Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 54-1 (2021) 696–701

A tracking performance analysis method for autonomous systems with neural networks ⋆

Attila Lelkó, Balázs Németh, Péter Gáspár

Systems and Control Laboratory, Institute for Computer Science and Control (SZTAKI), Eötvös Loránd Research Network (ELKH), Kende u. 13-17, H-1111 Budapest, Hungary.
E-mail: [attila.lelko;balazs.nemeth;peter.gaspar]@sztaki.hu

Abstract: Intelligent manufacturing and automated systems several complex control tasks must be carried out. A possible way for improving the performance level of the systems is the application of machine-learning-based agents, e.g. neural networks in the control loop. A novel challenge of these complex systems is to provide analysis and synthesis methods, with which their performance levels can be assessed. The paper proposes analysis method for tracking control systems, whose control loop contains feed-forward neural networks. Through the method the asymptotic stability and the tracking performance through decay rate are assessed. The proposed method is based on the linear approximation of the closed-loop system, and thus, a polytopic set of linear systems is resulted. Using the resulted polytopic system an analysis method based on an optimization is formed, whose result approximates the decay rate of the system. The effectiveness of the method is illustrated through a benchmark example, i.e. the torque control of a one-degree-of-freedom robotic arm.

Keywords: performance analysis, neural network, control systems, robotic arm

1. INTRODUCTION AND MOTIVATION

Increasing complexity in automated manufacturing and logistic systems leads to the application of non-conventional control methods, especially data-driven and learning-based approaches (Bukkapatnam et al. (2019); Zheng et al. (2018)). Several different approaches can be found in control theory, where the controller is realized with neural networks (Hagan and Demuth (1999)). The effectiveness of neural-network-based controllers over a traditional PD controller in the case of a robotic manipulator is investigated in He et al. (2018). Neural-network can be used for predictive control applications (Brüggemann and Possieri (2021)), it can be investigated in adaptive control problems (Haiyang et al. (2016)) or an example on neural-network-based modeling and system identification process is detailed in Zhang et al. (2021). Neural networks have also used for the control of heat pumps (Xu et al. (2020), for cyber-physical production systems (Bampoula et al. (2021)) or for water injection wells (Hassan et al. (2021)). These are just some illustrations of the large variety of neural network applications for the solution of control problems.

However, reliability questions are also arising with respect to these methods, which prevents the applications in safety critic systems. Designing a controller with a standard backpropagation algorithm might not provide theoretical guarantees on stability or performances. During the test phase the effectiveness of the trained neural network through experimental scenarios (e.g. simulations) can be illustrated. Nevertheless, if the input of the neural network significantly differs from the signals of the training set, the achieved performance level can be degraded or the stability of the system might be lost. It provides a strong motivation for developing analysis methods, with which the stability and the performance level of the controlled systems with neural networks in safety-critical systems can be verified.

A possible way for achieving guarantees is the using of Hamilton-Jacobi reachability methods, which work in conjunction with an arbitrary learning algorithm (Fisac et al. (2019)). It leads to a least restrictive, safety-preserving control law, which intervenes only when the computed safety guarantees require it, or confidence in the computed guarantees decays in light of new observations. Moreover, another way for the verification of the neural networks is based on the use of realization theory. For example, Defourneau and Petreczky (2019) proposed that the input-output behavior of a continuous-time recurrent neural network can be represented by a rational or polynomial non-linear system. The resulted nonlinear system can be used for the analysis of the neural network. The control synthesis and analysis methods of neural networks through Linear Fractional Transformation have also been provided, see e.g. Bendtsen and Trangbaek (2000, 2002). Verification

⋆ The paper funded by the National Research, Development and Innovation Office (NKFIH) under OTKA Grant Agreement No. K 135512. The research was supported by the Ministry of Innovation and Technology NRDI Office within the framework of the Autonomous Systems National Laboratory Program.
The work of Balázs Németh was partially supported by the János Bolyai Research Scholarship of the Hungarian Academy of Sciences and the ÚNKP-20-5 New National Excellence Program of the Ministry for Innovation and Technology from the source of the National Research, Development and Innovation Fund.

10.1016/j.ifacol.2021.08.081

and reachability analysis through semidefinite programming have also been provided by Fazlyab et al. (2019).

The goal of this paper is to provide an novel analysis method on stability and tracking performance of control systems, which contain feed-forward neural networks with one hidden layer. The method is based on the approximation of the neural networks in the form of polytopic set of discrete linear systems. The core of the analysis is an optimization method, whose constraints contain stability condition on the polytopic system and the result of the optimization is in correlation with the decay rate for tracking performance. The effectiveness of the analysis is illustrated by an application example on the control analysis of a robotic arm. The advantage of the proposed method is that the formulation of the approximating linear systems can be automated and the optimization method is independent from the application of the neural network.

The paper is organized as follows. The approximation method for achieving polytopic set of linear systems is proposed in Section 2. The optimization-based analysis method is proposed in Section 3. Section 4 illustrates the effectiveness of the analysis method and finally, the achievements and the future challenges of the method are concluded in Section 5.

2. METHOD FOR THE APPROXIMATION OF NEURAL-NETWORK-BASED CONTROL SYSTEMS

The aim of this section is to find an approximation of the neural-network-based control systems, which can be used in the analysis of stability and performances through model-based tools. In this section a polytopic linear representation is developed, whose elements contain the plant and the controller in connection. First, the linearization method of the neural networks is introduced. Second, the method is applied to create the linear representation of the plant and the controller. Third, the polytopic state-space representation closed-loop system is formed.

The mathematical representation of a neuron in a network is described by the form:

$$\gamma = \sigma\left(\sum_{i=1}^{\Omega} w_i \chi_i + b\right), \tag{1}$$

where γ is the output of the neuron, χ_i, $i = 1 \ldots \Omega$ are the inputs, w_i are scalar weights on the inputs and b scalar is a bias. σ represents the activation function, which can have various forms, e.g. linear, logistic or rectified linear etc. Neural networks can contain several neurons, which are ordered to layers, i.e. input, hidden and output layers. The M number of layers are connected to each other, which leads to the mathematical form of

$$\begin{aligned}\gamma_M =& f(\chi) = \sigma_M(W_M \sigma_{M-1}(W_{M-1} \ldots \\ & \sigma_2(W_2\sigma_1(W_1\chi + b_1) + b_2) \ldots + b_{M-1}) + b_M),\end{aligned} \tag{2}$$

where γ_M is the output of the network, χ is the vector of inputs, W_j, $j = 1 \ldots M$ represents the matrix of weights w_i for layer j (Hornik (1991)).

Relation (2) shows that the mathematical description of a feed-forward neural network $f(\chi)$ is highly nonlinear. The goal of the linearization is to create linear representation of the system. The linearization is based on the Taylor series expansion (M. Kraus (2008); A. Krantz (1991)). The derivative of function $f(\chi)$ in the variable of χ is formed as

$$\begin{aligned}f'(\chi) =& \sigma'_M(W_M\chi_{M-1} + b_M)\cdot \\ & W_M\sigma'_{M-1}(W_{M-1}\chi_{M-2} + b_{M-1})W_{M-1} \ldots \\ & \sigma'_1(W_1\chi + b_1)W_1,\end{aligned} \tag{3}$$

where χ_j, represents the output vector of the j^{th} layer, which is also the input vector of the $(j+1)^{th}$ layer. Since σ_j can be highly nonlinear, the derivative for given χ input can be computed. During the linearization the derivative of the network for several χ vectors must be computed, which results in a polytopic set of systems. In this paper the analysis of neural networks with one hidden layer is carried out. Thus, the representation for $M = 2$ is resulted as

$$f(\chi) = W_2\sigma(W_1\chi + b_1) + b_2, \tag{4}$$

whose derivative in χ_0 is

$$f'(\chi_0) = W_2\sigma'(W_1\chi_0 + b_1)W_1, \tag{5}$$

where σ' is a diagonal matrix, which means that the neurons operate independently from each other. In the representation (5) the form of W_1, W_2 and σ' are the same for all inputs, but χ_0 has high impact on f'. The Taylor series expansion with the consideration of the linear term is formed as

$$\gamma_M \approx f(\chi_0) + f'(\chi_0)(\chi - \chi_0). \tag{6}$$

Through the rearrangement of (6) the linear representation of the neural network for a given χ_0 is resulted, such as

$$\Delta\gamma_M \approx T(\chi_0)\Delta\chi, \tag{7}$$

where $T(\chi_0)$ is the matrix with the functions of $f(\chi_0)$, $\Delta\gamma_M = \gamma_M - f(\chi_0)$ and $\Delta\chi = \chi - \chi_0$. Thus, the result of the approximation is a linear system representation for χ_0 input. The linearization for various χ_0 leads to a polytopic system, which contains several linear representations.

In case of neural-network-based plant and controller the state-space representation of the closed-loop system is formulated as follows. The input vector of the neural network of the plant is

$$\chi_{0,p} = \left[u_k \ldots u_{k-d_{p,u}},\ y_{k-1} \ldots y_{k-d_{p,y}}\right]^T, \tag{8}$$

where $d_{p,u}, d_{p,y} > 0$ are constant values, which represent the previous elements of the control input and the plant output. The dynamics of the system using (7) is formed as

$$\Delta y = T_p(\chi_{0,p})\Delta\chi_p = T_{p,u}\Delta u_p + T_{p,y}\Delta y_p, \tag{9}$$

where Δy is the output of the plant and $T_p(\chi_{0,p})$ represents the dynamics of the system. The vector $\Delta\chi_p$ contains the vectors $\Delta u_p = \left[u_k \ldots u_{k-d_{p,u}}\right]^T$ and $\Delta y_p = \left[y_{k-1} \ldots y_{k-d_{p,y}}\right]^T$. Since Δy_p is in connection with Δy due to z^{-1} of the signal, such as $\Delta y_p = [z^{-1} \ldots z^{-d_{c,p}}]\Delta y$. The transfer function between u_k and y_k is formulated as

$$P_0(z^{-1}) = \frac{T_{p,u}[1\ \ z^{-1} \ldots z^{-d_{p,u}}]^T}{1 - T_{p,y}[z^{-1} \ldots z^{-d_{c,p}}]^T}. \tag{10}$$

The formulation of the state space representation for the controller is resulted through the same process. Its input vector is formed as

$$\chi_{0,c} = \left[r_k \dots r_{k-d_{c,r}}, y_{k-1} \dots y_{k-d_{c,y}}, u_{k-1} \dots u_{k-d_{c,u}}\right]^T, \tag{11}$$

which means that the input vector of the neural-network-based controller contains the reference signal, the measured output of the plant and the control input. In case of the reference signal the values are considered backwards, i.e. from the actual r_k value to $r_{k-d_{c,r}}$, where $d_{c,r} > 0$ is a constant. The measured signals y are considered with the past values between y_{k-1} and $y_{k-d_{c,y}}$, $d_{c,y} > 0$. The control input of the system is the output of the neural-network-based controller $u = \gamma_M$ and the past control inputs are considered as $u_{k-1} \dots u_{k-d_{c,u}}$, $d_{c,u} > 0$.

For the given $\chi_{0,c}$ the difference between actual input vector and $\chi_{0,c}$ is formed as

$$\Delta\chi_c = \begin{bmatrix}\Delta r_c\\ \Delta y_c\\ \Delta u_c\end{bmatrix} = \chi_c - \chi_{0,c}, \tag{12}$$

where $\Delta r_c, \Delta y_c$ and Δu_c vectors are related to the corresponding elements of (11). Similarly, row vector T_c can be divided into $T_c(\chi_{0,c}) = [T_{c,r},\ T_{c,y},\ T_{c,u}]$. The linear system around the linearization of $\chi_{0,c}$ is approximated in the form of $\Delta u = T_c(\chi_{0,c})\Delta\chi_c = T_{c,r}\Delta r_c + T_{c,y}\Delta y_c + T_{c,u}\Delta u_c$, where $\Delta\gamma_M = \Delta u$ is the control input vector of the system. In the resulted expression Δu and Δu_c are not independent from each other, because Δu_c contains the shifted values of Δu, such as $\Delta u_c = [z^{-1} \dots z^{-d_{c,u}}]\Delta u$, where z^{-1} represents shift operator to backwards. Δr_c and Δy_c can also express using shift operators, such as $\Delta r_c = [1\ \ z^{-1} \dots z^{-d_{c,r}}]\Delta r$, $\Delta y_c = [z^{-1} \dots z^{-d_{c,y}}]\Delta y$. Thus, the linear system has two main input sources, such as the reference signal r_k and the output of the plant y_k with the transfer functions

$$C_{r,0}(z^{-1}) = \frac{T_{c,r}[1\ \ z^{-1} \dots z^{-d_{c,r}}]^T}{1 - T_{c,u}[z^{-1} \dots z^{-d_{c,u}}]^T}, \tag{13a}$$

$$C_{y,0}(z^{-1}) = \frac{T_{c,y}[z^{-1} \dots z^{-d_{c,y}}]^T}{1 - T_{c,u}[z^{-1} \dots z^{-d_{c,u}}]^T}, \tag{13b}$$

where $C_{r,0}(z), C_{y,0}(z)$ are transfer functions, which are constant for a given $\chi_{0,c}$, but their values depend on the selection of $\chi_{0,c}$.

Finally, the through the approximation of the plant and the controller the closed-loop system is formed, which is illustrated in Figure 1. The reference signal r_k is the input of the system and the output is y_k. The transfer function of the closed-loop system between r_k and y_k for given $\chi_{0,c}, \chi_{0,p}$ is formed as

$$W_0(z^{-1}) = \frac{C_{r,0}(z^{-1})P_0(z^{-1})}{1 - C_{y,0}(z^{-1})P_0(z^{-1})}. \tag{14}$$

The resulted system (14) is an approximation of the dynamics of the controlled system, which contains neural networks. This system is valid for a given $\chi_{0,c}, \chi_{0,p}$ pair. Achieving an accurate approximation of the closed-loop system requires that the linearizaton must be carried out for several pairs of $\chi_{0,c}, \chi_{0,p}$. It results in a polytopic set of linear systems, such as

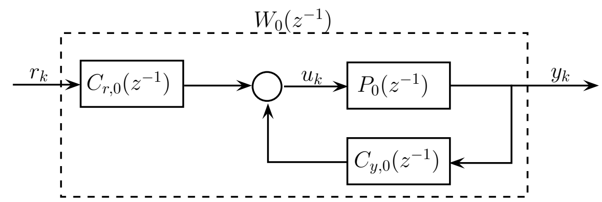

Fig. 1. Illustration of the closed-loop system

$$x_{k+1} = Ax_k + Br_k, \tag{15a}$$
$$y_k = Cx_k + Dr_k, \tag{15b}$$

where

$$A \in \mathbf{Co}\{A_1, \dots A_N\},\quad B \in \mathbf{Co}\{B_1, \dots B_N\}, \tag{15c}$$
$$C \in \mathbf{Co}\{C_1, \dots C_N\},\quad D \in \mathbf{Co}\{D_1, \dots D_N\}. \tag{15d}$$

A_i, B_i, C_i, D_i are the resulted system matrices for a given $\chi_{0,c}, \chi_{0,p}$ pair and N is the number of pairs. The matrices are resulted through the transformation of $W_0(z^{-1})$, see (14).

3. DEVELOPING AN ANALYSIS METHOD ON THE TRACKING PERFORMANCE OF THE SYSTEM

In this section an analysis method is developed to examine the tracking performance of the closed-loop system, together with the evaluation of the global exponential stability of the system. The analysis uses the state space approximation of the system, which is provided above. Thus, the examination on the neural-network-based system is transformed to an analysis problem on the approximating polytopic closed-loop system, whose elements are related to different reference signals. Due to the proposed linearization method, the polytopic representation significantly depends on the references, i.e. the values of r and the number of the linear systems. Since the stability and performance analysis is based on the linearized system, the consequences of the analysis are determined by the polytopic representation itself. Therefore, the selection of the linear systems has high impact on the analysis.

The stability and tracking performance of the system are analyzed through Lyapunov methods. It is considered that N number of linear systems are contained by the polytopic representation. The discrete polytopic system is quadratic stable if the following criteria is guaranteed:

$$A_i^T P A_i - P < 0 \qquad \forall i \in N, \tag{16}$$

where $P > 0$ positive definite matrix is the coefficient in the quadratic single Lyapunov function $V(x_k) = x_k^T P x_k$. Thus, it is necessary to find P for guaranteeing the quadratic stability of the polytopic system.

The tracking performance of the polytopic system is examined based on the decay rate α. In case of continuous systems, it can be formed as the largest Lyapunov exponent of the system (Boyd et al. (1997)) and thus, decay rate characterizes the lower bound of the convergence rate of the tracking control, such as $\lim_{t\to\infty} e^{\alpha t}||x|| = 0$. Lyapunov function $V(x)$ can be used for establish a lower bound on the decay rate. If $\frac{dV(x_i)}{dt} \leq -2\alpha V(x_i)$ for all trajectories, then $V(x_i(t)) \leq V(x_0)e^{-2\alpha t}$, which leads to $||x_i(t)|| \leq e^{-\alpha t}\sqrt{\kappa(P)}||x_i(0)||$ for all trajectories, where

$\kappa(P)$ is a function, which expresses the ratio of the largest and smallest eigenvalues of P (see Zhai et al. (2004)). The condition that $V(x_i(t)) \leq V(x_i(0))e^{-2\alpha t}$ for all trajectories is equivalent to the criteria $A^T P + PA + 2\alpha P \leq 0$, see Boyd et al. (1997) . In case of discrete time systems, the condition $||x_k|| \leq \alpha^k \sqrt{\kappa(P)}||x_0||$ for all trajectories is requested (Yong-Mei Ma and Guang-Hong Yang (2008)), which leads to the condition $A_i^T P A_i - P + \alpha P \leq 0$.

The goal of the analysis is to approximate decay rate, with which the supremum of the convergence rate of the tracking control. Thus, it is necessary to maximize α to approximate the tracking capability of the system, such as

$$\max \alpha \tag{17a}$$
subject to
$$\alpha > 0, \tag{17b}$$
$$P > 0, \tag{17c}$$
$$A_i^T P A_i - P + \alpha P \leq 0 \qquad \forall i \in N. \tag{17d}$$

The result of the analysis method is an optimization, where the largest Lyapunov exponent must be approximated. The approximation is more comprehensive as the number of the systems in the polytop N is increased. A possible way for the selection of N is the application of scenario-based techniques, with which the probability for guaranteeing the conditions can be scaled, see e.g. Calafiore (2013).

4. SIMULATION EXAMPLES FOR THE ILLUSTRATION OF THE METHOD

In this section an examples are presented, which highlights the effectiveness of the analysis method. The example is the control problem of a simplified one-degree-of-freedom rotational robotic arm. In the example the arm is able to rotate around an axis and the goal of the controller is to set the orientation of the arm through a control torque acted upon the arm. The differential equation of the system in described by

$$\ddot{y} + 2\ddot{y} + 10 \sin y = u, \tag{18}$$

where y is the angle of the arm and u is the control input torque. Since the arm placed in vertical orientation, the nonlinearity of the system is derived from the effects of gravity.

In the example the control task is carried out by a neural network, which has been trained through a supervised learning method based on various samples (Beale et al. (2019)). The objective for the neural network controller is that the arm must track the reference model

$$\ddot{y}_r + 5\ddot{y}_r + 10y_r = 10r. \tag{19}$$

The reference signal r for the training is selected as a random signal within the range $-0.7 \leq r \leq 0.7$ and the sampling interval is between $0.1s$ and $2s$. During the learning process the size of the hidden layer is selected for 13. The operation of the controlled system in the case of a random reference signal with steps is shown in Figure 2. It can be seen that the y output of the achieved closed-loop system is close to the output signal of the reference system y_r (19) and to the reference signal r.

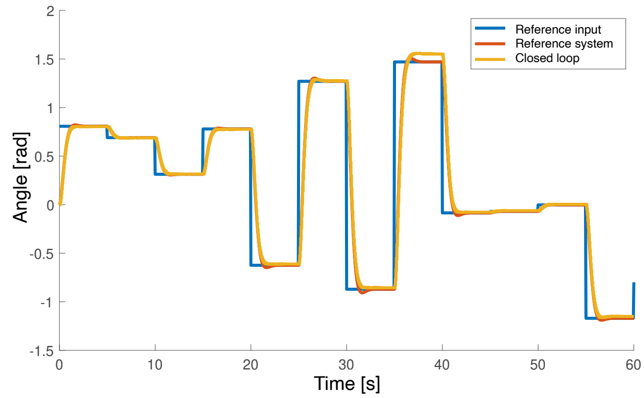

Fig. 2. Output of the designed close loop system and the output of the reference system in case of random step reference signal

Although the resulted simulation provides accurate tracking, the system is also examined using the proposed analysis to achieve more established consequences. In order to carry out the analysis the neural network have to be linearly approximated in several operating points. The way these points are selected is crucial to get desired results. This means that the operating space where these points are selected from should be as close to the operating space in the real application as possible. During training it is necessary that the distribution of the training data is similar to the real data, and this is the same in case of this validation.

A very simple example is shown here in order to point out the importance of the distribution of the operating points. In this example only the neural network plant model in considered. The exact dynamics of the plant is known in this case, the describing differential equation is (18). One way to compare the dynamic behavior of the neural network plant model with the exact nonlinear system is to compare the distribution of the dominant poles in case of the linear approximations. The poles in case of the linear approximations of the nonlinear plant depend on y the angle of the robotic arm, this dependence can be derived from the describing equation (18) of the system. The discrete poles can be computed as

$$p_{1,2}(y) = e^{(-1 \pm \sqrt{1-10\cos y})T}, \tag{20}$$

where T is sampling time of the discrete system. After determining the poles of the exact system, the trained neural network model is approximated at uniformly distributed random operating points from the operating space $u_k, u_{k-1} \in [-10, 10]$ and $y_k, y_{k-1} \in [-\pi, \pi]$. In every point the linear approximation of the network and based on that the poles can be calculated. The results can be seen in Figure 3. It can be seen that the poles of the neural network approximations are similar but not exactly coincide with the exact poles. The reason of the difference is that the operating points were chosen randomly, however during the training process the last two values of the plant output (y_{k-1} and y_k) are not independent. The difference of the two is proportional to the angular velocity of the robotic arm. In the completely random case, the maximum of this difference can be 2π an the maximal angular velocity is $2\pi/T$, which can be very large in case of small sampling

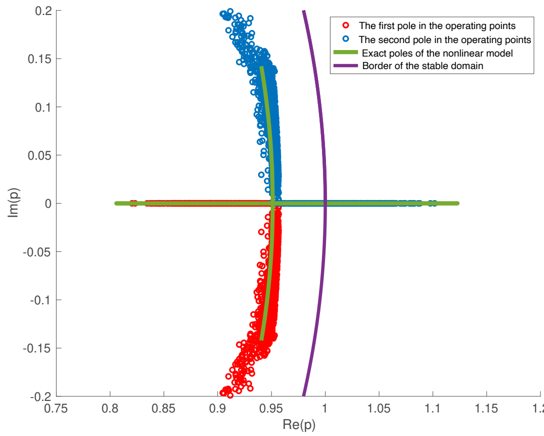

Fig. 3. The distribution of the exact plant and the neural network approximation poles in case of random operating points

times. If the neural network has not been trained with high velocity training data, it is possible that the validation fails at validation data with high velocity.

If there are strict restrictions on the operating points, the validation will result in much better approximation. If the angular velocity is chosen in the $[-10, 10]$ interval (which corresponds to the used training data) the results of the analysis can be seen in Figure 4. In this case the exact and the neural network poles are completely overlap each other which results in the similar dynamic behavior.

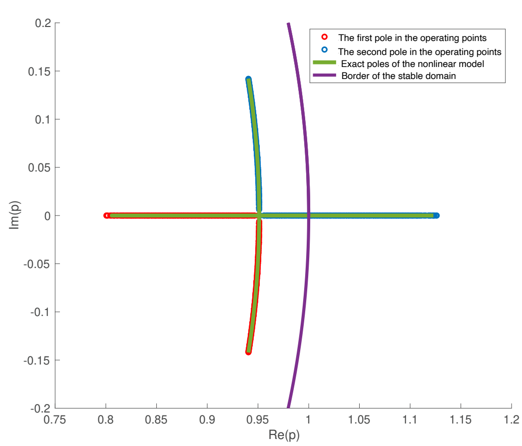

Fig. 4. The distribution of the exact plant and the neural network approximation poles in case of restricted operating space

The main point of this example is to emphasize the importance of the correct selection of the operating points which have to be in balance with the training data and the specifications. Based on these considerations the closed-loop controlled system has been linearized in several operating conditions in the operation range $r \in [-0.4, 0.4]$, $u \in [-3, 3]$, $y \in [-0.6, 0.6]$ and $\dot{y} \in [-2, 2]$. The number of the investigated points is $N = 500$ in this example.

The poles of the reference system are $2.5 \pm \frac{\sqrt{15}}{2}\mathrm{i}$ based on (19), these values as reference values are shown in Figure 5 in red and the poles resulted from the linearization of the neural network system are shown in blue. It can be seen that overall the poles of the approximations are close to the reference system which results in a similar dynamic behavior in the investigated operating domain.

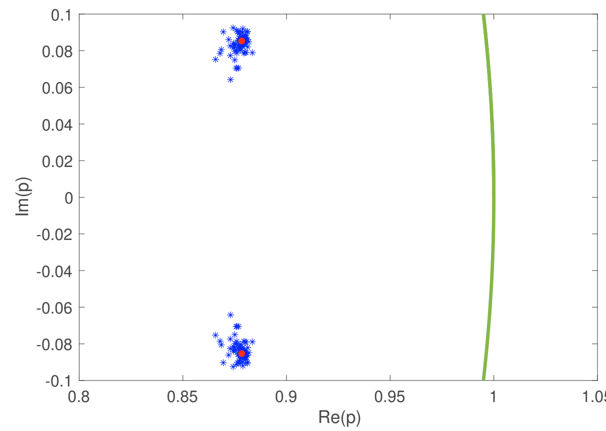

Fig. 5. Reference system poles (red), the dominant poles of the neural network approximations (blue) and the edge of the stable domain (green)

Poles contain information only about the denominator of the transfer function. It characterizes the most important dynamics of a system, but a detailed illustration can be the comparison of the Bode diagrams of the reference system to Bode diagram of the approximating systems. This comparison can be seen in Figure 6. It can be seen that the approximations result in similar Bode diagrams as in the case of the reference system. The largest difference is at high frequencies close to the sampling frequency of the discrete system.

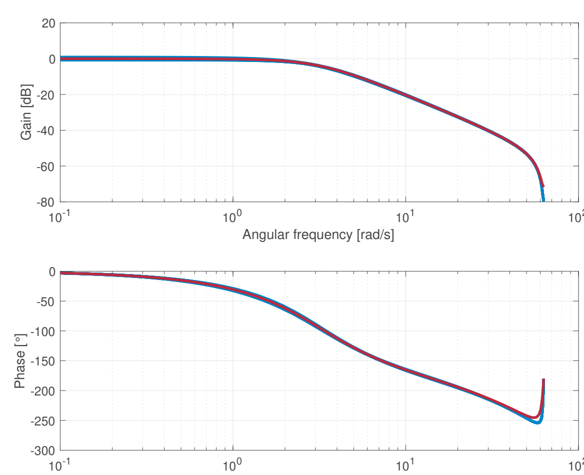

Fig. 6. The Bode diagrams of the reference system (red) and the neural network approximations (blue)

The investigation of the poles and the Bode diagrams are only valid near an operating point, however the system are continuously changing between these approximating systems. Through the polytopic representation these variations as the convex combination of the different linear approximations can be considered. In order to determine the stability and the worst-case decay rate of the system, (17) optimization problem has been solved. The optimization is resulted in a feasible solution, the approximating polytopic system is stable and the decay rate is resulted as 0.27, which is close to the decay rate is 0.34 of the reference system.

5. CONCLUSIONS

The presented examples on the analysis method illustrated that the optimization is effective for providing tracking performance evaluation of the feed-forward neural-network-based control system. The effectiveness is illustrated through an example on a robotic arm control, where the positioning of the arm must be guaranteed.

A future challenge of the method is to extend the proposed results for neural networks, which contain high number of hidden layers and feedback inside of the neural network. It requires an improved mathematical formulation of the neural network and to find connections between its mathematical description and the linearization formulas. Moreover, it is requested to extend the current method for achieving the evaluation of non-tracking types of performances and objectives in manufacturing systems.

REFERENCES

A. Krantz, X.R. (1991). *Elementary Introduction to the Theory of Pseudodifferential Operators*. New York: Routledge.

Bampoula, X., Siaterlis, G., Nikolakis, N., and Alexopoulos, K. (2021). A deep learning model for predictive maintenance in cyber-physical production systems using lstm autoencoders. *Sensors*, 21(3), 972.

Beale, M.H., Hagan, M.T., and Demuth, H.B. (2019). *Deep Learning Toolbox User's Guide*. MathWorks.

Bendtsen, J.D. and Trangbaek, K. (2000). Transformation of neural state space models into LFT models for robust control design. Technical report, Aalborg Universitet.

Bendtsen, J.D. and Trangbaek, K. (2002). Robust quasi-LPV control based on neural state-space models. *IEEE Transactions on Neural Networks*, 13(2), 355–368. doi: 10.1109/72.991421.

Boyd, S., Ghaoui, L.E., Feron, E., and Balakrishnan, V. (1997). *Linear Matrix Inequalities in System and Control Theory*. Society for Industrial and Applied Mathematics, Philadelphia.

Brüggemann, S. and Possieri, C. (2021). On the use of difference of log-sum-exp neural networks to solve data-driven model predictive control tracking problems. *IEEE Control Systems Letters*, 5(4), 1267–1272. doi: 10.1109/LCSYS.2020.3032083.

Bukkapatnam, S.T., Afrin, K., Dave, D., and Kumara, S.R. (2019). Machine learning and AI for long-term fault prognosis in complex manufacturing systems. *CIRP Annals*, 68(1), 459 – 462.

Calafiore, G.C. (2013). Direct data-driven portfolio optimization with guaranteed shortfall probability. *Automatica*, 49(2), 370 – 380. doi: https://doi.org/10.1016/j.automatica.2012.11.012.

Defourneau, T. and Petreczky, M. (2019). Realization theory of recurrent neural networks and rational systems. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, 8048–8053.

Fazlyab, M., Morari, M., and Pappas, G.J. (2019). Probabilistic verification and reachability analysis of neural networks via semidefinite programming. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, 2726–2731. doi:10.1109/CDC40024.2019.9029310.

Fisac, J.F., Akametalu, A.K., Zeilinger, M.N., Kaynama, S., Gillula, J., and Tomlin, C.J. (2019). A general safety framework for learning-based control in uncertain robotic systems. *IEEE Transactions on Automatic Control*, 64(7), 2737–2752.

Hagan, M. and Demuth, H. (1999). Neural networks for control. *Proceedings of the 1999 American Control Conference, San Diego, CA*, 1642–1656.

Haiyang, Z., Yu, S., Deyuan, L., and Hao, L. (2016). Adaptive neural network pid controller design for temperature control in vacuum thermal tests. In *2016 Chinese Control and Decision Conference (CCDC)*, 458–463. doi:10.1109/CCDC.2016.7531028.

Hassan, A., Abdulraheem, A., and Awadh, M. (2021). New approach to evaluate the performance of highly deviated water injection wells using artificial neural network. *Journal of Petroleum Science and Engineering*, 196, 107770.

He, W., Yan, Z., Sun, Y., Ou, Y., and Sun, C. (2018). Neural-learning-based control for a constrained robotic manipulator with flexible joints. *IEEE Transactions on Neural Networks and Learning Systems*, 29(12), 5993–6003.

Hornik, K. (1991). Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4(2), 251 – 257.

M. Kraus, J. Kunovský, M.P.V.t. (2008). Taylor series in control theory. In *Tenth International Conference on Computer Modeling and Simulation (uksim 2008)*.

Xu, X., Liu, J., Wang, Y., Xu, J., and Bao, J. (2020). Performance evaluation of ground source heat pump using linear and nonlinear regressions and artificial neural networks. *Applied Thermal Engineering*, 180, 115914.

Yong-Mei Ma and Guang-Hong Yang (2008). Stabilization with decay rate analysis for discrete-time linear systems subject to actuator saturation. In *2008 American Control Conference*, 1887–1892.

Zhai, G., Chen, X., and Lin, H. (2004). Stability and l_2 gain analysis for discrete-time lti systems with controller failures. *IFAC Proceedings Volumes*, 37(11), 545 – 550. 10th IFAC/IFORS/IMACS/IFIP Symposium on Large Scale Systems 2004: Theory and Applications, Osaka, Japan, 26-28 July, 2004.

Zhang, M., Wang, X., Yang, D., and Christensen, M.G. (2021). Artificial neural network based identification of multi-operating-point impedance model. *IEEE Transactions on Power Electronics*, 36(2), 1231–1235. doi: 10.1109/TPEL.2020.3012136.

Zheng, P., Wang, H., Sang, Z., Zhong, R.Y., Liu, Y., Chao Li and, K.M., Yu, S., and Xu, X. (2018). Smart manufacturing systems for Industry 4.0: conceptual framework, scenarios, and future perspectives. *Frontiers of Mechanical Engineering*, 13, 137–150.