

基于组稀疏卡尔曼滤波的多步轨迹预测方法

王娜^{1,2,3}, 罗亮¹, 彭锐¹, 张鑫海¹

(1. 天津工业大学控制科学与工程学院, 天津, 300387; 2. 天津市电气装备智能控制重点实验室, 天津, 300387;
3. 微光机电系统技术教育部重点实验室(天津大学), 天津, 300072)

摘要 提出一种基于组稀疏卡尔曼滤波的机动轨迹多步预测方法。首先引入组稀疏编码, 通过一次学习建立简单的多步线性回归预测模型, 克服了传统方法未能充分利用历史数据而导致预测精度降低的问题; 再利用最小角回归算法来计算该模型的稀疏系数, 进一步改善模型系数估计的准确性; 然后改进了卡尔曼滤波算法, 并结合上述组稀疏编码算法, 来确保预测结果的精确性; 最后通过与传统BP、长短时记忆网络和组稀疏编码方法的仿真比较, 验证了所提方法的有效性。

关键词 多步轨迹预测; 组稀疏编码; 卡尔曼滤波; 最小角回归

DOI 10.3969/j.issn.2097-1915.2023.06.010

中图分类号 TN953 **文献标志码** A **文章编号** 2097-1915(2023)06-0070-08

A Multi-Step Trajectory Prediction Method Based on Group Sparse Kalman Filtering

WANG Na^{1,2,3}, LUO Liang¹, PENG Kun¹, ZHANG Xin-hai¹

(1. School of Control Science and Engineering, Tiangong University, Tianjin 300387, China;
2. Tianjin Key Laboratory of Intelligent Control of Electrical Equipment, Tianjin 300387, China;
3. Key Laboratory of Micro Optical Electronic Mechanical System Technology of Ministry of Education, Tianjin University, Tianjin 300072, China)

Abstract A multi-step trajectory prediction method based on group sparse coding Kalman filtering for mobile target is proposed in this paper. Firstly, a group sparse coding is introduced, and just at that time, a simple multi-step linear prediction model is obtained by one learning, overcoming the problem that prediction accuracy is low due to the inadequate historical data with the traditional method. And then, the minimum angle regression algorithm is utilized for calculating the sparse coefficients of the above model to further improve the estimation accuracy of the model coefficients. The basic Kalman filtering algorithm is modified in combination with the group sparse coding method to ensure the precision in the prediction output. Finally, the effectiveness of the presented approach is verified by the simulation comparison among the traditional BP network, long short time memory network and the group sparse coding method.

Key words multi-step trajectory prediction; group sparse coding; Kalman filtering; least angle regression

收稿日期: 2023-03-23

基金项目: 天津市重点研发计划项目(19YFHBQY00040); 微光机电系统技术教育部重点实验室(天津大学)开放基金(MOMST2016-4)

作者简介: 王娜(1977-), 女, 河北衡水人, 讲师, 博士, 研究方向为数据驱动建模、智能故障诊断、目标预测与跟踪、多目标优化与控制等。

E-mail: wangna@tiangong.edu.cn

引用格式: 王娜, 罗亮, 彭锐, 等. 基于组稀疏卡尔曼滤波的多步轨迹预测方法[J]. 空军工程大学学报, 2023, 24(6): 70-77. WANG Na, LUO Liang, PENG Kun, et al. A Multi-Step Trajectory Prediction Method Based on Group Sparse Kalman Filtering[J]. Journal of Air Force Engineering University, 2023, 24(6): 70-77.

多步轨迹预测即利用运动目标的历史轨迹信息构建预测模型,并输出该模型对运动目标未来多个时刻的轨迹估计值。目前的研究以模型法和深度学习为主,前者一般利用卡尔曼滤波及其改进为基础来构建预测模型。如文献[1]提出的基于传统卡尔曼滤波(Kalman filtering, KF)的飞行轨迹预测方法。相比传统卡尔曼滤波,扩展卡尔曼滤波能够有效处理非线性模型的估计,如文献[2]基于扩展卡尔曼滤波(extended Kalman filtering, EKF)实现对不确定情况下的无人驾驶车辆的轨迹预测;文献[3]基于扩展卡尔曼滤波方法和差分自回归滑动平均模型对上海中心大厦实测加速度幅值数据的实时变化趋势进行预测。相比前两类滤波方法,文献[4]所提的多项式卡尔曼滤波方法能够有效针对信息缺失、非线性、多机动情况,过程较为简单,计算效率高,机动自适应较强,但通常易受模型误差和数据噪声影响,对于复杂的运动目标轨迹预测精度不高。上述卡尔曼滤波方法主要基于单模型法,所构建的运动模型存在数据处理难度大及预测精度低等问题,对此文献[5]提出改进的交互式多模型轨迹预测算法。此外,上述卡尔曼滤波类方法由于只能输出一步预测的估计值,故难以单独应用于多步预测中。

与传统模型法基于参数构建的思想不同,神经网络基于数据驱动实现自主学习,对于数据特性考虑较为全面,可以实现复杂非线性时间序列的准确、多步预测。浅层学习神经网络中,文献[6]在BP神经网络基础上,引入双三角函数变换思想,实现对船舶轨迹的精确预测;文献[7]利用改进的狼群算法优化径向基神经网络学习过程,实现对无人艇动态障碍物的航行轨迹预测。深度学习不依赖运动目标的机理特性,数据容易获得,且可利用历史数据进行中长期轨迹预测,预测时间长且精度较高,故得到广泛应用。如文献[8]利用长短时记忆网络(long short-term memory, LSTM)与一维卷积神经网络(one-dimensional convolutional neural network, 1DCNN)结合,实现对导弹轨迹的预测。文献[9]利用门控循环单元网络(gated recurrent unit, GRU)进行飞机轨迹预测,与传统的循环神经网络(recurrent neural network, RNN)和反向传播网络(back propagation, BP)相比,预测性能更优。文献[10]引入双向传播机制和Mogriifier数据耦合模块,改进传统GRU网络,使其预测性能进一步提高。综上所述,神经网络类方法能够较充分继承历史信息,更适合处理时间序列的预测,但其内部参数较多,训练耗时,且随着时间序列的数量级增加,容易出现梯度消失和梯度爆炸问题。

此外,神经网络自身的结构和参数的确定受人为因素影响而缺乏客观依据,因而导致预测误差偏大,泛化能力较差。相比神经网络,稀疏编码(sparse coding, SC)能以少量的输入特征矩阵与稀疏向量的加权和形式对原始输出数据矩阵进行重构,无需人为调节参数,计算过程简单,效率较高^[11]。但稀疏编码直接用于多步预测时,求解稀疏向量时计算量较大,为此,组稀疏编码(group sparse coding, GSC)把输入和输出的系数矩阵以对角矩阵形式表示,仅通过一次计算即可输出多个稀疏向量,有效降低了计算量,简化了多步预测的计算过程^[12]。目前稀疏编码类方法通常用于结构健康检测和图像处理等领域,而在轨迹预测领域的应用较少,如文献[13]针对Lamb波边界反射引起的码间串扰问题,采用移不变稀疏编码方法进行信息恢复,提出了基于Lamb波的数据传输与缺陷检测同步实现方法;文献[14]针对在小样本人脸表情数据库上识别模型过拟合问题,提出基于特征优选和字典优化的组稀疏表示分类方法。

综上所述,本文将组稀疏编码^[15]与卡尔曼滤波结合,提出了一种基于组稀疏卡尔曼滤波的多步轨迹预测方法(multi-step trajectory prediction method via group sparse Kalman filtering, MSTPM-GSKF)。

1 超前 T 步轨迹预测问题描述

为方便起见,在单维坐标系下,给定轨迹序列 $\mathbf{X}_g = \{x_i\}$, 利用其建立预测模型, $i = 1, 2, \dots, h$, h 为轨迹序列的长度; x_i 为第 i 个轨迹数据。超前 T 步轨迹预测即利用前 m 个轨迹数据 $\mathbf{X}_i = (x_i, x_{i-1}, \dots, x_{i-(m-1)})$, 对该轨迹未来第 $i+1$ 到第 $i+T$ 时刻的轨迹 $\hat{\mathbf{X}}_{i+T}$ 进行预测:

$$\hat{\mathbf{X}}_{i+T} = f(x_i, x_{i-1}, \dots, x_{i-(m-1)}) \quad (1)$$

式中: $\hat{\mathbf{X}}_{i+T} = (\hat{x}_{i+1}, \hat{x}_{i+2}, \dots, \hat{x}_{i+T})$, \hat{x}_{i+t} 为第 $i+t$ 个预测数据; t 为预测时间间隔, $t = 1, 2, \dots, T$, T 为预测步长。

2 GSC 的基本原理

2.1 SC 基本原理

给定轨迹序列 \mathbf{X}_g , 稀疏编码是以稀疏向量 $\boldsymbol{\alpha}$ 和输入特征矩阵 \mathbf{D} 乘积的线性组合表示对应的输出特征矩阵 \mathbf{X} ^[16], 即:

$$\mathbf{X} = \mathbf{D}\boldsymbol{\alpha} = \sum_{i=1}^M \mathbf{d}_i \alpha_i \quad (2)$$

式中: $\mathbf{X}=[x_1, x_2, \dots, x_p]^T$, x_i 为 \mathbf{X} 中第 i 个轨迹点, $i=1, 2, \dots, p$, p 为轨迹点个数; $\mathbf{D}=[\mathbf{d}_1, \mathbf{d}_2, \dots, \mathbf{d}_M]$, $\mathbf{d}_i=[x_{i1}, x_{i2}, \dots, x_{ip}]^T$ 为 \mathbf{D} 中的第 i 个列向量, $i=1, 2, \dots, M$, M 为 \mathbf{D} 中列向量集合的个数, x_{ij} 为 \mathbf{d}_i 中第 j 个轨迹点, $j=1, 2, \dots, p$; $\boldsymbol{\alpha}=[\alpha_1, \alpha_2, \dots, \alpha_M]^T$ 为稀疏向量, α_i 为 $\boldsymbol{\alpha}$ 中第 i 个稀疏系数, $i=1, 2, \dots, M$ 。

稀疏编码中, 给定输入特征矩阵 \mathbf{D} , 稀疏向量 $\boldsymbol{\alpha}$ 直接影响着输出特征矩阵 \mathbf{X} 的精确性, 因此对它的求解尤为重要, 一般通过松弛算法转化为优化问题求解^[17]:

$$\boldsymbol{\alpha} = \arg_{\boldsymbol{\alpha}} \min \frac{1}{2} \|\mathbf{X} - \mathbf{D}\boldsymbol{\alpha}\|_2^2 + \lambda \|\boldsymbol{\alpha}\|_1 \quad (3)$$

式中: λ 为惩罚参数。

对于 $\boldsymbol{\alpha}$ 的求解, 传统最小二乘 (least square, LS) 算法过程较简单, 但 LS 对于输入多重共线性及输入变量数大于数据个数等特殊情况下, 难以获得稳定解^[18], 且求解效率较低。对此, 最小角回归算法 (least angle regression, LAR) 在误差平方和函数中加入 1 范数, 可以递阶地获得重要输入变量, 提高特征选择的准确性, 有效改善上述问题, 故本文采用 LAR 方法求解^[15]。

LAR 算法的基本思想如下: 首先根据余弦相似度函数计算每个输入特征向量与输出特征向量的相关性, 根据最大相关性值获得对应的输入特征向量, 然后计算所有剩余的输入特征向量与当前残差向量的相关系数, 从中筛选出绝对值最大的相关系数, 选择其对应的输入特征向量, 并将其与第一个获得的输入特征向量组合, 获取其角平分线, 并在该角平分线方向继续选择输入, 直至所有输入被选择完毕, 其原理如下:

图 1 中 \mathbf{d}_1^b 和 \mathbf{d}_2^b 为标准化后的列向量, $\hat{\mathbf{X}}_z$ 为零均值化后的输出特征向量, $\hat{\mathbf{X}}_0=[0, \dots, 0]_{1 \times p}^T$ 为初始估计, e_1 为第 1 次迭代时的残差, \mathbf{h}_1 和 \mathbf{h}_2 为第 1 次和第 2 次迭代时搜索方向。

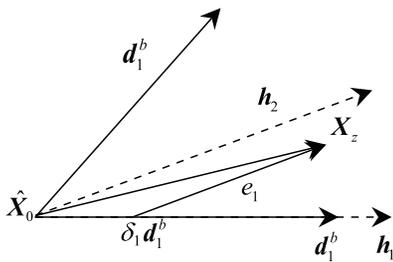


图 1 最小角回归原理

由图 1 所示, 首先在所有列向量中找到与残差 $e_0=\mathbf{X}_z$ 相关性最大的列向量, 记为 \mathbf{d}_1^b , 然后以步长 δ_1 逼近 \mathbf{X}_z , 同时计算残差 e_1 , 直到出现第 2 个列向

量 \mathbf{d}_2^b , 使得 \mathbf{d}_2^b 和 \mathbf{d}_1^b 与 e_1 的相关性相同, 继续以 \mathbf{d}_2^b 和 \mathbf{d}_1^b 的角平分线 \mathbf{h}_2 为新的搜索方向对 \mathbf{X}_z 逼近, 直到出现第 3 个列向量, 使得此时 3 个列向量与残差的相关性相同, 再次以此 3 个列向量的角平分线为新的搜索方向逼近 \mathbf{X}_z , 以此类推, 直到所有列向量都被选中。其具体步骤如下:

Step 1 对于给定的 $\mathbf{D}=[\mathbf{d}_1, \mathbf{d}_2, \dots, \mathbf{d}_M]$, $\mathbf{d}_i=[x_{i1}, x_{i2}, \dots, x_{ip}]^T$, $i=1, 2, \dots, M$, 将 \mathbf{D} 零均值化并转化为单位向量, 获得 $\mathbf{D}_b=[\mathbf{d}_1^b, \mathbf{d}_2^b, \dots, \mathbf{d}_M^b]$, 其中

$$\mathbf{d}_i^b = [x_{i1}^b, x_{i2}^b, \dots, x_{ip}^b]^T, x_{ij}^b = \frac{x_{ij} - \bar{d}_i^b}{\sqrt{\sum_{j=1}^p (x_{ij} - \bar{d}_i^b)^2}}, \bar{d}_i^b =$$

$\sum_{j=1}^p x_{ij} / p$, $i=1, 2, \dots, M$; 同时将 \mathbf{X} 零均值化, 记为 $\mathbf{X}^0=[x_1^0, x_2^0, \dots, x_p^0]^T$, 即对于第 i 个数据 x_i , 其零均值化为 $x_i^0 = x_i - \bar{\mathbf{X}}$, $\bar{\mathbf{X}} = \sum_{i=1}^p x_i / p$ 。

Step 2 初始化 $\boldsymbol{\alpha}$ 和估计结果 $\hat{\mathbf{X}}$, 即 $\boldsymbol{\alpha}_0=[0, \dots, 0]_{1 \times M}^T$, $\hat{\mathbf{X}}_0=[0, \dots, 0]_{1 \times p}^T$; 初始化活动矩阵 \mathbf{I}_c 和非活动矩阵 \mathbf{I}_q , 即 $\mathbf{I}_c = \boldsymbol{\Phi}$, $\mathbf{I}_q = \mathbf{D}_b$, 其中 \mathbf{I}_c 由 \mathbf{D}_b 中已被选择的列向量组成, 记 $\mathbf{I}_c = [\mathbf{d}_1^b, \mathbf{d}_2^b, \dots, \mathbf{d}_k^b]$, 其中 \mathbf{d}_i^b 为 \mathbf{I}_c 中的第 i 个列向量, $i=1, 2, \dots, k$, k 为迭代次数, $k=1, 2, \dots, M$; \mathbf{I}_q 由未被选择的列向量组成, 记 $\mathbf{I}_q = [\mathbf{d}_1^b, \mathbf{d}_2^b, \dots, \mathbf{d}_q^b]$, \mathbf{d}_j^b 为 \mathbf{I}_q 中的第 j 个列向量, $j=1, 2, \dots, q$, q 为 \mathbf{I}_q 中列向量个数。

Step 3 迭代开始, 令 $k=1$, 更新残差 $\mathbf{e}_{k-1} = \mathbf{X}^0 - \hat{\mathbf{X}}_{k-1}$, 计算 \mathbf{e}_{k-1} 与 \mathbf{I}_q 中各列向量的相关系数 $\mathbf{C}_k = \mathbf{e}_{k-1}^T \mathbf{I}_q = [c_1^k, c_2^k, \dots, c_q^k]$ 。

Step 4 计算最大相关系数 $c_{\max}^k = \max\{|c_j^k|\}$, $j=1, 2, \dots, q$, 将对应的列向量 \mathbf{d}_j^b 从 \mathbf{I}_q 中加入 \mathbf{I}_c 。

Step 5 根据函数 $s_i^k = \begin{cases} 1, & c_i^k \geq 0 \\ -1, & c_i^k < 0 \end{cases}$, $i=1, 2, \dots, k$, 结合 \mathbf{I}_c 构建系数矩阵 $\mathbf{G}_k = [s_1^k \mathbf{d}_1^b, s_2^k \mathbf{d}_2^b, \dots, s_k^k \mathbf{d}_k^b]$, 计算搜索系数 $\boldsymbol{\mu}_k = \boldsymbol{\eta}_k (\mathbf{G}_k^T \mathbf{G}_k)^{-1} \mathbf{L} = [\mu_1^k, \mu_2^k, \dots, \mu_k^k]^T$ 和 $\boldsymbol{\eta}_k = (\mathbf{L}^T (\mathbf{G}_k^T \mathbf{G}_k)^{-1} \mathbf{L})^{-1/2}$, 其中 $\mathbf{L} = [1, \dots, 1]^T$, μ_i^k 为 $\boldsymbol{\mu}_k$ 中第 i 个系数, $i=1, 2, \dots, k$ 。

Step 6 确定搜索方向 $\mathbf{h}_k = \mathbf{G}_k \boldsymbol{\mu}_k$, 计算 \mathbf{h}_k 与 \mathbf{I}_q 中各列向量的相关系数 $\boldsymbol{\varphi}_k = \mathbf{h}_k^T \mathbf{I}_q = [\varphi_1^k, \varphi_2^k, \dots, \varphi_q^k]$ 。

Step 7 计算搜索步长

$$\delta_k = \min^+ \left\{ \frac{c_{\max}^k - c_j^k}{\eta_k - \varphi_j^k}, \frac{c_{\max}^k + c_j^k}{\eta_k + \varphi_j^k} \right\}, j=1, 2, \dots, q, \min^+$$

表示取正数部分的最小值。

Step 8 更新估计结果 $\hat{\mathbf{X}}_k = \hat{\mathbf{X}}_{k-1} + \delta_k \mathbf{h}_k$ 和稀疏

系数 $\alpha_i^k = \alpha_i^{k-1} + \delta_k s_i \mu_i^k$, α_i^k 为 α_k 中的第 i 个系数。

Step 9 令 $k = k + 1$, 返回 Step 3, 直到 $\mathbf{I}_q = \Phi$, 迭代结束, 获得稀疏向量集合 $\alpha_k = [\alpha_1^k, \alpha_2^k, \dots, \alpha_M^k]^T$ 。

2.2 GSC 基本原理

组稀疏编码是将多个输入特征矩阵和输出特征矩阵排列成对角矩阵形式的线性组合, 从而对输出特征矩阵进行稀疏表示的过程^[12]:

$$\begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \\ \vdots \\ \mathbf{X}_J \end{bmatrix} = \begin{bmatrix} \mathbf{D}_1 & & & \\ & \mathbf{D}_2 & & \\ & & \ddots & \\ & & & \mathbf{D}_J \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_J \end{bmatrix} \quad (4)$$

式中: \mathbf{D}_i 为第 i 个输入特征矩阵; \mathbf{X}_i 为 \mathbf{D}_i 对应的输出特征矩阵; α_i 为对应的第 i 个稀疏向量, $i = 1, 2, \dots, J$, J 为输入或输出特征矩阵的个数。

3 IKF 基本原理

传统卡尔曼滤波算法利用当前残差 $\mathbf{V}(k)$ 对预测值 $\hat{\mathbf{X}}(k/k-1)$ 进行动态修正, 得到当前状态的估计值 $\hat{\mathbf{X}}(k/k)$ ^[19]。但本文需要利用估计过程得到修正后的预测值, 而对于预测时刻 k 而言, 由于无法提前获得观测值 $\mathbf{Z}(k)$, 则难以继续进行迭代计算, 因此, 本文利用前一时刻的残差 $\mathbf{V}(k-1)$ 对 $\hat{\mathbf{X}}(k/k-1)$ 进行修正, 以减小缺失观测值 $\mathbf{Z}(k)$ 带来的影响, 形成 IKF 算法, 该改进过程的具体描述如下: 在基本 KF 中, 首先建立状态方程和观测方程, 进行状态的一步预测:

$$\hat{\mathbf{X}}(k/k-1) = \mathbf{F}\mathbf{X}(k-1/k-1) \quad (5)$$

式中: \mathbf{F} 为状态转移矩阵; k 为迭代次数。

一步预测协方差阵 $\mathbf{P}(k/k-1)$ 为:

$$\mathbf{P}(k/k-1) = \mathbf{F}\mathbf{P}(k/k-1)\mathbf{F}^T + \mathbf{Q} \quad (6)$$

\mathbf{Q} 为过程协方差阵, 则卡尔曼增益矩阵 $\mathbf{K}(k)$ 为:

$$\mathbf{K}(k) = \hat{\mathbf{P}}(k/k-1)\mathbf{H}^T\mathbf{S}(k)^{-1} \quad (7)$$

其中 \mathbf{H} 为观测矩阵, $\mathbf{S}(k)$ 为:

$$\mathbf{S}(k) = \mathbf{H}\hat{\mathbf{P}}(k/k-1)\mathbf{H}^T + \mathbf{R} \quad (8)$$

最后协方差阵 $\mathbf{P}(k/k)$ 更新为:

$$\mathbf{P}(k/k) = [\mathbf{I}_n - \mathbf{K}(k+1)\mathbf{H}]\mathbf{P}(k/k-1) \quad (9)$$

状态更新 $\hat{\mathbf{X}}(k/k)$ 为:

$$\hat{\mathbf{X}}(k/k) = \hat{\mathbf{X}}(k/k-1) + \mathbf{K}(k)\mathbf{V}(k) \quad (10)$$

其中 $\mathbf{V}(k)$ 为残差, 即:

$$\mathbf{V}(k) = \mathbf{Z}(k) - \mathbf{H}\mathbf{X}\hat{\mathbf{X}}(k/k-1) \quad (11)$$

由式(7)可知, 在卡尔曼增益 $\mathbf{K}(k)$ 中, $\hat{\mathbf{P}}(k/k-1)\mathbf{H}^T$ 和 $\mathbf{S}(k)^{-1}$ 的乘积表示了预测误差和残差之间的比例关系, 因此, $\mathbf{V}(k)$ 和 $\mathbf{K}(k)$ 的乘积计算了 k

时刻的预测误差, 从而在式(10)中对 $\hat{\mathbf{X}}(k/k-1)$ 进行修正, 减小了建模误差的影响。

当得到 k 时刻预测误差与 $k-1$ 时刻残差的比例关系后, 可利用 $\mathbf{V}(k-1)$ 对预测值进行修正, 故本文进行以下改进:

1) 将式(7)的 $\mathbf{S}(k)^{-1}$ 替换为 $\mathbf{S}(k-1)^{-1}$, 获得式(12):

$$\mathbf{K}_{\text{new}}(k) = \hat{\mathbf{P}}(k/k-1)\mathbf{H}^T\mathbf{S}(k-1)^{-1} \quad (12)$$

2) 把式(10)中 $\mathbf{V}(k)$ 替换为 $\mathbf{V}(k-1)$, 获得式(13):

$$\hat{\mathbf{X}}(k/k) = \hat{\mathbf{X}}(k/k-1) + \mathbf{K}_{\text{new}}(k)\mathbf{V}(k-1) \quad (13)$$

将式(13)替换式(10), 并将式(12)置于式(7)之后, 其余步骤同原基本卡尔曼滤波算法。

上述改进的优点在于, 在原 KF 算法中, 式(10)的结果是对当前状态的最优估计, 而改进后, 用式(13)得到的结果为目标状态修正后的超前一步预测值, 该结果相比前者更加符合状态预测的意义。

4 MSTPM-GSKF 算法描述

针对多步轨迹预测问题, 本文提出一种基于组稀疏卡尔曼滤波的多步轨迹预测方法 MSTPM-GSKF, 首先利用组稀疏编码直接输出多步预测结果, 避免了迭代多步中预测误差的累计, 及繁琐的调参以及模型训练过程, 然后与 IKF 结合, 修正其预测值, 输出最终的预测结果, 具体过程如下:

首先为方便起见, 仅考虑单维坐标系的情况, 给定该坐标系下的 m 个输入数据, 建立超前 T 步预测模型, 如图 2 所示。

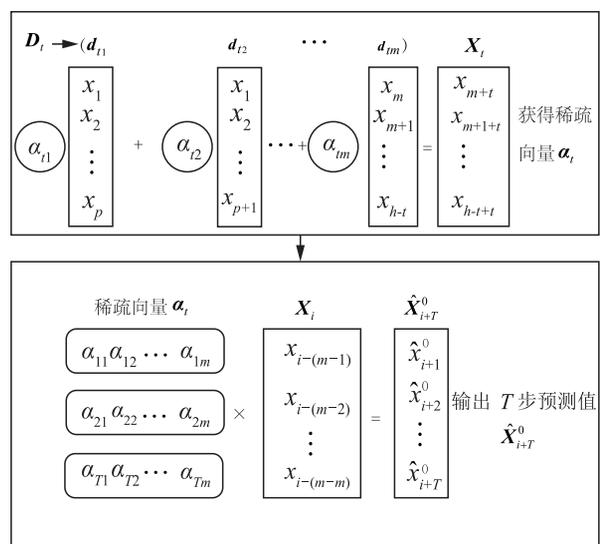


图2 基于 GSC 的超前 T 步预测基本原理

图 2 中, $\mathbf{D}_t = [d_{t1}, d_{t2}, \dots, d_{tm}]$ 为预测时间间隔为 t 时对应的输入特征矩阵, d_{ik} 为 \mathbf{D}_t 中的第 k 个列向量, $k = 1, 2, \dots, m$; $\alpha_t = [\alpha_{t1}, \alpha_{t2}, \dots, \alpha_{tm}]^T$ 为 \mathbf{D}_t

对应的稀疏向量, α_{tk} 为 d_{tk} 对应的稀疏系数; $\mathbf{X}_i = [x_{m+i}, x_{m+1+i}, \dots, x_{h-t+i}]$ 为 \mathbf{D}_i 对应的输出特征向量; i 表示当前时刻, \mathbf{X}_i 为输入数据; $\hat{\mathbf{X}}_{i+T}^0 = [\hat{x}_{i+1}^0, \hat{x}_{i+2}^0, \dots, \hat{x}_{i+T}^0]^T$ 为超前 T 步的预测值。

图 2 的具体过程如下: 首先, 利用窗口长度为 m , 滑动步长为 1 的滑动窗口根据不同的预测时间间隔 t , 将 $h-t$ 个轨迹数据划分为多个轨迹序列, 其序列个数为 $p = h-t-m+1$, 进而按时间顺序进行排列, 构成对应的输入特征矩阵 \mathbf{D}_i :

$$\mathbf{D}_i = \begin{bmatrix} x_1 & x_1 & \cdots & x_m \\ x_2 & x_3 & \cdots & x_{m+1} \\ \vdots & \vdots & & \vdots \\ x_r & x_{r+1} & \cdots & x_{h-t} \end{bmatrix} \quad (14)$$

其次, 根据输入数据进行时间间隔为 t 的时间延迟, 构建对应的输出特征向量 \mathbf{X}_i :

$$\mathbf{X}_i = [x_{m+i}, x_{m+1+i}, \dots, x_h]^T \quad (15)$$

再次, 利用所构建的 T 个 \mathbf{X}_i 及 \mathbf{D}_i 进行组稀疏编码, 并将所求解的稀疏向量 α_i 按顺序排列, 构成系数矩阵 \mathbf{A} :

$$\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_T]^T \quad (16)$$

最后, 获得基于 GSC 的初始 T 步预测值 $\hat{\mathbf{X}}_{i+T}^0 = [\hat{x}_{i+1}^0, \hat{x}_{i+2}^0, \dots, \hat{x}_{i+T}^0]^T$ 。

由卡尔曼滤波的计算过程可发现, 当输入数据为 $m \times 1$ 的列向量时, 其输出也应为 $m \times 1$ 的列向量, 故其状态转移矩阵 \mathbf{F} 应为 $m \times m$ 的方阵, 且当预测步长为 T 时, 为保证输入输出数据个数一致, 即预测步长最大为 $T = m$, 因此, 本节对于状态转移矩阵的建立如下。

对于单步预测, 即预测步长 $T = 1$ 时, 状态转移矩阵 \mathbf{A} 为:

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \\ \alpha_{T1} & \alpha_{T2} & \cdots & \alpha_{T(m-1)} & \alpha_{Tm} \end{bmatrix} \quad (17)$$

则输出 $\hat{\mathbf{X}}_{i+T}^0 = [x_{i-(m-1)}, x_{i-(m-2)}, \dots, x_{i+T}^0]^T$ 。

对于两步预测, 即预测步长 $T = 2$ 时, 状态转移矩阵 \mathbf{A} 为:

$$\mathbf{A} = \begin{bmatrix} 1 & \cdots & 0 & 0 & 0 \\ 0 & \cdots & 0 & 0 & 0 \\ \vdots & & \vdots & \vdots & \vdots \\ 0 & \cdots & 1 & 0 & 0 \\ \alpha_{(T-1)1} & \cdots & \alpha_{(T-1)(m-2)} & \alpha_{(T-1)(m-1)} & \alpha_{(T-1)m} \\ \alpha_{T1} & \cdots & \alpha_{T(m-2)} & \alpha_{T(m-1)} & \alpha_{Tm} \end{bmatrix} \quad (18)$$

输出 $\hat{\mathbf{X}}_{i+T}^0 = [x_{i-(m-1)}, x_{i-(m-2)}, \dots, \hat{x}_{i+(T-1)}^0, \hat{x}_{i+T}^0]^T$ 。

同理, 依次类推, 得到预测步长 $T = m$ 时的输出状态 $\hat{\mathbf{X}}_{i+T}^0$:

$$\hat{\mathbf{X}}_{i+T}^0 = \begin{bmatrix} \hat{x}_{i+1}^0 \\ \hat{x}_{i+2}^0 \\ \vdots \\ \hat{x}_{i+T}^0 \end{bmatrix} = \begin{bmatrix} \alpha_{11} & \alpha_{12} & \cdots & \alpha_{1m} \\ \alpha_{21} & \alpha_{22} & \cdots & \alpha_{2m} \\ \vdots & \vdots & & \vdots \\ \alpha_{T1} & \alpha_{T2} & \cdots & \alpha_{Tm} \end{bmatrix} \begin{bmatrix} x_{i-(m-1)} \\ x_{i-(m-2)} \\ \vdots \\ x_i \end{bmatrix} \quad (19)$$

在上述产生的多步预测模型基础上, 分析式 (14)~(19) 可知, 预测过程中系数矩阵 \mathbf{A} 不变, 当目标具有较强机动性时, 会产生较大的预测误差, 故利用改进卡尔曼滤波的修正式 (13) 对式 (19) 得到的初始预测结果 $\hat{\mathbf{X}}_{i+T}^0$ 进行修正, 以获得最终预测结果 $\hat{\mathbf{X}}_{i+T} = [\hat{x}_{i+1}, \hat{x}_{i+2}, \dots, \hat{x}_{i+T}]^T$, 从而进一步提高预测模型的精度。

综上所述, 基于 MSTPM-GSKF 的多步轨迹预测方法如下:

Step 1 采集机动目标在单维坐标系下的 h 个历史轨迹数据 $\mathbf{X}_g = \{x_1, x_2, \dots, x_h\}$ 。

Step 2 针对不同预测时间间隔 t , 构建输入特征矩阵 \mathbf{D}_i 及其相应的输出特征向量 \mathbf{X}_i 。

Step 3 利用 GSC 求解稀疏向量 α_i , 构成系数矩阵 \mathbf{A} , 建立 T 步预测模型。

Step 4 对上述模型输入前 m 个历史数据 $\mathbf{X}_i = [x_i, x_{i-1}, \dots, x_{i-(m-1)}]^T$, 获得 T 步预测初始值 $\hat{\mathbf{X}}_{i+T}^0 = [\hat{x}_{i+1}^0, \hat{x}_{i+2}^0, \dots, \hat{x}_{i+T}^0]^T$ 。

Step 5 利用式 (13) 将 $\hat{\mathbf{X}}_{0,i+T}$ 修正, 输出 T 步最终预测结果 $\hat{\mathbf{X}}_{i+T} = [\hat{x}_{i+1}, \hat{x}_{i+2}, \dots, \hat{x}_{i+T}]^T$ 。

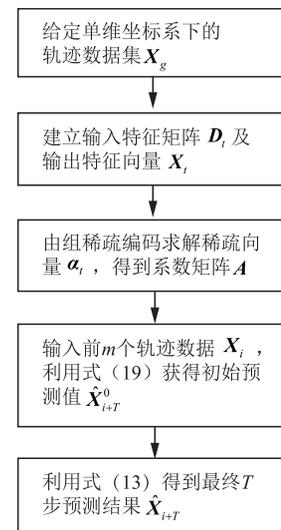


图 3 基于 MSTPM-GSKF 的多步轨迹预测流程图

由图 3 可知, 本文将 GSC 方法得到的多步预测结果 $\hat{\mathbf{X}}_{i+T}^0$ 作为 IKF 算法的一步预测初始值, 然后运行 IKF 算法, 由公式实现卡尔曼滤波过程, 它的

本质仍然是卡尔曼滤波过程,而卡尔曼滤波过程的稳定性和收敛性早已得到证明,故所提基于 MST-PM-GSKF 的多步轨迹预测方法稳定且收敛。

5 仿真研究

本文利用 X-Y-Z 三维坐标系下的非线性函数轨迹验证所提方法对机动目标轨迹预测的有效性,其参数方程如下:

$$\begin{cases} x=2\sin(0.1t) \\ y=\sin(t)\cos(0.1t) \\ z=2\sin(t)\cos(2t) \end{cases} \quad (20)$$

其中 $0 \leq t \leq 10\pi$, 时间点数量为 1 000 个, 时间间隔为 0.01π , 坐标单位为 m, 取前 800 个数据点作为训练集来构建预测模型, 后 200 个数据点作为测试集来输出预测结果。分别在 X、Y、Z 方向上, 对每一维的轨迹进行多步预测, 最后组成三维空间的预测轨迹。

预测误差指标采用均方根误差 (root mean square error, RMSE):

$$RMSE = \sqrt{\left(\sum_{i=1}^N (x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 + (z_i - \hat{z}_i)^2\right) / N} \quad (21)$$

式中: N 为轨迹点个数, x_i 、 y_i 和 z_i 分别为 X 轴、Y 轴和 Z 轴方向的第 i 个轨迹点的真实值, \hat{x}_i 、 \hat{y}_i 和 \hat{z}_i 分别为 X 轴、Y 轴和 Z 轴方向的第 i 个轨迹点的预测值。

当输入数据点的个数 m 从 2~8 进行单步预测时, 数学描述为 $\hat{x}_{i+1} = f(x_i, x_{i-1}, \dots, x_{i-(m-1)})$, $m = 1, 2, \dots, 8$ 分别对应输入数据点为 2, 3, ..., 8, 所对应的预测均方根误差比较如表 1 所示。

表 1 $m = 2 \sim 8$ 时 MSTPM-GSKF 的单步预测误差比较

数据点	2	3	4	5	6	7	8
RMSE/ 10^{-3} m	3.7	3.5	3.5	3.5	3.3	3.3	3.3

由表 1 可知, 在轨迹点个数 $m = 6$ 时轨迹预测的均方根误差最低, 为 0.003 3 m。当 $m = 7$ 和 8 时, 随着输入轨迹点的增加, 预测的均方根误差均保持不变, 仍为 0.003 3 m, 故可以确定, 当 $m = 6$ 作为系统输入较适宜。

输入个数为 6 时, 将预测步长从 1 增加到 6, 获得轨迹的预测均方根误差如表 2 所示。

表 2 $m = 6$ 时 MSTPM-GSKF 方法的多步预测误差比较

步长	1	2	3	4	5	6
RMSE/ 10^{-3} m	3.3	5.1	7.0	9.5	12.2	15.8

由表 2 可知, 在预测步长均匀增加下, 所提方法

的预测 RMSE 也随之增加, 如步长 $T = 1$ 增加到 $T = 2$ 时, 预测误差由 3.3 mm 增加到 5.1 mm, 增加了 1.8 mm。步长 $T = 2$ 增加到 $T = 3$ 时, 预测误差增加了 1.9 mm, 而且随着预测步长的增加, 预测误差逐渐增加, 在 $T = 4$ 时, 误差精度为 10^{-3} m, 之后随输入个数增加, 预测误差逐渐增加, 但仍保持在 10^{-2} m 左右。因此, 预测步长为 4 是预测误差变化趋势的重要转折点, 但为便于和 LSTM 等其他神经网络方法作仿真比较, 这里取预测步长 $T = m = 6$ 。

为验证 $m = 6, T = 6$ 时所提方法的鲁棒性, 对测试数据集依次添加 1~100 dB 信噪比的高斯白噪声, 输出的轨迹预测误差曲线如图 4 所示。

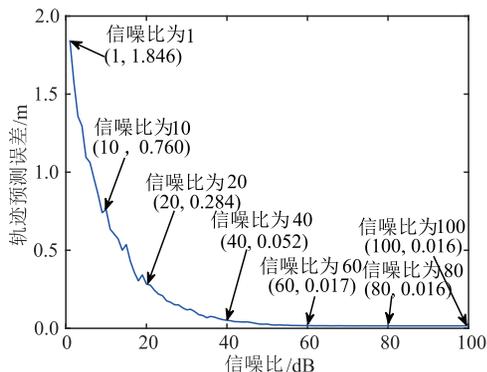


图 4 MSTPM-GSKF 方法在不同信噪比下的轨迹预测误差

由图 4 可知, 随着信噪比的增加, MSTPM-GSKF 方法的轨迹预测误差逐渐降低, 当取信噪比为 1 dB 时, 预测误差最大, 为 1 846 mm; 信噪比为 20 dB 时预测误差为 284 mm, 误差曲线开始变得平缓, 在信噪比为 40 dB 时预测误差为 52 mm, 误差曲线进一步平缓; 之后, 信噪比在 60~100 dB 范围内, 预测误差曲线进入稳定状态, 预测误差的值基本不变, 为 16 mm 左右, 体现出本方法在由低到高的信噪比变化下, 具有较强的鲁棒性, 而且收敛速度较快。

为进一步验证所提方法的有效性, 设 $m = 6, T = 6$ 下, 将 GSC、BP、LSTM 和所提 MSTPM-GSKF 在 X、Y、Z 单轴及整体预测 RMSE 进行比较, 见表 3, BP 和 LSTM 网络的学习率 η 取 0.1, 训练次数取 500。

表 3 BP、LSTM、GSC 和 MSTPM-GSKF 在 X、Y、Z 轴及整体预测误差的比较 单位: m

方法	X 轴	Y 轴	Z 轴	三维
BP ^[20]	0.041 5	0.083 3	0.080 0	0.122 7
LSTM ^[21]	0.057 2	0.057 3	0.177 8	0.195 4
GSC ^[12]	0.030 6	0.021 3	0.038 6	0.053 6
MSTPM-GSKF	0.001 6	0.004 1	0.015 2	0.015 8

表 3 可见, 和其他 3 种方法相比, 所提 MST-PM-GSKF 方法预测误差均为最小, 在 X、Y、Z 轴

的 RMSE 分别为 1.6 mm、4.1 mm、15.2 mm, 预测精度最高为 10^{-3} m, 最低为 10^{-2} m, 而 GSC 方法的预测精度最高和最低均为 10^{-2} m, BP 网络的预测精度最高为 10^{-2} m, 最低为 10^{-1} m, LSTM 网络的预测精度最高为 10^{-2} m, 最低为 10^{-1} m。通过上述比较可知, 所提方法在单轴和三维坐标系下的预测精度均较高, 显示出它的有效性。

分析原因, 这是因为 BP 和 LSTM 神经网络中, 内部采用 tanh 和 sigmoid 激活函数, 待调参数较多, 如神经元个数、学习率、批次等, 而训练结果的准确性直接受参数的影响, 当某次训练时这些参数选取不当而导致训练结果不佳时, 只能试凑选取新的参数进行下一次训练。这种试凑选择的主观性易导致训练结果不稳定, 甚至会难以收敛。

而 GSC 只需要计算稀疏向量的系数 α 即可确定最终输出结果, α 采用最小角回归算法估计求解, 一般给定初值后, 通过简单的计算, 即可获得稳定的全局解, 避免了 BP 和 LSTM 中由于初值选取和试凑选择的主观性而易于陷入局部极值的情况, 确保了结果的准确性, 因此预测误差较小。所提 MSTPM-GSKF 方法在预测中, 利用 IKF 的修正过程对 GSC 的预测结果进行误差补偿, 因此预测精度高于 GSC, 在上述方法中预测性能最好。

结合本文第 4 节关于所提 MSTPM-GSKF 方法的稳定和收敛性证明, 作进一步的仿真验证, 在输入个数为 6, 预测步长为 6 时, 获得 BP、LSTM、GSC 和 MSTPM-GSKF 的三维轨迹预测结果, 受输入个数选取所限, 实际轨迹点取 198 个, 见图 5。

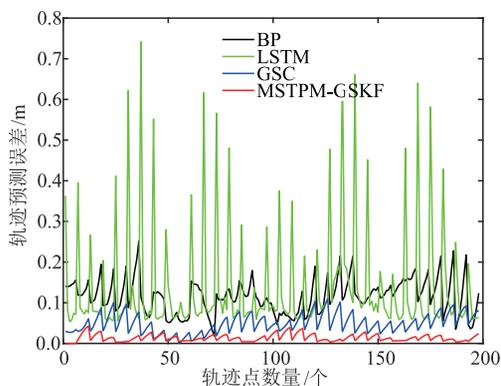


图 5 BP、LSTM、GSC 和 MSTPM-GSKF

在 198 个轨迹点下的三维轨迹预测误差

由图 5 可知不同方法在每个轨迹点的预测误差均具有一定波动性, 其中 LSTM 的波动性最大, BP 次之, 然后是 GSC 方法, 所提 MSTPM-GSKF 方法的波动性最小, 说明它具有较高稳定性。

此外, 本文为进一步加强稳定性的验证, 采用均值和方差指标来评价上述方法的预测稳定性, 各方法的指标比较见表 4。

表 4 GSC、LSTM、GRU 和 MSTPM-GSKF 方法轨迹预测误差的均值和方差比较

稳定性指标	GSC	BP	LSTM	MSTPM-GSKF
均值/m	0.048 4	0.115 2	0.137 9	0.013 3
方差/ m^2	0.000 5	0.001 8	0.019 2	0.000 07

由表 4 可知, 所提 MSTPM-GSKF 方法的均值为 0.013 3 m, 方差为 0.000 07 m^2 , 远远低于其他 3 种方法, 具有较高的稳定性。

X-Y-Z 三维坐标系下, 上述 4 种方法的预测轨迹和实际轨迹的对比如图 6 所示。

由图 6 可知, 在机动性较强的实际点 (0.312 9, 0.987 7, 2) 处, BP 法预测值为 (0.338 2, 1.096 7, 2.081 0), 距该点距离为 0.138 1 m; LSTM 法预测值为 (0.372 9, 0.998 5, 1.953 7), 距该点距离为 0.076 6; GSC 预测为 (0.366 1, 0.978 7, 1.987 8), 距该点距离为 0.055 4。而本文所提方法 MSTPM-GSKF 的预测结果为 (0.314 5, 0.993 6, 2.023 5), 距该点距离仅为 0.024 3 m, 在 3 种方法中预测误差最小, 对于具有较强机动性运动目标的适应能力较好, 故预测性能最好。

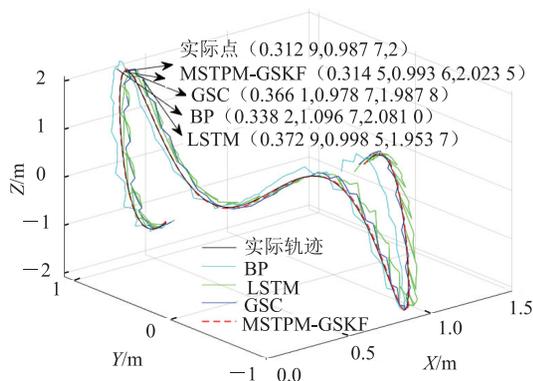


图 6 BP、LSTM、GSC 和 MSTPM-GSKF 的三维轨迹多步预测结果比较

6 结语

本文首先利用 LAR 算法一次直接求解稀疏编码方法中的稀疏系数, 并且结合卡尔曼滤波一次学习多步预测过程, 然后利用改进卡尔曼滤波来修正预测结果, 确保其精确性, 形成了基于组稀疏卡尔曼滤波的多步轨迹预测方法, 最后通过与传统浅层、深度学习神经网络和组稀疏编码方法的仿真比较, 验证了所提方法的有效性。

本方法不仅在建模时避免了人为因素对模型结构的影响, 而且在预测中, 当待预测目标的运动特性与所用训练数据的特性相差较大时, BP 和 LSTM 依然根据此训练模型进行预测, 参数失配会导致较

大的预测误差,而本方法利用改进卡尔曼滤波对组稀疏编码的预测值进行了动态修正,有效减小了建模误差产生的影响。因此对于机动目标的运动轨迹具有较强的预测性能。

参考文献

- [1] 乔少杰,韩楠,朱新文,等.基于卡尔曼滤波的动态轨迹预测算法[J].电子学报,2018,46(2):418-423.
- [2] 赵梓烨,刘海鸥,陈慧岩.分布式电驱动无人高速履带车辆越野环境轨迹预测方法研究[J].兵工学报,2019,40(4):680-688.
- [3] 徐洪俊,张锦东,张其林.基于时间序列模型和扩展卡尔曼滤波算法的结构响应预测[C]//第十九届全国现代结构工程学术研讨会论文集.天津:天津大学,2019:389-393.
- [4] 姜佰辰,关键,周伟,等.基于多项式卡尔曼滤波的船舶轨迹预测算法[J].信号处理,2019,35(5):741-746.
- [5] 王开园,许志,唐硕,等.一种基于飞行任务的临近空间短距滑翔飞行器弹道预示方法[J].宇航学报,2021,42(1):50-60.
- [6] 高天航,徐力,靳廉洁,等.考虑航艏向与数据变化差异的船舶轨迹预测[J].交通运输系统工程与信息,2021,21(1):90-94.
- [7] 石韬.基于机器学习的无人艇自主避碰方法研究[D].哈尔滨:哈尔滨工程大学,2021.
- [8] 宋波涛,许广亮.基于LSTM与1DCNN的导弹轨迹预测方法[J].系统工程与电子技术,2023,45(2):504-512.
- [9] 张宏鹏,黄长强,轩永波,等.用门控循环单元实时预测空战飞行轨迹[J].系统工程与电子技术,2020,42(11):2546-2552.
- [10] 张堃,杜睿怡,时昊天,等.基于Mogriifier-BiGRU的飞行器轨迹预测[J/OL].兵工学报.http://kns.cnki.net/kcms/detail//11.2176.tj.20230201.1130.001.html
- [11] 储晨阳,秦川,鞠平,等.基于优化稀疏编码的超短期负荷滚动多步预测[J].电工技术学报,2021,36(19):4050-4059.
- [12] WEI D, SHEN X B, SUN Q S, et al. Locality-Aware Group Sparse Coding on Grassmann Manifolds for Image Set Classification[J]. Neurocomputing, 2020, 385: 197-210.
- [13] 徐云飞,孙永顺,丁晓喜,等.基于lamb波的数据传输与缺陷检测同步实现方法[J].仪器仪表学报,2022,43(3):24-31.
- [14] 谢惠华,黎明,王艳,等.基于特征优选和字典优化的组稀疏表示表情识别[J].模式识别与人工智能,2021,34(5):446-454.
- [15] EFRON B, HASTIE T, JOHNSTONE I, et al. Least Angle Regression[J]. The Annals of Statistics, 2004, 32(2): 407-499.
- [16] HELMI A, FAKHR M W, ATIYA A F. Multi-Step Ahead Time Series Forecasting via Sparse Coding and Dictionary Based Techniques[J]. Applied Soft Computing, 2018, 69: 464-474.
- [17] CHAI Z L, SONG W, WANG H L, et al. A Semi-Supervised Auto-Encoder using Label and Sparse Regularizations for Classification[J]. Applied Soft Computing, 2019, 77: 205-217.
- [18] 丁克良,欧吉坤,赵春梅.正交最小二乘曲线拟合法[J].测绘科学,2007(3):17-19,192.
- [19] 张友民,戴冠中,张洪才.卡尔曼滤波计算方法研究进展[J].控制理论与应用,1995(5):529-538.
- [20] 王德明,王莉,张广明.基于遗传BP神经网络的短期风速预测模型[J].浙江大学学报(工学版),2012,46(5):837-841+904.
- [21] 王鑫,吴际,刘超,等.基于LSTM循环神经网络的故障时间序列预测[J].北京航空航天大学学报,2018,44(4):772-784.

(编辑:徐敏)