b_y)^2 = v_{xyd}^2. \quad (16)$$

联立式(13)-(15), 利用Levenberg-Marquardt(LM)算法可求解各系数. 对式(11)求导数, 可得机器人期望出球速度

$$V_{\text{back}} = [V_{x\text{back}} \ V_{y\text{back}} \ V_{z\text{back}}]^T = [b_x \ b_y \ b_z]^T.$$

5.2 能量变化函数(Energy functions)

在第3节中根据乒乓球的来球轨迹和机器人的最优击球参数, 确定了击球点的搜索区域. 对于任何选定的击球点和给定的落点, 可按照4.1中的方法计算回球速度. 搜索优化球拍运动的轨迹点, 将其作为击球点, 以进一步降低机器人回球难度. 在球拍击球过程中, 主要考虑球拍3个击球参数: 旋转角 θ_s 、俯仰角 θ_p 和 Y 轴的球拍速度 V_y . 击球过程中, 在球拍作用下, 乒乓球的动能和动量都会发生变化. 动能变化是由球与球拍碰撞过程中的能量损失和球拍击球引起的. 在排除了碰撞前后乒乓球动能损失的影响后, 主要与球拍击球速度有关. 动量变化是指乒乓球速度方向发生变化, 是由于球拍拍面反弹造成的, 主要与球拍的旋转角和俯仰角有关.

根据以上分析, 定义相关的能量函数代表乒乓球被击打前后的能量变化, 以其函数值定量地衡量此击球点的回球难度.

在选定的搜索区间 $[p_{j-1}, p_{j+1}]$ 内, 选择 n 个平均分布的离散轨迹点作为决策论域 $U2 = \{p_{j1}, p_{j2}, \dots, p_{jn}\}$. 由于论域内的相邻轨迹点间距已经比较小, 故决策后的离散点可以代替连续轨迹上的全局最优解. 令 $V_c = (V_{cx}, V_{cy}, V_{cz})^T$ 代表乒乓球在击球点的来球速度, $V_b = (V_{bx}, V_{by}, V_{bz})^T$ 代表乒乓球的期望回球速度, $p_h = (x_h, y_h, z_h)$ 为选定的击球点坐标, 根据球拍的镜面反射原理, 定义能量函数如式(17)-(19)所示:

$$F_1 = e^{\frac{1}{2} m_s V_b^T V_b - \frac{1}{2} k_\eta m_s V_c^T V_c}, \quad (17)$$

$$F_2 = e^{-\frac{\sqrt{k_\eta} V_{cx} V_{bx} + V_{cy} V_{by}}{\sqrt{k_\eta V_{cx}^2 + V_{cz}^2} \sqrt{V_{bx}^2 + V_{bz}^2}}}, \quad (18)$$

$$F_3 = e^{-\frac{\sqrt{k_\eta} V_{cx} V_{bx} + V_{cy} V_{by}}{\sqrt{k_\eta V_{cx}^2 + V_{cy}^2} \sqrt{V_{bx}^2 + V_{by}^2}}}, \quad (19)$$

其中: k_η 为乒乓球拍反弹系数, m_s 为乒乓球质量. F_1 表示乒乓球被击打前后的能量变化, F_2 和 F_3 表示乒乓球被击打前后的动量变化.

以式(17)–(19)定义的能量函数为指标函数

$$V2 = \{F_1, F_2, F_3\}. \quad (20)$$

定义相关因素重要程度模糊子集

$$A_2 = \left(\frac{k_7}{V_{y\max} - V_{y\min}}, \frac{k_8}{\theta_{p\max} - \theta_{p\min}}, \frac{k_9}{\theta_{s\max} - \theta_{s\min}} \right), \quad (21)$$

其中: $[V_{y\min}, V_{y\max}]$, $[\theta_{p\min}, \theta_{p\max}]$, $[\theta_{s\min}, \theta_{s\max}]$ 为乒乓球机器人关节的运动范围, $k_i (i = 7, 8, 9)$ 是球拍各运动分量的权重系数, 由实验确定其值.

根据决策算法选定最优方案 p_{ji} , 将其作为击球点发送至控制计算机, 由控制计算机规划球拍运动并控制机器人回球.

6 实验与结果(Experiments and results)

本文的实验平台是一个5自由度乒乓球机器人, 如图4所示. 智能相机将乒乓球的图像坐标发送至视觉计算机. 视觉计算机接收球的图像坐标并计算其三维坐标. 视觉计算机主频2.99 GHz, 内存2 GB. 控制计算机接收视觉计算机的控制参数并控制机器人的运动^[14–15]. 控制计算机主频1.87 GHz, 内存1 GB. 在控制坐标系 $\{M\}$ 中, 乒乓球机器人各平移关节的运动范围如表1所示, 来球轨迹和选择的击球点必须控制此范围内. 另外, 智能相机的采样周期为10 ms, 乒乓球的速度应控制在8 m/s以下.

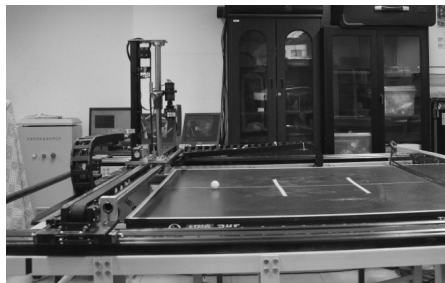


图4 5自由度乒乓球机器人

Fig. 4 Robotic table tennis with five degrees of freedom

表 1 关节参数

Table 1 Parameters of the robot's joints

关节名称	运动范围/mm	最大运动速度/(m·s ⁻¹)
X	-800 ~ 800	3.5
Y	0 ~ 1500	4
Z	200 ~ 400	1

6.1 求解期望出球速度(Computation of desired outgoing velocity)

为了验证第4.1节中提出的求解出球速度算法的有效性, 笔者采集了多组人打球时乒乓球的飞行轨迹, 进行仿真实验. 根据人打球时的击球点、击打后乒乓球的落点和落球速度, 利用第4.1节中算法计算乒乓球在击球点的出球速度, 并与利用乒乓球轨迹点计算得到的实际出球速度做对比. 在其中的6次实验中, 对比结果如表2所示.

表 2 实际出球速度与模型计算结果对比

Table 2 Comparison of the actual outgoing velocity and the velocity computed with the proposed method

No.	实际出球速度 (V_x, V_y, V_z)/(m·s ⁻¹)	模型计算结果 (V_x, V_y, V_z)/(m·s ⁻¹)
1	7.35, 0.23, -0.66	7.27, 0.21, -0.79
2	7.25, -0.04, -0.67	7.25, -0.05, -0.64
3	6.93, 0.07, -0.60	7.10, 0.03, -0.76
4	6.96, -0.22, -0.54	6.84, -0.15, -0.49
5	6.96, 0.07, -0.63	6.94, 0.06, -0.74
6	6.67, -0.10, -0.58	6.67, -0.06, -0.57

由表2可看出, 利用第4.1节中算法求解得到的出球速度误差可控制在0.1 m/s以内, 故可用计算得到的出球速度代表实际出球速度选择击球点.

6.2 最优击球点选择(Optimal striking position selection)

经过多次击球试验, 选定最佳击球区域为 $P_{x\min} = 1.850$ m, $P_{y\min} = 0.11$ m, $P_{z\min} = 0.22$ m, $P_{x\max} = 2.100$ m, $P_{y\max} = 0.485$ m, $P_{z\max} = 0.265$ m, 其中心点坐标为 $(P_{x\text{avg}}, P_{y\text{avg}}, P_{z\text{avg}}) = (1.975, 0.298, 0.243)$ m. 选定最佳来球速度为 $V_{x\min} = 1.90$ m/s, $V_{y\min} = -0.14$ m/s, $V_{z\min} = 0.30$ m/s, $V_{x\max} = 2.30$ m/s, $V_{y\max} = 0.30$ m/s, $V_{z\max} = 1.7$ m/s, 其平均速度为 $(V_{x\text{avg}}, V_{y\text{avg}}, V_{z\text{avg}}) = (2.10, 0.08, 1.00)$ m/s.

设 $p_{x\text{land}}$ 为在某次击球过程中, 乒乓球来球落点的 X 轴坐标. 选以下轨迹点上的乒乓球位置和速度为决策论域 $U1$: 单位 m.

$$U1 = \{p_{x\text{land}} + 0.07, p_{x\text{land}} + 0.14, p_{x\text{land}} + 0.21, p_{x\text{land}} + 0.28, p_{x\text{land}} + 0.35, p_{x\text{land}} + 0.42\}.$$

在视觉坐标系 $\{V\}$ 中, 决策的击球点在轨迹上的位置如图5所示. 在图5中, 乒乓球的来球落点坐标为(1610.5, 325.7, 20) mm, 搜索区域为 $([1899.2, 2123.1], [350.3, 370.4], [97.8, 180.0])$ mm, 决策得到的击球点坐标为(2050.9, 363.5, 156.3) mm. 由图5可看出, 利用本文算法决策得到的击球点基本符合人

的经验结果.

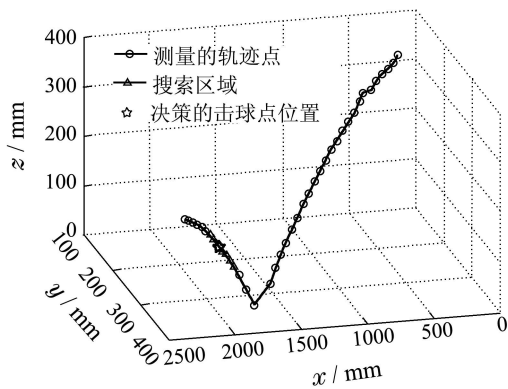


图5 决策的击球点在轨迹上的位置

Fig. 5 Selected striking positions on the trajectory

为了进一步对比本文算法与文献[8-9]中固定高度平面确定击球点的算法. 利用两种算法分别计算击球点, 计算期望出球速度并按照文献[12]中的方法计算球拍参数. 考虑到乒乓球来球轨迹的多样性, 笔者设计了3类实验, 分别在乒乓球反弹后竖直方向速度较小时, 反弹后竖直方向速度较大时和一般情况下对比两种算法.

在实验过程中, 利用固定高度平面算法计算击球点时, 根据打球经验, 将截止平面高度设为220 mm. 表3和表4分别是在乒乓球反弹后竖直方向速度分量较小和较大的两种情况下本文算法与固定高度平面算法的计算结果, 表5是在一般情况下两种算法的计算结果. 由表5知, 在一般情况下, 两种算法得到的击球点位置接近, 本文算法得到的球拍参数略优于固定高度平面算法. 由表3, 当乒乓球反弹后竖直方向速度分量较小时, 固定高度平面算法得到的击球点坐标与最优击球区域相比明显偏大, 球拍参数变化也较大, 则机器人迎球加速段距离较短, 且乒乓球被击打前后能量变化较大, 故回球难度较大, 可能造成回球失败. 由表4, 当乒乓球反弹后竖直方向速度分量较大时, 利用固定高度平面算法得到的击球点坐标与最优击球区域相比明显偏小, 则机器人在迎球阶段需要以较大的加速度到达击球点, 会加重机器人关节的运动负担, 而且选择的击球点过于靠前, 从预测点到击打点的预测剩余时间较短, 机器人可能没有充足的时间进行击球准备, 造成击球失败. 由上可知, 本文的决策算法能适应来球轨迹灵活多变的情况, 根据来球轨迹选择回球难度较小的击球点.

表3 V_z 较小时本文算法与文献[8-9]算法的击球参数

Table 3 Striking parameters computed with the proposed method and the method in [8-9] in the case of small V_z

No.	本文算法结果		文献[8-9]算法结果	
	击球点坐标 (x, y, z)/mm	球拍参数 (V_y, θ_s, θ_p)/(($m \cdot s^{-1}$), ($^\circ$), ($^\circ$))	击球点坐标 (x, y, z)/mm	球拍参数 (V_y, θ_s, θ_p)/(($m \cdot s^{-1}$), ($^\circ$), ($^\circ$))
1	2050.9, 363.5, 156.3	0.99, 7.61, -2.06	2290.9, 381.3, 220.0	1.42, 7.36, -7.35
2	2030.9, 151.4, 123.4	0.98, -2.27, -0.22	2277.5, 99.4, 220.9	1.43, -3.55, 5.06
3	2039.4, 350.1, 196.9	1.13, 7.83, -3.87	2153.4, 360.5, 220.0	1.30, 7.73, -6.24
4	2002.0, 315.2, 159.2	1.06, 6.83, -1.75	2220.3, 330.3, 220.2	1.43, 6.67, -6.74
5	2000.5, 329.9, 173.0	1.15, 7.08, -2.02	2155.7, 340.0, 220.3	1.31, 6.69, -4.67

表4 V_z 较大时本文算法与文献[8-9]算法的击球参数

Table 4 Striking parameters computed with the proposed method and the method in [8-9] in the case of large V_z

No.	本文算法结果		文献[8-9]算法结果	
	击球点坐标 (x, y, z)/mm	球拍参数 (V_y, θ_s, θ_p)/(($m \cdot s^{-1}$), ($^\circ$), ($^\circ$))	击球点坐标 (x, y, z)/mm	球拍参数 (V_y, θ_s, θ_p)/(($m \cdot s^{-1}$), ($^\circ$), ($^\circ$))
1	1905.5, 368.4, 317.9	2.11, 9.23, -2.97	1756.1, 360.2, 220.0	1.93, 9.64, 2.08
2	1979.7, 334.8, 368.2	2.14, 7.98, -2.55	1784.0, 326.8, 220.9	1.88, 8.54, 3.99
3	1955.8, 507.2, 314.0	2.18, 14.28, 0.50	1862.2, 486.9, 220.1	2.03, 14.30, 3.47
4	1811.2, 161.1, 287.2	2.00, 1.42, 0.67	1732.5, 173.6, 220.5	1.90, 1.85, 3.09
5	1917.3, 45.4, 304.3	2.01, -2.63, 2.32	1789.4, 74.1, 220.1	1.86, -1.95, -1.51

表5 一般情况下本文算法与文献[8-9]算法的击球参数

Table 5 Striking parameters computed with the proposed method and the method in [8-9] in the normal case

No.	本文算法结果		文献[8-9]算法结果	
	击球点坐标 (x, y, z)/mm	球拍参数 (V_y, θ_s, θ_p)/(($m \cdot s^{-1}$), ($^\circ$), ($^\circ$))	击球点坐标 (x, y, z)/mm	球拍参数 (V_y, θ_s, θ_p)/(($m \cdot s^{-1}$), ($^\circ$), ($^\circ$))
1	2001.4, 152.5, 186.7	0.90, 3.84, -0.09	2093.2, 159.1, 220.3	1.03, 4.14, -1.62
2	2007.2, 99.2, 215.8	0.95, 2.74, -0.82	2022.2, 100.0, 220.5	0.97, 2.78, -1.118
3	2010.2, 290.6, 188.0	0.84, 6.81, -1.75	2131.5, 302.5, 220.4	0.99, 6.76, -3.83
4	1894.8, 136.4, 227.3	0.96, 4.00, 0.26	1876.0, 134.9, 220.8	0.94, 3.96, 0.61
5	1957.4, 61.8, 185.2	0.76, 1.52, 0.35	2058.2, 67.5, 220.6	0.88, 1.72, -1.06

利用上述两种击球点选择算法进行分次击球实验. 实验过程中, 笔者控制乒乓球落点, 让其均匀分布在机器人一边的球桌上, 统计机器人成功回球次数. 在50次分次击球实验中, 本文算法成功回球39次, 固定高度平面成功回球27次. 上述实验结果表明, 本文的击球点选择算法有助于降低回球难度, 提高机器人回球率.

7 结论(Conclusions)

本文提出了一种利用多目标模糊决策确定击球点的方法. 首先, 根据机器人的最优击打区间和来球速度确定了包含击球点的搜索区域. 其次, 利用击球点和回球目标参数求解期望出球速度, 并根据球拍参数定义了能量函数, 分别代表乒乓球被击打前后的能量和动量变化. 以能量函数为优化指标在搜索区域内决策得到击球点. 最后, 在不同的实验情况下, 对比本文算法与固定高度平面算法, 验证了本文算法的有效性. 实验表明, 本文算法能够适应各种不同的来球轨迹, 选择合适的击球点, 提高回球率.

参考文献(References):

- [1] ZHANG Z, XU D, YU J. Reseach and latest development of ping-pong robot player [C] // *Proceedings of the 7th World Congress on Intelligent Control and Automation*. Chongqing: IEEE, 2008: 4881 - 4886.
- [2] ANDERSSON R L. *A Robot Ping-Pong Player: Experiment in Real Time Intelligent Control* [M]. Cambridge, CA: MIT Press, 1988.
- [3] ANDERSSON R L. Aggressive trajectory generator for a robot ping-pong player [J]. *IEEE Control Systems Magazine*, 1989, 9(2): 15 - 21.
- [4] ACOSTA L, RODRIGO J, MENDEZ J A, et al. Ping-pong player prototype, a PC-based, low-cost, ping pong robot [J]. *IEEE Robotics and Automation Magazine*, 2003, 10(4): 44 - 52.
- [5] MATSUSHIMA M, HASHIMOTO T, TAKEUCHI M, et al. A learning approach to robotic table tennis robotics [J]. *IEEE Transactions on Robotics and Automation*, 2005, 21(4): 767 - 771.
- [6] MATSUSHIMA M, HASHIMOTO T, MIYAZAKI F. Learnig to the robot table tennis task-ball control and rally with a human [C] // *Proceedings of IEEE Conference on Systems, Man and Cybernetics*. Washington D C, USA: IEEE, 2003: 2962 - 2969.
- [7] MIYAZAKI F, MATSUSHIMA M, TAKEUCHI M. Learning to dynamically manipulate: a table tennis robot controls a ball and rallies with a human being [M] // *Advances in Robot Control*. Berlin, Germany: Springer Berlin Heidelberg, 2006.
- [8] ZHANG Z, XU D, TAN M. Visual mearurement and prediction of ball trajectory for table tennis robot [J]. *IEEE Transactions on Instrumentation and Measurement*, 2010, 59(12): 3195 - 3205.
- [9] 彭博, 洪永潮, 杜森森, 等. 乒乓球机器人击打点的预测方法[J]. *江南大学学报(自然科学版)*, 2007, 6(4): 433 - 437. (PENG Bo, HONG Yongchao, DU Sensen, et al. An approach to hit point prediction for ping-pong robot [J]. *Journal of Jinan University (Natural Science Edition)*, 2007, 6(4): 433 - 437.)
- [10] MULLING K, KOBER J, PETERS J. A biomimetic approach to robot table tennis [C] // *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*. Taipei, Taiwan: IEEE, 2010: 1921 - 1926.
- [11] MULLING K, PETERS J. A computational model of human table tennis for robot application [M] // *Autonome Mobile Systeme*. Berlin, Germany: Springer Berlin Heidelberg, 2009.
- [12] ZHANG P Y, LU T S. Real-time motion planning for a volleyball robot task based on a multi-agent technique [J]. *Journal of Intelligent and Robotic Systems*, 2007, 49(4): 355 - 366.
- [13] ZHANG P Y, LU T S. On trajectory planning of a volleyball robot based on blackboard model [J]. *Mechanical Science and Technology*, 2005, 24(5): 581 - 583.
- [14] YANG P, XU D, WANG H, et al. Control system design for a 5-DOF table tennis robot [C] // *Proceedings of the 11th International Conference on Control, Automation Robotics and Vision*. Singapore: IEEE, 2010: 1731 - 1735.
- [15] YANG P. *Research on motion planning and control strategy for cartesian robots*[D]. Beijing: Chinese Academy of Science, 2011.

作者简介:

苏 虎 (1986-), 男, 博士研究生, 目前研究方向为智能控制与机器学习, E-mail: suhumathe@yahoo.com.cn;

徐 德 (1965-), 男, 研究员, 目前研究方向为机器人视觉测量与控制、智能控制, E-mail: de.xu@ia.ac.cn;

陈国栋 (1985-), 男, 博士研究生, 目前研究方向为先进机器人控制、机器视觉, E-mail: dylancgd@hotmail.com;

方灶军 (1981-), 男, 助理研究员, 目前研究方向为机器人与自动化, E-mail: fzaoj@163.com.