arXiv:2304.02975v1 [eess.SY] 6 Apr 2023

深度长短期记忆网络：稳定性性质与实验验证

Fabio Bonassi[1,*] , Alessio La Bella[1],Giulio Panzani[1],MarcelloFarina[1],Riccardo Scattolini[1]

摘要—**本研究旨在探讨使用增量输入到状态稳定（δISS）深层长短期记忆网络（LSTM）辨识非线性动力系统。我们证明，可以利用网络权重的适当充分条件来设计一种训练过程，从数据中学习经证明具有δISS特性的LSTM模型。我们在真实的线控制动装置上测试了所提出的方法，利用实验采集的输入输出数据辨识系统模型。结果表明，建模性能令人满意。**

索引词—**循环神经网络、增量输入到状态稳定性、非线性系统辨识、线控制动。**

I. 引言

近年来，机器学习与控制系统领域之间日益频繁的思想交流，促成了机器学习技术和工具在动态系统辨识与控制背景下的富有成效的传播。我们特别指的是神经网-络（NNs）：尽管自九十年代以来，人们就一直在研究其在控制系统设计背景下的使用 [1],，但用于实现和训练这些网络的大型且信息丰富的数据集、廉价的计算能力以及开源软件工具的可用性，又重新激发了人们对这些技术的兴趣 [2]。

在众多可用的神经网络架构中，循环神经网络（RNN）被认为最适用于系统辨识 [3]，，因为其状态特性使其能够存储过往数据。长短期记忆网络（LSTM，[4]）和门控循环单元（GRU，[5]）尤其成功且广受欢迎；它们的门控架构降低了受到众所周知的梯度消失和梯度爆炸问题影响的可能性，而这些问题通常困扰着循环神经网络 [6]。相比之下，这两种架构在学习动态系统方面展现出了卓越的能力 [7]。

在控制领域，控制神经网络可以通过多种多样的方式加以应用 [8], [9]，，包括间接的数据驱动控制综合。具体而言，从通过实验收集的输入输出数据中辨识出的动态系统神经网络模型 [10], 随后可用于综合基于模型的控制器，例如采用模型预测控制（模型预测控制） [11]。

这种策略能够自然地将机器学习方法（用于系统辨识）与传统的基于模型的控制设计策略相结合，因此已获得广泛关注，学术界中例如见 [12], [13], [14] ，工业界中例如见 [15], [16], [17]。根据间接数据驱动控制范式辨识出系统的门控循环神经网络模型后，便会采用确定性等价原理（CEP），从而可以基于辨识出的模型设计控制器，例如采用模型预测控制策略。在此背景下，若能验证模型的增量输入到状态稳定性（δISS， [18]），设计者便可综合出具有闭环稳定性保证 [14], [19]且能够对恒定参考信号实现鲁棒零误差跟踪的控制律 [20], [21]。正如 [9],中所讨论的，这一性质还有助于增强模型抵御输入扰动的鲁棒性、确保即使初始化有误也能保持建模性能的一致性，并提升安全性 [22]。

近期文献开始研究如何训练可证明具有 δISS 性质的 RNN。文献 [20] 针对神经 NARX 网络（即使用前馈神经网络作为非线性回归器的 ARX 模型），提出了保证网络具有 δISS 性质的充分条件。文献 [23],研究了 GRU 的 δ ISS 性质，并提出了适用于浅层 （即单层）和深层 （即多层）GRU 的 δISS 性质的条件。文献 [14], 给出了 LSTM 具有 δISS 性质的充分条件，但仅限于浅层网络；而浅层网络可能不适合识别复杂的真实系统 [3]。

本文的目标有两个方面：(*i*) 为深度LSTM网络建立δ ISS条件，(*ii*) 应用所提出的框架，对一个具有非平凡动力学特性的真实系统进行辨识，即一个线控制动装置。

本文的结构安排如下。第II节介绍深度LSTM网络的状态空间表述，并在 δ节推导其III节中的充分条件IV节将所提出的方法应用于真实的线控制动装置进行测试，从而辨识出系统的 δISS模型。

符号说明

本文采用以下符号。给定向量 $v \in \mathbb{R}^n$，用 v' 表示其转置，用 $[v]_i$ 表示其第 i 个分量，并用 $\|v\|_p$ 表示其 p- 范数。两个维度相同的向量 v 和 w 之间的 Hadamard（逐元素）乘积记为 $v \circ w$。给定一个 $n \times m$ 矩阵 A，用 $\|A\|_p$ 表示其诱导 p-范数。

为简洁起见，对于随时间变化的向量，时间索引 k 记为下标，例如 v_k。从时间索引 k_1 到 $k_2 \geqslant k_1$的向量序列记为$v_{k_1:\, k_2}$，即$v_{k_1:\, k_2} = \{v_{k_1},\ v_{k_1+1},\ \ldots\ldots,\ v_{k_2}\}$。序列的$\ell_{p,q}$ 范数定义为$\|v_{k_1:\, k_2}\|_{p,q} =$

* 通讯作者[1] 作者就职于电子、信息与生物工程系、米兰理工大学，地址：意大利米兰 20133，Ponzio 路 34/5。电子邮箱：name.surname@polimi.it

本文是获 2023 年欧洲控制会议（ECC'23）接收的一篇论文的扩展版本。版权可能在未事先通知的情况下转让，转让后此版本可能不再可访问。

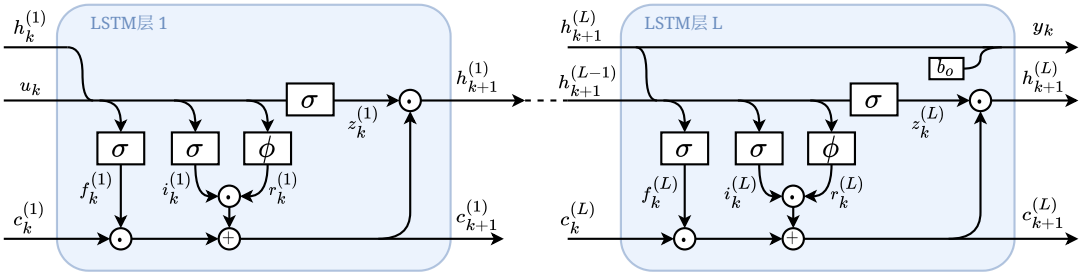


图 1。 深度 LST 的示意图 M 架构，由 L 个 LSTM 层串联而成，其中每一层都接收更新后的隐藏状态 of 前一层作为输入。

$\|[\|v_{k_1}\|_p, \ldots\ldots, \|v_{k_2}\|_p]'\|_q$，其中一个值得注意的情形是$\ell_{p,\infty}$范数，此时$\|v_{k_1:k_2}\|_{p,\infty} = \max_{k\in\{k_1,\ldots\ldots,k_2\}} \|v_k\|_p$。

最后，我们分别用$\sigma(x) = \frac{1}{1+e^{-x}}$和 $\phi(x) =$tanh(x) 表示 sigmoid 和 tanh 激活函数。注意，当应用于向量时，这些激活函数均按元素逐一应用。

II. 深度LSTM架构

受深度循环网络 [3], 众所周知的卓越建模能力启发，本文聚焦于用于非线性系统辨识的深度 LSTM。

深度LSTM由以下内容串联而成： $L > 1$LSTM层，其中每一层都由浅层LSTM的状态空间方程描述，见图 1。特别地，每一层 $l \in \{1, \ldots, L\}$ 由以下方程控制：

$$\begin{cases} c_{k+1}^{(l)} = f_k^{(l)} \circ c_k^{(l)} + i_k^{(l)} \circ r_k^{(l)}, \\ h_{k+1}^{(l)} = z_k^{(l)} \circ \phi(h_k^{(l)}), \end{cases} \tag{1a}$$

其中 $c_k^{(l)} \in \mathbb{R}^{n_c^{(l)}}$ 和 $h_k^{(l)} \in \mathbb{R}^{n_c^{(l)}}$ 分别为细胞状态和隐藏状态， $n_c^{(l)}$表示所选定的该层神经元数量。该层的状态向量是细胞状态和隐藏状态的拼接，即 $x_k^{(l)} = [c_k^{(l)\prime}, h_k^{(l)\prime}]' \in \mathbb{R}^{n_x^{(l)}}$，其中 $n_x^{(l)} = 2n_c^{(l)}$。

术语 $f_k^{(l)}$， $i_k^{(l)}$，以及 $z_k^{(l)}$ 出现在（1a）中的是所谓的 门 网络的门，它们可以调节层内信息的流动和信息存储，而 $r_k^{(l)}$ 是所谓的 压缩输入。其形式为

$$\begin{aligned} f_k^{(l)} &= \sigma\big(W_f^{(l)} u_k^{(l)} + U_f^{(l)} h_k^{(l)} + b_f^{(l)}\big), \\ i_k^{(l)} &= \sigma\big(W_i^{(l)} u_k^{(l)} + U_i^{(l)} h_k^{(l)} + b_i^{(l)}\big), \\ z_k^{(l)} &= \sigma\big(W_z^{(l)} u_k^{(l)} + U_z^{(l)} h_k^{(l)} + b_z^{(l)}\big), \\ r_k^{(l)} &= \phi\big(W_r^{(l)} u_k^{(l)} + U_r^{(l)} h_k^{(l)} + b_r^{(l)}\big). \end{aligned} \tag{1b}$$

请注意，由于 Sigmoid 函数 $\sigma(\cdot)$ 在$(0,\ 1)$中有界，因此门是次酉正元素构成的向量：分量接近 1 的门称为开启，即允许信息流动的门；分量接近 0 的门则称为关闭。同样，由于双曲正切函数 $\phi(\cdot)$ 在 $(-1,\ 1)$中有界，因此压缩后的输入是一个分量处于该范围内的向量。

在（1b）中引入的门和压缩输入还取决于 $u_k^{(l)}$，即层的输入向量。在深层架构中，这样的输入确实定义为

$$u_k^{(l)} = \begin{cases} u_k & \text{if } l = 1, \\ h_{k+1}^{(l-1)} & \text{if } l \in \{2, \ldots, L\}. \end{cases} \tag{1c}$$

具体而言，第一层由模型的输入 u_k 提供输入，而后续任意层 $l \in \{2, \ldots, L\}$ 的输入定义为紧邻的前一层更新后的隐藏状态，即 $h_{k+1}^{(l-1)}$。最后，模型的输出定义为最后一层隐藏状态的仿射变换，

$$y_k = U_o h_k^{(L)} + b_o. \tag{1d}$$

请注意，每一层都由一组权重参数化

$$\Phi^{(l)} = \{W_f^{(l)}, U_f^{(l)}, b_f^{(l)}, W_i^{(l)}, U_i^{(l)}, b_i^{(l)}, W_z^{(l)}, U_z^{(l)}, b_z^{(l)}, W_r^{(l)}, U_r^{(l)}, b_r^{(l)}\}.$$

因此，令 $\Phi = \{U_o,\ b_o\} \cup \bigcup_{l=1}^{L} \Phi^{(l)}$ 为网络的权重集合，并用 $x_k = [x_k^{(1)\prime}, \ldots\ldots, x_k^{(L)\prime}]'$ 表示深度 LSTM 模型的状态向量，（1）可简洁地表示为

$$\Sigma(\Phi): \begin{cases} x_{k+1} = \varphi(x_k, u_k; \Phi) \\ y_k = \psi(x_k; \Phi) \end{cases}, \tag{2}$$

其中 φ 和 ψ 是由方程（1）产生的适当函数。现在，我们提出一个关于模型输入有界性的常用假设。

假设 *1*： 输入向量 u_k 的范数有界于 1，即 $u_k \in \mathcal{U}$，其中

$$\mathcal{U} = \{u \in \mathbb{R}^{n_u} : \|u\|_\infty \leqslant 1\}. \tag{3}$$

请注意，当控制变量受到饱和限制时，假设 1 可通过对输入进行适当的归一化轻松满足，这在深度学习中是一种常见做法 [2]。因此，可以提出以下命题。

命题 *1*： 对于任意层 $l \in \{1, \ldots, L\}$，压缩输入 $r_k^{(l)}$ 以及门 $f_k^{(l)}$、 $i_k^{(l)}$和 $z_k^{(l)}$均有界，具体如下：

$$-1 < -\bar{\phi}_r^{(l)} \leqslant [r_k^{(l)}]_j \leqslant \bar{\phi}_r^{(l)} < 1, \tag{4a}$$

$$0 < 1 - \bar{\sigma}_f^{(l)} \leqslant [f_k^{(l)}]_j \leqslant \bar{\sigma}_f^{(l)} < 1, \tag{4b}$$

$$0 < 1 - \bar{\sigma}_i^{(l)} \leqslant [i_k^{(l)}]_j \leqslant \bar{\sigma}_i^{(l)} < 1, \tag{4c}$$

$$0 < 1 - \bar{\sigma}_z^{(l)} \leqslant [z_k^{(l)}]_j \leqslant \bar{\sigma}_z^{(l)} < 1, \tag{4d}$$

$\forall j \in \{1, ..., n_c^{(l)}\}$，其中界定义如下：

$$\bar{\phi}_r^{(l)} = \phi(\|W_r^{(l)} \quad U_r^{(l)} \quad b_r^{(l)}\|_\infty), \tag{5a}$$
$$\bar{\sigma}_f^{(l)} = \sigma(\|W_f^{(l)} \quad U_f^{(l)} \quad b_f^{(l)}\|_\infty), \tag{5b}$$
$$\bar{\sigma}_i^{(l)} = \sigma(\|W_i^{(l)} \quad U_i^{(l)} \quad b_i^{(l)}\|_\infty), \tag{5c}$$
$$\bar{\sigma}_z^{(l)} = \sigma(\|W_z^{(l)} \quad U_z^{(l)} \quad b_z^{(l)}\|_\infty). \tag{5d}$$

证明：参见附录I-A. ■

鉴于这些界，可以定义式（1）的不变集。在下一节中，将针对该不变集分析 δISS性质。我们回顾一下，若一个集合 $\mathcal{X}$ 满足以下条件，则称其为不变集：对于任意 $u \in \mathcal{U}$，$x \in \mathcal{X} \implies \varphi(x,\ u) \in \mathcal{X}$，其中 $\varphi(\cdot)$ 定义于式（2）中。

命题 *2*：深度 LSTM模型的一个不变集（1）是

$$\mathcal{X} = \bigtimes_{l=1}^{L} \mathcal{X}^{(l)}, \tag{6a}$$

其中，$\times_{l=1}^{L}$ 表示笛卡尔积，$\mathcal{X}^{(l)}$ 是

$$\mathcal{X}^{(l)} = \mathcal{C}^{(l)} \times \mathcal{H}^{(l)}, \text{ where} \tag{6b}$$
$$\mathcal{C}^{(l)} = \left\{ c \in \mathbb{R}^{n_c^{(l)}} : \|c\|_\infty \leqslant \frac{\bar{\sigma}_i^{(l)}\ \bar{\phi}_r^{(l)}}{1 - \bar{\sigma}_f^{(l)}} = \bar{c}^{(l)} \right\}, \tag{6c}$$
$$\mathcal{H}^{(l)} = \left\{ h \in \mathbb{R}^{n_c^{(l)}} : \|h\|_\infty \leqslant \phi(\bar{c}^{(l)}) = \bar{h}^{(l)} \right\}, \tag{6d}$$

以及 $\bar{\sigma}_f^{(l)}$，$\bar{\sigma}_i^{(l)}$，$\bar{\sigma}_z^{(l)}$，以及 $\bar{\sigma}_r^{(l)}$定义同（5）。

证明：见附录I-B。 ■

现在我们可以分析 δ深度LSTM模型的ISS性质（2）。

III. 稳定性性质

定义 *1*（$\mathcal{K}_\infty$ 函数）：连续函数 $\Psi(s) : \mathbb{R}_{\geqslant 0} \mapsto \mathbb{R}_{\geqslant 0}$ 属于 $\mathcal{K}_\infty$ 类，当且仅当它关于自变量严格递增，$\Psi(0) = 0$，且 $\Psi(s) \longrightarrow \infty$。

定义 *2*（$\mathcal{KL}$ 函数）：连续函数 $\Psi(s) : \mathbb{R}_{\geqslant 0} \times \mathbb{R}_{\geqslant 0} \mapsto \mathbb{R}_{\geqslant 0}$ 属于 $\mathcal{KL}$ 类，如果它关于其第一个参数属于 $\mathcal{K}_\infty$ 类，并且对于任意 s、$\Psi(s,\ t) \xrightarrow[t\to\infty]{} 0$。

我们将 $x_{0:k}(x_0,\ u_{0:k})$ 记作（2）在以 x_0 初始化并输入序列 $u_{0:k}$时的状态轨迹。（区域）δ本文所考虑的ISS 因此可定义如下。

定义 *3*（*δISS [18]*）：若存在函数 β，其属于$\mathcal{KL}$ 类，且 γ 属于 $\mathcal{K}_\infty$ 类，使得对于任意一对初始状态$x_{a,\ 0} \in \mathcal{X}$ 和 $x_{b,\ 0} \in \mathcal{X}$，以及任意一对输入序列$u_{a,\ 0:k} \in \mathcal{U}$ 和 $u_{b,\ 0:k} \in \mathcal{U}$，均有：系统（2）相对于输入集 $\mathcal{U}$和不变集 $\mathcal{X}$ 是增量输入到状态稳定（δISS）的。

$$\|x_{a,k} - x_{b,k}\|_2 \leqslant \beta(\|x_{a,0} - x_{b,0}\|_2, k) + \gamma(\|u_{a,0:k} - u_{b,0:k}\|_{2,\infty}), \tag{7}$$

其中 $x_{\alpha,\ k}$ 是 $x_k(x_{\alpha,\ 0},\ u_{\alpha,\ 0:k})$的简写。

值得注意的是，这一性质意味着，将系统（2）初始化为 $\mathcal{X}$ 中两个不同的初始状态，并向其输入从集合 $\mathcal{U}$ 中提取的任意两个输入序列，便可获得状态轨迹，其间的距离渐近地不再取决于初始条件。此外，这些输入序列越接近，状态轨迹之间距离的上界就越小。

δISS 性质是保障 RNN 在受到输入扰动时安全且鲁棒的有效工具[9]。事实上，可以利用这一性质计算模型输出可达集的保守界限；或者至少为该集合的数值计算提供理论依据。在 [14], [24], [21], 模型的 δISS 也已被证明有助于合成标称闭环稳定的预测控制律。

本文旨在分析 δ深度LSTM的ISS性质，从而为基于这些模型设计预测控制律铺平道路，并保证闭环系统的稳定性。特别地，本文将证明命题1 和2 表明，若 δ命题2中提出的ISS充分条件 [14]在每一层均得到满足，则深度LSTM是 δISS系统。

定理*1*：深度LSTM（ δ2）关于集合 $\mathcal{X}$ 和 $\mathcal{U}$ 具有ISS性质的充分条件是，对于每一层 $l \in \{1,\ .....,\ L\}$，

$$\bar{\sigma}_f^{(l)} + \bar{\sigma}_z^{(l)} \bar{\alpha}^{(l)} + \frac{1}{4}\bar{h}^{(l)} \|U_z^{(l)}\|_2 - \frac{1}{4}\bar{\sigma}_f^{(l)} \bar{h}^{(l)} \|U_z^{(l)}\|_2 - 1 < 0, \tag{8a}$$
$$\frac{1}{4}\bar{\sigma}_f^{(l)} \bar{h}^{(l)} \|U_z^{(l)}\|_2 - 1 < 0, \tag{8b}$$

其中，$\bar{\alpha}^{(l)}$ 定义为

$$\bar{\alpha}^{(l)} = \frac{1}{4}\bar{c}^{(l)} \|U_f^{(l)}\|_2 + \bar{\sigma}_i^{(l)} \|U_r^{(l)}\|_2 + \frac{1}{4}\bar{\phi}_r^{(l)} \|U_i^{(l)}\|_2, \tag{9}$$

$\bar{\sigma}_f^{(l)}$，$\bar{\sigma}_i^{(l)}$，$\bar{\sigma}_z^{(l)}$和 $\bar{\phi}_r^{(l)}$的定义与（5）中相同，且 $\bar{c}^{(l)}$ 和$\bar{h}^{(l)}$ 的定义分别如（6）所示。

Proof: See Appendix I-C. ■

注意，定理1中提出的充分条件对应于 $2L$ 关于深度LSTM 模型权重的非线性不等式。为简洁起见，这组不等式记为

$$\nu(\Phi) < 0. \tag{10}$$

请注意，这一结果与 [25],其中δ级联连续时间 δISS 系统的研究相一致。条件（10）可以通过两种方式使用：（*i*）事后评估经过训练的 δ模型是否能够保证 ISS，（*ii*）在训练过程中，强制满足 $\nu(\Phi) < 0$，从而保证 δ正在训练的网络具有 ISS。基于这一思路，附录II 详细介绍了一种训练可证明具有δISS 的深度 LSTM 模型的方法。

IV. 数值示例

所提出方法的建模性能通过线控制动（BBW）执行器的数据进行了测试。线控制动执行器是线控技术在车辆制动部件上的一种应用：通过在物理上将驾驶员指令与实际制动压力解耦，它们被有效应用于

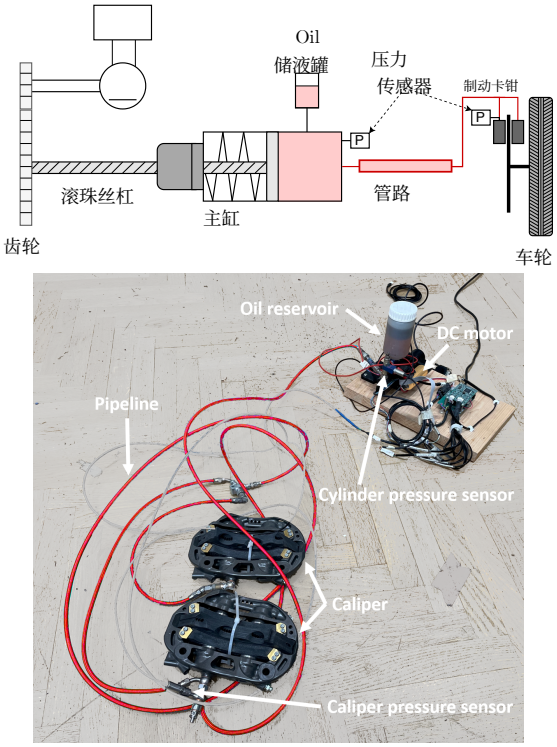


图2。线控制动原型装置示意图。

车辆高层控制系统，例如防抱死制动系统（ABS）和电子稳定程序（ESP）。

线控制动执行器如图所示 2，并附有其示意图。该系统由与滚珠丝杠机构耦合的直流电机构成，可将电机的旋转运动转换为主缸的直线运动。通过改变主缸位置，在制动卡钳中建立压力；最终，制动片压向制动盘，从而产生所需的制动扭矩。线控制动控制系统的最终目标是精确跟踪制动压力参考值。一种有效且广泛采用的控制策略包含两个嵌套回路 [26]：内回路控制主缸位置 x_p，通过作用于直流电机电流 i_m；外回路控制回路制动压力，通过作用于内部回路位置参考值实现。尽管关注的变量是卡钳压力 P_{cal}，为简化整个系统布局，通常使用主缸压力 P_{cyl} 作为反馈。

所考虑设置的特殊之处在于管路非常长，如图 2中以红色标出，这导致主缸与卡钳压力之间存在不可忽略的动态差异， [27]。出于控制设计和虚拟传感的目的，这促使我们开发一种能够正确描述主缸位置 x_p、缸内压力 P_{cyl}以及卡钳处压力 P_{cal}之间关系的模型。我们在此提出通过 δISS 深度LSTM网络来学习该模型。选择使用类似LSTM的非线性动态模型，与BBW的特征十分契合，即存在非线性以及其单输入多输出（SIMO）特性。

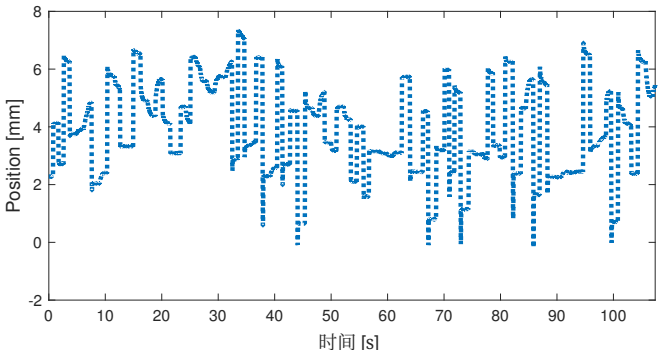


图 3。性能测试：应用于深度 LSTM 模型的输入序列（即活塞位置）。

A. 实验计划和数据采集

为收集必要数据并完成模型训练过程，已在该装置上开展了 6 次开环实验。在每次持续180秒的实验中，均为电机电流参考值生成了一个多电平伪随机二进制信号，随后由该装置的电子控制单元执行该信号。因此，电流参考值是一串具有随机幅值（范围为 0A 至 10A）和随机持续时间（介于 0.5s 和1.5s 之间）的阶跃信号。对于每次实验，这些随机变量均从不同的分段均匀概率分布中抽取。通过这种方式，提高了所收集数据的信息丰富度。

在实验过程中，x_p、P_{cyl}和 P_{cal} 以 200Hz 的采样频率进行了测量。因此，每项实验包含 36000 (x_p, P_{cyl}, P_{cal}) 个数据点。

随后根据附录中提供的详细信息，将实验划分为训练、验证和测试数据集III。

B. 深度*LSTM*训练

训练过程使用运行于Python 3.10上的 Py-Torch1.12 完成。采用了一个具有$L = 2$ 层、每层 $n_c^{(1)} = n_c^{(2)} = 8$ 个单元的深度LSTM。请注意，该模型有一个输入（活塞位置 x_p）和两个输出（制动钳和主缸中的压力，即分别为 P_{cal} 和 P_{cyl}）。

该模型通过最小化仿真均方误差（MSE）并满足条件（8），按照附录 II中所述的过程进行训练，而附录 III 详细介绍了训练过程的实现。所得建模性能可通过下图进行评估。具体而言，图 4 概述了测试数据及其仿真轨迹，采用图 3中的测试输入，并展示了学习得到的 LSTM 模型和线性模型的结果。线性模型采用经典的子空间辨识方法进行辨识，该方法适用于系统的单输入多输出（SIMO）特性。事实上，LSTM 模型的有效性显而易见：尽管线性模型对两种压力总体上都能较好地拟合，但其稳态误差和低估更为明显，甚至导致不切实际的负压力值。为了量化 LSTM

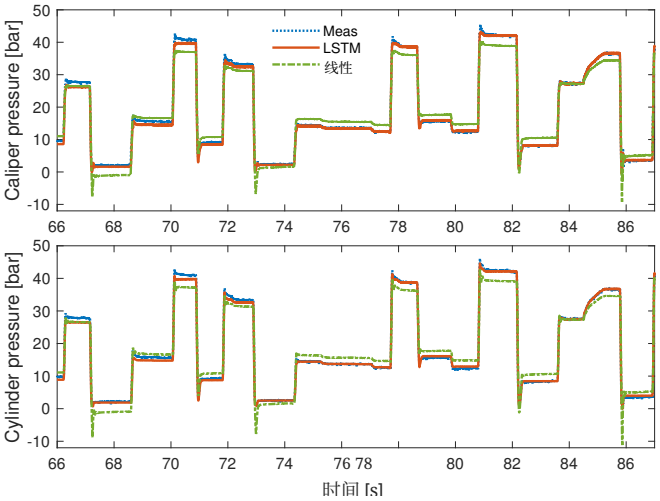


图4.性能测试：训练好的深度 LSTM模型的开环预测（红色实线）与线性模型（绿色点划线）和真实值（蓝色点线）的比较。

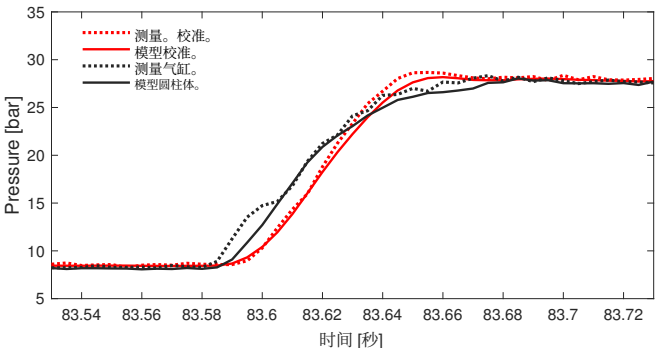


图5. 性能测试：高度动态瞬态期间的LSTM建模细节。

性能 使用FIT 指数，参见附录 II，得到 FIT $= 88.6\%$，表明建模性能良好。

最后，为了评估网络对缸压和卡钳压力特征动态过程的精确建模能力，图5 展示了高度动态瞬态过程中的测试数据细节：训练得到的模型能够正确描述系统的因果关系，即缸压先于卡钳压力升高；此外，模型还能很好地描述各自的动态特性，尤其是与主缸压力相比，卡钳压力的阻尼较小。

V. 结论

本文研究了使用深度长短期记忆网络（LSTM）对非线性动态系统进行辨识。提出了保证其增量输入到状态稳定性（δISS）的充分条件。这些条件由网络权重上的若干非线性不等式组成，并且可以在训练过程中加以约束。

所提出的方法经过测试，以从实验采集的数据中识别线控制动系统。

装置，取得了非常令人满意的结果。未来的研究方向将涉及利用所识别的网络，为所分析的基准系统综合一种计算效率高的控制系统，例如 [19] 中提出的内部模型控制策略。

附录 I 证明

A. 命题1的证明

首先，注意到，对于任意层 $l \in \{1, \ldots, L\}$，均有 $h_k^{(l)}$ 是有界于单位值的，即 $\|h_k^{(l)}\|_\infty \leqslant 1$，这是因为它被定义为两个次酉向量的逐元素乘积，见（1a）。根据假设 1，并结合输入向量的定义（1c），有 $\|u_k^{(l)}\|_\infty \leqslant 1$。

因此，可以通过逐层应用 5 中的论证来得到界，这些论证见于 [14,第 2 节。2]。 ■

命题2的证明

由于对于每一层 $l \in \{1, \ldots\ldots, L\}$，输入向量$u_k^{(l)}$ 均为次酉，根据逐层应用 [14,第 2 节2] 中得到的结果，则（1a）具有$\mathcal{X}^{(l)}$ 中报告的（6）所示不变集。因此，（6a）是深度 LSTM 模型的不变集。 ■

定理1的证明

考虑一个一般层 $l \in \{1, \ldots\ldots, L\}$。由于其输入向量有界于单位范围内，满足条件（8）意味着根据 [14, 命题 2], 该层是 δISS [14, 。也就是说，考虑初始状态对 $x_{a,\,0} \in \mathcal{X}$ 和$x_{b,\,0} \in \mathcal{X}$，以及输入序列对 $u_{a,\,0:k} \in \mathcal{U}_{0:k}$ 和$u_{b,\,0:k} \in \mathcal{U}_{0:k}$，并令 $x_{\alpha,\,k}$ 简写为 $x_k(x_{\alpha,\,0},\ u_{\alpha,\,0:k})$，则有

$$\begin{bmatrix} \|c_{a,k+1}^{(l)} - c_{b,k+1}^{(l)}\|_2 \\ \|h_{a,k+1}^{(l)} - h_{b,k+1}^{(l)}\|_2 \end{bmatrix} \leqslant \mathfrak{A}^{(l)} \begin{bmatrix} \|c_{a,k}^{(l)} - c_{b,k}^{(l)}\|_2 \\ \|h_{a,k}^{(l)} - h_{b,k}^{(l)}\|_2 \end{bmatrix} + \mathfrak{B}^{(l)} \|u_{a,k}^{(l)} - u_{b,k}^{(l)}\|_2, \tag{11}$$

其中 $\mathfrak{A}^{(l)}$ 是一个施尔稳定的2×2矩阵，见 [14], 定义为

$$\mathfrak{A}^{(l)} = \begin{bmatrix} \bar{\sigma}_f^{(l)} & \bar{\alpha}^{(l)} \\ \bar{\sigma}_z^{(l)} \bar{\sigma}_f^{(l)} & \bar{\sigma}_z^{(l)} \bar{\alpha}^{(l)} + \frac{1}{4} \bar{h} \|U_z^{(l)}\|_2 \end{bmatrix}. \tag{12}$$

回顾 $u_\alpha^{(l)}$ 的定义，k 由式（1c）给出，可得

$$\begin{bmatrix} \|c_{a,k+1}^{(1)} - c_{b,k+1}^{(1)}\|_2 \\ \|h_{a,k+1}^{(1)} - h_{b,k+1}^{(1)}\|_2 \\ \vdots \\ \|c_{a,k+1}^{(L)} - c_{b,k+1}^{(L)}\|_2 \\ \|h_{a,k+1}^{(L)} - h_{b,k+1}^{(L)}\|_2 \end{bmatrix} \leqslant \mathfrak{A} \begin{bmatrix} \|c_{a,k+1}^{(1)} - c_{b,k+1}^{(1)}\|_2 \\ \|h_{a,k+1}^{(1)} - h_{b,k+1}^{(1)}\|_2 \\ \vdots \\ \|c_{a,k+1}^{(L)} - c_{b,k+1}^{(L)}\|_2 \\ \|h_{a,k+1}^{(L)} - h_{b,k+1}^{(L)}\|_2 \end{bmatrix} + \mathfrak{B} \|u_{a,k} - u_{b,k}\|_2, \tag{13a}$$

where

$$\mathfrak{A} = \begin{bmatrix} \mathfrak{A}^{(1)} & 0_{2,2} & \ldots & 0_{2,2} \\ \mathfrak{B}^{(2)} \tilde{\mathfrak{A}}^{(1)} & \mathfrak{A}^{(2)} & \ldots & 0_{2,2} \\ \vdots & & \ddots & \vdots \\ \mathfrak{B}^{(L)} (\prod_{l=L-1}^{2} \tilde{\mathfrak{A}}^{(l)}) \mathfrak{A}^{(1)} & \mathfrak{B}^{(L)} (\prod_{l=L-1}^{3} \tilde{\mathfrak{A}}^{(l)}) \mathfrak{A}^{(2)} & \ldots & \mathfrak{A}^{(L)} \end{bmatrix}, \tag{13b}$$

and

$$\mathfrak{B} = \begin{bmatrix} \mathfrak{B}^{(1)} \\ \mathfrak{B}^{(2)}\tilde{\mathfrak{B}}^{(1)} \\ \vdots \\ \mathfrak{B}^{(L)}\left(\prod_{l=L-1}^{1}\tilde{\mathfrak{B}}^{(l)}\right) \end{bmatrix}, \tag{13c}$$

其中，矩阵 $\tilde{\mathfrak{A}}^{(l)}$ 和 $\tilde{\mathfrak{B}}^{(l)}$ 定义为 $\tilde{\mathfrak{A}}^{(l)} = [0,\ 1]\mathfrak{A}^{(l)}$ 以及 $\tilde{\mathfrak{B}}^{(l)} = [0,\ 1]\mathfrak{B}^{(l)}$。对式（13）进行迭代，可得

$$\begin{bmatrix} \|c_{a,k}^{(1)} - c_{b,k}^{(1)}\|_2 \\ \|h_{a,k}^{(1)} - h_{b,k}^{(1)}\|_2 \\ \vdots \\ \|c_{a,k}^{(L)} - c_{b,k}^{(L)}\|_2 \\ \|h_{a,k}^{(L)} - h_{b,k}^{(L)}\|_2 \end{bmatrix} \leqslant \mathfrak{A}^k \begin{bmatrix} \|c_{a,0}^{(1)} - c_{b,0}^{(1)}\|_2 \\ \|h_{a,0}^{(1)} - h_{b,0}^{(1)}\|_2 \\ \vdots \\ \|c_{a,0}^{(L)} - c_{b,0}^{(L)}\|_2 \\ \|h_{a,0}^{(L)} - h_{b,0}^{(L)}\|_2 \end{bmatrix} + \sum_{\tau=0}^{k-1} \mathfrak{A}^{k-\tau-1}\mathfrak{B}\|u_{a,k} - u_{b,k}\|_2 \tag{14}$$

需要指出的是

$$\left\| \begin{bmatrix} \|c_{a,k}^{(1)} - c_{b,k}^{(1)}\|_2 \\ \|h_{a,k}^{(1)} - h_{b,k}^{(1)}\|_2 \\ \vdots \\ \|c_{a,k}^{(L)} - c_{b,k}^{(L)}\|_2 \\ \|h_{a,k}^{(L)} - h_{b,k}^{(L)}\|_2 \end{bmatrix} \right\|_2 = \left\| \begin{bmatrix} \|x_{a,k}^{(1)} - x_{b,k}^{(1)}\|_2 \\ \vdots \\ \|x_{a,k}^{(L)} - x_{b,k}^{(L)}\|_2 \end{bmatrix} \right\|_2 = \|x_{a,k} - x_{b,k}\|_2. \tag{15}$$

此外，矩阵 $\mathfrak{A}$ 是施尔稳定的，因为它是一个分块对角矩阵，其主对角线上各分块均为施尔稳定矩阵。因此，存在 $\lambda \in (0,\ 1)$ 和 $\mu > 0$，使得

$$\|x_{a,k} - x_{b,k}\|_2 \leqslant \mu\lambda^k\|x_{a,0} - x_{b,0}\|_2 + \|(I_{2L,2L} - \mathfrak{A})^{-1}\mathfrak{B}\|_2\|u_{a,0:k} - u_{b,0:k}\|_{2,\infty}, \tag{16}$$

也就是说，深度LSTM是 δISS。 ■

附录II 训练过程

本附录简要介绍本文采用的深度 LSTM 模型训练算法（2），该算法基于所谓的截断时间反向传播（TBPTT） [3],。有关更多详情，读者可参阅 [28]。这一过程以直观的方式解决训练问题，即寻找权重$\Phi^\star$，使其在训练数据上最小化自由运行仿真误差。总之，TBPTT 通过从实验计划期间收集的可用输入输出序列中提取 N_s 个部分重叠的子序列来实现。记这些子序列为 $\left(u_{0\ T_s:}^{\{i\}},\ y_{0\ :\ T_s}^{\{i\}}\right)$，其索引为 $i \in \mathcal{I} = \{1、\ldots\ldots、\ N_s\}$，长度固定为 T_s，这是一个设计选择。子序列集合 $\mathcal{I}$ 被划分为训练集 $\mathcal{I}_{tr}$和验证集 $\mathcal{I}_{val}$；前者用于实际训练模型，后者则用于量化网络在迭代训练过程中的建模性能。

训练过程以迭代和批处理方式进行。在每次迭代（也称为轮次）中，训练子序列集合，即 $\mathcal{I}_{tr}$，会被随机划分在 B 个子集 $\mathcal{I}_{tr}^{\{b\}}$ 中，每个子集包含 $b \in \{1, \ldots, B\}$，称为批次。然后，对于每个批次 b，损失函数

$$\mathcal{L}(\Phi) = \text{MSE}\left(\mathcal{I}_{tr}^{\{b\}}; \Phi\right) + \rho(\nu(\Phi)), \tag{17}$$

进行评估，并以符号方式计算其相对于权重的梯度，即 $\nabla_\Phi(\mathcal{L}(\Phi))$，然后执行一次梯度下降步骤。最简单的形式是按 $\Phi \leftarrow \Phi - \gamma\nabla_\Phi(\mathcal{L}(\Phi))$更新权重，其中 γ是足够小的学习率，参见 [2]。也可以采用基于梯度动量的更高级技术，加快该过程的收敛。

注意，损失函数 (17) 由两项组成。前一项是深度LSTM在批次 $\mathcal{I}_{tr}^{\{b\}}$ 上自由运行仿真所得的均方误差（均方误差），即

$$\text{MSE}\left(\mathcal{I}_{tr}^{\{b\}}; \Phi\right) = \sum_{i \in \mathcal{I}_{tr}^{\{b\}}} \sum_{k=\tau_w+1}^{T_s} \frac{\|y_k(x_0, u_{0:k}^{\{i\}}) - y_k^{\{i\}}\|_2^2}{|\mathcal{I}_{tr}^{\{b\}}|(T_s - \tau_w)}, \tag{18}$$

其中，$\tau_w \geqslant 0$ 为洗脱期，用于适应由随机初始状态x_0导致的初始瞬态，参见 [3]。

式（17）中的第二项是经过适当设计的正则化项，用于强制满足 δISS 充分条件（8），见 [9], [28]。此类正则化函数的一个示例是分段线性函数，即

$$\rho(\nu) = \frac{1}{n_\nu}\sum_{i=1}^{n_\nu}\left[\bar{\pi}\max([\nu]_i + \varepsilon_\nu, 0) + \underline{\pi}\min([\nu]_i + \varepsilon_\nu, 0)\right], \tag{19}$$

其中，n_ν 是 $\nu(\Phi)$的元素数量，即不等式的数量，$\varepsilon_\nu > 0$ 是约束裕度，而$\bar{\pi} \gg \underline{\pi} > 0$ 表示代价系数。还可以采用更高级的正则化函数，例如广义分段函数 [28]，。当验证集上的性能，即验证指标均方误差($\mathcal{I}_{val}$； Φ)不再提升时，该迭代训练过程最终停止，并得到训练后的权重，记为 $\Phi^\star$。

最后，需要在独立测试集上对训练模型的性能进行客观量化。为此，可以采用 FIT性能指标 [%]，其定义为 [9], [28],

$$\text{FIT} = 100\left(1 - \sum_{k=\tau_w+1}^{T_{te}} \frac{\|y_k - y_k^{\{te\}}\|_2}{\|y_k^{\{te\}} - \bar{y}^{\{te\}}\|_2}\right), \tag{20}$$

其中 $y_{0\ :T_{te}}^{\{te\}}$ 表示测试数据集的输出序列，$y_k = y_k(x_0,\ u_{0\ :k}^{\{te\}})$ 表示长短期记忆模型（2）的输出，而 $\bar{y}^{\{te\}}$ 表示 $y_{0\ T_{te}:}^{\{te\}}$的平均值。请注意，FIT 指数为 100% 表示理想的建模性能。

附录III 学习BBW装置的黑箱模型

如第四节所述，已在BBW装置上进行了6次开环实验：所收集序列的一个示例如图6所示，其中前60秒的实验1 结果已报告。

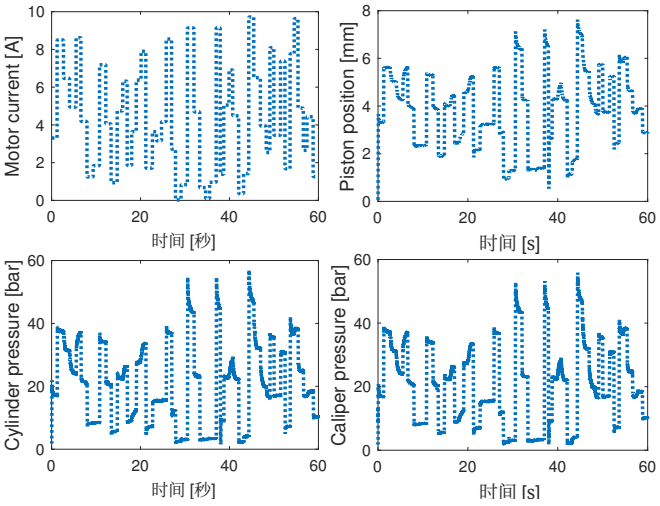


图6。实验1的部分结果：施加电流参考值（左上）、测得的活塞位置（右上）、测得的主缸压力（左下）和测得的卡钳压力（右下）。

收集这些序列后，将其归一化 [28], 并按如下方式划分：实验1 划分为 40% 验证集和 60% 测试集，而其余实验则划分为 25% 验证集和 75%训练集。然后，根据附录 II 中讨论的 TBPTT 算法，$N_{tr} = 3500$ 训练子序列和$N_{val} = 1000$ 验证子序列（长度为 T_s =200）从这些数据集中提取出来。前者收集在训练集 $\mathcal{I}_{tr}$中，后者收集在验证集 $\mathcal{I}_{val}$中。请注意，与之不同的是，测试序列不会被划分为子序列，因为它用于在训练过程结束时，通过足够长的开环仿真评估模型的建模性能。

损失函数随后定义为（17），其中 δISS约束项 $\rho(\nu(\Phi))$ 定义为（19）中的形式，其中 $\bar{\pi} = 2 \cdot 10^{-4}$， $\underline{\pi} = 10^{-6}$，以及 $\varepsilon_\nu = 0.02$。

训练过程采用RMSProp优化算法 [2], 以及由50 个训练子序列组成的批次，这意味着 $B = 70$ 批次需要在每个训练轮次中进行处理。在图7中，报告了训练损失函数（17）以及验证数据集上的均方误差指标，即MSE ($\mathcal{I}_{val}$; Φ)。整个训练过程共进行了694 轮次，并在验证指标MSE($\mathcal{I}_{val}$; Φ) 停止改善时终止，从而得到训练后的权重 $\Phi^\star$。所得深度LSTM模型满足δISS充分条件（8），其中最大值$_i[\nu(\Phi^\star)]_i < -0.09$。

致谢

本项目获得欧盟地平线2020研究与创新计划下玛丽·斯克沃多夫斯卡-居里项目资助，资助编号为 953348。

本研究部分获得意大利研究部在2017年国家利益研究项目计划（PRIN）框架下的资助，资助编号为 2017YKXYXJ。

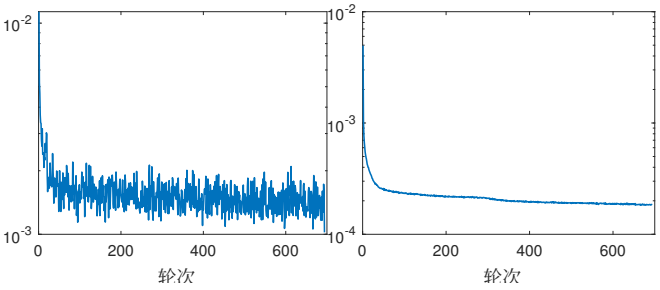


图 7. 训练过程中损失函数（左）和验证数据集上的均方误差（右）的变化。

参考文献

[1] K. J. Hunt 等，“控制系统中的神经网络——综述”，Automatica，第 28 卷，第 6 期，第 1083–1112 页，1992 年。[2] Y. Bengio、I. Goodfellow 和 A. Courville，《深度学习》。美国马萨诸塞州：麻省理工学院出版社，2017 年，第 1 卷。[3] F. M. Bianchi、E. Maiorino、M. C. Kampffmeyer、A. Rizzi 和 R. Jenssen，《用于短期负荷预测的循环神经网络：概述与比较分析》。施普林格，2017 年。[4] S. Hochreiter 和 J. Schmidhuber，“长短期记忆”，Neural Computation，第 9 卷，第 8 期，第 1735–1780 页，1997 年 11 月。[5] K. Cho 等，“使用 RNN 编码器-解码器学习统计机器翻译的短语表示”，载于《自然语言处理经验方法会议》（EMNLP 2014），2014 年。[6] R. Pascanu、T. Mikolov 和 Y. Bengio，“循环神经网络训练的困难”，载于《国际机器学习会议》。PMLR，2013 年，第 1310–1318 页。[7] N. Mohajerin 和 S. L. Waslander，“使用循环神经网络对动态系统进行多步预测”，IEEE神经网络与学习系统汇刊，第 30 卷，第 11 期，第 3370–3383 页，2019 年。[8] J. Xu、M. Kovatsch、D. Mattern、F. Mazza、M. Harasic、A. Paschke 和 S. Lucia，“智能制造中的人工智能综述：深度学习挑战与解决方案”，Applied Sciences，第 12 卷，8239，2022 年 16 月，第 p. 号。[9] F. Bonassi 、 M. Farina 、 J. Xie 和 R. Scattolini，“关于用于基于学习的控制的循环神经网络：近期成果与未来发展的思路”，Journal of Process Control，第 114 卷，104，2022 年 92pp. – 。[10] J. Schoukens 和 L. Ljung，“非线性系统辨识：面向用户的路线图”，IEEE控制系统杂志，第 39 卷，第 6 期，第 28–99 页，2019 年。[11] J. B. Rawlings、D. Q. Mayne 和 M. Diehl，《模型预测控制：理论、计算与设计》。威斯康星州麦迪逊：Nob Hill Publishing，2017 年，第 2 卷。[12] Z. Wu、A. Tran、D. Rincon 和 P. D. Christofides，“基于机器学习的非线性过程预测控制。第一部分：理论”，AIChE Journal，第 65 卷，第 11 期，2019 年。[13] A. Levin 和 K. Narendra，“使用神经网络控制非线性动力系统：可控性与稳定化”，IEEE神经网络汇刊，第 4 卷，第 2 期，第 192–206 页，1993 年。[14] E. Terzi、F. Bonassi、M. Farina 和 R. Scattolini，“使用长短期记忆网络的学习型模型预测控制”，International Journal of Robust and Nonlinear Control，第 31 卷，第 18 期，第 8877–8896 页，2021 年。[15] N. Lanzetti、Y. Z. Lian、A. Cortinovis、L. Dominguez、M. Mercang¨oz 和 C. Jones，“用于流程工业的基于循环神经网络的模型预测控制”，载于《2019 年第 18 届欧洲控制会议》（欧洲控制会议）。IEEE，2019 年，第 1005–1010 页。[16] Z. K. Nagy，“使用经优化设计的人工神经网络对酵母发酵生物反应器进行基于模型的控制”，Chemical engineering journal，第 127 卷，第 1–3 期，第 95–109 页，2007 年。[17] W. C. Wong、E. Chee、J. Li 和 X. Wang，“用于连续制药生产的基于循环神经网络的模型预测控制”，Mathematics，第 6 卷，第 11 期，第 242 页，2018 年。[18] F. Bayer、M. B¨urger 和 F. Allg¨ower，“离散时间增量输入到状态稳定性：鲁棒非线性模型预测控制的框架”，载于《2013 年欧洲控制会议》（欧洲控制会议）。IEEE，2013 年，第 2068–2073 页。

[19] F. Bonassi 和 R. Scattolini，“基于循环神经网络的未知非线性稳定系统内模控制”，《欧洲控制杂志》，第 100632 页，2022年。 [20] F. Bonassi、M. Farina 和 R. Scattolini，“NARX结构下离散时间前馈神经网络的稳定性”，载于第*19*届*IFAC*系统辨识研讨会（*SYSID 2021*），2021年。 [21] F. Bonassi、M. Farina、J. Xie 和 R. Scattolini，“一种针对由神经NARX模型学习的系统的无偏差非线性模型预测控制方案”，载于第*61*届*IEEE*决策与控制会议（*CDC*），2022年。 [22] F. Bonassi、E. Terzi、M. Farina 和 R. Scattolini，“LSTM神经网络：输入到状态稳定性与概率安全验证”，载于动力学与控制学习。PMLR，2020年，第85–94页。 [23] F. Bonassi、M. Farina 和 R. Scattolini，“门控循环单元神经网络的稳定性性质”，《系统与控制快报》，第157卷，第105049页，2021年。 [24] F. Bonassi、C. F. Oliveira da Silva 和 R. Scattolini，“基于GRU神经网络学习的系统的无偏差跟踪非线性模型预测控制”，载于第*3*届*IFAC*非线性系统建模、辨识与控制会议（*MICNON 2021*），2021年。 [25] D. Angeli，“增量稳定性性质的李雅普诺夫方法”，《*IEEE*自动控制汇刊》，第47卷，421，2002 3，410第 – 页。[26] F. Todeschini、M. Corno、G. Panzani、S. Fiorenti 和 S. M. Savaresi，“运动摩托车线控制动执行器的自适应级联控制”，《*IEEE/ASME*机电一体化汇刊》，第20卷，第3期，第1310–1319页，2014年。 [27] M. Corno、F. Todeschini、G. Panzani 和 S. M. Savaresi，“赛车摩托车线控制动执行器的建模与参数辨识”，载于面向控制的建模与辨识：理论与实践。工程技术学会，2015年，第329–363页。 [28] F. Bonassi，“协调深度学习与控制理论：用于基于模型的控制设计的循环神经网络”，2023年，博士学位论文。