Robust recurrent neural network for system identification with prior knowledge

Daniel Frank[1]

[1] University of Stuttgart, Institute for Parallel and Distributed Systems,
Universitätsstraße 32 70569 Stuttgart, Germany
daniel.frank@ipvs.uni-stuttgart.de

1 Introduction

Our goal is to identify an unknown nonlinear system from input-output measurements where we assume to have partial knowledge about the system in the form of a linearized model. The underlying true system consists of nonlinear differential equations as well as some unknown environmental effects. We have access to a linear model that approximates the nonlinear differential equation and a set of input and output measurements. We use a recurrent neural network (RNN) to compensate for the error between the linearized model and the true system and train it by the input-output measurements. The RNN can be seen as a linear, time-invariant (LTI) system with nonlinear disturbance, which allows us to give system theoretic guarantees on the learned input-output behavior. To give such guarantees the parameters are constrained which leads to worse prediction accuracy when compared to unconstrained neural networks [1]. By incorporating prior knowledge in the neural network our hypothesis is to improve prediction accuracy and still be able to give rigorous guarantees. The input-output measurements are from the set $\mathcal{D} := \{(w_p, z_p)_i\}_{i=0}^{N}$ where the input and output is a sequence denoted by w_p and z_p respectively, an element at time step $k = 1, \dots, T$ is denoted as $w_p^k \in \mathbb{R}^{n_{w_p}}$, $z_p^k \in \mathbb{R}^{n_{z_p}}$.

We further assume to know a linear model

$$\begin{pmatrix} x_{\text{lin}}^{k+1} \\ \hat{w}_p^k \end{pmatrix} = \begin{pmatrix} A_{\text{lin}} & B_{\text{lin}} \\ I & 0 \end{pmatrix} \begin{pmatrix} x^k \\ z_p^k \end{pmatrix}, \tag{1}$$

where the state x_{lin}^k is observed at the output y^k. The state matrix $A_{\text{lin}} \in \mathbb{R}^{n_x \times n_x}$ and the input matrix $B_{\text{lin}} \in \mathbb{R}^{n_x \times n_{w_p}}$ are an approximation of the true nonlinear system. A LTI system in feedback interconnection with a static nonlinearity is a generalized version of a RNN and can be described as:

$$\begin{pmatrix} x_{\text{rnn}}^{k+1} \\ u^k \\ z_u^k \end{pmatrix} = \begin{pmatrix} A & B_1 & B_2 \\ C_1 & D_{11} & D_{21} \\ C_2 & D_{21} & 0 \end{pmatrix} \begin{pmatrix} x_{\text{rnn}}^k \\ y^k \\ w^k \end{pmatrix} \tag{2a}$$

$$w^k = \Delta(z^k), \tag{2b}$$

where $\Delta := \mathbb{R}^{n_z} \mapsto \mathbb{R}^{n_z}$ is a static nonlinearity. We neglect the D_{22} block in (2a) to avoid a direct dependency between z^k and w^k. For $A = B_1 = C_1 = D_{11} = D_{22} = 0$, $B_2 = I$, $\Delta(\cdot) = \tanh(\cdot)$ and $h^{t-1} = x_{\text{rnn}}^k$ we recover the standard RNN known from the deep learning literature [2].

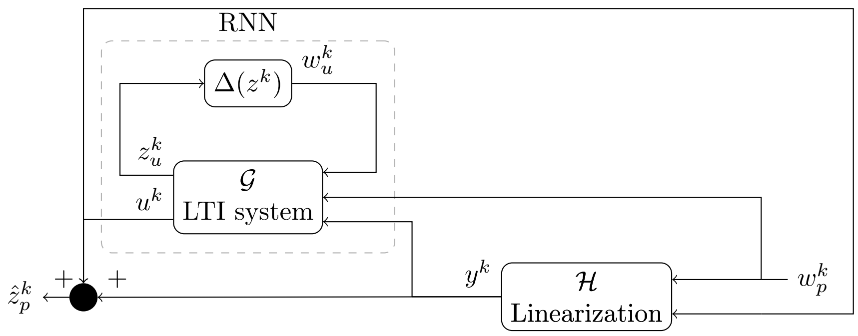

Figure 1: Interconnected RNN with linearization an unknown system.

The RNN (2) should learn the difference between the linearized model and the true system, the output u can therefore be seen as an error between the prediction $\hat{w}_p$ and the true ground truth output w_p.

We evaluate the interconnection shown in Figure 1, which allows us to incorporate prior knowledge with a recurrent neural network, on a four-degrees-of-freedom ship motion model [3]

2 Related work

Output feedback synthesis was introduced in [4] and recently connected to RNNs in [5, 6]. In [5] a RNN was used to control an unstable equilibrium point of a linear system. It was shown that the interconnection could stabilize the system and guarantee stability. In a follow-up work [6] the authors extended the RNN to equilibrium networks.

RNNs are analyzed in [7] for their finite stability gain and an H_2-gain, which refers to a worst-case amplification and the performance to white noise input sequences respectively.

References

[1] D. Frank, D. A. Latif, M. Muehlebach, and S. Staab, "Robust recurrent neural network to identify ship motion in open water with performance guarantees–technical report," *arXiv preprint arXiv:2212.05781*, 2022.

[2] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning.* MIT press, 2016.

[3] A. Baier and S. Staab, "A Simulated 4-DOF Ship Motion Dataset for System Identification under Environmental Disturbances," 2022.

[4] C. Scherer, P. Gahinet, and M. Chilali, "Multiobjective output-feedback control via lmi optimization," *IEEE Transactions on automatic control*, vol. 42, no. 7, pp. 896–911, 1997.

[5] F. Gu, H. Yin, L. El Ghaoui, M. Arcak, P. Seiler, and M. Jin, "Recurrent neural network controllers synthesis with stability guarantees for partially observed systems," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, pp. 5385–5394, 2022.

[6] N. Junnarkar, H. Yin, F. Gu, M. Arcak, and P. Seiler, "Synthesis of stabilizing recurrent equilibrium network controllers," in *2022 IEEE 61st Conference on Decision and Control (CDC)*, pp. 7449–7454, IEEE, 2022.

[7] P. Pauli, J. Berberich, and F. Allgöwer, "Robustness analysis and training of recurrent neural networks using dissipativity theory," *at-Automatisierungstechnik*, vol. 70, no. 8, pp. 730–739, 2022.