

# 基于轨迹基元优化匹配的初次迭代控制信号提取方法

孔令新, 徐建明, 臧永灿, 朱自立, 王耀东

浙江工业大学 信息工程学院, 杭州 310023

E-mail: xujm@zjut.edu.cn

**摘要:** 在传统迭代学习控制 (ILC) 系统中, 只要参考轨迹发生变化, 需要重新经历一段训练学习过程, 且初次 ILC 往往达不到期望加工精度. 本文研究迭代学习控制中的初次迭代控制信号提取问题, 提出了一种基于轨迹基元优化匹配组合算法的初次迭代控制信号提取方法. 首先在一定相似度条件下, 基于 Kabsch 算法从轨迹基元库中搜寻出与当前期望轨迹空间形状相似的轨迹组合. 进而, 通过时间尺度变换和轨迹基元控制信号拼接提取出当前期望轨迹的初次迭代控制信号. 最后通过仿真验证了所提方法的有效性.

**关键词:** 迭代学习控制, 初次迭代控制信号, 轨迹基元, 优化匹配

## The Extraction of Initial Iterative Control Signals Based on Trajectory Primitives Matching

KONG Ling-xin, XU Jian-ming, ZANG Yong-can, ZHU Zi-li, WANG Yao-dong

Information Engineering Institute, Zhejiang University of Technology, Hangzhou 310023, China

E-mail: xujm@zjut.edu.cn

**Abstract:** In the traditional Iterative Learning Control (ILC) system, as long as the reference trajectory changes, the traditional ILC need to experience a period of learning, and the initial ILC cannot often reach an expectation processing precision. It is studied in this paper that the problem of initial iterative control signal extraction in ILC, which propose a method of initial iterative control signal extraction based on the trajectory primitive optimization matching and combination algorithm. Firstly, under the condition of a certain similarity, the combination trajectory, which is similar to the current desired trajectory with identical spacial pattern, is found in the library of trajectory primitives based on the Kabsch algorithm. Then, the initial iterative control signals of the desired trajectory are extracted by the time scale transformation and the trajectory primitive combination. Finally, the simulation is carried out to demonstrate the effectiveness of this method.

**Key Words:** Iterative Learning Control, Initial Iterative Control Signal, Trajectory Primitive, Optimal Matching

### 1 引言

迭代学习控制 (Iterative Learning Control, 简称ILC) 适合于一类具有重复运行特性的被控对象, 其任务是寻找控制信号, 使得被控系统的实际输出轨迹在有限时间区间上沿整个期望输出轨迹实现零误差的完全跟踪, 并且整个控制过程要求快速完成<sup>[1]</sup>. 然而, ILC 初次迭代控制信号往往默认为 0, 若能获取期望的初次迭代控制信号, 将有效降低达到控制精度要求的迭代学习次数.

目前, J.-X. Xu and T. Zhu等<sup>[2-4]</sup>提出通过寻找在幅值、时间尺度上与目标轨迹存在比例关系的轨迹, 然后对控制信号进行分析、变换获得期望轨迹的近似控制信号. Pieter Janssens等<sup>[5]</sup>人通过基于之前学习的相似轨迹和扰动模型设计出面向线性时不变系统的迭代学习控制初始化方法. Arif 等<sup>[6]</sup>基于以往作业经验信息和局部加权回归方法, 提出了一种估计当前作业初次迭代控制输入信号的ILC方法, 能有效降低初次迭代跟踪误差. Freeman 等<sup>[7]</sup>提出了一种基于经验和模型, 从以往具有相同长度参考轨迹的作业控制信息中, 提取初始迭代控制信号的优化方法, 并应用于龙门架机器人的轨迹跟踪控制. Hoelzle 等<sup>[8]</sup>提出了基于基本任务

的 ILC 方法, 将跟踪轨迹分解为一系列基本任务(或运动基元), 通过对基本任务进行迭代学习训练获取相应的控制信号; 新作业任务由一系列基本任务串联组合而成, 这样可提取基本任务的控制信号用于新作业任务的 ILC. 这类研究策略从挖掘适合当前作业的初次迭代控制输入信号出发, 试图减小初次迭代控制信号偏差、提高初次迭代控制精度和 ILC 收敛速度; 尤其是基于基本任务的 ILC 方法为解决作业区间和作业任务同时变化情况下的 ILC 问题提供了一种新的研究思路, 但针对一个新任务如何确定由哪些基本任务组合而成, 作业任务相似性数学描述、分析和度量等问题还没有涉及.

针对这些问题, 本文采用非均匀有理 B 样条曲线(Non-Uniform Rational B-Splines, 简称NURBS)描述期望轨迹, 基于 NURBS 轨迹分割算法<sup>[9]</sup>、Kabsch算法<sup>[10]</sup>给出轨迹空间形状相似性定义和轨迹基元优化匹配算法; 在满足一定匹配相似度下, 依次寻找相似轨迹基元, 建立相似轨迹基元序列并合成空间形状相似而时间尺度不同的相似期望轨迹组合; 通过相似轨迹基元的控制信号与期望轨迹的控制信号之间的时间尺度关系, 直接获取分段的初次迭代控制信号, 并采用过渡带依次拼接分段的初次迭代控制信号, 进而获得当前期望轨

此项工作得到国家自然科学基金资助, 项目批准号: 61374103.

迹的初次迭代控制信号, 可避免 ILC 从零开始学习的不足, 将有效缩短达到伺服控制精度的迭代学习次数; 最后通过仿真实例验证方法的有效性.

## 2 轨迹基元匹配的ILC系统结构

设基坐标系(或 {B} 坐标系)下描述的期望轨迹:

$$l_d(x(t), y(t), z(t)) = \begin{bmatrix} x_d(t) \\ y_d(t) \\ z_d(t) \end{bmatrix} \quad (1)$$

其中,  $x_d(t)$ 、 $y_d(t)$  和  $z_d(t)$  分别为笛卡尔空间  $X$ 、 $Y$  和  $Z$  轴的期望轨迹.

假设运动控制系统通过 ILC 方法, 已建立了基于 NURBS 模型的期望轨迹和相应控制信号的数据库; 通过轨迹基元匹配试图从轨迹数据库中搜寻与当前期望轨迹 (1) 相似的轨迹基元组合轨迹, 进而利用数据库中轨迹基元控制信号获得当前期望轨迹 (1) 的初次迭代控制信号:

$$U_{0d}(t) = \begin{bmatrix} u_{0dx}(t) \\ u_{0dy}(t) \\ u_{0dz}(t) \end{bmatrix} \quad (2)$$

其中,  $u_{0dx}(t)$ 、 $u_{0dy}(t)$  和  $u_{0dz}(t)$  分别为  $x_d(t)$ 、 $y_d(t)$  和  $z_d(t)$  的初次迭代控制信号.

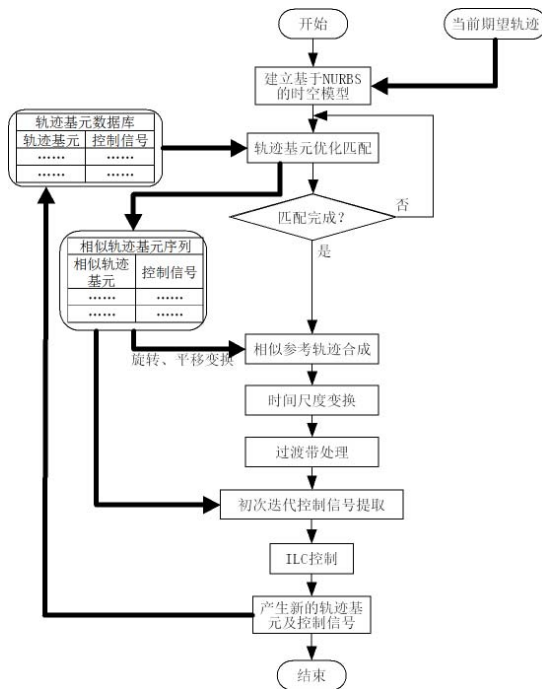


图 1: 轨迹基元匹配的 ILC 系统结构框图

基于轨迹基元匹配的 ILC 系统结构如图 1 所示. 首先采用 NURBS 模型描述当前期望轨迹; 研究基于 Kabsch 算法的轨迹空间形状相似性描述方法和轨迹基元优化匹配算法, 依次寻找相似轨迹基元, 建立相似轨迹基元序列并合成空间位置相似而时间尺度不同的轨迹组合; 通过相似轨迹基元的控制信号与期望轨迹

的控制信号之间的时间尺度关系, 直接获取分段的初次迭代控制信号<sup>[3]</sup>, 并采用过渡带依次拼接分段的初次迭代控制信号, 进而获得当前期望轨迹 (1) 的初次迭代控制信号 (2). 在针对当前期望轨迹的 ILC 过程中能够产生新的期望轨迹及其期望控制信号, 不断丰富轨迹基元数据库.

## 3 轨迹优化匹配组合算法

对于任一条轨迹, 运行时间为  $T$ , 则此轨迹记作:  $l(x(t), y(t), z(t))$ . 为了建立统一的轨迹优化匹配组合算法, 将轨迹总运行时间  $T$  归一化  $v = \frac{t}{T} \in [0, 1]$ ,  $v$  是归一化后的 NURBS 轨迹的节点矢量参数, 则轨迹  $l(t)|_{t \in [0, T]}$  可记作:  $l(v)|_{v \in [0, 1]}$ .

### 3.1 基于 Kabsch 算法的匹配相似性描述方法

针对两条 NURBS 轨迹, 我们设定一个统一的取点方式, 即按在节点矢量上取均匀参数的取点的方式, 按这种方式取点使得在 NURBS 曲率高的附近取的点密集, 在其曲率低的附近取的点稀疏, 这样取出的  $L_d$  和  $L_l$  两个序列概括了两条轨迹的基本特征, 即  $L_d$  和  $L_l$  两系列点能分别代表两条轨迹, 并且  $L_d$  和  $L_l$  序列含有相同个数的点. Kabsch 算法用来计算使两系列点的均方根偏差 (RMSD) 最小时的最优平移向量、旋转矩阵, 最小均方根偏差为  $lrm$ <sup>[12]</sup>.

最小均方根偏差  $lrm$  包含了两个序列  $L_d$  和  $L_l$  中的所有点, 因此, 我们可以将  $lrm$  作为轨迹相似性指标; 以下给出 NURBS 轨迹相似性以及叠合的定义.

**定义:**  $l_i(v)|_{v \in [0, 1]}$  通过旋转  $R$  和平移 <sup>$d$</sup>   $P_{IORG}$  变换为  $l_i''(v)|_{v \in [0, 1]}$  后,  $l_d(v)|_{v \in [0, 1]}$  与  $l_i''(v)|_{v \in [0, 1]}$  之间的最小均方根偏差  $lrm < \varepsilon$ ; 则称在相似度  $\varepsilon$  下轨迹  $l_d(v)|_{v \in [0, 1]}$  与  $l_i(v)|_{v \in [0, 1]}$  相似, 即  $l_d(v)|_{v \in [0, 1]} \stackrel{\varepsilon}{\sim} l_i(v)|_{v \in [0, 1]}$ , 并且轨迹  $l_d(v)|_{v \in [0, 1]}$  与  $l_i''(v)|_{v \in [0, 1]}$  叠合, 即  $l_d(v)|_{v \in [0, 1]} \stackrel{\varepsilon}{\approx} l_i''(v)|_{v \in [0, 1]}$ .

设  $l_d(v)|_{v \in [0, 1]}$  和  $l_i''(v)|_{v \in [0, 1]}$  的一阶导矢曲线分别为  $\dot{l}_d(v)|_{v \in [0, 1]}$  和  $\dot{l}_i''(v)|_{v \in [0, 1]}$ , 若  $l_d(v)|_{v \in [0, 1]} \stackrel{\varepsilon}{\sim} l_i''(v)|_{v \in [0, 1]}$ , 易知  $\dot{l}_d(v)|_{v \in [0, 1]} \stackrel{\varepsilon}{\sim} \dot{l}_i''(v)|_{v \in [0, 1]}$ , 即相似的两条轨迹, 各自的速度变化趋势也是相似的. 设轨迹  $l_d(v)|_{v \in [0, 1]}$  与  $l_i''(v)|_{v \in [0, 1]}$  的运行时间分别为  $T_d$  和  $T_l$ , 并且归一化之前两条轨迹的节点矢量  $U_d$  和  $U_l$  如式 (3) 所示, 控制点分别为  $P_{d_i}$  和  $P_{l_i}$ . 这两条轨迹的一阶导矢  $\dot{l}_d(v)|_{v \in [0, 1]}$  和  $\dot{l}_i''(v)|_{v \in [0, 1]}$  均为  $p-1$  阶的 NURBS 曲线, 其节点矢量分别为  $\dot{U}_d$  和  $\dot{U}_l$ , 如式 (4) 所示, 控制点分别为  $\dot{P}_{d_i}$  和  $\dot{P}_{l_i}$ , 如式 (5) 所示. 其中, 式中  $i = (0, 1, \dots, n_v, n_v + 1)$ ,  $n_v + 1$  为控制点的个数.

$$U_d = \underbrace{[0, \dots, 0]}_{p+1}, v_{p+1}, \dots, v_{m_v-p-1}, \underbrace{[1, \dots, 1]}_{p+1} \cdot T_d$$

$$U_l = \underbrace{[0, \dots, 0]}_{p+1}, v_{p+1}, \dots, v_{m_v-p-1}, \underbrace{[1, \dots, 1]}_{p+1} \cdot T_l \quad (3)$$

其中,  $m_v = n_v + p + 1$ .

$$\dot{U}_d = \underbrace{[0, \dots, 0]}_p, v_{p+1}, \dots, v_{m_v-p-1}, \underbrace{[1, \dots, 1]}_p \cdot T_d$$

$$\dot{U}_l = \underbrace{[0, \dots, 0]}_p, v_{p+1}, \dots, v_{m_v-p-1}, \underbrace{[1, \dots, 1]}_p \cdot T_l \quad (4)$$

$$\begin{aligned} \dot{P}_{d_i} &= \frac{p}{T_d(v_{i+p+1} - v_{i+1})} (P_{d_{i+1}} - P_{d_i}) \\ \dot{P}_{l_i} &= \frac{p}{T_l(v_{i+p+1} - v_{i+1})} (P_{l_{i+1}} - P_{l_i}) \end{aligned} \quad (5)$$

令  $k = \frac{T_l}{T_d}$ , 因为  $l_d(v)|_{v \in [0,1]}$  与  $l'_l(v)|_{v \in [0,1]}$  叠合, 所以  $P_{d_i}$  和  $P_{l_i}$  叠合, 并且  $\dot{U}_d$  和  $\dot{U}_l$  相同, 由公式(5)可得, 一阶导矢的控制点形成的凸包缩放了  $k$  倍, 即  $k \cdot \dot{P}_{l_i}$  形成的凸包与  $\dot{P}_{d_i}$  形成的凸包相叠合, 又一阶导矢的基函数均等同于原基函数, 所以  $\dot{l}_d(t)|_{t \in [0, T_d]}$  与  $\dot{l}'_l(t)|_{t \in [0, T_l/k]}$  之间缩放了  $k$  倍, 即  $\dot{l}_d(t)|_{t \in [0, T_d]} \approx \dot{l}'_l(t)|_{t \in [0, T_l/k]}$ , 同理,  $\ddot{l}_d(v)|_{v \in [0,1]} \approx \ddot{l}'_l(v)|_{v \in [0,1]}$ , 即调整其中一条轨迹的运行时间, 使其与另一条轨迹的运行时间相同, 则相似的两条轨迹上的运行速度及加速度均一致。

### 3.2 优化匹配组合方法

轨迹库 CurveBank 中的所有轨迹和期望轨迹均用 NURBS 描述, 由归一化后的节点矢量参数与  $v$  是一一对应的, 且对一条 NURBS 轨迹可以进行任意分割。设

轨迹库 CurveBank 中含有  $N$  条 NURBS 轨迹, NURBS 轨迹的分割参数  $v_1^i$  和  $v_2^i$ , ( $0 \leq v_1^i \leq v_2^i \leq 1$ ) 用来分割轨迹库 CurveBank 中的第  $i$  条轨迹  $l_i(v)|_{v \in [0,1]}$  ( $1 \leq i \leq n$ ,  $i \in N$ ), 轨迹  $l_i(v)|_{v \in [v_1^i, v_2^i]}$  (其中  $l_i(v)|_{v=v_1^i}$  和  $l_i(v)|_{v=v_2^i}$  分别为轨迹  $l_i(v)|_{v \in [v_1^i, v_2^i]}$  的起止点) 作为临时基元。我们将一条期望 NURBS 轨迹表示为  $l_d(v)|_{v \in [0,1]}$ , 则起点  $l_d(v)|_{v=0}$  与参数  $\bar{v}$  ( $0 < \bar{v} < 1$ ) 对应的点  $l_d(v)|_{v=\bar{v}}$  之间的轨迹段记为  $l_d(v)|_{v \in [0, \bar{v}]}$ 。

优化匹配的原则是在满足一定相似度  $\varepsilon$  的情况下, 期望轨迹的分割段数最少。通过类似二分查找法来匹配期望轨迹, 把与期望轨迹匹配的所有临时基元依次找出; 优化匹配算法流程如图 2 所示。其中,  $b$  为可变搜索步长,  $\sigma$  为最小搜索步长,  $n$  为匹配完后期望轨迹的分割段数。  $\bar{v}$  为期望轨迹  $l_d(v)|_{v \in [0,1]}$  的分割参数,  $l_i(v)|_{v \in [v_1^i, v_2^i]}$  表示轨迹库 CurveBank 中任意的临时基元。将所有与期望轨迹优化匹配的轨迹段依次存储在  $l_j(v)|_{v \in [0,1]}$  中, 所有  $l_j(v)|_{v \in [0,1]}$  组成了优化匹配基元库, 与  $l_j(v)|_{v \in [0,1]}$  对应的期望轨迹段为  $l_{d_j}(v)|_{v \in [0,1]}$ 。

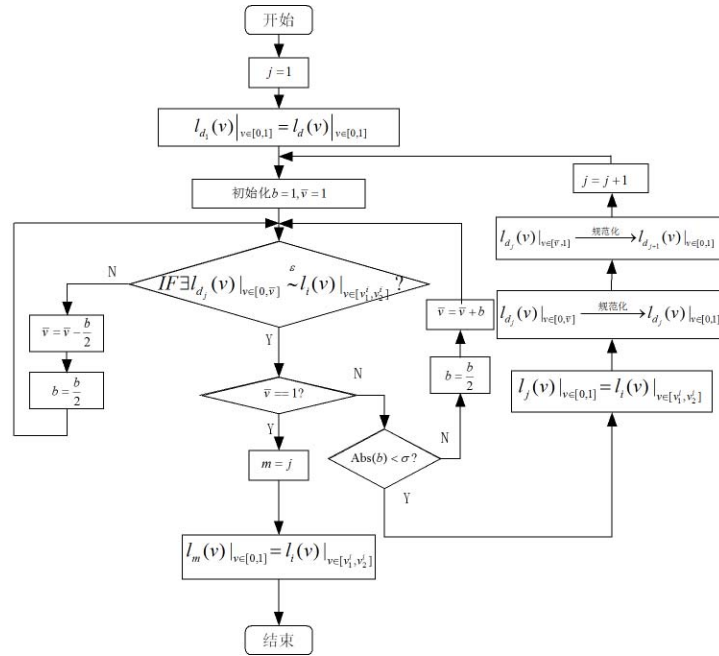


图 2: 优化匹配算法的结构流程图

对于确定的轨迹库 CurveBank 和已知的期望轨迹  $l_d(v)|_{v \in [0,1]}$ , 算法步骤如下:

**Step1:** 寻找与整条期望曲线  $l_d(v)|_{v \in [0,1]}$  最相似的一条曲线, 令  $l_{d_j}(v)|_{v \in [0,1]} = l_d(v)|_{v \in [0,1]}$  ( $j=1, l_{d_1}(v)|_{v \in [0,1]}$  表示第 1 段参考轨迹), 分割参数  $\bar{v}$  和搜索步长  $b$  初始化。

**Step2:** 由文献 [12] 得到两条曲线的最小均方根偏差  $lrm$ , 给定一个相似度  $\varepsilon > 0$ , 若  $lrm < \varepsilon$ , 则两条曲线相似, 跳入 Step3; 反之不相似, 跳入 Step4。

**Step3:** 若  $\bar{v} = 1$ , 跳入 Step6; 若  $\bar{v} \neq 1$ , 则判断搜索步长是否达到最小。若没达到, 则令搜索步长  $b = b/2$ ,

分裂参数  $\bar{v} = \bar{v} + b$ , 跳入 Step2; 若达到则跳入 Step5。

**Step4:** 令搜索步长  $b = b/2$ , 分裂参数  $\bar{v} = \bar{v} - b$ , 跳入 Step2。

**Step5:** 此时可得被分割的第  $j$  段期望曲线  $l_{d_j}(v)|_{v \in [0, \bar{v}]}$  和与其对应的匹配基元  $l_j(v)|_{v \in [0,1]}$ , 这样, 第  $j$  次匹配结束, 期望曲线  $l_{d_1}(v)|_{v \in [0, \bar{v}]}$  被分成两段期望曲线  $l_{d_j}(v)|_{v \in [0,1]}$  与  $l_{d_{j+1}}(v)|_{v \in [0,1]}$ 。将  $l_{d_{j+1}}(v)|_{v \in [0,1]}$  作为整条期望曲线, 令  $j = j + 1$ , 分割参数  $\bar{v}$  和搜索步长  $b$  初始化, 跳入 Step2。

**Step6:** 此时可得所有期望曲线段  $l_{d_i}(v)|_{v \in [0,1]}$  和与其对应的匹配基元  $l_j(v)|_{v \in [0,1]}$  ( $j = 1, \dots, n$ ), 优化匹

配结束。

在用相同的曲线库与同一期望曲线匹配时,若曲线越短,则越容易找到与其相似的曲线段。在已知的相似度  $\varepsilon$  和最小搜索步长  $\sigma$  下,若两曲线相似,则将分割点往后移,搜索更长的相似曲线段,若不相似,则将分割点往前移,搜索较短的相似曲线段;这样,每次匹配完成后都能得到最长的相似曲线段,从而保证匹配完后期望曲线被分割的段数最少。经过上述优化匹配算法后,分割点分别为  $l_{d_2}(v)|_{v=0}$ 、 $l_{d_3}(v)|_{v=0}$ 、 $\dots$ 、 $l_{d_n}(v)|_{v=0}$ ,与各个期望轨迹段匹配的各基元分别为  $l_1(v)|_{v \in [0,1]}$ 、 $l_2(v)|_{v \in [0,1]}$ 、 $\dots$ 、 $l_n(v)|_{v \in [0,1]}$ 。

对匹配基元  $l_j(v)|_{v \in [0,1]}$  ( $j = 1, \dots, n$ ) 进行旋转和平移变换得到  ${}^j l_j(v)|_{v \in [0,1]}$  ( ${}^j l_j(v)|_{v \in [0,1]}$  为匹配基元  $l_j(v)|_{v \in [0,1]}$  在其质心坐标系  $\{j\}$  下的表示),如式 (6) 所示,与其对应的期望基元段叠合如式 (7) 所示,其中,  $R_j$  和  ${}^j P_{BORG}$  分别是基元  ${}^j l_j(v)|_{v \in [0,1]}$  与对应的期望曲线段  $l_d(v)|_{v \in [0,1]}$  之间最优的旋转矩阵和平移向量。

$${}^j l_j(v)|_{v \in [0,1]} = l_j(v)|_{v \in [0,1]} + {}^j P_{BORG} \quad (6)$$

$$l_d(v)|_{v \in [0,1]} \stackrel{\varepsilon}{\approx} l_j''(v)|_{v \in [0,1]} \quad (7)$$

其中,  $l_j''(v)|_{v \in [0,1]} = R_j {}^j l_j(v)|_{v \in [0,1]} + {}^B P_{d_j,ORG}$ 。组合匹配基元  $l_j(v)|_{v \in [0,1]}$  ( $j = 1, \dots, n$ ), 获得 ILC 参考轨迹  $l_d(v)|_{v \in [0,1]}$  的相似参考轨迹,如式 (8), 其中  $\bigcup_{j=1}^n l_j''(v)|_{v \in [0,1]}$  表示将  $n$  条曲线  $l_j''(v)|_{v \in [0,1]}$  依次按各自质心在世界坐标系下的组合。

$$l_d(v)|_{v \in [0,1]} \stackrel{\varepsilon}{\approx} \bigcup_{j=1}^n l_j''(v)|_{v \in [0,1]} \quad (8)$$

设一条运行时间为  $T_d$  的期望轨迹被分成运行时间为  $T_{d_j}$  ( $j = 1, \dots, n$ ) 的  $n$  段轨迹分别记为  $l_{d_1}(t)|_{t \in [0, T_{d_1}]}$ 、 $l_{d_2}(t)|_{t \in [0, T_{d_2}]}$ 、 $\dots$ 、 $l_{d_n}(t)|_{t \in [0, T_{d_n}]}$ , 匹配基元  $l_j(v)|_{v \in [0,1]}$ , ( $j = 1, \dots, n$ ) 的运行时间为  $T_j$  ( $j = 1, \dots, n$ )。令  $k_j = \frac{T_j}{T_{d_j}}$ , 经过如式 (9) 所示的时间尺度变换后,  $l_j(t_j)|_{t_j \in [0, T_j]}$  与  $l_{d_j}(t_{d_j})|_{t_{d_j} \in [0, T_{d_j}]}$  的运行时间保持一致。

$$l_j''(t_j)|_{t_j \in [0, T_j]} \stackrel{\varepsilon}{\approx} l_{d_j}''(t_{d_j})|_{t_{d_j} \in [0, T_{d_j}]} \quad (9)$$

### 3.3 初次迭代控制信号的获取

#### 3.3.1 轨迹基元控制信号的获取

由式 (6) 和 (7), 可以得

$$l_j''(t_j)|_{t_j \in [0, T_j]} = R_j \cdot {}^j l_j(t_j)|_{t_j \in [0, T_j]} + {}^B P_{d_j,ORG} \quad (10)$$

因此, 根据线性系统叠加定理, 可得出第  $j$  段相似参考轨迹  $l_j''(t_j)|_{t_j \in [0, T_j]}$  的初次迭代控制信号为:

$$u_{0j}(t_j)|_{t_j \in [0, T_j]} = R_j^{-1} \cdot {}^j u_j(t_j)|_{t_j \in [0, T_j]} + u\{{}^B P_{d_j,ORG}\} \quad (11)$$

式中,  ${}^j u_j(t_j)|_{t_j \in [0, T_j]}$  是轨迹基元  ${}^j l_j(x(t_j), y(t_j), z(t_j))$  的控制信号,  $u\{{}^B P_{d_j,ORG}\}$  是  ${}^B P_{d_j,ORG}$  的控制信号。设

定  $P_x, P_y, P_z$  分别是 X、Y、Z 轴的传递函数, 则

$$R_j^u = \begin{bmatrix} R_{j11} & R_{j12} \frac{P_y(s)}{P_x(s)} & R_{j13} \frac{P_z(s)}{P_x(s)} \\ R_{j21} \frac{P_x(s)}{P_y(s)} & R_{j22} & R_{j23} \frac{P_z(s)}{P_y(s)} \\ R_{j31} \frac{P_x(s)}{P_z(s)} & R_{j32} \frac{P_y(s)}{P_z(s)} & R_{j33} \end{bmatrix}$$

此外, 由 (6) 式可知:

$${}^j l_j(t_j)|_{t_j \in [0, T_j]} = l_j(t_j)|_{t_j \in [0, T_j]} + {}^j P_{BORG} \quad (12)$$

而且,

$${}^j l_j(t_j) = \begin{bmatrix} {}^j x_j(t_j) \\ {}^j y_j(t_j) \\ {}^j z_j(t_j) \end{bmatrix}, l_j(t_j) = \begin{bmatrix} x_j(t_j) \\ y_j(t_j) \\ z_j(t_j) \end{bmatrix}$$

设定

$${}^j P_{BORG} = m_j^x \begin{bmatrix} I_x \\ 0 \\ 0 \end{bmatrix} + m_j^y \begin{bmatrix} 0 \\ I_y \\ 0 \end{bmatrix} + m_j^z \begin{bmatrix} 0 \\ 0 \\ I_z \end{bmatrix},$$

其中,  $m_j^x, m_j^y, m_j^z$  分别是  ${}^j P_{BORG}$  在  $x, y, z$  轴方向的坐标值,  $I_x, I_y, I_z$  分别是  $x, y, z$  轴方向上的单位平移基元, 其对应的控制信号为  $u_{I_x}, u_{I_y}, u_{I_z}$ 。那么  ${}^j l_j(t_j)$  对应的在  $x, y, z$  轴上的描述为:

$${}^j x_j(t_j) = x_j(t_j) + m_j^x I_x(t_j) \quad (13)$$

$${}^j y_j(t_j) = y_j(t_j) + m_j^y I_y(t_j) \quad (14)$$

$${}^j z_j(t_j) = z_j(t_j) + m_j^z I_z(t_j) \quad (15)$$

依据线性系统叠加定理, 可以得到其对应的控制信号满足下面的关系:

$${}^j u_{x_j}(t_j) = u_{x_j}(t_j) + m_j^x u_{I_x}(t_j) \quad (16)$$

$${}^j u_{y_j}(t_j) = u_{y_j}(t_j) + m_j^y u_{I_y}(t_j) \quad (17)$$

$${}^j u_{z_j}(t_j) = u_{z_j}(t_j) + m_j^z u_{I_z}(t_j) \quad (18)$$

其中,  ${}^j u_{x_j}(t_j), u_{x_j}(t_j), {}^j u_{y_j}(t_j), u_{y_j}(t_j), {}^j u_{z_j}(t_j), u_{z_j}(t_j)$  分别指的是  ${}^j x_j(t_j), x_j(t_j), {}^j y_j(t_j), y_j(t_j), {}^j z_j(t_j), z_j(t_j)$  的控制信号。

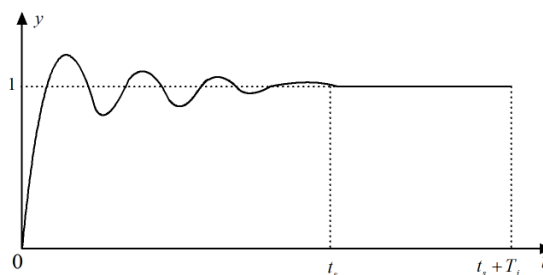


图 3: 单位阶跃响应

注: 在这里, 可以通过下面的方法获取单位平移基元及其控制信号。如图 3 所示, 当  $t > t_s$  时, 系统的输出才趋于稳态, 所以取  $I' \approx 1(t \in (t_s, t_s + T_j))$ , 这里  $t_s$  指的是调节时间,  $T_j$  是经历轨迹  $l_j(x(t), y(t), z(t))$  所用的时间。我们令  $I(0, T_j) = I'(t_s, t_s + T_j)$  作为单位平移基元, 其对应的控制信号为  $u_I$ 。按照此方法, 可以获得  $I_x, I_y, I_z$  对应的  $u_{I_x}, u_{I_y}, u_{I_z}$ 。类似的, 也可以得出  $u\{{}^j P_{BORG}\}$  的控制信号。

### 3.3.2 基于时间尺度变换的轨迹控制信号的获取

在轨迹基元匹配过程中, 通过优化匹配组合算法, 将同一条期望轨迹  $l_d(t_d)|_{t_d \in [0, T_d]}$  分割成两种不同的情况, 第一次期望轨迹被分割为  $n_1$  段, 记为  $l_{d_{1j}}(t_{d_{1j}})|_{t_{d_{1j}} \in [0, T_{d_{1j}}]}$ , ( $\sum_{j=1}^{n_1} T_{d_{1j}} = T_d, j = 1, 2, \dots, n_1$ ), 其对应的分割点为  $g_1^j$  ( $j = 0, 1, \dots, n_1$ ); 第二次期望轨迹被分割为  $n_2$  段, 记为  $l_{d_{2j}}(t_{d_{2j}})|_{t_{d_{2j}} \in [0, T_{d_{2j}}]}$ , ( $\sum_{j=1}^{n_2} T_{d_{2j}} = T_d, j = 1, 2, \dots, n_2$ ), 对应的分割点为  $g_2^j$  ( $j = 0, 1, \dots, n_2$ ); 这里, 两次分割的期望轨迹的始末端点值分别记为  $g_1^0, g_1^{n_1}, g_2^0, g_2^{n_2}$ , 定义为特殊的分割点, 且  $g_1^0 = g_2^0, g_1^{n_1} = g_2^{n_2}$ , 但  $g_1^j$  和  $g_2^j$  不完全相同, 得到两条不同的与期望轨迹相匹配的相似轨迹组合如图 4 所示. 其对应的时间变量分别为  $t_{d_{1j}}, t_{d_{2j}}$ , 与其对应得两条相似轨迹组合分别为相似轨迹组合 1 和相似轨迹组合 2, 分别记为  $l'_{1j}(t_{1j})|_{t_{1j} \in [0, T_{1j}]}$ , ( $\sum_{j=1}^{n_1} T_{1j} = T_1, j = 1, 2, \dots, n_1$ ),  $l'_{2j}(t_{2j})|_{t_{2j} \in [0, T_{2j}]}$ , ( $\sum_{j=1}^{n_2} T_{2j} = T_2, j = 1, 2, \dots, n_2$ ). 两条相似轨迹组合与期望轨迹具有相同形状, 但时间尺度不一样, 它们与期望轨迹之间存在的时间尺度关系分别为:  $t_{1j} = k_1^j t_{d_{1j}}, t_{2j} = k_2^j t_{d_{2j}}$ .

合并两组相似轨迹组合的分割点, 得到同一分割点集记为  $G$ , 如式 (19) 所示:

$$G = \{g_1^0, \dots, g_1^i, \dots, g_1^{n_1}\} \cup \{g_2^0, \dots, g_2^i, \dots, g_2^{n_2}\} \\ = \{g_d^0, g_d^1, \dots, g_d^r, \dots, g_d^s\} \quad (19)$$

其中,  $\max(n_1 + 1, n_2 + 1) \leq s \leq n_1 + n_2 + 2, g_d^0, g_d^s$  分别是轨迹  $l'_{1j}(t_{1j})|_{t_{1j} \in [0, T_{1j}]}$  的始末端点. 在分割点  $G$  相应的位置上对两条相似轨迹组合重新分割, 如图 5 所示, 此轨迹分别记为  $l''_{1j}(t_{1j})|_{t_{1j} \in [0, T_{1j}]}$  ( $\sum_{j=1}^{n_1} T_{1j} = T_1, j = 1, 2, \dots, n_1$ ),  $l''_{2j}(t_{2j})|_{t_{2j} \in [0, T_{2j}]}$  ( $\sum_{j=1}^{n_2} T_{2j} = T_2, j = 1, 2, \dots, n_2$ ).

取  $G$  中的任意一点  $g_d^r$  ( $0 < r < s$ ), ① 当  $g_1^j < g_d^r < g_1^{j+1}$  时, 在区间  $[g_1^j, g_d^r]$  和  $[g_d^r, g_1^{j+1}]$  上,  $l'_{1j}(t_{1j})|_{t_{1j} \in [0, T_{1j}]}$  对应的时间尺度关系分别为:  $\bar{k}_1^i = k_1^j, \bar{k}_1^{i+1} = k_1^{j+1}$ ; ② 当  $g_d^r = g_1^j$  时, 在区间  $[g_1^j, g_d^r]$  或  $[g_d^r, g_1^{j+1}]$  上,  $l'_{1j}(t_{1j})|_{t_{1j} \in [0, T_{1j}]}$  对应的时间尺度关系为:  $\bar{k}_1^i = k_1^j$ .

同理, 可得到  $l'_{2j}(t_{2j})|_{t_{2j} \in [0, T_{2j}]}$  的各分割点之间的时间尺度对应关系  $\bar{k}_2^i$ , ( $i = 1, \dots, s$ ).

通过轨迹基元匹配, 可得到两条相似轨迹组合, 分别取出相似轨迹组合的第  $i$  段, 它们在空间中的时间尺度不一样, 文献 [3] 得到期望轨迹的控制信号:

$$u_d = [I \quad I] (A_l^T \cdot A_l)^{-1} A_l^T \cdot \bar{u}_l \quad (20)$$

式中:

$$\bar{u}_l = [u_1^T, u_2^T]^T, \quad A_l = \begin{bmatrix} (\bar{k}_1^i)^{-1} \cdot I & I \\ (\bar{k}_2^i)^{-1} \cdot I & I \end{bmatrix}$$

注: 通过轨迹基元匹配, 得到  $a$  ( $2 \leq a$ ) 条时间尺度不一样的相似轨迹组合, 类似的, 由式 (20) 得到期望轨迹

的控制信号, 其中,

$$\bar{u}_l = [u_1^T, \dots, u_a^T]^T, \quad A_l = \begin{bmatrix} (\bar{k}_1^i)^{-1} \cdot I & I \\ \vdots & \vdots \\ (\bar{k}_a^i)^{-1} \cdot I & I \end{bmatrix}$$

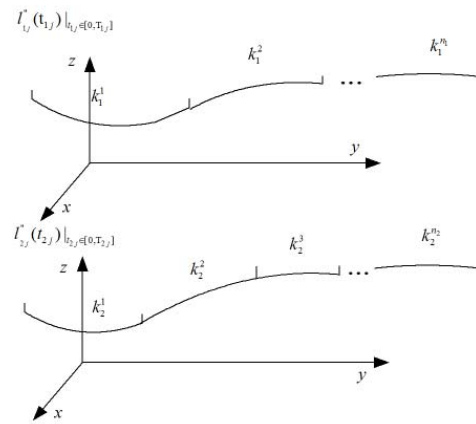


图 4: 原始相似轨迹组 1 和相似轨迹组 2

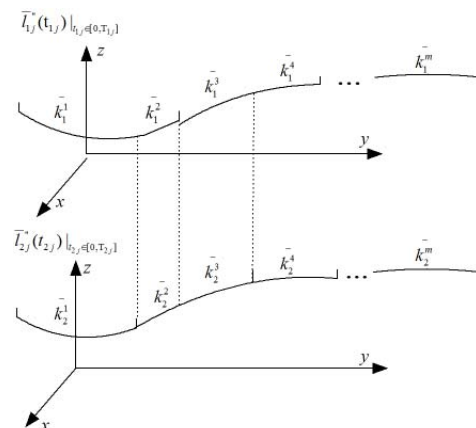


图 5: 重新分割后的相似轨迹组 1 和相似轨迹组 2

### 3.3.3 初次迭代控制信号的获取

通过轨迹基元匹配, 轨迹基元组成了期望轨迹的相似轨迹组合, 但在拼接的部分可能会出现如图 6 所示三种情况: ① 前后轨迹段交叉, ② 前后轨迹段不连续, ③ 前后轨迹段连续衔接. 因此, 前后轨迹段基元拼接处, 通过线性插值的方法对控制信号引入过渡带, 实现控制信号的无扰切换.

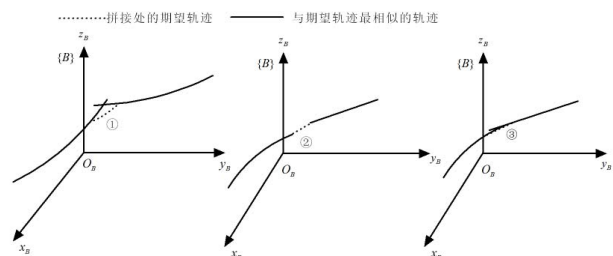


图 6: 拼接处可能出现的情况

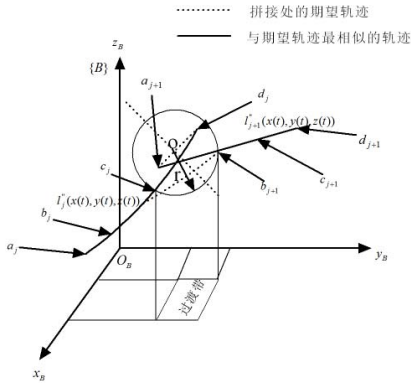


图 7: 第①种情况的拼接处的过渡带

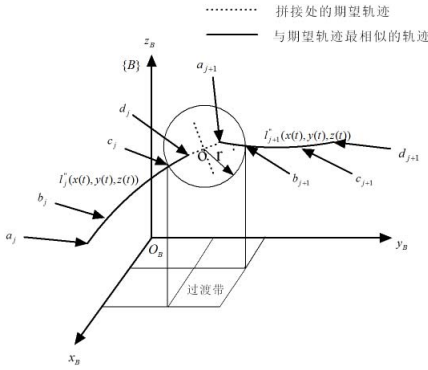


图 8: 第②种情况的拼接处的过渡带

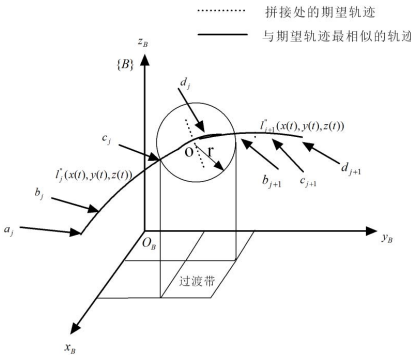


图 9: 第③种情况的拼接处的过渡带

针对第①种情况,如图7所示,取 $l_j''(x(t_j), y(t_j), z(t_j))$ 的末端 $d_j$ 和 $l_{j+1}''(x(t_{j+1}), y(t_{j+1}), z(t_{j+1}))$ 的始端 $a_{j+1}$ 的中心点 $O$ ,并以该点为球心, $r_j$ 为半径画球,分别交于 $c_j$ 、 $b_{j+1}$ 。 $a_j$ 是 $l_j''(x(t_j), y(t_j), z(t_j))$ 的始端点, $b_j$ 是上一个过渡带与 $l_j''(x(t_j), y(t_j), z(t_j))$ 的交点, $c_{j+1}$ 是下一个过渡带与 $l_{j+1}''(x(t_{j+1}), y(t_{j+1}), z(t_{j+1}))$ 的交点, $d_{j+1}$ 是 $l_{j+1}''(x(t_{j+1}), y(t_{j+1}), z(t_{j+1}))$ 的末端点。则过渡带的时间区间为:

$$T_j^{tr} = t_{b_{j+1}} - t_{c_j} \quad (21)$$

其中, $t_{b_{j+1}}$ 为点 $b_{j+1}$ 在期望轨迹 $l_d''(x(t_d), y(t_d), z(t_d))$ 上的时刻, $t_{c_j}$ 为点 $c_j$ 在期望轨迹 $l_d''(x(t_d), y(t_d), z(t_d))$ 上

的时刻.经过过渡带处理后的近似轨迹的时间长度与期望轨迹的时间长度的误差为:

$$\Delta T_j^{tr} = T_j^{tr} - (t_{cd_j} + t_{ab_{j+1}}) \quad (22)$$

其中, $t_{cd_j}$ 是轨迹 $l_j''(x(t_j), y(t_j), z(t_j))$ 上点 $c_j$ 和点 $d_j$ 之间的时间区间长度, $t_{ab_{j+1}}$ 是轨迹 $l_{j+1}''(x(t_{j+1}), y(t_{j+1}), z(t_{j+1}))$ 上点 $a_{j+1}$ 和点 $b_{j+1}$ 之间的时间区间长度.在过渡带中 $c_j$ 和 $b_{j+1}$ 对应的控制信号分别为 $u_{c_j}$ 和 $u_{b_{j+1}}$ ,则在控制信号 $u_{c_j}$ 和 $u_{b_{j+1}}$ 用线性插值的方法进行无扰切换.经过处理的控制信号存放在 $u_j^{tr}(t), t \in [0, T_j^{tr}]$ .

针对第②、③种情况,如图8、9所示,类似的,可以得到过渡带的时间区间如式(21)所示,经过过渡带处理后的相似轨迹组合的时间长度与期望轨迹的时间长度的误差如式(22)所示,但是第①种情况的 $\Delta T_j^{tr} < 0$ ,第②种情况的 $\Delta T_j^{tr} > 0$ ,第③种情况的 $\Delta T_j^{tr} = 0$ .假设期望轨迹 $l_d''(x(t_d), y(t_d), z(t_d))$ 由 $m$ 段轨迹元经过仿射变换近似获得.若 $\sum_{j=1}^{n-1} \Delta T_j \approx 0$ ,则期望轨迹的初次迭代控制信号为:

$$\bar{u}_{0_d} = \{u_1, \bar{u}_1^{tr}(t_1), \bar{u}_2^{tr}(t_2), \bar{u}_3^{tr}(t_3), \dots, \bar{u}_m^{tr}(t_m)\} \quad (23)$$

其中, $u_1(t)$ 是轨迹 $l_1''(x(t_1), y(t_1), z(t_1))$ 的控制信号, $t_1 \in [0, t_{c_1}]$ , $\bar{u}_1^{tr}(t_1) = u_1^{tr}(t_1 - t_{c_1})$ , $t_1 \in (t_{c_1}, t_{c_1} + T_1^{tr})$ 为轨迹 $l_1''(x(t_1), y(t_1), z(t_1))$ 和 $l_2''(x(t_2), y(t_2), z(t_2))$ 的过渡带控制信号.

$$\bar{u}_2(t) = u_2(t - (t_{c_1} + T_1^{tr} - t_{b_2})),$$

$$t \in (t_{c_1} + T_1^{tr}, t_{c_1} + T_1^{tr} + t_{c_2} - t_{b_2}],$$

$$\bar{u}_2^{tr}(t) = u_2(t - (t_{c_1} + T_1^{tr} + t_{c_2} - t_{b_2})),$$

$$t \in (t_{c_1} + T_1^{tr} + t_{c_2} - t_{b_2}, t_{c_1} + T_1^{tr} + t_{c_2} - t_{b_2} + T_2^{tr}],$$

$$\bar{u}_3(t) = u_3(t - (t_{c_1} + T_1^{tr} + t_{c_2} - t_{b_2} + T_2^{tr} - t_{b_3})),$$

$$t \in (t_{c_1} + T_1^{tr} + t_{c_2} - t_{b_2} + T_2^{tr},$$

$$t_{c_1} + T_1^{tr} + t_{c_2} - t_{b_2} + T_2^{tr} + t_{c_3} - t_{b_3}],$$

⋮

$$\bar{u}_n(t) = u_n(t_n - \sum_{j=1}^{n-1} (t_{c_j} + T_j^{tr} - t_{b_{(j+1)}})),$$

$$t \in (\sum_{j=1}^n (t_{c_j} + T_j^{tr}) - t_{b_{(j+1)}}), T_d]$$

其中, $T = T^* - \sum_{j=1}^{m-1} \Delta T_j^{tr}$ , $T^*$ 为整段相似参考轨迹的时间长度,且当 $\sum_{j=1}^{m-1} \Delta T_j^{tr} < 0$ 时, $\bar{u}_m = u_m(T^*)$ , $t_m \in [T^*, T]$ .

#### 4 仿真实验

设XYZ运动控制平台各轴的设定速度至实际位移之间的线性动态特性可近似为如式(24)所示<sup>[15]</sup>.

$$P_x(s) = \frac{1.066}{0.1876s^2 + s}$$

$$P_y(s) = \frac{1.1746}{0.2687s^2 + s}$$

$$P_z(s) = \frac{2.0523}{0.3138s^2 + s} \quad (24)$$

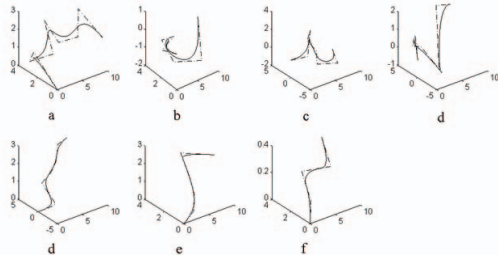


图 10(a) ~ (f): 三维轨迹库中的七条轨迹

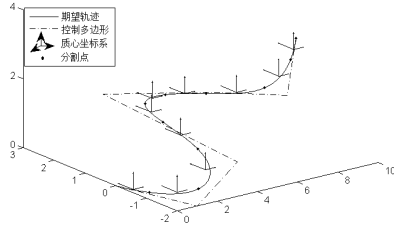


图 11: 分段期望轨迹 1 及其质心坐标系

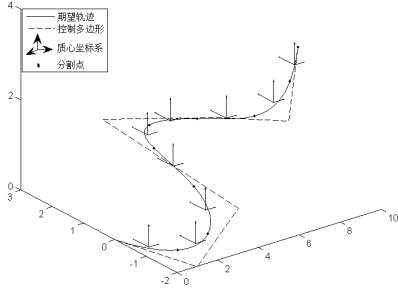


图 12: 分段期望轨迹 2 及其质心坐标系

首先针对应用背景, 按照以往的经验提取出经验轨迹, 建立轨迹数据库, 将期望的轨迹与轨迹数据库中的所有轨迹进行优化匹配组合. 三维轨迹库中有 7 条 NURBS 描述的轨迹由图 10(a) ~ (f) 中的实线表示, 各控制多边形由虚线表示. 经优化匹配算法分割之后的各段期望轨迹及其各自质心坐标系如图 11、12 所示, 图 11 期望轨迹被分割为 10 段, 图 12 期望轨迹被分割为 9 段, “.” 为期望轨迹上的分割点. 图 13(a) ~ (j) 分别表示与图 11 中各期望轨迹段对应的匹配基元及各自质心坐标系, 图 14(a') ~ (i') 分别表示与图 12 中各期望轨迹段对应的匹配基元及各自质心坐标系. 由此可见, 经过优化匹配组合方法后, 可以得到

$$l_{d_1}(v)|_{v \in [0,1]} \approx \bigcup_{j=1}^{10} l_{1j}''(v)|_{v \in [0,1]},$$

$$l_{d_2}(v)|_{v \in [0,1]} \approx \bigcup_{j=1}^9 l_{2j}''(v)|_{v \in [0,1]}$$

即两条组合轨迹分别与期望轨迹相似. 期望轨迹采用本文提出的方法和采用传统方法迭代 1 次的跟踪效果分别如图 15、16 所示, 此外, 如图 17、18 所示, 期望轨迹  $x_d(t)$  采用本文提出的方法第 1 次跟踪误差的 RMS 为 0.1096, 在第 3 次为 0.0048; 采用传统方法的第 1 次跟踪误差的 RMS 为 0.505, 第 5 次为 0.0062; 期望轨迹  $y_d(t)$  采用本文提出的方法第 1 次跟踪误差的 RMS 为 0.018, 在第 3 次为 0.0065; 采用传统方法的第 1 次跟踪误差的 RMS 为 0.0633, 第 5 次为 0.0069; 期望轨迹  $z_d(t)$  采用本文提出的方法的第 1 次跟踪误差的 RMS 为 0.071, 在第 3 次为 0.013; 采用传统方法的第 1 次跟踪误差的 RMS 为 0.157, 第 3 次为 0.031.

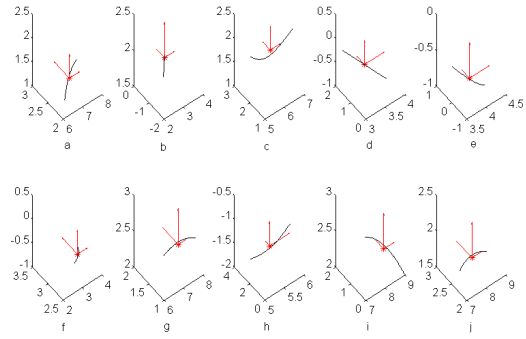


图 13(a) ~ (j): 相似的轨迹 1 的基元及其质心坐标系

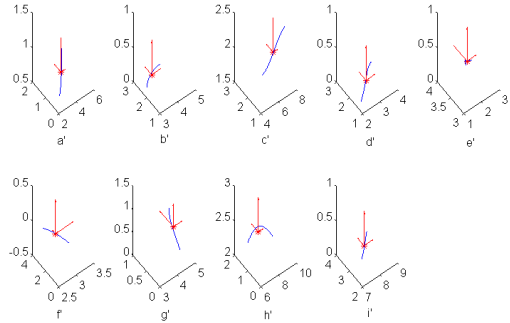


图 14(a') ~ (i'): 相似的轨迹 2 基元及其质心坐标系

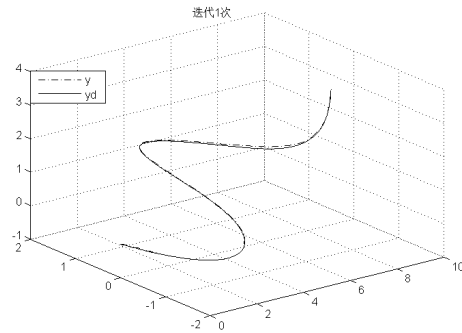


图 15: 采用本方法迭代 1 次的跟踪效果

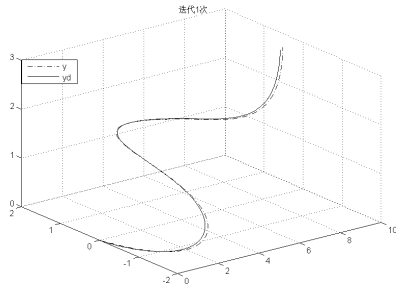


图 16: 采用传统方法迭代 1 次的跟踪效果

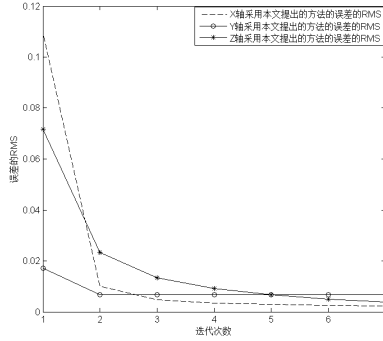


图 17: XYZ轴采用本文方法误差的 RMS 值

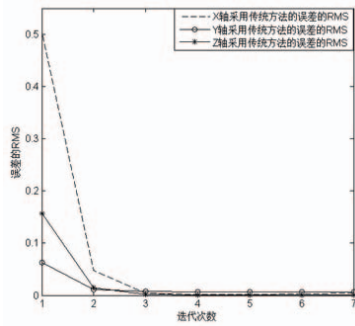


图 18: XYZ轴采用传统方法误差的 RMS 值

## 5 结论

本文通过优化匹配算法, 寻找到与期望轨迹匹配的相似轨迹组合, 利用轨迹库中已存储的且存在一定时间尺度关系的轨迹, 获得期望轨迹的初次迭代控制信号。仿真实验结果表明, 本文提出的基于轨迹基元优化匹配算法的初次迭代控制信号提取方法能够显著降低初次迭代控制的跟踪误差, 有效地解决了传统迭代学习控制方法从零开始学习的问题。

## 参考文献

[1] S. Arimoto, S. Kawamura, F. Miyazaki. Bettering operation of robots by learning[J]. *Journal of Robot System*, 1984, 1(2): 123 – 140.

[2] J. X. Xu, T. Zhu. Dual-scale direct learning control of trajectory tracking for a class of nonlinear uncertain systems[J]. *IEEE Transaction on Automatic Control*, 1999, 44(10): 1884 – 1888.

[3] J. X. Xu. Direct learning of control efforts for trajectories with different time scales[J]. *IEEE Transaction on Automatic Control*. 1998, 43(7): 1027 – 1030.

[4] J. X. Xu. Direct learning of control efforts for trajectories with different magnitude scales[J]. *Automatica*, 1997, 33(12): 2191 – 2195.

[5] P. Janssens, G. Pipeleers, J. Swevers. Initialization of ILC based on a previously learned trajectory[C]. *American Control Conference*. Canada: ACC, 2012, 1: 610 – 614.

[6] M. Arif, T. Ishihara and H. Inooka. Incorporation of experience in iterative learning controllers using locally weighted learning. *Automatica*, vol. 37, no. 6, pp. 881 – 888, 2001.

[7] C. T. Freeman, M. A. Alsubaie, Z. Cai, E. Rogers and P.L. Lewin. "Initial input selection for iterative learning control", *Journal of Dynamic Systems, Measurement, and Control*, vol.133, no.5, pp. 054504, 2011.

[8] D. J. Hoelzle, A. G. Alleyne, A. J. Wagoner Johnson. Basis task approach to iterative learning control with applications to micro-robotic deposition[J]. *IEEE Transactions on Control Systems Technology*, 2011, 19(5): 1138 – 1148.

[9] 施法中. 计算机辅助几何设计与非均匀有理 B 样条 CAGD & NURBS. 北京航空航天大学出版社, 北京, 275 – 277, 1994.

[10] Kabsch W. A solution for the best rotation to relate two sets of vectors. *Acta Crystallographica Section A: Crystal Physics, Diffraction, Theoretical and General Crystallography*, 32(5): 922–923, 1976.

[11] J. -X. Xu, and S. K. Panda, T. H. Lee. Introduction to ILC: Concepts, Schematics, and Implementation. *Real-time Iterative Learning Control*, London. U. K.: Springer-Verlag, 7 – 27, 2009.

[12] Umeyama, Shinj, Least-Squares Estimation of Transformation Parameters Between Two Point Patterns. *IEEE Trans. Pattern Anal. Mach. Intell.* 13(4): 376 – 380, 1991.

[13] Wasnik I, Sawlekar R, Kazi F, El al. Dual quaternion applications to bioinformatics for kinematical analysis of Protein Backbone Advances in Computing, Communications and Informatics (ICACCI), *International Conference on IEEE*, 2043 – 2048, 2013.

[14] J. J. Craig 著, 负超等译. 机器人学导论[M], 北京: 机械工业出版社, 2006. (J. J. Craig Author, C. Yun Interpreter. *Introduction to Robotics*[M], Beijing: China Machine Press, 2006.)

[15] J. -X. Xu, and S. K. Panda, T. H. Lee. Robust Optimal ILC Design for Precision Servo: Application to an XY Table. *Real-time Iterative Learning Control*, London. U. K.: Springer-Verlag, 29 – 44, 2009.