Adaptive Control of Multivariable Networked Systems with uncertain Time Delays

Martin Steinberger, *Member, IEEE*, Martin Horn, *Member, IEEE*, and Antonella Ferrara, *Fellow, IEEE*

***Abstract*—A discrete-time adaptive control approach for uncertain linear multivariable networked systems is proposed. It is capable of dealing with unknown time-delays introduced by a communication network between plant and controller. Based on the idea of model reference adaptive control, two adaptive laws are presented to reduce the conservativeness that is usually introduced when the time-delays are unknown. The stability of the closed loop system is proven. Simulation examples provide a comparison of the proposed techniques to a non-adaptive algorithm.**

***Index Terms*—Networked Control, Time Delay System, Adaptive Control, discrete-time Sliding Mode Control.**

I. Introduction

In the emerging field of networked control, controllers are linked to plants via (wireless) communication networks. This allows to gain more flexibility but also introduces new challenges related to network protocols and controller design, see, e.g., [1], [2] for an overview. This has led to a big variety of different approaches dealing with time-delays introduced by the communication medium in a networked control system (NCS). For example, [3] uses Lyapunov-Krasovskii analysis to ensure stability of the closed loop system affected by time-delays. In [4], a prediction based scheme is used, [5] proposes the structure of filtered Smith predictors to treat the effect of time-delays. An approach using over-approximation techniques is presented in [6] to deal with time-varying network delays. A good overview of different approaches is given in [7]. In addition, simulations of NCS subject to time-delays have to be done with care as pointed out in [8].

Since sliding mode control is known to exhibit outstanding robustness properties, different steps towards networked sliding mode control were pursued in the last years. In [9], event triggered sliding mode approaches are proposed to reduce the network load. Furthermore, continuous-time sliding mode techniques are proposed for systems with input and/or output delays to achieve output tracking [10], [11] and an estimation of the associated time delay [12]. A different approach was followed in [13], [14] where discrete-time sliding mode algorithms were introduced to deal with time-varying delays and external perturbation in networked systems. This approach was extended in [15] for multi-variable systems yielding controllers that can be implemented in a spatially distributed fashion. A main ingredient for those approaches is a buffering mechanism that ensures a constant time-delay that is utilized in the controller design.

To reduce the conservatism introduced by buffering, adaptive control [16]–[18] is considered in the present paper. For a comprehensive overview of existing adaptive control methods see [19] and the references therein. Adaptive methods are applied in the context of event-triggered networked control [20], [21], where the network delay is typically not considered. In [22], the output regulation of linear systems subject to known input time delays is covered. Bounded unknown state delays can be treated by using the adaptive tracking design presented in [23]. In contrast, unknown time delays in the input channel will be considered in the present work. This goal is also followed in [24] for continuous-time systems by employing a reduction approach. However, the assumptions imposed on the input delay are very strict. This may be circumvented by using an alternative technique proposed in [25], where a multiple-model adaptive approach is followed by generating a set of adaptive controllers for all possible actuator delays. Eventually, one controller is selected based on an actual performance index.

As shown in literature, adaptive sliding mode techniques are very effective to achieve desired control performance, see, e.g., [26]–[28]. However, discrete-time versions are available to a lesser extent. They have advantages in real-world networked control systems where the sampled states are sent in discrete packets over a communication medium. The notion of ideal sliding motion is replaced by quasi-sliding modes, where the trajectories are forced to a (quasi-sliding-mode) band in a finite number of steps. Usually this is also referred to as discrete-time sliding mode, see, e.g., [29], [30]. In [30], discrete-time MRAC is used to accomplish a desired sliding behavior. A gain adapting approach using equivalent control is proposed in [31]. An adaptive sliding mode controllers design based on transfer functions is shown in [32] and [33].

In the present paper, a new adaptive algorithm based on the ideas of [34] is proposed for linear multi-input discrete-time networked systems in state space form that are subject to uncertain time-delays. The major contributions of the paper are the extension to multivariable systems in networked envi-

Manuscript received ??, 2020; This work was supported by the LEAD project “Dependable Internet of Things in Adverse Environments” funded by Graz University of Technology. The financial support by the Christian Doppler Research Association, the Austrian Federal Ministry for Digital and Economic Affairs and the National Foundation for Research, Technology and Development is gratefully acknowledged.

Martin Steinberger and Martin Horn are with the Institute of Automation and Control, Graz University of Technology, 8010 Graz, Austria (e-mail: martin.steinberger@tugraz.at; martin.horn@tugraz.at).

Martin Horn is with the Christian Doppler Laboratory for Model Based Control of Complex Test Bed Systems, Institute of Automation and Control, Graz University of Technology, 8010 Graz, Austria.

Antonella Ferrara is with the Dipartimento di Ingegneria Industriale e dell’Informazione, University of Pavia, 27100 Pavia, Italy (e-mail: antonella.ferrara@unipv.it).

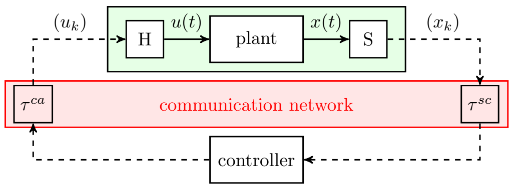

Fig. 1. Networked control system: a continuous-time plant is connected to a controller via a communication network that is subject to unknown delays. S represents a sample element, H a zero-order hold element.

ronments and a novel stabilizing adaptive control algorithm. The proposed control approach consists of the following main parts:

(i) First, a virtual output variable is designed. From a sliding mode perspective, this output can be regarded as the so-called sliding variable. Two ways how to design this virtual output for unknown but time-invariant multi-input networked systems are presented and compared - the stabilization and the reference tracking approach. Consequently, the dynamics can be split into dynamics related to the virtual output and the remaining dynamics.

(ii) In a second step, adaptive laws are proposed to force the virtual output to zero and, as a consequence, to stabilize the closed feedback loop that is subject to unknown time-delays in the communication network. This can be achieved by exploiting model reference adaptive control for the previously designed virtual output. Hence, the adaptive laws operate on a space of reduced dimension which allows computationally cheap implementations.

(iii) The stability of the closed loop consisting of multi-input plant, communication network, controller and adaptation laws, is proven. A comparison between the presented adaptive control approaches and a non-adaptive algorithm is conducted using simulation examples to highlight the achieved properties.

II. Problem Statement

Consider a continuous-time linear multi-input plant

$$\frac{dx(t)}{dt} = \tilde{A}x(t) + \tilde{B}u(t) \tag{1}$$

with states $x(t) \in \mathbb{R}^n$ and input vector $u(t) \in \mathbb{R}^m$ at time t, and constant system matrices $\tilde{A} \in \mathbb{R}^{n\times n}$, $\tilde{B} \in \mathbb{R}^{n\times m}$ that might be partially unknown due to some uncertain parameters or completely unknown. The plant state is sampled with a constant sampling period h resulting in a sequence (x_k) with elements $x_k = x(kh)$ for $k \in \mathbb{N}$. The states are sent over a network with delay τ^{sc} to a controller as shown in Fig. 1. The calculated control signals are forwarded to the zero-order hold element at the plant through the network that introduces an additional delay τ^{ca}. Since the arrival times of the packets (u_k) do not have to coincide with the sampling times, controller and zero-order hold element operate in an event-triggered way.

Assumption 1 (Network): The overall time delay $\tau = \tau^{sc} + \tau^{ca} \in \mathbb{R}$ is constant but unknown. It is upper and lower bounded, i.e. $\underline{\tau} \leq \tau \leq \overline{\tau}$, with $\overline{\tau} > h$ (large delay case).

Remark 1: Assumption 1 is reasonable because of the existence of a minimum time for transmission of data packets in real networks due to physical constraints and the used protocols. Unstable plants should be controlled over reliable networks, e.g., using WirelessHART [35] that ensures constant delays for a specific task class. The delay only changes if, e.g., additional tasks with higher priority are started or stopped.

Remark 2: It is possible to combine both delays to an equivalent round-trip delay τ due to the fact that they are assumed to be constant. In the case of variable-time delays, one would have to impose additional assumptions on the change rate of the delays as pointed out in [7].

Due to the upper bound of the communication delay, one can combine the discretized plant and the network as explained in detail in [7]. This yields $x_{k+1} = e^{\tilde{A}h}x_k + M_0(\tau)\,u_{k-\overline{d}} + M_1(\tau)\,u_{k-\overline{d}+1} + \ldots + M_{\overline{d}-1}(\tau)\,u_{k-1} + M_{\overline{d}}(\tau)\,u_k$ with matrices M_j depending on the unknown network delay τ, where $\overline{d} = \lceil \overline{\tau}/h \rceil$ represents the smallest integer larger than $\overline{\tau}/h$. Hence, a model of the networked plant can be stated as

$$v_{k+1} = \mathcal{A}v_k + \mathcal{B}u_k \qquad \text{with} \tag{2a}$$

$$\mathcal{A} = \begin{bmatrix} e^{\tilde{A}h} & M_{\overline{d}-1} & M_{\overline{d}-2} & \cdots & M_2 & M_1 & M_0 \\ 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & I_m & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & I_m & \ddots & 0 & 0 & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \ddots & I_m & 0 & 0 \\ 0 & 0 & 0 & \cdots & 0 & I_m & 0 \end{bmatrix}, \tag{2b}$$

$$\mathcal{B} = \begin{bmatrix} M_{\overline{d}}^T & I_m & 0 & 0 & \cdots & 0 & 0 \end{bmatrix}^T, \tag{2c}$$

and augmented state vector

$$v_k = \begin{bmatrix} x_k^T & u_{k-1}^T & u_{k-2}^T & \cdots & u_{k-\overline{d}}^T \end{bmatrix}^T \in \mathbb{R}^{n+m\overline{d}}, \tag{3}$$

which combines the plant states and the old control signals. Constant I_m symbolizes the identity matrix with dimension m. System matrix $\mathcal{A} \in \mathbb{R}^{(n+m\overline{d})\times(n+m\overline{d})}$ and input matrix $\mathcal{B} \in \mathbb{R}^{(n+m\overline{d})\times m}$ are constant but unknown because of the unknown network delay, see Assumption 1. Note that the matrix dimensions increase with the maximal admissible network delay $\overline{d}$. The main goal is to design a control algorithm that

(a) allows plant states x_k to track given reference values $\hat{x}_k$ under the presence of unknown time-delays induced by the network and unknown system data $\tilde{A}$ and $\tilde{B}$;

(b) minimizes the computational complexity and the corresponding number of adaptation laws to a minimum by splitting the dynamics into two parts;

(c) ensures stability of the closed loop system for all times.

Assumption 2 (Plant): The pair $(\tilde{A}, \tilde{B})$ is controllable. Sampling is done with a non-pathological sampling period h in the sense of [36] so that the full rank of the input matrix is preserved during discretization.

Remark 3: Assumption 2 ensures that the remaining dynamics can be specified arbitrarily by the design of the corresponding virtual output, see also Proposition 1.

III. Adaptive Tracking

A control algorithm based on the idea of model reference adaptive control [16], [18], [37] for multi-input NCS (2) is introduced in this section. Hence, the augmented state vector (3) is rearranged such that

$$z_k = \begin{bmatrix} z_{1,k}^T & z_{2,k}^T \end{bmatrix}^T \quad \in \mathbb{R}^{n+m\overline{d}} , \tag{4}$$

with $z_{1,k} = \begin{bmatrix} x_k^T & u_{k-\overline{d}}^T & \cdots & u_{k-2}^T \end{bmatrix}^T \in \mathbb{R}^{n+m(\overline{d}-1)}$ and $z_{2,k} = u_{k-1} \in \mathbb{R}^m$. Using (4), model (2) turns into

$$z_{k+1} = Az_k + Bu_k \qquad \text{with} \tag{5a}$$

$$A = \begin{bmatrix} e^{\tilde{A}h} & M_0 & M_1 & M_2 & \cdots & M_{\overline{d}-2} & M_{\overline{d}-1} \\ 0 & 0 & I_m & 0 & \ddots & 0 & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \ddots & 0 & 0 \\ 0 & 0 & 0 & 0 & \ddots & I_m & 0 \\ 0 & 0 & 0 & 0 & \cdots & 0 & I_m \\ 0 & 0 & 0 & 0 & \cdots & 0 & 0 \end{bmatrix}, \tag{5b}$$

$$B = \begin{bmatrix} M_{\overline{d}} & 0 & \cdots & 0 & 0 & 0 & I_m \end{bmatrix}^T , \tag{5c}$$

where matrices M_j, with $j \in \{0, 1, \ldots, \overline{d}\}$, depend on τ.

Assumption 3 (Lower bound of delay):
The unknown network delay τ is lower bounded by $\underline{\tau} = h \leq \tau$

As a consequence of Assumption 3, $M_{\overline{d}}(\tau) = 0$ due to the structure of $M_{\overline{d}}$, see [7]. This yields a perfectly known B (5c). A generalization of [34] to the case of multiple-input systems is proposed and applied to (5) to design a tracking controller for the considered NCS. Hence, (5b) and (5c) are split to comply with (4) resulting in

$$\begin{bmatrix} z_{1,k+1} \\ z_{2,k+1} \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} z_{1,k} \\ z_{2,k} \end{bmatrix} + \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} u_k \tag{6}$$

where $B_1 = 0$ due to Assumption 3. Matrices A_{11}, A_{12}, A_{21} and A_{22} represent submatrices of A in accordance with the dimensions of (4). A virtual output variable

$$\sigma_k = Cz_k = \begin{bmatrix} C_1 & I_m \end{bmatrix} z_k \tag{7}$$

is designed by specifying matrix $C \in \mathbb{R}^{m\times(n+m\overline{d})}$ such that CB has full rank. The resulting dynamics is given by

$$\sigma_{k+1} = A_1 z_{1,k} + A_2 \sigma_k + CB\, u_k \tag{8}$$

with matrices $A_1 = A_{21} - A_{22}C_1 + C_1A_{11} - C_1A_{12}C_1$ and $A_2 = A_{22} + C_1A_{12}$. Similar to (8), the reference dynamics

$$\sigma_{M,k+1} = A_{M1} z_{1,k} + A_M \sigma_{M,k} + B_M r_k \tag{9}$$

is defined with respect to $z_{1,k}$, σ_k and, in addition, to reference r_k incorporated via a non-singular matrix B_M. From a sliding-mode perspective, a so-called discrete-time quasi-sliding motion is achieved for $A_{M1} = 0$, $A_M = 0$ and $B_M = I_m$ as, e.g., in [29] for the single-input case. This can be regarded as a non-switching discrete-time reaching law that forces the virtual output to zero in a finite number of steps. As shown in [14], non-switching reaching laws may have some advantages for networked sliding mode control. Especially, the achievable size of the resulting quasi-sliding mode band is reduced for the non-switching approach in [14], whereas the considered switching reaching law yields a larger band.

The error between actual and desired output is defined as

$$e_k = \sigma_k - \sigma_{M,k} \ . \tag{10}$$

A. Nominal control

Following the idea of MRAC [18], [34], a nominal controller is designed first. Combining (5), (7), (9) and (10) yields the error dynamics $e_{k+1} = A_M e_k + (A_1 - A_{M1})z_{1,k} + (A_2 - A_M)\sigma_k - B_M r_k + CBu_k$ that should be, in the end, independent of the states and references. This is possible by using a nominal control law $\bar{u}_k = \bar{\theta}_1 z_{1,k} + \bar{\theta}_2 \sigma_k + \bar{\theta}_3 r_k$ with

$$\bar{\theta}_1 = -(CB)^{-1}(A_1 - A_{M1}), \tag{11a}$$
$$\bar{\theta}_2 = -(CB)^{-1}(A_2 - A_M), \;\; \bar{\theta}_3 = (CB)^{-1} B_M \tag{11b}$$

that can be written in vector form as

$$u_k = \bar{\theta}\omega_k + \bar{\theta}_3\, r_k \ , \tag{12}$$

where matrix $\bar{\theta} = \begin{bmatrix} \bar{\theta}_1 & \bar{\theta}_2 \end{bmatrix} \in \mathbb{R}^{m\times(n+m\overline{d})}$ depends on the actual values of A_1, A_2 that are unknown because A is unknown. Vector $\omega_k^T = \begin{bmatrix} z_{1,k}^T & \sigma_k^T \end{bmatrix} \in \mathbb{R}^{n+m\overline{d}}$ consists of known entities. An implementable version of control law (12) is

$$u_k = \theta_k\omega_k + \bar{\theta}_3\, r_k \tag{13}$$

whereas θ_k can be seen as the sum of $\bar{\theta}$ (which is unknown due to the unknown network delay τ) and an additional part $\phi_k = \begin{bmatrix} \phi_{1,k} & \phi_{2,k} \end{bmatrix}$, i.e.

$$\theta_k = \begin{bmatrix} \theta_{1,k} & \theta_{2,k} \end{bmatrix} = \bar{\theta} + \phi_k \ . \tag{14}$$

Using (14) and (13) instead of the nominal control (12) results in the error dynamics that can be written as

$$e_{k+1} = A_M e_k + CB\,\phi_k\omega_k, \;\; e_k = W_M\left[B_M^{-1}CB\phi_k\omega_k\right] \tag{15}$$

using transfer matrix

$$W_M(z) = (zI_m - A_M)^{-1} B_M \ . \tag{16}$$

Matrices A_M and B_M represent the system matrix and input matrix of an equivalent state-space representation of (16). Notation $W_M[\cdot]$ in (15) implies that m inputs $B_M^{-1}CB\phi_k\omega_k$ result in m outputs e_k.

Assumption 4 (Properties of W_M):
Transfer matrix $W_M(z)$ is stable and diagonal with identical entries, i.e. $A_M = \mathrm{diag}_m(a_M) = a_M I_m$, $B_M = \mathrm{diag}_m(b_M)$ with $0 \leq a_M < 1$, $b_M = 1 - a_M$. Hence, the desired error dynamics exhibits no steady state error with respect to e_k.

B. Adaptive Law

In a second step, an adaptive law is designed to guarantee that e_k converges to zero.

Proposition 1: (Controllability of (A_{11}, A_{12}))
Let Assumptions 1, 2 hold. Then, the pair (A_{11}, A_{12}) is controllable for any value of delay τ.

Proof: The proof is given in Appendix A. ■

This proposition serves as a basis for the following theorem for adaptive tracking.

Theorem 1 (Adaptive tracking):
Let Assumptions 1-3 hold for the networked control system with m inputs represented by (5). The desired transfer matrices $W_M(z)$ (dimension $m \times m$) and $W_N(z)$ (dimension $n + m\overline{d} \times n + m\overline{d}$) comply with Assumption 4.

(a) The virtual output (7) is designed by selecting matrix $C = [C_1\ I_m]$ such that $(A_{11} - A_{12}C_1)$ is Schur for all delays τ corresponding to Assumptions 1 and 3.

(b) Adaptation laws

$$\theta_{k+1} = \theta_k - \epsilon_k \xi_k^T \Gamma_1 \quad \in \mathbb{R}^{m\times(n+m\bar{d})}, \tag{17a}$$

$$\Upsilon_{k+1} = \Upsilon_k + B_M^{-1} C B \epsilon_k \eta_k^T \Gamma_2 \quad \in \mathbb{R}^{m\times m} \tag{17b}$$

with

$$\xi_k = W_N(z)\,[\omega_k] \quad \in \mathbb{R}^{n+m\bar{d}}, \tag{18a}$$

$$\eta_k = W_M(z)\,[\theta_k\omega_k] - \theta_k W_N(z)\,[\omega_k] \quad \in \mathbb{R}^{m}, \tag{18b}$$

$$\epsilon_k = \frac{e_k - \Upsilon_k \eta_k}{1 + \alpha \tilde{\xi}_k^T \Gamma \tilde{\xi}_k} \quad \in \mathbb{R}^{m}, \tag{18c}$$

$$\tilde{\xi}_k = [\xi_k^T \quad \eta_k^T]^T \quad \in \mathbb{R}^{n+m\bar{d}+m}, \tag{18d}$$

$$\Gamma = \begin{bmatrix} \Gamma_1 & 0 \\ 0 & \Gamma_2 \end{bmatrix}, \qquad \begin{matrix} \Gamma_1 = \Gamma_1^T > 0 \\ \Gamma_2 = \Gamma_2^T > 0 \end{matrix} \tag{18e}$$

with $\Gamma_1 \in \mathbb{R}^{(n+m\bar{d})\times(n+m\bar{d})}$, $\Gamma_2 \in \mathbb{R}^{m\times m}$ and

$$2\alpha I_m > B_M^{-1} C B\ , \tag{19}$$

where α is a positive constant.

Then, the closed loop system consisting of the networked plant (5) and controller (13) is stable despite unknown system data A. The tracking error e_k (10) tends to zero asymptotically under the presence of an unknown network delay $\underline{\tau} \le \tau \le \overline{\tau}$. Parameters θ_k and Υ_k tend to constant values.

Proof: The proof is given in Appendix B. ∎

Remark 4: Please note that $CB = I_m$ holds due to the structure of (5c), the definition of σ_k (7), and Assumption 3. As a consequence, a strict positive real requirement as in [30] is not needed.

Remark 5: The adaptive laws are inspired by [37] and extended to multivariable systems with additional uncertain network delays under consideration of a suitably constructed virtual output σ_k.

Remark 6: Relations (17a) and (17b) are of lower dimension than classical ones, cf. [19], because only the dynamics related to the virtual output is stabilized in this approach. For $\sigma_k = 0$ for $k \ge k^\star$, the remaining dynamics is fixed by the choice of matrix $C = [C_1\ I_m]$. Matrix C_1 is designed such that $(A_{11} - A_{12}C_1)$ is Schur for all τ and, e.g., the corresponding spectral radius is minimized.

Remark 7: It is also possible to, e.g., include x_k into $z_{2,k}$ and define $z_{1,k}$ differently so that

$$z_{1,k} = \begin{bmatrix} u_{k-2}^T & u_{k-3}^T & \cdots & u_{k-\bar{d}}^T \end{bmatrix}^T \quad \in \mathbb{R}^{m(\bar{d}-1)}, \tag{20a}$$

$$z_{2,k} = \begin{bmatrix} x_k^T & u_{k-1}^T \end{bmatrix}^T \quad \in \mathbb{R}^{n+m}. \tag{20b}$$

This yields the corresponding partitioning of matrix A as

$$A_{11} = \begin{bmatrix} 0 & 0 & \cdots & 0 & 0 & 0 \\ I_m & 0 & \cdots & 0 & 0 & 0 \\ 0 & I_m & \ddots & \vdots & \vdots & \vdots \\ 0 & \ddots & \ddots & 0 & 0 & 0 \\ 0 & \ddots & \ddots & I_m & 0 & 0 \\ 0 & \cdots & \cdots & 0 & I_m & 0 \end{bmatrix},\ A_{12} = \begin{bmatrix} 0 & I_m \\ 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix},$$

$$A_{21} = \begin{bmatrix} M_{\bar{d}-2} & M_{\bar{d}-3} & \cdots & M_0 \\ 0 & 0 & \cdots & 0 \end{bmatrix},\ A_{22} = \begin{bmatrix} e^{\tilde{A}h} & M_{\bar{d}-1} \\ 0 & 0 \end{bmatrix},$$

and matrices $B_1 = 0$, $B_2 = \begin{bmatrix} M_{\bar{d}}^T & 0 \end{bmatrix}^T$ with appropriate dimensions. As a result, a modified virtual output $\sigma_k = C z_k = C_1 z_{1,k} + C_2 z_{2,k}$ with $C_1 \in \mathbb{R}^{m\times(m(\bar{d}-1))}$ and $C_2 \in \mathbb{R}^{m\times(n+m)}$ can be designed allowing to further improve the performance at the price of a more sophisticated design problem for C_1 and C_2. The adaptive laws are then employed for $s_k = C_2 z_{2,k} \in \mathbb{R}^{m\times(n+m)}$; the remaining dynamics is linked to states $\begin{bmatrix} z_{1,k}^T & s_k^{\perp T} \end{bmatrix}^T$ where $s_k^\perp = C_2^\perp z_{2,k} \in \mathbb{R}^{n\times(n+m)}$ with $C_2^\perp$ spanning the right null space of C_2.

IV. Adaptive Stabilization

In the previous section, adaptive tracking of virtual output of (5) is considered. An alternative approach is to define the virtual output with respect to a desired reference vector for the states $\hat{z}_k$ of the networked plant, i.e.

$$\sigma_k = C(z_k - \hat{z}_k) = \begin{bmatrix} C_1 & I_m \end{bmatrix}(z_k - \hat{z}_k)\ , \tag{21}$$

and aim for stabilizing $\sigma_k = 0$. Signals $\hat{z}_k$ are chosen such that they are consistent for (5) as shown, e.g., in [38]. Subsequently, the same line of calculations as in the tracking case is followed to get dynamics $\sigma_{k+1} = A_1 z_{1,k} + A_2 \sigma_k + CB\,u_k - C\hat{z}_{k+1}$ and the dynamics of error (10) as

$$\begin{aligned} e_{k+1} = & A_M e_k + (A_1 - A_{M1}) z_{1,k} + (A_2 - A_M)\sigma_k - \\ & B_M r_k + CBu_k - C\hat{z}_{k+1}\ , \end{aligned} \tag{22}$$

with matrices A_1, A_2 according to (5). The nominal control is then given by $\bar{u}_k = \bar{\theta}_1 z_{1,k} + \bar{\theta}_2 \sigma_k + \bar{\theta}_3 \hat{z}_{k+1} + \bar{\theta}_4 r_k$, $\bar{\theta}_3 = (CB)^{-1} C$, $\bar{\theta}_4 = (CB)^{-1} B_M$ and $\bar{\theta}_1$, $\bar{\theta}_2$ as in (11). Using control law

$$u_k = \theta_k \omega_k + \bar{\theta}_4\, r_k \qquad \text{with} \tag{23a}$$

$$\theta_k = \bar{\theta} + \phi_k\ ,\ \omega_k^T = \begin{bmatrix} z_{1,k}^T & \sigma_k^T & \hat{z}_{k+1}^T \end{bmatrix} \in \mathbb{R}^{2(n+m\bar{d})}, \tag{23b}$$

$\theta_k = \begin{bmatrix} \theta_{1,k} & \theta_{2,k} & \theta_{3,k} \end{bmatrix}$, $\bar{\theta} = \begin{bmatrix} \bar{\theta}_1 & \bar{\theta}_2 & \bar{\theta}_3 \end{bmatrix}$ and $\phi_k = \begin{bmatrix} \phi_{1,k} & \phi_{2,k} & \phi_{3,k} \end{bmatrix}$ yields the same structure of error dynamics (15) but with different dimensions of the matrices involved. This is stated in the following corollary that is a direct consequence of Theorem 1.

Corollary 1 (Adaptive stabilization):
Consider Theorem 1 where the control law is replaced by (21), controller (23a), $\theta_k \in \mathbb{R}^{m\times 2(n+m\bar{d})}$ and $\omega_k \in \mathbb{R}^{2(n+m\bar{d})}$ as in (23b), $\tilde{\xi}_k \in \mathbb{R}^{2(n+m\bar{d})+m}$, $\Gamma \in \mathbb{R}^{(2(n+m\bar{d})+m)\times(2(n+m\bar{d})+m)}$, and transfer matrix $W_N(z)$ of dimension $2(n+m\bar{d}) \times 2(n+m\bar{d})$. Let condition (19) hold.

Then, the closed networked control system consisting of augmented plant (5) and controller (23a) is stable despite unknown system data A. The tracking error e_k (10) tends to zero asymptotically under the presence of an unknown network delay $\underline{\tau} \le \tau \le \overline{\tau}$. Parameters θ_k, Υ_k tend to constant values.

Proof: The proof is a direct consequence of Appendix B adopted to the use of (21) and the different matrix dimensions involved. Because the remaining dynamics for $\sigma_k \to \sigma_{M,k}$ is $z_{1,k+1} = (A_{11} - A_{12}C_1) z_{1,k} + A_{12}\sigma_{M,k} + B_1 u_k + A_{12} C \hat{z}_{k+1}$, the same considerations have to be taken into account as in Theorem 1 to achieve stability. ∎

Remark 8: Note that one have to deal with a larger matrix $\theta \in \mathbb{R}^{m\times 2(n+m\bar{d})}$ in (17a) instead of $\theta \in \mathbb{R}^{m\times(n+m\bar{d})}$ in Theorem 1.

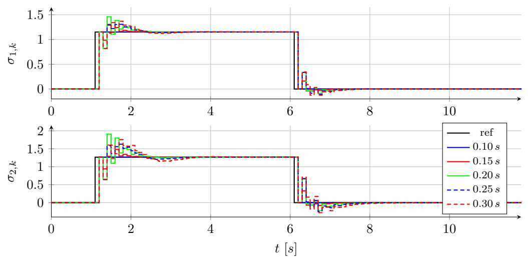

Fig. 2. Example 1: Tracking σ_k for different network delays τ

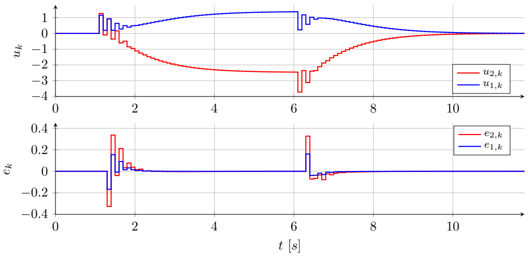

Fig. 3. Example 1: Control signals u_k and resulting error signals e_k for network delay $\tau = 0.15\,s$.

V. Simulation Examples

In this section, two examples are used to show the differences between the adaptive tracking and adaptive stabilization approach presented for NCS. First, sampling period $h = 0.1\,s$ and system data

$$\tilde{A} = \begin{bmatrix} 1.2 & -0.5 & 0.5 \\ -1.0 & -0.4 & -1.2 \\ 0.2 & -0.8 & -1.0 \end{bmatrix}, \quad \tilde{B} = \begin{bmatrix} 0.3 & 1.5 \\ -1.2 & -1.7 \\ 0.4 & 0.4 \end{bmatrix} \tag{24}$$

is used. The eigenvalues of the continuous-time plant are 1.6562, -0.1362 and -1.7200. A delay equal to the sampling period is chosen to calculate nominal values θ_{nom} and Υ_{nom}. Desired values for $x_{1,k}$ and $x_{2,k}$ are converted into references r_k for σ_k using [38], $a_M = 0$ and parameter α is chosen as $\underline{\alpha} + 1$, where $\underline{\alpha}$ corresponds to the minimal value of α given by condition (19). Matrix C_1 is designed such that $(A_{11} - A_{12}C_1)$ is Schur for all τ and the corresponding spectral radius is minimized. In addition, the largest imaginary part is penalized in the underlying optimization problem to avoid strong oscillations during parameter adaptation.

Figure 2 presents a comparison of the tracking performance for different (unknown) delays τ. Control signals u_k, errors e_k as well as the relative change of the adapted parameters $\theta_{1,k}$, $\theta_{2,k}$, Υ_k to their nominal values are shown in Fig. 3 and 4 for the case $\tau = 0.15\,s$. As shown in Theorem 1, Stability is achieved for all admissible delays, σ_k converges to its desired reference and the adaptation parameters converge to constant values. Figure 5 shows results achieved for the second algorithm that stabilizes $\sigma_k = 0$, according to Corollary 1. Figure 6 allows a comparison of the tracking approach (track σ_k, red) and the stabilization approach (stab σ_k, green) with results obtained with a basic linear time-invariant non-adaptive state controller that is shown in blue (non ad). This basic controller is designed in the same way as C_1 for the remaining dynamics for the other approaches, i.e. the spectral radius

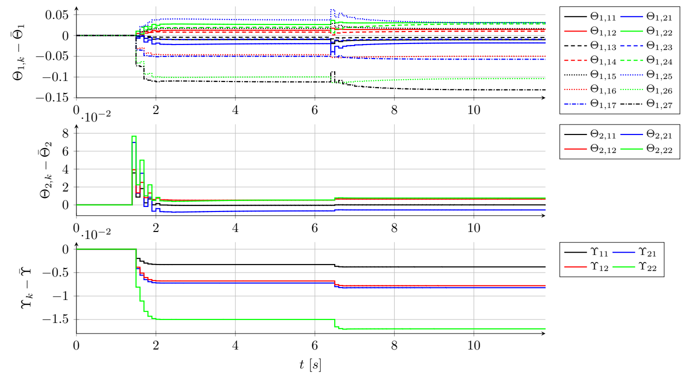

Fig. 4. Example 1: Relative changes in the adaptation parameters, i.e. $\Theta_{i,rc}$, $i \in \{1, 2\}$, and Υ_{rc} where r and c are the numbers of the corresponding rows and columns in matrices $\Theta_i = \Theta_{i,k} - \bar{\Theta}_i$ and $\Upsilon = \Upsilon_k - \bar{\Upsilon}$ for a network delay $\tau = 0.15\,s$.

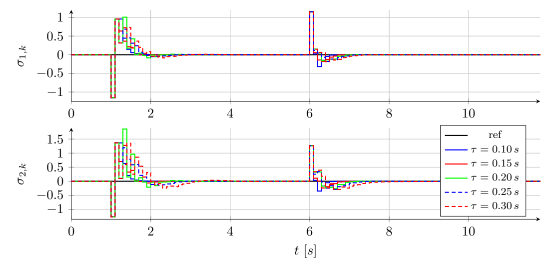

Fig. 5. Example 1: Stabilization of $\sigma_k = 0$ for different network delays τ.

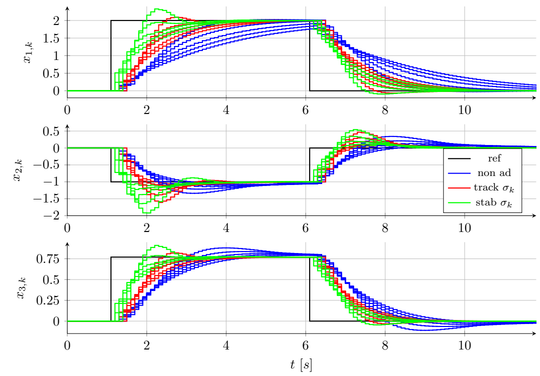

Fig. 6. Example 1: Comparison of the approaches in terms of the plant states x_k. The curves with the same color represent results for different actual network delays τ as stated in Fig. 2.

is minimized to get one controller for all possible τ that guarantees stability. Adaptive tracking results in improved performance when compared to the basic controller. Using the stabilization approach allows to further speed up the response time of the closed loop but resulting in a larger overshoot as seen in Fig. 6. The differences become clearly visible by inspecting the resulting error between desired $\hat{x}_k$ and actual plant states x_k, using measure $\sqrt{\sum (x_k - \hat{x}_k)^2}$ as well as the related control energy as shown in Table I. No specification of a maximal admissible overshoot was taken into account in the design of C_1. Please note that example 1 represents a case with moderate improvements compared to the basic controller. Figure 7 plots a representative case when the plant dynamics

TABLE I
RESULTS FOR THE NON-ADAPTIVE CONTROLLER (NON AD) IN COMPARISON TO ADAPTIVE TRACKING (TRACK) AND ADAPTIVE STABILIZATION (STAB) IN TERMS OF TRACKING ERROR AND CONTROL ENERGY.

example	example 1			example 2		
approach	non ad	track	stab	non ad	track	stab
$\sqrt{\sum(x_k-\hat{x}_k)^2}$	23.18	18.96	16.10	31.43	16.28	13.23
$\sqrt{\sum u_k^2}$	38.16	42.19	44.33	5.58	15.94	19.83

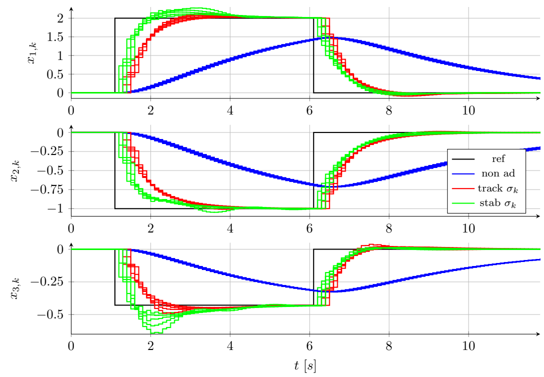

Fig. 7. Example 2: Comparison of the approaches in terms of the plant states x_k. The curves with the same color represent results for different actual network delays τ as stated in Fig. 2.

does not allow fast tuning. In this second example,

$$\tilde{A}=\begin{bmatrix}-0.5 & -1.0 & 1.1\\ -0.7 & -0.8 & -1.9\\ 0.5 & 0.1 & 1.8\end{bmatrix}, \quad \tilde{B}=\begin{bmatrix}-0.9 & 1.3\\ 0.6 & -0.5\\ 1.0 & 0.0\end{bmatrix} \tag{25}$$

with corresponding eigenvalues 2.0903, -1.3501 and -0.2402. It emphasizes that fact that, in any case, the proposed adaptive methods outperforms the basic controller that has to be designed for the worst case scenario. The related measures can be found in Table I.

VI. Summary and Outlook

Adaptive controllers for linear multivariable networked plants with constant but uncertain network-induced transmission delays are proposed. The developed controllers and adaptation laws allow to stabilize the dynamics related to a virtual output, independent of the unknown time-delay present in the network. The remaining dynamics can be influenced by a proper design ot the virtual output as outlined in the paper.

Two algorithms are introduced and compared to each other. They differ both in terms of the definition of the virtual output (that is related to the states) and with respect to the choice of the desired reference values for σ_k. Using adaptive stabilization of $\sigma_k = 0$ yields faster transient responses compared to adaptive tracking of σ_k at the price of increased control energy. Only computationally cheap manipulations are needed and no online optimization is involved. As shown by means of simulation examples, the proposed algorithms enable to considerably improve the closed loop performance when compared non-adaptive to controllers that have to be designed for worst case scenarios.

Further developments may take into account additional external perturbations and packet dropout occurring due to imperfect network communication.

Appendix A
Proof of Proposition 1

Submatrices $A_{11}\in\mathbb{R}^{\delta\times\delta}$ and $A_{12}\in\mathbb{R}^{\delta\times m}$ with $\delta = n + m(\overline{d}-1)$ in (6) for model (5) read as

$$A_{11}=\begin{bmatrix} e^{\tilde{A}h} & M_0 & M_1 & M_2 & \cdots & M_{\overline{d}-2}\\ 0 & 0 & I_m & 0 & \ddots & 0\\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots\\ 0 & 0 & 0 & 0 & \ddots & 0\\ 0 & 0 & 0 & 0 & \ddots & I_m\\ 0 & 0 & 0 & 0 & \cdots & 0\end{bmatrix}, A_{12}=\begin{bmatrix} M_{\overline{d}-1}\\ 0\\ \vdots\\ 0\\ 0\\ I_m\end{bmatrix}. \tag{26}$$

The controllability matrix $S\in\mathbb{R}^{\delta\times\delta m}$ for the pair (A_{11},A_{12}) is defined as $S = \begin{bmatrix}A_{12} & A_{11}A_{12} & A_{11}^2A_{12} & \cdots & A_{11}^{n\delta-1}A_{12}\end{bmatrix}$. Using the specific structure of (26) yields

$$\begin{aligned} A_{11}A_{12} &= \begin{bmatrix}K_1^T & 0 & \cdots & 0 & I_m & 0\end{bmatrix}^T ,\\ K_1 &= e^{\tilde{A}h}M_{\overline{d}-1} + M_{\overline{d}-2} ,\\ A_{11}^2A_{12} &= \begin{bmatrix}K_2^T & 0 & \cdots & I_m & 0 & 0\end{bmatrix}^T ,\\ K_2 &= e^{\tilde{A}h}K_1 + M_{\overline{d}-3} , \quad \ldots ,\\ A_{11}^{\overline{d}-1}A_{12} &= \begin{bmatrix}K_{\overline{d}-1}^T & 0 & \cdots & 0 & 0 & 0\end{bmatrix}^T , \qquad (27)\\ K_{\overline{d}-1} &= e^{\tilde{A}h}K_{\overline{d}-2} + M_0 . \qquad (28)\end{aligned}$$

As a result, S exhibits a very special structure such that

$$S=\begin{bmatrix} M_{\overline{d}-1} & K_1 & \cdots & K_{\overline{d}-3} & K_{\overline{d}-2} & K_{\overline{d}-1} & \tilde{K}\\ 0 & 0 & \cdots & 0 & I_m & 0 & 0\\ 0 & 0 & \iddots & I_m & 0 & 0 & 0\\ \vdots & \iddots & \iddots & \iddots & \vdots & \vdots & \vdots\\ 0 & I_m & 0 & 0 & 0 & 0 & 0\\ I_m & 0 & 0 & 0 & 0 & 0 & 0\end{bmatrix} \tag{29}$$

with $K_j = \sum_{i=0}^{j} e^{i\tilde{A}h}M_{\overline{d}-1-j+i}(\tau)$ and a matrix $\tilde{K}$ considered later. Matrices B, $e^{\tilde{A}h}$ and matrices M_j [7] are full rank, due to Assumption 2. Matrix $K_{\overline{d}-1}$ consists of the summation of different K_j multiplied by powers of $e^{\tilde{A}h}$, e.g. for $\overline{d}=5$ one gets $K_4 = M_0 + e^{\tilde{A}h}M_1 + e^{2\tilde{A}h}M_2 + e^{3\tilde{A}h}M_3 + e^{4\tilde{A}h}M_4$. For a constant τ either (i) one M_ℓ or (ii) two matrices M_ℓ and $M_{\ell-1}$ can be different from zero. In case of (i), e.g. $e^{\tilde{A}h}M_1$ with $e^{\tilde{A}h}\in\mathbb{R}^{n\times n}$ and $M_1\in\mathbb{R}^{n\times m}$ are full rank. Hence, $e^{\tilde{A}h}M_1$ has also rank m because of the rank properties of a matrix product [39], i.e. $\operatorname{rank} e^{\tilde{A}h} \operatorname{rank} M_1 - n \le \operatorname{rank}\left(e^{\tilde{A}h}M_1\right) \le \min\left\{\operatorname{rank} e^{\tilde{A}h}, \operatorname{rank} M_1\right\}$. This holds for for all ℓ in (i). In case (ii) two matrices M_ℓ and $M_{\ell-1}$ are different from zero, e.g. $M_0 + e^{\tilde{A}h}M_1$. Then, M_0 can be subtracted of the first n rows in (29) redirecting to (i). As a result, $K_{\overline{d}-1}$ has full rank, independent of τ. All remaining columns of (29) are gained by multiplications of $K_{\overline{d}-1}$ by powers of $e^{\tilde{A}h}$ such that $\begin{bmatrix}K_{\overline{d}-1} & \tilde{K}\end{bmatrix} = \begin{bmatrix}I_n & e^{\tilde{A}h} & \cdots & e^{(\delta-\overline{d})\tilde{A}h}\end{bmatrix}K_{\overline{d}-1}$. This matrix has full rank because $h\neq 0$ and $A\neq 0$ due to Assumption 2. This shows that (29) has full rank for any value of τ and so (A_{11}, A_{12}) is controllable. This completes the proof. ■

APPENDIX B
PROOF OF THEOREM 1

The proof is given in three parts: In (i), the error dynamics is modified, in (ii) it is shown that e_k tends to zero asymptotically. Finally, the properties of the remaining dynamics as well as concluding manipulations are shown in (iii). It follows the main ideas as in [34] but takes care of additional aspects that arise due to the networked multi-input system.

Part (i): An auxiliary signal $s_{A,k}$ is defined such that

$$s_{A,k} = \Upsilon_k \eta_k + \varphi_k \tag{30}$$

with time-varying matrix $\Upsilon_k \in \mathbb{R}^{m\times m}$, η_k as in (18b) and vector $\varphi_k \in \mathbb{R}^m$. It is used to stabilize error dynamics (15) utilizing an auxiliary error signal

$$\epsilon_k = e_k - s_{A,k} \ . \tag{31}$$

Combining (16), (30) and (31) results in

$$\epsilon_k = W_M \left[B_M^{-1} CB \phi_k \omega_k\right] - \Upsilon_k \eta_k - \varphi_k \ . \tag{32}$$

Next, the first term on the right hand side in (32) is investigated. Relation (15) is multiplied by $(CB)^{-1}B_M$ yielding

$$\underbrace{(CB)^{-1}B_M e_{k+1}}_{\tilde{e}_{k+1}} = A_M \underbrace{(CB)^{-1}B_M e_k}_{\tilde{e}_k} + B_M\, \phi_k \omega_k \tag{33}$$

where a new error variable $\tilde{e}_k$ such that $\tilde{e}_k = W_M[\phi_k\omega_k]$ where the structures of A_M and B_M are exploited. From the definition of $\tilde{e}_k$, see (33) with constant matrix $(CB)^{-1}B_M$, one gets $e_k = CBB_M^{-1}\tilde{e}_k = B_M^{-1}CB\, W_M[\phi_k\omega_k]$ and (32) changes into

$$\epsilon_k = B_M^{-1}CB\Big\{W_M\left[\phi_k\omega_k\right] - (CB)^{-1}B_M \Upsilon_k \eta_k\Big\} - \varphi_k \ . \tag{34}$$

Combining new variable

$$\psi_k = I_m - (CB)^{-1} B_M \Upsilon_k \tag{35}$$

with (14), (18b) and (34) yields

$$\begin{aligned}
\epsilon_k =& \ B_M^{-1}CB\Big\{W_M\left[\phi_k\omega_k\right] - W_M(z)\left[\theta_k\omega_k\right] + \\
& \theta_k W_N(z)\left[\omega_k\right] + \psi_k\eta_k\Big\} - \varphi_k \\
=& \ B_M^{-1}CB\Big\{ - W_M(z)\left[\bar{\theta}\omega_k\right] + \\
& \bar{\theta}W_N(z)\left[\omega_k\right] + \phi_k W_N(z)\left[\omega_k\right] + \psi_k\eta_k\Big\} - \varphi_k \ .
\end{aligned} \tag{36}$$

Due to the fact that $\bar{\theta}$ is constant and Assumption 4 is introduced, relation

$$W_M(z)\left[\bar{\theta}\omega_k\right] = \bar{\theta}W_N(z)\left[\omega_k\right] \tag{37}$$

holds. This can be seen by using (15) in (37) such that $\left(zI_m - A_M\right)^{-1} B_M \bar{\theta}\mathcal{Z}\{\omega_k\} = \bar{\theta}\left(zI_{n+m\bar{d}} - A_N\right)^{-1}\tilde{B}_N\mathcal{Z}\{\omega_k\}$, where $\mathcal{Z}\{\omega_k\}$ symbolizes the z-transformation applied component by component to vector ω_k. Consequently,

$$\operatorname*{diag}_{m}\left(\frac{b_M}{z-a_M}\right)\bar{\theta} = \bar{\theta}\operatorname*{diag}_{n+m\bar{d}}\left(\frac{b_M}{z-a_M}\right) \tag{38}$$

has hold for all non-square matrices $\bar{\theta} \in \mathbb{R}^{m\times n+m\bar{d}}$. By Assumption 4, the diagonal elements of W_M and W_N are equivalent, i.e. the left diagonal matrix in (38) multiplies the rows of $\bar{\theta}$ and the right diagonal matrix multiplies the columns of $\bar{\theta}$. Hence, (38) is fulfilled and (36) turns into

$$\begin{aligned}
\epsilon_k =& B_M^{-1}CB\Big\{\phi_k W_N(z)\left[\omega_k\right] + \psi_k\eta_k\Big\} - \varphi_k \\
=& B_M^{-1}CB\tilde{\phi}_k\tilde{\xi}_k - \alpha\,\tilde{\xi}_k^T\Gamma\tilde{\xi}_k\,\epsilon_k \quad \text{with}
\end{aligned} \tag{39}$$

$$\varphi_k = \alpha\,\tilde{\xi}_k^T\Gamma\tilde{\xi}_k\,\epsilon_k \ , \ \tilde{\phi}_k = \begin{bmatrix}\phi_k & \psi_k\end{bmatrix} \in \mathbb{R}^{m\times(n+m\bar{d}+m)} \tag{40}$$

and $\tilde{\xi}_k$ as stated in (18d) and Γ as (18e), cf. [34], [40].

Part (ii): An adaptation law depending on the auxiliary error ϵ_k and $\tilde{\xi}_k$ is defined as

$$\Delta\tilde{\phi}_k = \tilde{\phi}_{k+1} - \tilde{\phi}_k = -\epsilon_k\tilde{\xi}_k^T\Gamma \quad \in \mathbb{R}^{m\times(n+m\bar{d}+m)} \ . \tag{41}$$

To show stability, the squared Frobenius norm of the weighed parameter matrix $\tilde{\phi}_k$ is used as a Lyapunov function candidate, i.e. $V_k = \left\|\tilde{\phi}_k L\right\|_F^2 = \operatorname{tr}\left(\tilde{\phi}_k LL^T\tilde{\phi}_k^T\right)$, with $\Gamma^{-1} = LL^T$. Such a separation of Γ^{-1} is always possible because Γ is assumed to be positive definite. The forward difference of V_k taking into account (41) and symmetry properties of the trace operator is given by $\Delta V = V_{k+1} - V_k = \operatorname{tr}\left(\tilde{\phi}_{k+1}\Gamma^{-1}\tilde{\phi}_{k+1}^T - \tilde{\phi}_k\Gamma^{-1}\tilde{\phi}_k^T\right) = \operatorname{tr}\left(-2\epsilon_k\tilde{\xi}_k^T\tilde{\phi}_k^T + \epsilon_k\tilde{\xi}_k^T\Gamma\tilde{\xi}_k\epsilon_k^T\right)$ and using (39) yields

$$\begin{aligned}
\Delta V =& \operatorname{tr}\left(-2\frac{B_M^{-1}CB\tilde{\phi}_k\tilde{\xi}_k\left(\tilde{\phi}_k\tilde{\xi}_k\right)^T}{1+\alpha\,\tilde{\xi}_k^T\Gamma\tilde{\xi}_k} + \right. \\
& \left. \frac{B_M^{-1}CB\tilde{\phi}_k\tilde{\xi}_k\tilde{\xi}_k^T\Gamma\tilde{\xi}_k\left(\tilde{\phi}_k\tilde{\xi}_k\right)^T\left(B_M^{-1}CB\right)^T}{\left(1+\alpha\,\tilde{\xi}_k^T\Gamma\tilde{\xi}_k\right)^2}\right) .
\end{aligned} \tag{42}$$

Definitions $P_k = B_M^{-1}CB\tilde{\phi}_k\tilde{\xi}_k\left(\tilde{\phi}_k\tilde{\xi}_k\right)^T$ and $N = 2\alpha I_m - B_M^{-1}CB$ allow further simplifications resulting in

$$\Delta V = -\frac{2\operatorname{tr}\left(P_k\right)}{\left(1+\alpha\,\tilde{\xi}_k^T\Gamma\tilde{\xi}_k\right)^2} - \frac{\tilde{\xi}_k^T\Gamma\tilde{\xi}_k\operatorname{tr}\left(P_kN^T\right)}{\left(1+\alpha\,\tilde{\xi}_k^T\Gamma\tilde{\xi}_k\right)^2} \tag{43}$$

Corresponding to condition (19) in Theorem 1 and Assumption 3, $N > 0$ and symmetric because $CB > 0$ and symmetric. With that, $B_M^{-1}CB$ is split into $B_M^{-1}CB = X^TX > 0$. Exploiting the properties of the trace operator [39] yields $\operatorname{tr}\left(P_k\right) = \operatorname{tr}\left(X\tilde{\phi}_k\tilde{\xi}_k\tilde{\xi}_k^T\tilde{\phi}_k^TX^T\right) \geq 0$ and for the second term on the right hand side of equation (43) $\operatorname{tr}\left(P_kN\right) = \operatorname{tr}\left(NB_M^{-1}CB\tilde{\phi}_k\tilde{\xi}_k\left(\tilde{\phi}_k\tilde{\xi}_k\right)^T\right) = \operatorname{tr}\left(Y^TY\tilde{\phi}_k\tilde{\xi}_k\tilde{\xi}_k^T\tilde{\phi}_k^T\right) = \operatorname{tr}\left(Y\tilde{\phi}_k\tilde{\xi}_k\tilde{\xi}_k^T\tilde{\phi}_k^TY^T\right) \geq 0$. Hence, (43) gives $\Delta V \leq -\frac{2}{\left(1+\alpha\,\tilde{\xi}_k^T\Gamma\tilde{\xi}_k\right)^2}\operatorname{tr}\left(P_k\right) \leq 0$ so that $\tilde{\phi}_k$ is bounded for any bounded initial value $\tilde{\phi}_0$. V_k is non-increasing and bounded below and has, therefore, a limit V_∞ as $k \to \infty$, i.e. $\lim_{k\to\infty}\left\|\tilde{\phi}_kL\right\|_F^2 = V_\infty < \infty$, and so, the squared Frobenius norm of scaled $\tilde{\phi}_k$ converges to a limit. The difference between initial value V_0 and this limit is $\lim_{k\to\infty}\sum_{i=1}^{k}\left|\Delta V_i\right| = V_0 - V_\infty < \infty$, causing that ΔV_i has to tend to zero. Implied by this, P_k has to approach zero as can be seen in the first line of (43). According to the definition of P_k and the fact that $B_M^{-1}CB$ is non-singular, it follows that this is only possible if $\tilde{\phi}_k\xi_k$ tends to zero. Then ϵ_k is zero for $k \to \infty$, cf. (39), $\Delta\tilde{\phi}_k \to 0$ and $\tilde{\phi}_k$ tends to a constant value.

Part (iii): As a result of (ii), parameter matrix $\tilde{\theta}_k$ tends to a constant value θ_c. Combining (30), (31) and (40) gives $\epsilon_k = e_k - \Upsilon_k\eta_k - \alpha\,\tilde{\xi}_k^T\Gamma\tilde{\xi}_k\,\epsilon_k$, see (18c) in Theorem 1. In addition, $\eta_k = W_M(z)\,[\theta_k\omega_k] - \theta_k W_N(z)\,[\omega_k] \to 0$ as $\theta_k \to \theta_c$ and $\epsilon_k \to 0$. This means that the error $e_k \to 0$ for $k \to \infty$ and $\sigma_k \to \sigma_{M,k}$. The remaining dynamics reads as

$$z_{1,k+1} = (A_{11} - A_{12}C_1)z_{1,k} + A_{12}\sigma_{M,k} + B_1u_k \qquad (44)$$

where (6) and (7) are used. Matrix $(A_{11} - A_{12}C_1)$ in (44) has to be Schur for all delays to assure bounded-input bounded-state stability, which is assured by the assumptions made in the theorem. Combining (18d), (18e), (40) and (41) yields $\tilde{\phi}_{k+1} - \tilde{\phi}_k = -\epsilon_k \begin{bmatrix}\xi_k^T & \eta_k^T\end{bmatrix}\begin{bmatrix}\Gamma_1 & 0\\ 0 & \Gamma_2\end{bmatrix}$, $\phi_{k+1} - \phi_k = -\epsilon_k\xi_k^T\Gamma_1$ as in (17a) and $\psi_{k+1} - \psi_k = -\epsilon_k\eta_k^T\Gamma_2$ that is converted into (17b) using (35) so that $\Upsilon_{k+1} - \Upsilon_k = B_M^{-1}CB\,\epsilon_k\eta_k^T\Gamma_2$. This completes the proof. ■

References

[1] D. Zhang, P. Shi, Q.-G. Wang, and L. Yu, "Analysis and synthesis of networked control systems: A survey of recent advances and challenges," *ISA Transactions*, vol. 66, pp. 376–392, 2017.

[2] P. Park, S. C. Ergen, C. Fischione, C. Lu, and K. H. Johansson, "Wireless Network Design for Control Systems: A Survey," *IEEE Communications Surveys and Tutorials*, vol. 20, no. 2, pp. 978–1013, 2018.

[3] K. Liu and E. Fridman, "Networked-based stabilization via discontinuous lyapunov functionals," *International Journal of Robust and Nonlinear Control*, vol. 22, no. 4, pp. 420–436, 2012.

[4] G. P. Liu, "Predictive controller design of networked systems with communication delays and data loss," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 57, no. 6, pp. 481–485, 2010.

[5] M. Steinberger and M. Horn, "From classical to Networked Control: Retrofitting the Concept of Smith Predictors," *arXiv 2010.05486*, 2020.

[6] M. Cloosterman, L. Hetel, N. van de Wouw, W. Heemels, J. Daafouz, and H. Nijmeijer, "Controller synthesis for networked control systems," *Automatica*, vol. 46, no. 10, pp. 1584 – 1594, 2010.

[7] W. P. M. H. Heemels and N. van de Wouw, "Stability and stabilization of networked control systems," in *Networked Control Systems*, ser. Lecture Notes in Control and Information Sciences, Bemporad A. and Heemels M., Johansson M., Ed. Springer, London, 2010, vol. 406.

[8] M. Steinberger, M. Tranninger, M. Horn, and K. H. Johansson, "How to Simulate Networked Control Systems with Variable Time Delays?" *21st IFAC World Congress, IFAC-PapersOnLine*, vol. 53, no. 2, pp. 3098–3103, 2020.

[9] G. P. Incremona, A. Ferrara, and L. Magni, "Asynchronous networked MPC with ISM for uncertain nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 62, no. 9, pp. 4305–4317, 2017.

[10] C. L. Coutinho, T. R. Oliveira, and J. P. V. S. Cunha, "Output-feedback sliding-mode control via cascade observers for global stabilisation of a class of nonlinear systems with output time delay," *International Journal of Control*, vol. 87, no. 11, pp. 2327–2337, 2014.

[11] G. Liu, A. Zinober, and Y. B. Shtessel, "Second-Order SM Approach to SISO Time-Delay System Output Tracking," *IEEE Transactions on Industreal Electronics*, vol. 56, no. 9, pp. 3638–3645, SEP 2009.

[12] Y. Deng, V. Lechappe, S. Rouquet, E. Moulay, and F. Plestan, "Super-Twisting Algorithm-Based Time-Varying Delay Estimation With External Signal," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 12, pp. 10 663–10 671, DEC 2020.

[13] J. Ludwiger, M. Steinberger, M. Rotulo, M. Horn, A. Luppi, G. Kubin, and A. Ferrara, "Towards networked sliding mode control," in *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, 2017, pp. 6021–6026.

[14] J. Ludwiger, M. Steinberger, M. Horn, G. Kubin, and A. Ferrara, "Discrete time sliding mode control strategies for buffered networked systems," in *2018 IEEE 57th Annual Conference on Decision and Control (CDC)*, 2018, pp. 6735–6740.

[15] J. Ludwiger, M. Steinberger, and M. Horn, "Spatially distributed networked sliding mode control," *IEEE Control Systems Letters*, vol. 3, no. 4, pp. 972–977, 2019.

[16] G. Goodwin and K. Sin, *Adaptive Filtering Prediction and Control*. Dover, 1984.

[17] K. Aström and B. Wittenmark, *Adaptive Control*, 2nd ed. Dover, 2008.

[18] K. Narendra and A. Annaswamy, *Stable Adaptive Systems*. Dover, 2012.

[19] G. Tao, "Multivariable adaptive control: A survey," *Automatica*, vol. 50, no. 11, pp. 2737–2764, 2014.

[20] S. Al Issa, A. Chakravarty, and I. Kar, "Improved event-triggered adaptive control of non-linear uncertain networked systems," *IET Control Theory and Applications*, vol. 13, no. 13, pp. 2146–2152, SEP 3 2019.

[21] X. Zhang, Q. Han, and X. Yu, "Survey on recent advances in networked control systems," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 5, pp. 1740–1752, 2016.

[22] W. Gao and Z.-P. Jiang, "Adaptive Optimal Output Regulation of Discrete-time Linear Systems subject to Input Time-delay," in *2018 Annual American Control Conference (ACC)*, ser. Proceedings of the American Control Conference, 2018, pp. 4484–4489, American Control Conference, Milwaukee, WI, JUN 27-29, 2018.

[23] S. Li, D.-P. Li, and Y.-J. Liu, "Adaptive neural network tracking design for a class of uncertain nonlinear discrete-time systems with unknown time-delay," *Neurocomputing*, vol. 168, pp. 152–159, NOV 30 2015.

[24] K. Abidi, H. J. Soo, and I. Postlethwaite, "Discrete-time adaptive control of uncertain sampled-data systems with uncertain input delay: a reduction," *IET Control Theory and Applications*, vol. 14, no. 13, pp. 1681–1691, SEP 3 2020.

[25] C. Tan, G. Tao, and H. Yang, "A Multiple-Model MRAC Scheme for Discrete-time Multivariable Systems with Uncertain Actuation Delays," in *2018 IEEE Conference on Decision and Control (CDC)*, 2018, pp. 6283–6288.

[26] J. P. V. S. Cunha, L. Hsu, R. R. Costa, and F. Lizarralde, "Output-feedback model-reference sliding mode control of uncertain multivariable systems," *IEEE Transactions on Automatic Control*, vol. 48, no. 12, pp. 2245–2250, 2003.

[27] J. A. Moreno, D. Y. Negrete, V. Torres-Gonzalez, and L. Fridman, "Adaptive continuous twisting algorithm," *International Journal of Control*, vol. 89, no. 9, pp. 1798–1806, 2016.

[28] T. R. Oliveira, V. H. P. Rodrigues, and L. Fridman, "Generalized model reference adaptive control by means of global HOSM differentiators," *IEEE Transactions on Automatic Control*, vol. 64, no. 5, pp. 2053–2060, 2018.

[29] S. Chakrabarty and A. Bartoszewicz, "Improved robustness and performance of discrete time sliding mode control systems," *ISA Transactions*, vol. 65, pp. 143–149, 2016.

[30] G. Bartolini, A. Ferrara, and V. Utkin, "Adaptive sliding mode control in discete-time systems," *Automatica*, vol. 31, no. 5, pp. 769–773, 1995.

[31] P. Lee and J. Oh, "Improvements on VSS-type self-tuning control for a tracking controller," *IEEE Transactions on Industial Electronics*, vol. 45, no. 2, pp. 319–325, 1998.

[32] C. Chan, "Discrete adaptive sliding-mode tracking controller," *Automatica*, vol. 33, no. 5, pp. 999–1002, 1997.

[33] X. Chen, "Adaptive sliding mode control for discrete-time multi-input multi-output systems," *Automatica*, vol. 42, no. 3, pp. 427–435, 2006.

[34] M. Steinberger, M. Horn, and A. Ferrara, "Discrete-time Model Reference Adaptive Sliding Mode Control for Systems in State-Space Representation," in *IEEE Conference on Decision and Control*, 2019, pp. 6007–6012.

[35] A. Saifullah, Y. Xu, C. Lu, and Y. Chen, "End-to-End Communication Delay Analysis in Industrial Wireless Networks," *IEEE Transactions on Computers*, vol. 64, no. 5, pp. 1361–1374, 2015.

[36] R. Kalman, B. Ho, and N. Narendra, "Controllability of linear dynamical systems," *Contributions to Differential Equations*, vol. 1, pp. 198–213, 1963.

[37] K. Narendra and Y.-H. Lin, *Applications of adaptive control*. Academic Press, 1980, ch. Design of stable model reference adaptive controllers, pp. 69–130.

[38] M. Steinberger, I. Castillo, M. Horn, and L. Fridman, "Robust Output Tracking of Constrained Perturbed Linear Systems via Model Predictive Sliding Mode Control," *International Journal of Robust and Nonlinear Control*, vol. 30, no. 3, pp. 1258–1274, 2020.

[39] D. S. Bernstein, *Matrix Mathematics: Theory, Facts, and Formulas*. Princeton University Press, 2009.

[40] Y.-H. Lin and K. Narandra, "A new error model for adaptive systems," *IEEE Transactions on Automatic Control*, vol. 25, no. 3, pp. 585–587, 1980.