

空间 3R 机械手逆向运动学的多模块神经网络求解

肖 帆 李 光 游雨龙
湖南工业大学机械工程学院, 株洲, 412007

摘要:提出用多模块神经网络的方法求解空间 3R 机械手的逆运动学多解。通过几何分析,将关节空间划分为多个只有唯一逆运动学解的关节子空间,每个子空间均用 3 个单输出的 BP 神经网络训练和求解。通过仿真试验并与其他方法对比,表明该方法不仅可以准确地划分逆运动学解的取值范围,还可以快速求得高精度的逆运动学多解。

关键词:机器人逆运动学多解;BP 神经网络;关节子空间;3R 机械手

中图分类号:TP242.2

DOI:10.3969/j.issn.1004-132X.2019.10.014

开放科学(资源服务)标识码(OSID):



Multiple Module Neural Network Solving for Inverse Kinematics
of Space 3R Manipulators

XIAO Fan LI Guang YOU Yulong

College of Mechanical Engineering, Hunan University of Technology, Zhuzhou, Hunan, 412007

Abstract: A multiple module neural network method was proposed to solve the inverse kinematics multiple solutions of space 3R manipulators. The joint spaces were divided into several joint subspaces with unique inverse kinematics solution through geometric analysis. Each subspace was trained and solved by 3 single output BP neural networks. After the simulation and comparison with other methods, results show that the method herein may accurately divide the value ranges of inverse kinematics solutions, and quickly obtain high precision inverse kinematics multiple solutions.

Key words: inverse kinematics multiple solution of robot; BP neural network; joint subspace; 3R manipulator

0 引言

机器人逆运动学求解作为机器人离线编程、轨迹规划、控制算法设计等其他课题研究的基础,一直是机器人学中的一个经典问题,同样也是研究热点^[1]。逆运动学求解的实质是完成机器人工作空间到关节空间的映射,它们之间的映射关系是非线性且多耦合的,问题比较复杂。很多学者在该领域做了大量研究与探索,提出了许多理论与方法。传统方法有解析法^[2]、几何法^[3]和迭代法^[4]等。在求机器人逆运动学解问题中,解析法计算较为复杂,但是在机器人满足 PEIPER^[5]、DUFFY^[6]分别提出的结构特点时,可以得到全部解;几何法针对机器人的某些特殊结构进行简化,再进行求解,一般无法单独使用甚至根本无法使用^[7];迭代法受到初始值选取的约束,只能求得一组逆运动学解。

虽然传统方法中解析法和几何法能求机器人逆运动学的多解,但是需要满足特殊结构。随着机器人结构复杂化和计算机发展的突飞猛进,现代智能算法^[8-10]被应用来求逆运动学解。

神经网络对连续函数具有非常好的拟合能力及泛化能力,尤其是后者,使其被广泛应用于机器人逆运动学求解中。通常,在使用神经网络求机器人的逆运动学解时,采用如下思路进行:通过正向运动学,得出关节空间 Q 对应的工作空间位姿 T ,从而得到学习样本中输入与输出关系 $T \rightarrow Q$ 。在这种学习样本获取方法中,无法准确地建立一组位姿对应多组逆解,而只能建立一组位姿对应一组逆解的训练方式,这会导致在存在逆运动学多解的区域用训练好的神经网络求解时得到的是一个折中的结果。

文献[9]提出在关节子空间下,使用 6 个子神经网络组成的多层感知结构,解决了平面二连杆机器人逆运动学多解问题。文献[10]使用三层 BP 神经网络,通过增加方位角作为输入参数,消除了输入-输出间的映射错误关系。虽然文献[9]和文献[10]提出了应用神经网络解决机器人逆运动学多解的方法,但是二者都是在工作空间为二维平面运动的机器人中实现的,都没有解决三维工作空间中逆运动学多解的问题。末端执行器在三维工作空间的逆运动学多解问题远比二维平面的复杂。文献[11]提出多神经网络结构的方式,

求得了高精度的空间 3R 机械手的逆运动学完备解。

本文通过对空间 3R 机械手进行几何分析,确定了在关节空间中逆运动学多解的分布规律,根据该规律将关节空间划分为 4 个子空间,每个子空间用 3 个 BP 神经网络训练及求解,并在每个神经网络的输入中,增加绕 Z 轴旋转的矩阵元素作为特征参数;并进行了仿真验证。

1 三连杆机器人运动学分析

1.1 正向运动学模型

图 1a 所示是空间 3R 机械手,其中 (p_x, p_y, p_z) 表示末端执行器的空间位置,关节 1 的轴线为铅垂方向且和关节 2 的轴线垂直,其交点与基座的距离为 L_1 ;关节 2 和关节 3 的轴线水平且平行,距离为 L_2 ;末端执行器与关节 3 轴线的距离为 L_3 。根据标准 D-H 坐标^[12]建立图 1b 所示连杆坐标系,相应的连杆参数列于表 1。其中, $L_1=783\text{ mm}$, $L_2=702.5\text{ mm}$, $L_3=651\text{ mm}$, $\alpha_1=\pi/2$ 。根据表 1 所示连杆参数可求得其正向运动学方程为

$$\left. \begin{aligned} p_x &= \cos\theta_1 [L_2 \cos\theta_2 + L_3 \cos(\theta_2 + \theta_3)] \\ p_y &= \sin\theta_1 [L_2 \cos\theta_2 + L_3 \cos(\theta_2 + \theta_3)] \\ p_z &= L_1 + L_2 \sin\theta_2 + L_3 \sin(\theta_2 + \theta_3) \end{aligned} \right\} \quad (1)$$

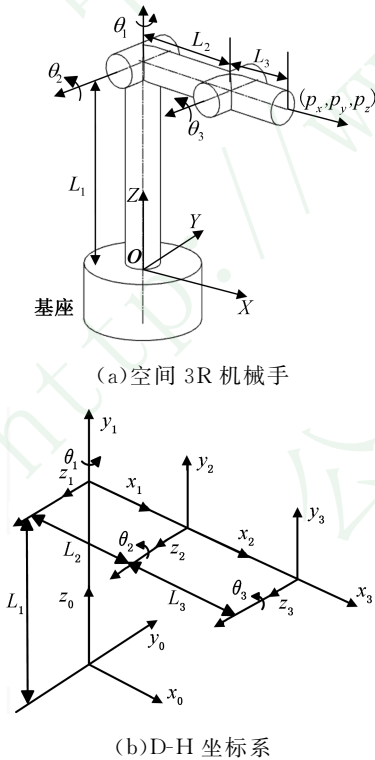


图 1 空间 3R 机械手及其连杆坐标系
Fig.1 Space 3R manipulator and its linkage coordinate system

表 1 空间 3R 机械手连杆参数

Tab.1 Spatial 3R manipulator connecting rod parameters

连杆	θ_i	d_i	a_i	α_i
1	θ_1	L_1	0	α_1
2	θ_2	0	L_2	0
3	θ_3	0	L_3	0

1.2 基于几何分析的关节子空间划分

对于空间 3R 机械手,其后面两个连杆相当于平面二连杆,根据文献[9],平面内一点,二连杆机器人有两种到达途径。令 θ_1 为任意定值 q ,只活动关节 2 和关节 3 时,三连杆机械手的工作空间为一个平面,取平面内一点 E (点 E 的坐标以基坐标系为参考),末端执行器到达 E 点的关节组合形式将有两种,组合 1 为 $\theta_2 \in (0, 2\pi)$, $\theta_3 \in (-\pi, 0)$;组合 2 为 $\theta_2 \in (0, 2\pi)$, $\theta_3 \in (0, \pi)$ 。本文将观察视角始终固定在 $\theta_1=q$ 时的平面的法向量方向上(图 2),并将该平面法向量方向记为“ k_1+ ”,此时观察由组合 1 得到的机械手空间形态,记为“ k_1+, k_2+ ”,组合 2 的记为“ k_1+, k_2- ”。当 $\theta'_1=q \pm \pi$ 时,与 $\theta_1=q$ 的平面重合,但是法向量方向改变为“ k_1- ”,此时末端执行器到达点 E 时,观察到的组合 1 变为“ k_1-, k_2- ”,组合 2 变为“ k_1-, k_2+ ”。

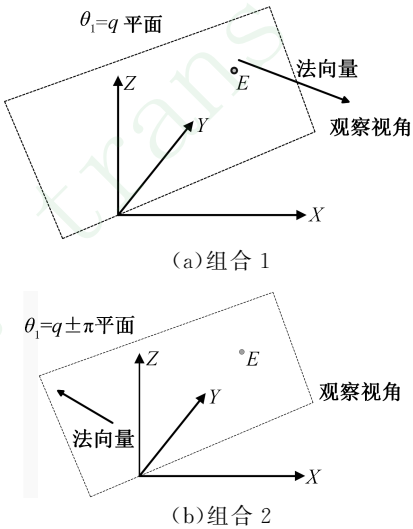


图 2 观察视角

Fig.2 Observation perspective

根据上述记法,将表 1 中的关节空间划分为表 2 中的 4 个子空间。

表 2 关节子空间

Tab.2 Joint subspaces

区域	θ_1	θ_2	θ_3	空间形态
Zone ₁	$[0, 165^\circ]$	$[15^\circ, 165^\circ]$	$[-135^\circ, 0]$	k_1+, k_2+
Zone ₂	$[0, 165^\circ]$	$[15^\circ, 165^\circ]$	$[0, 135^\circ]$	k_1+, k_2-
Zone ₃	$[-165^\circ, 0]$	$[15^\circ, 165^\circ]$	$[-135^\circ, 0]$	k_1-, k_2-
Zone ₄	$[-165^\circ, 0]$	$[15^\circ, 165^\circ]$	$[0, 135^\circ]$	k_1-, k_2+

1.3 特征参数

三连杆机械手的工作空间可以理解为二连杆形成的平面工作空间绕 Z 轴旋转形成的三维图，因此，表 2 中的 4 个区域在 Z 轴上的位置仍然有无穷多组解。参考文献[9]对接近点的描述可知，以 ϵ (ϵ 为最小正实数) 为半径、Z 轴为旋转轴的邻域内，在用神经网络训练时也会存在映射错误。

假设每个区域的 θ_{1i} 都平分成 $n-1$ 份，则每个区域对应的工作空间将由 n 个平面组成，在第 i 个平面中，末端执行器的位置都相同地增加一个绕 Z 轴旋转 θ_{1i} 的旋转矩阵 R ：

$$\theta_{1i} = \arctan2(p_y, p_x) \tag{2}$$

$$R = \text{rotz}(\theta_{1i}) \tag{3}$$

式(2)用于 Zone₁ 和 Zone₂ 中时，每个平面只需选择一个满足 $p_y > 0, p_x \neq 0$ 的点即可；式(2)用于 Zone₃ 和 Zone₄ 中时，同样只需选择一个满足 $p_y \leq 0, p_x \neq 0$ 的点即可。

最终 BP 神经网络的输入由矩阵 $T = [R \ P]$ 中的元素确定，其中 $P = [p_x \ p_y \ p_z]^T$ 。

2 BP 神经网络及其 MATLAB 实现

2.1 BP 神经网络的结构

BP 神经网络具有多层结构，分别是输入层、一层或多层隐含层、输出层。各层神经元之间是全连接的，层内无连接。隐含层各神经元节点采用 Sigmoid 函数^[13] 作为激励函数。图 3 所示是一个三层的 BP 神经网络，其中阈值未画出。

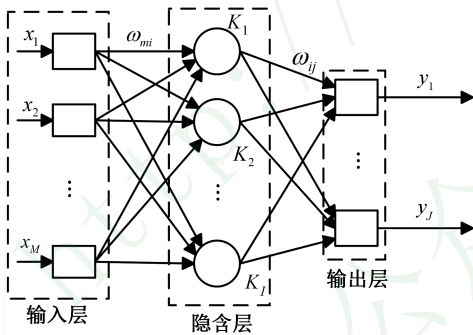


图 3 三层 BP 神经网络

Fig.3 Three-layer BP neural network

图 3 中，输入层有 M 个神经元，输入向量 $X = (x_1, x_2, \dots, x_M)^T$ ；隐含层有 I 个神经元；输出层有 J 个神经元，输出向量 $Y = (y_1, y_2, \dots, y_J)^T$ ； ω_{mi} 是输入层与隐含层之间的权值， ω_{ij} 是隐含层与输出层之间的权值，隐含层各神经元阈值为 a_i ($i=1, 2, \dots, I$)，输出层各神经元阈值为 b_j ($j=1, 2, \dots, J$)。

根据图 3 可以得到网络的输出：

$$y_j = \sum_{i=1}^I \omega_{ij} f(\sum_{m=1}^M \omega_{mi} x_m + a_i) + b_j \tag{4}$$
$$j = 1, 2, \dots, J$$

式中， $f(\cdot)$ 为隐含层的激励函数。

2.2 MATLAB 神经网络工具箱

在 MATLAB 中可通过以下步骤训练 BP 神经网络^[14]。

(1) 创建前向神经网络：

$$net = \text{feedforwardnet}(\text{hiddenSizes}, \text{trainFcn}) \tag{5}$$

其中， hiddenSizes 表示隐含层的规模，列数表示隐含层数，每列的数值表示各隐含层神经元个数； trainFcn 是训练权值的算法，默认为“trainlm”，即 LM 算法 (Levenberg-Marquadt 算法)。LM 算法将梯度下降法与高斯-牛顿法相结合，既有高斯-牛顿法的局部收敛性，又具有梯度下降法的全局特性^[15]。

(2) 参数设定。 $\text{net.trainParam.epochs}$ 为迭代最大轮数； $\text{net.trainParam.max_fail}$ 为训练最大的失败次数，默认为 6； $\text{net.divideParam.trainRatio}$ 为训练样本占总样本百分比，默认 75%； $\text{net.divideParam.valRatio}$ 为校验样本占总样本百分比，默认 15%； $\text{net.divideParam.testRatio}$ 为测试样本占总样本百分比，默认 15%。 $\text{net.trainParam.goal}$ 为目标误差。

(3) 训练 BP 神经网络：

$$[net, TR] = \text{train}(net, P_{\text{train}}, T_{\text{train}}) \tag{6}$$

式中， P_{train} 为训练样本的输入矩阵； T_{train} 为训练样本的输出矩阵； TR 为记录训练的结果。

(4) 预测数据：

$$Y = \text{net}(T_{\text{test}}) \tag{7}$$

式中， Y 为网络预测结果； T_{test} 为需要预测的输入。

2.3 样本选取

采用以下方式得到训练样本，在 Zone _{i} ($i=1, 2, 3, 4$) 中，先将 θ_j ($j=1, 2, 3$) 平均分成 $M_\theta, I_\theta, H_\theta$ ，然后将 θ_j 进行组合，如图 4 所示。

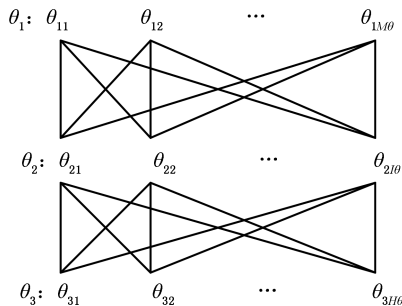


图 4 关节角的组合

Fig.4 Combination of joint angle

Zone _{i} 中的关节角组合均为 $M_\theta \times I_\theta \times H_\theta$ 组，记为 Q_i ($i=1, 2, 3, 4$)。

2.4 网络的输入输出

将 Q_i 代入式(1)得到末端执行器的位置,并将结果代入式(2)、式(3),最终得到神经网络的训练样本,输入 $T_i = [R_H \ P^T]$, $i=1,2,3,4$, R_H 是矩阵 R 中每一列拼接在一起组成的行向量, θ_{ij} ($j=1,2,3$) 作为训练样本的输出,即每个网络都是 12 个输入神经元, 1 个输出神经元。

LM 算法训练权值时,在每轮计算中均会存储一个 $M \times P \times N$ (M 为输出神经元个数, P 为样本个数, N 为权值与阈值的总个数)的雅可比矩阵,雅可比矩阵的大小直接影响网络的训练速度。每个网络只用一个关节角作为输出与多个关节角作为输出相比,可以大大地提高训练速度,减小各网络的计算开支,加之 LM 算法具有收敛快、误差小的特点,从而使得每个网络可以在更短的训练时间内收敛。

2.5 预测效果评定

用 θ_{pij} ($i=1,2,3;j=1,2,\cdots,N$) 表示预测得到的关节角, θ_{Tij} ($i=1,2,3;j=1,2,\cdots,N$) 表示实际的关节角, (x_{Tj}, y_{Tj}, z_{Tj}) 表示实际位置, (x_{pj}, y_{pj}, z_{pj}) 表示预测位置,其中 i 是关节角序号, N 是测试样本数量。评价指标公式如下:

$$e_j = \theta_{Tij} - \theta_{pij} \tag{8}$$
$$E_j = \sqrt{(x_{Tj} - x_{pj})^2 + (y_{Tj} - y_{pj})^2 + (z_{Tj} - z_{pj})^2} \tag{9}$$

式中, e_j 为预测的关节角与实际关节角间的误差; E_j 为实际笛卡儿坐标与预测笛卡儿坐标之间的绝对误差。

3 输入点区域识别及逆运动学求解

3.1 输入点的识别方法

由于在整个工作空间中,部分区域将不存在四组逆运动学解,在进行预测前,需要确定如何选用训练好的 BP 神经网络。本文使用投影识别方法确定输入点所在的工作子空间。

(1)工作子空间轮廓的投影方法。在 OXY 平面中,边界条件为 θ_1 的取值范围;在 OXZ 平面中,投影轮廓根据表 1 中各关节子空间取值范围代入式(1)得到,其中 $Zone_1$ 和 $Zone_2$ 的 θ_1 都取 0° , $Zone_3$ 和 $Zone_4$ 的 θ_1 都取 π (或 $-\pi$)。

(2)识别点的投影方法。先垂直投影至 OXY 平面,投影点坐标为 (p_x, p_y) ,投影角度为 $\theta = \arctan2(p_y, p_x)$;再绕 Z 轴顺时针旋转 θ 投影至 OXZ 平面,若 $p_y > 0$,投影后的坐标为 $(\sqrt{p_x^2 + p_y^2}, p_z)$;若 $p_y < 0$,投影后的坐标为 $(-\sqrt{p_x^2 + p_y^2}, p_z)$;若 $p_y = 0$,投影后的坐标为 (p_x, p_z) 。

(3)识别点的判据。若 θ 在 $Zone_i$ 的 θ_1 范围内,且在 OXZ 的投影坐标也属于 $Zone_i$ 的轮廓范围,则输入点属于 $Zone_i$,其中有一条不符合则不属于 $Zone_i$ 。

(4)输入点识别步骤如下:①根据 $L_2 - L_3 \leq \sqrt{p_x^2 + p_y^2 + p_z^2} \leq L_2 + L_3$ 筛选出在工作空间极限作业半径内的输入点;②在步骤①得到的点中,筛选出投影点符合在 OXY 平面内轮廓的输入点;③在步骤②得到的点中,筛选出投影点符合在 OXZ 平面内轮廓的输入点;④判断输入点所属区域。

3.2 逆运动学求解

对每个工作子空间都建立一个区域识别器 Recognition Unit,简称为 RU_i ($i=1,2,3,4$), RU_i 与 net_i 串联且每个 net_i 中有 3 个子神经网络,分别为 net_{ij} ($i=1,2,3,4;j=1,2,3$)。当位置输入时,先通过每个识别器进行识别,符合其条件的,添加旋转 4 矩阵,再输入到与之对应的 BP 神经网络中求逆运动学解。

$Zone_1$ 和 $Zone_2$ 中, $p_y > 0$ 的部分仍然用式(2)求 θ_{1i} , $p_y \leq 0$ 的部分用式(10)求 θ_{1i} ; $Zone_3$ 和 $Zone_4$ 中, $p_y \leq 0$ 的部分仍然用式(2)求 θ_{1i} , $p_y > 0$ 的部分用式(11)求 θ_{1i} 。位于 Z 轴上的点,以实际需要给予一个 θ_{1i} :

$$\theta_{1i} = \pi + \arctan2(p_y, p_x) \tag{10}$$

或

$$\theta_{1i} = \arctan2(p_y, p_x) - \pi \tag{11}$$

图 5 为逆运动学求解流程图,图中的 net_i 均表示已训练好的 BP 神经网络。 net_i 若有输出,表示该点在相应的关节子空间有逆运动学解, net_i 没有输出,则表示该点在相应的关节子空间中不存在对应的逆运动学解。输出 Q_i 的个数,表示存在逆向运动学解的组数。

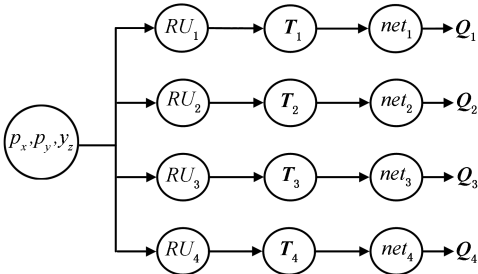


图 5 逆运动学求解流程

Fig.5 Inverse kinematics solution flow

4 算例

4.1 BP 神经网络的训练及结果分析

取 2.3 节的 $M_\theta=18, I_\theta=16, H_\theta=14, Zone_i$ 均

得到 4 032 组样本,使用 MATLAB 神经网络工具箱创建 BP 神经网络,篇幅所限,仅列出 Zone₁中 net_{1j} (1,2,3)的训练和求解结果进行分析。

每个神经网络的参数如下:net₁₁中,hidden-Sizes = [8, 8, 8]; net. trainParam. epochs = 6 000; net₁₂ 中,hiddenSizes = [13, 12, 10], net. trainParam.epochs = 6 000; net₁₃ 中,hiddenSizes = [13,12,11],net.trainParam.epochs = 6 000。

每个 BP 神经网络中的样本均分为训练样本、校验样本、测试样本,各样本所占比例为 2.2 节中的默认值, net. trainParam. goal 均设为 10⁻¹⁰,其余参数均设为默认值。

图 6~图 8 所示分别为 Zone₁中 BP 神经网络的训练过程、测试样本中各关节角预测误差、测试样本中预测位置与实际位置间的绝对误差。

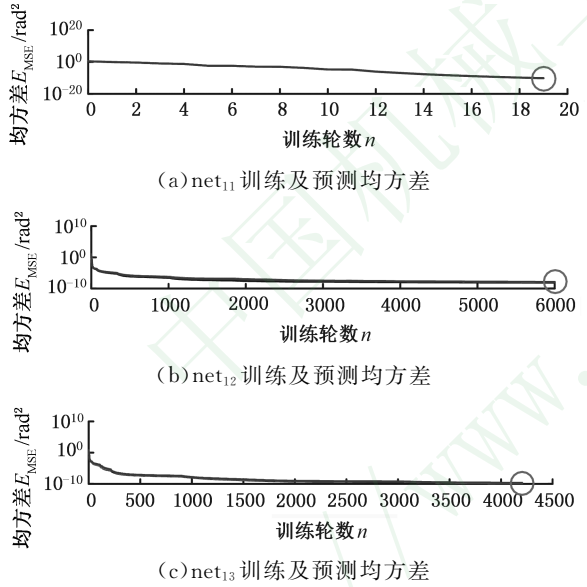


图 6 各 BP 神经网络的训练过程

Fig.6 Training process of each BP neural network

在每个区域中虽然使用了 3 个 BP 神经网络训练,但是各区域内网络依次训练的时间之和均在 30 min 内,与文献[11]相比,网络训练的轮数或时间均大大减少。文献[11]中关节角预测误差分别为±0.06°,±0.15°,±0.05°,而本文的关节角误差分别为±0.002°,±0.05°,±0.02°,位置误差在 0.04 mm 内。通过对比,显然本文所用方法具有更快的收敛速度和更高的预测精度。

4.2 预测点的识别及其逆运动学求解结果

在工作空间中选 3 个点(表 3)按图 5 的流程求逆运动学解,最终结果显示,P₁有四组逆运动学解,P₂在 Zone₁和 Zone₄中有逆运动学解,P₃没有逆运动学解,说明 P₃不在工作空间内。表 4 为 P₁、P₂分别用训练好的 BP 神经网络求得的逆运动学解和预测位置。

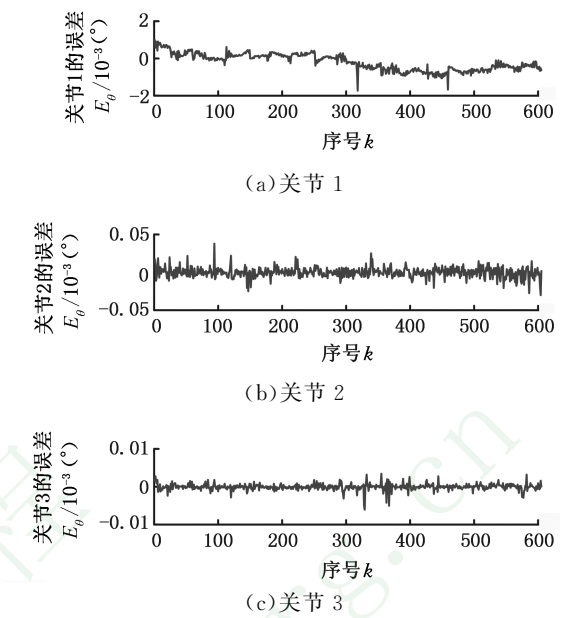


图 7 各关节角预测误差

Fig.7 Error of each joint angle prediction

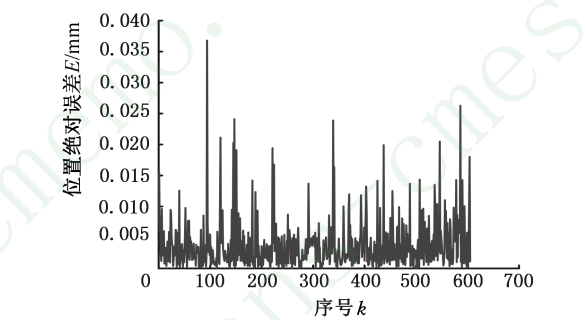


图 8 位置绝对误差

Fig.8 Absolute error of position

表 3 预测点

Tab.3 Prediction points

P_i	p_x (mm)	p_y (mm)	p_z (mm)
P_1	300	200	1 700
P_2	-834	722	600
P_3	-100	-100	100

表 4 机械手两个工作位置的逆运动学多解

Tab.4 Inverse kinematics multiple solutions for two working positions of a robot

P_i	Zone _i	预测角度(°)	预测位置(mm)
P_1	Zone ₁	(33.70,109.80,-86.64)	(300.04,200.03,1 700.00)
	Zone ₂	(33.69,27.27,86.64)	(299.95,199.96,1 700.02)
	Zone ₃	(-146.31,152.73,-86.64)	(299.97,199.99,1 700.01)
	Zone ₄	(-146.31,70.20,86.64)	(299.99,200.00,1 700.00)
P_2	Zone ₁	(139.12,23.42,-68.65)	(-834.02,722.02,600.16)
	Zone ₄	(-40.88,156.58,68.65)	(-833.99,-722.00,599.97)

5 结论

(1)本文提出了一种基于几何分析的多模块

神经网络求解空间 3R 机械手逆运动学多解的方法。

(2)进行了几何分析,将关节空间划分为4个只有唯一逆运动学解的子空间,对各子空间用3个BP神经网络进行训练和求解。仿真试验结果表明,该方法具有训练时间短、预测精度高的优点。

(3)对于已训练好的BP神经网络,可以通过输入点区域识别的方法进行选择。

参考文献:

- [1] 熊超,张鹏超,冯博琳,等.基于并行RBF神经网络的机器人逆运动求解[J].组合机床与自动化加工技术,2017(10):34-36.
XIONG Chao, ZHANG Pengchao, FENG Bolin, et al. Solution to Inverse Kinematics of Robot Based on Parallel RBF Neural Networks[J]. Combined Machine Tool and Automatic Machining, 2017(10):34-36.
- [2] 姜宏超,刘士荣,张波涛.六自由度模块化机械臂的逆运动学分析[J].浙江大学学报(工学版),2010,44(7):1348-1354.
JIANG Hongchao, LIU Shirong, ZHANG Botao. Inverse Kinematics Analysis for 6 Degree-of-freedom Modular Manipulator[J]. Journal of Zhejiang University (Engineering Edition), 2010, 44(7): 1348-1354.
- [3] 王英石.冗余机器人的运动学及轨迹规划的研究[D].天津:南开大学,2014.
WANG Yingshi. Research on the Kinematics and Trajectory Planning of Redundant Robots [D]. Tianjin: Nankai University, 2014.
- [4] 房立金,高瑞.一般6R机器人逆运动学算法的改进[J].机械科学与技术,2018(9):1325-1330.
FANG Lijin, GAO Rui. Modified Inverse Kinematics Algorithm for General 6-DOF Robots[J]. Mechanical Science and Technology, 2018(9):1325-1330.
- [5] PEIPER D L. The Kinematics of Manipulators under Computer Control[D]. California: Stanford University, 1968.
- [6] DUFFY J. Analysis of Mechanisms and Robot Manipulators[M]. London: Edward Arnold, 1980: 369-409.
- [7] 王俊龙,张国良,敬斌,等.一种新的六自由度机械臂运动学反解方法研究[J].计算机工程与应用,2013,49(22):266-270.
WANG Junlong, ZHANG Guoliang, JING Bin, et al. Research on Way of Inverse Kinematic Solution of Six DOF Manipulator[J]. Computer Engineering and Applications, 2013, 49(22):266-270.

- [8] 林阳,赵欢,丁汉.基于多种群遗传算法的一般机器人逆运动学求解[J].机械工程学报,2017,53(3):1-8.
LIN Yang, ZHAO Huan, DING Han. Solution of Inverse Kinematics for General Robot Manipulators Based on Multiple Population Genetic Algorithm[J]. Journal of Mechanical Engineering, 2017, 53(3):1-8.
- [9] DAYA B, KHAWANDI S, AKOUM M. Applying Neural Network Architecture for Inverse Kinematics Problem in Robotics[J]. Journal of Software Engineering and Applications, 2010, 3(3): 230-239.
- [10] DUKA A V. Neural Network Based Inverse Kinematics Solution for Trajectory Tracking of a Robotic Arm[J]. Procedia Technology, 2014, 12: 20-27.
- [11] RAHEEM F A, KAREEM A R, HUMAIDI A J. Inverse Kinematics Solution of Robot Manipulator End-effector Position Using Multi-neural Networks [J]. Engineering and Technology Journal, 2016, 34: 1360-1368.
- [12] 宋孟军,张明路,张建华.一种新型移动机器人运动学坐标系快速构建方法研究[J].中国机械工程,2011,22(17):2017-2021.
SONG Mengjun, ZHANG Minglu, ZHANG Jianhua. Research on Rapid Construction Method of Kinematics Coordinate System from a New Kind of Mobile Robot[J]. China Mechanical Engineering, 2011, 22(17): 2017-2021.
- [13] 张尧.激活函数导向的RNN算法优化[D].杭州:浙江大学,2017.
ZHANG Yao. Activation Function Awareness of RNN Algorithm Optimization[D]. Hangzhou: Zhejiang University, 2017.
- [14] 陈明. MATLAB神经网络原理与实例精解[M].北京:清华大学出版社,2013:10-173.
CHEN Ming. Theory and Example of Neural Network on MATLAB[M]. Beijing: Tsinghua University Press, 2013:10-173.
- [15] 张峤,邓贵仕. Levenberg-Marquardt神经网络在煤矿作业人员人因可靠性评价中应用研究[J].大连理工大学学报,2015,55(4):424-430.
ZHANG Qiao, DENG Guishi. Investigation on Application of Levenberg-Marquardt Neural Networks to Human Reliability Evaluation of Coalmine Workers[J]. Journal of Dalian University of Technology, 2015, 55(4):424-430.

(编辑 王旻玥)

作者简介:肖帆,男,1988年生,硕士研究生。研究方向为机器人智能控制。李光(通信作者),男,1963年生,教授。E-mail: liguangw@126.com。