TOWARDS NEURAL NETWORK BASED CONTROL WITH GUARANTEES — APPLICATION TO A CHEMICAL REACTOR

Tim Zieger[1,2], Janine Matschek[1], Hoang Hai Nguyen[1], Thimo Oehlschlägel[2], Anton Savchenko[1], and Rolf Findeisen[1*]

[1] Laboratory for Systems Theory and Automatic Control, Otto von Guericke University Magdeburg, Germany

[2] Department of Powertrain Mechatronics, iav GmbH, Gifhorn, Germany

Merging machine learning and systems and control theory opens new possibilities to tackle uncertainty in an increasingly digitalized world where data and measurements become widely available. However, deriving guarantees, such as stability properties, within learning supported control is a challenging task. We outline a learning supported control approach that utilizes a special type of neural networks to approximate a baseline controller based on simulated or learned data. Applying recent results from neural network theory, we illustrate how stability guarantees can be embedded in the learning to guarantee closed loop nominal stability. The approach is underlined by a simulation example of a non-isothermal continuously stirred tank reactor.

Keywords: Process control, neural networks, model predictive control, stability, chemical process.

Introduction

The design of controllers for many chemical and biotechnological processes can be highly involved. Often the underlying dynamics are complicated, contain nonlinearities, interconnections and coupling of multiple subsystems, as well as lower level controllers. Despite these challenges provable performance and stability properties should be achieved. Machine learning techniques, specifically neural networks, recently gained popularity as valuable tools for both system modeling and control in many fields of applications, including process systems engineering, see e.g. (Akesson et al. 2005, Kittisupakorn et al. 2009, Lu and Tsai 2008, Mohanty 2009). For example, neural network controllers allow to approximate normal regulator designs, which could either be too expensive for real time application or insufficiently flexible for changing environmental conditions. However, providing (nominal) stability results for such controllers is in general challenging. In (Nguyen et al., 2019) ideas to achieve stability of the closed loop system using neural networks as controllers have been outlined. To do so, a special type of neural networks (Ciccone et al., 2018) is employed, supported by either Lyapunov functions or the small gain theorem to provide convergence to an invariant set or input-output stability, respectively. In the frame of this paper we outline how these results can be applied towards problems from chemical engineering. Specifically we apply the developed approach to a nonlinear chemical reactor operated at an unstable equilibrium point.

Basic idea and concept

We briefly outline the results and ideas presented in (Nguyen et al., 2019) to provide a learning procedure that allows for stability analysis and defines criteria how to obtain such.

* rolf.findeisen@ovgu.de

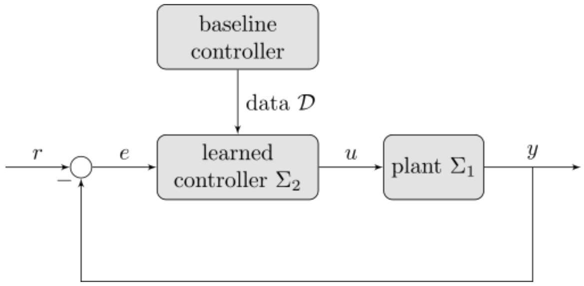

Fig. 1. Block diagram of a learning based controller operated in closed loop. Following along the lines of (Nguyen et al., 2019) stability of the closed loop can be achieved.

Consider the continuous-time nonlinear system

$$\Sigma_1: \dot{x}(t) = f\big(x(t), u(t)\big),\ x(t_0) = x_0, \quad y(t) = h\big(x(t), u(t)\big), \tag{1}$$

where $x \in R^n$, $u \in R^m$ denote the system states and inputs. Besides full state information, we assume that a baseline controller κ_b, which provides desirable closed-loop system performance (or corresponding measurement data), is given. The data set D used to train the neural network is provided in form of the tuples of control errors $e(t) = y(t) - r(t)$, with $r(t)$ being the reference signal and the corresponding inputs $u_b(t)$ generated by the baseline controller. The neural network is trained such that

$$\Sigma_2: u(t) = \kappa(e(t)) \simeq \kappa_b(e(t)),$$

c.f. Fig. 1. It is not compulsory that the baseline controller is theoretically proven to stabilize the system, rather we aim for proving closed loop stability under the neural network controller. The structure of the neural network we employ is a Non-Autonomous Input-output Stable Network (NAIS-Net) (Ciccone et al. 2018), which is a special type of a residual neural network. It comprises one or several blocks, where an input is persistently applied to all layers within each block. In this work, we use one block structure and the hyperbolic tangent (tanh) as nonlinear activation function. The latent states z of each layer i of NAIS-Net are passed through to the next layer, making the propagation function have the form $z(i+1) = z(i) + \sigma(Wz(i) + Hv + b)$, with the bias $b \in R^{n_z}$, the persistent input $v \in R^{n_v}$, as well as the state and input transfer matrix $W \in R^{n_z \times n_z}$ and $H \in R^{n_z \times n_v}$ (n_z and n_v are the number of nodes and the dimension of the input, respectively). The nonlinear mapping $\sigma: R^{n_z} \to R^{n_z}$ is a vector of element-wise hyperbolic tangent functions. The output of the network for a finite number of hidden layers N is $\phi = \Upsilon z(N) + b_N$, where $b_N \in R^{n_\phi}$ is the bias of the output layer, n_ϕ is the number of outputs and $\Upsilon \in R^{n_\Phi \times n_z}$ is the output transfer matrix.

Closed-loop System Stability via a Lyapunov function

Linearizing the system (1) around its equilibrium point and given a known reference r, the closed loop system with a NAIS-Net controller with finite number of layers becomes

$$\dot{x} = (A - B\Upsilon W^{-1}H)x + B\delta(x) + B(b_N - \Upsilon W^{-1}b). \tag{2}$$

Hereby, $A \in R^{n\times n}, B \in R^{n\times m}$ are the linearized time invariant system matrix and input matrix, respectively. The term $\delta(x) = \Upsilon(z(N) - \bar{z})$ is bounded by an affine function with slope $\beta \geq \bar{\rho}^N ||\Upsilon|| ||W^{-1}H||$ and $\bar{\rho}^N$ represents an intrinsic parameter of the NAIS-Net that is adjusted during training to be smaller than one. Based on those relations, there exists a maximum value β_{max} such that the system is still stable. This maximum value can be computed by $\beta_{max} = \sqrt{\alpha_{max}}$ and α_{max} is obtained by solving the following LMI

$$\begin{aligned} &\underset{\alpha, P}{\text{maximize}} && \alpha \\ &\text{subject to} && \alpha > 0, P \succ 0, \\ & && \begin{bmatrix} A_{\text{cl}}^\top P + PA_{\text{cl}} + \tau\alpha I & PB \\ B^\top P & -\tau I \end{bmatrix} < 0 \end{aligned}$$

where $\tau > 0$ is a fixed number resulting from the S-procedure (see Boyd et al., 1994). Thus, the closed loop system matrix is given by $A_{cl} = A - B\Upsilon W^{-1}H$, see also equation (2). Since the NAIS-Net is constructed such that $\bar{\rho} < 1$, β becomes smaller when we increase the number of layers. If β_{max} is known, we can choose the minimal number of layers $\underline{N}$ to satisfy $\beta < \beta_{max}$ as follows

$$N \geq \underline{N} = (\ln\beta_{max} - \ln||\Upsilon|| ||W^{-1}H||)/\ln\bar{\rho}$$

which guarantees closed loop stability.

Application Example to a non-isothermal CSTR

We apply the proposed approach to control a non-isothermal continuously stirred tank reactor with an irreversible reaction A→B. Based on standard modeling assumptions (e.g. Arrhenius reaction kinetics, stationarity of liquid volume, negligible shaft work) the nonlinear state equations representing the material and energy balances can be derived as follows:

$$V\dot{c}_A = F\big(c_{A,in} - c_A\big) - Vk_0 e^{\frac{-E}{RT}} c_A$$

$$V\rho c_p \dot{T} = \rho c_p F(T_{in} - T) - \frac{aF_c^{\,b+1}}{F_c + \frac{aF_c^{\,b}}{2\rho_c c_{pc}}}(T - T_{c,in}) - \Delta H_{rnx} V k_0 e^{\frac{-E}{RT}} c_A.$$

The state variables are the concentration of reactant A c_A and the temperature inside the reactor T, while the control input is the coolant flow rate F_c. The parameter values for the reactor volume V, flow rate F, inlet concentration $c_{A,in}$, the reaction constant k_0, activation energy E, density ρ, heat capacity c_p, inlet temperature T_{in}, the coolant temperature $T_{c,in}$, coolant density ρ_c, coolant heat capacity c_{pc}, and reaction heat ΔH_{rnx} are taken from (Marlin, T. E., 1995 Appendix C). Linearizing the nonlinear systems equations around the unstable equilibrium point $x_1 = c_{A,ss} = 1.06\,\frac{\text{kmol}}{\text{m}^3}$, $x_2 = T_{ss} = 360\text{K}$, $u = F_{c,ss} = 15\,\frac{\text{m}^3}{\text{min}}$ leads to the dynamic system and input matrices $A \approx [-1.89, -6.08e^{-2}; 1.16e^{2}, 2.62]$ and $B \approx [0; -3.265]$, respectively.

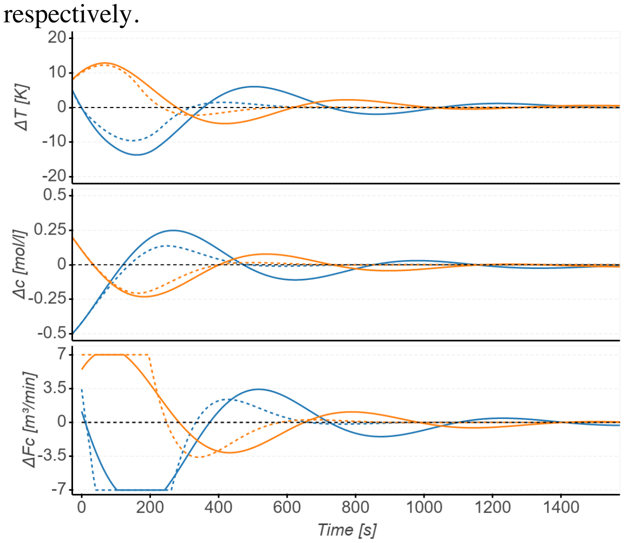

Fig. 2. Closed loop response while using the baseline controller (dotted lines) and the trained neural network controller (solid lines) with input restriction of $\pm 7\,\frac{m^3}{min}$.

The considered baseline controller is a model predictive controller. Besides its advantages (as e.g. direct consideration of constraints, multi-input-multi-output systems, nonlinear systems, ...) applying predictive control comes with a high computational burden, especially if long prediction horizons are necessary to prevent large overshooting of the system or to establish stability. We used the outlined approach based on 2^{14} data points for different initial conditions to train the network. The minimum number of required hidden layers to obtain stability is $N_{min} = 243$ with $\beta_{max} = 0.2722$ whereby each consists of 20 neurons. Note that the minimal slope is estimated by taking the minimizing state which originates in a set of equal distributed samples from the constraint state space. The neural network controlled system exhibits satisfactory and achieves provable stability, seen Fig. 2.

Conclusions

Providing stability results for learning based controllers is, even in the nominal case, challenging. Considering so-called non-autonomous input-output stable neural networks, a lower bound on the number of layers can be obtained, that allows to achieve global or local stability for linear and nonlinear controlled systems, respectively. We used these to design a neural network controller for a non-isothermal continuously stirred tank reactor, which shows nonlinear behavior. The training was based on a model predictive controller, which was not tuned to achieve provably stable behavior. Despite this and the fact that input constraints are imposed a posteriori, the learned controller achieves closed loop stability for the nominal system. Future work will consider performance criteria with respect to approximation quality and imposing such constraints during the training of the neural network. Furthermore, extensions to different network structures are under consideration.

References

Akesson, B.M., Toivonen, H.T., Waller, J.B., Nyström R.H. (2005) "Neural network approximation of a nonlinear model predictive controller applied to a pH neutralization process". In: Comp. & Chem. Eng.29.2, pp. 323–335

Boyd, S., El Ghaoui, L., Feron, E., & Balakrishnan, V. (1994) Linear Matrix Inequalities in System and Control Theory. SIAM Philadelphia, 1994.

Ciccone, M., Gallieri, M., Masci, J., Osendorfer, C., & Gomez, F. (2018). NAIS-Net: stable deep networks from non-autonomous differential equations. In Advances in Neural Information Processing Systems (pp. 3025-3035).

Nguyen H.H., Matschek, J., Zieger T., Savchenko, A., Findeisen, R. (2019). Towards Deep Learning Based Controllers with Nominal Closed Loop Stability Guarantees. Under review.

Kittisupakorn, P., Thitiyasook, P., Hussain, M. A., & Daosud, W. (2009). Neural network based model predictive control for a steel pickling process. Journal of Process Control, 19(4), 579-590.

Lu, C. H., & Tsai, C. C. (2008). Adaptive predictive control with recurrent neural network for industrial processes: An application to temperature control of a variable-frequency oil-cooling machine. IEEE Transactions on Industrial Electronics, 55(3), 1366-1375.

Marlin, T. E., (1995). Process control: designing processes and control systems for dynamic performance (Vol. 2). New York: McGraw-Hill.

Mohanty, S. (2009). Artificial neural network based system identification and model predictive control of a flotation column. Journal of Process Control, 19(6), 991-999.