Unknown Input Observer Design and Analysis for Networked Control Systems

Ahmad F Taha*, Ahmed Elmahdi#, Jitesh H. Panchal‡ and Dengfeng Sun†

Abstract—**The insertion of communication networks in the feedback loops of control systems is a defining feature of modern control systems. These systems are often subject to unknown inputs in a form of disturbances, perturbations, or attacks. The objective of this paper is to design and analyze an observer for networked dynamical systems with unknown inputs. The network effect can be viewed as either a perturbation or time-delay to the exchanged signals. In this paper, we (a) review an Unknown Input Observer (UIO) design for non-networked system, (b) derive the Networked Unknown Input Observer (N_{et}UIO) dynamics, (c) design a N_{et}UIO such that the effect of higher delay order terms are nullified and (d) establish stability-guaranteeing bounds on the networked-induced time-delay and perturbation. The formulation and results derived in this paper can be generalized to scenarios and applications where the signals are perturbed due to a different source of perturbation or delay.**

Keywords*—*Networked Control Systems, Unknown Input Observers, N_{et}UIO, Time-Delay Systems.

I. Introduction

THE objective of this paper is to analyze and design an observer for Networked Control Systems (NCS) with unknown inputs. Many modern control systems are becoming networked, where often a band-limited network is used as a mean of communication between sensors, actuators and controllers [1]. Estimators in general, and observers in specific, use the known plant's inputs and outputs to generate estimates for the state of the plant. The closed loop system is then controlled through a controller that often use the estimated plant state to generate control commands.

A. Literature Review

State-estimators and observers are used in power networks to precisely estimate the plant state (i.e., the bus voltages and phase angles), which is crucial for successful control and operation of the modern smart-grid. The generated real-time dynamic estimates of the bus voltages and angles facilitate calculating optimal power flows for transmission lines [2]. One of the main objectives behind utilizing state estimators and observers for dynamical systems is to augment or replace expensive sensors in a control system [3]. For that and various reasons, the analysis and design of dynamic robust observers for linear and nonlinear systems, for systems with known and unknown inputs and disturbances have received noticeable attention in the past few decades.

Luenberger was the first to propose, analyze and design observers [4]–[5]. The well-known Luenberger Observer is still utilized for various engineering applications. Furthermore, observers for systems with unknown inputs and disturbances, also called Unknown Input Observers (UIO), have been extensively studied since the late seventies. The following are some well known research efforts on UIOs: Bhattacharyya [6], Chang *et al.* [7], Chen *et al.* [8], Chen and Patton [9], Corless and Tu [10], Darouach *et al.* [11], Hui and Żak [12]. For more references on different UIO architectures, we direct the reader to [3, p. 431].
The study of UIOs is becoming more crucial to large-scale systems, as these systems are becoming more susceptible to high-disturbances, faults, cyber and physical attacks. The design of robust UIOs for systems under attacks would result in a better estimation of the state of the plant and thus better control and performance. UIOs can be used to employ Fault Detection and Isolation (FDI) mechanisms as proposed by Chen *et al.* [8]. Teixeira *et al.* [13] applied UIOs to design an FDI scheme to analyze power networks under cyber attacks. In [14], Darouach presented necessary and sufficient conditions for the existence of reduced order linear functional observers. As reported in [15], those conditions may not be satisfied in some cases and thus, the design of the linear functional observer may not be possible. In [16], Fernando *et al.* designed a minimum-order linear functional observer for LTI systems and in [17], the authors derived existence conditions for an unknown input functional observer design. In some cases, the proposed observer design may exist, even when an unknown input state observer does not exist.

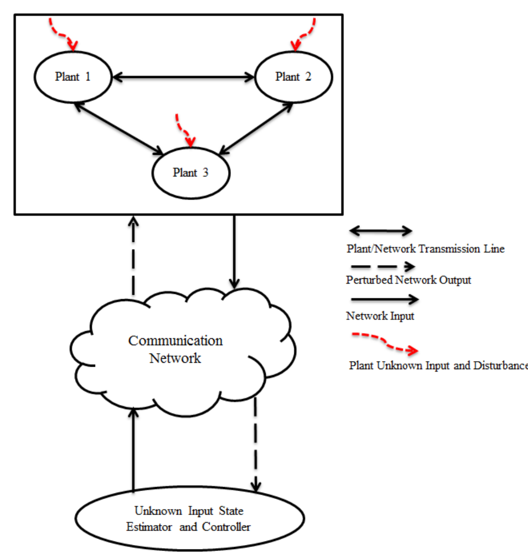

Fig. 1. Networked Control System with Unknown Inputs

The basic idea behind most FDI schemes is to generate weighted residual functions for each subsystem or node which is defined as the difference between the actual system outputs and the estimated ones [3], as in power networks. After choosing a suitable dynamic or static estimation threshold for the error of the residual functions, faulty nodes are isolated. For example, if a bus in a power network is continuously generating higher residuals than the threshold, this bus would be geographically identified and then physically isolated from the overall network.

*Ph.D. Candidate, School of Electrical and Computer Engineering, Purdue University, West Lafayette, Indiana 47907 tahaa@purdue.edu
#Research Associate, School of Aeronautics and Astronautics, Purdue University, West Lafayette, Indiana 47907 aelmahdi@purdue.edu
‡Assistant Professor, School of Mechanical Engineering, Purdue University, West Lafayette, Indiana 47907 panchal@purdue.edu
†Assistant Professor, School of Aeronautics and Astronautics, Purdue University, West Lafayette, Indiana 47907 dsun@purdue.edu

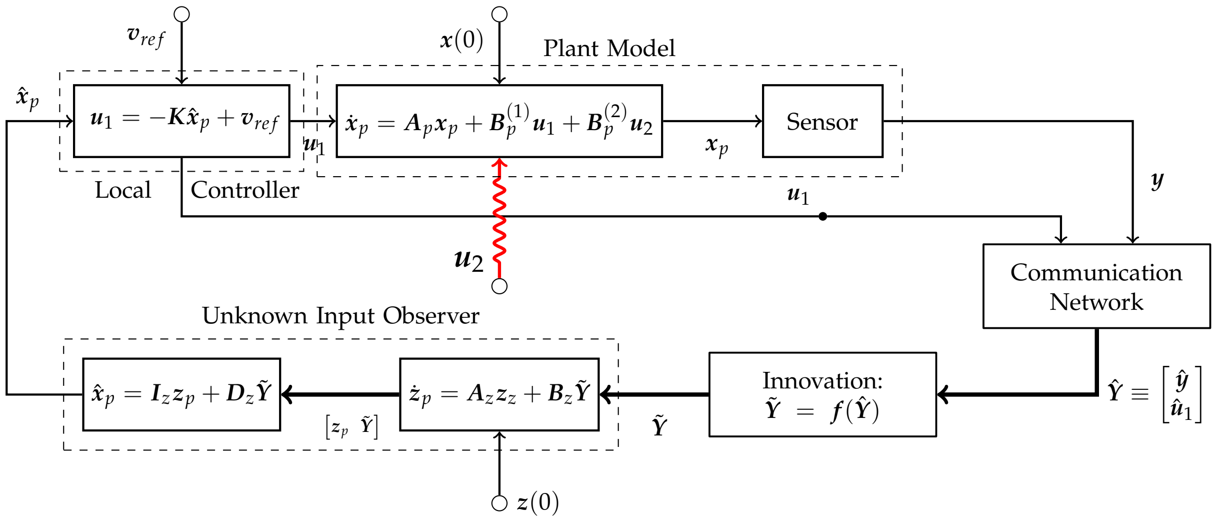

Fig. 2. Networked Unknown Input Observer (N_{et}UIO) Architecture for Systems with Unknown Inputs

B. Research Gaps

As mentioned earlier, observers use the known plant's inputs and outputs to generate estimates for the state of the plant. The closed loop system is then controlled through a controller that often uses the estimated plant state to generate control commands. Observers in different large-scale dynamical systems such as transportation networks, power plants, and remotely controlled mobile agents are often distributed. Hence, the UIO's inputs (i.e., the plant's input and output) are transmitted through a communication network, which is a key component in modern NCSs. Thus, most observers for systems with unknown inputs are Networked Unknown Input Observers (N_{et}UIO). The generic architecture considered in this paper is depicted in Figure 1. Without loss of generality, this model can be generalized to different system scenarios where the perturbation caused to the exchanged signals is due to another source, rather than a communication network.

Most of the developed UIOs in the literature are designed for non-networked systems as in [6]–[12]. The UIO's input is assumed to be transmitted without any disturbances, perturbations, or time-delays. Since communication networks are inserted in many modern decentralized control systems, the analysis and design of observers for *networked* systems with unknown inputs becomes a necessity. The network effect can be either modeled as a perturbation or time-delay to the transmitted signals. In our previous work, we analyzed the effect of perturbation of the signals exchanged through communication networks for decentralized observer-based control for systems with only known inputs.

C. Paper Overview and Contributions

In this paper, the proposed design targets the time-delay and the signals perturbation representations of a network. In Section II, we review an Unknown Input Observer (UIO) design for non-networked system from the UIO literature as in [3]. Then in Section III we derive the dynamics of the N_{et}UIO and we design the N_{et}UIO such that the effect of higher delay order terms are nullified, assuring that the effect of the unknown inputs to the plant is minimized. In Section IV, we derive a bound on the maximum allowable time-delay for the N_{et}UIO. The perturbation model of the network is analyzed and stability bounds for that model are derived in Sections V and VI. Sections VII and VIII includes numerical examples to illustrate the usefulness and applicability of the proposed model. Closing remarks, conclusions, and future work are presented in Section IX.

The major paper contributions are as follows:

1) Studying the effect of the communication network on state estimation for observers designed for plants with unknown inputs.
2) Designing UIO for networked systems such that the effect of higher delay terms and unknown input on state-estimation is minimized.
3) Deriving bounds on the stability of the N_{et}UIO for NCSs that includes the network effect as pure time-delay or perturbation.
4) Applying the derived bounds to a SISO LTI system and demonstrating the applicability and feasibility of the derived results.

II. System Modeling and Problem Formulation

The objective of the research presented in this paper is to study the effect of Unknown Input Observers (UIO) architecture for networked systems with unknown disturbances. Figure 2 shows the proposed architecture for a N_{et}UIO scheme. The non-networked UIO architecture used in this paper is presented in [3]. The input to the UIO block is the delayed or perturbed version of the plant's output (y), that is $\hat{y}$, and the delayed version of the known input u_1, that is $\hat{u}_1$. These two quantities are assumed to be known to the UIO block. The *innovation* block differs from one observer to another. The output of the UIO is the estimate of the plant state (x_p), that is $\hat{x}_p$. The local controller takes the a reference input (v_{ref}) and $\hat{x}_p$ as inputs. The control law is computed through a linear state feedback, but this could be changed according to the application under consideration. The unknown input for the plant is u_2 (unknown plant disturbances, nonlinearities and actuator faults). Although we assume that the signals exchanged through the network are perturbed due to the presence of the network itself, this assumption can be generalized to different cases where the signals are perturbed due to a different perturbation source. Hence, the established results in the paper can be generalized to any dynamical system that involves unknown input observers.

A. Observer Review for Non-Networked Systems with Unknown Inputs

The observer design and the state estimations for non-networked systems with unknown inputs used in this paper is based on a projector operator approach used in [3]. In this paper, we assume a Linear Time-Invariant (LTI) class of systems. The modeled plant can be a linearized representation of a nonlinear plant. The unknown input models a combination of unmeasured disturbances, lumped uncertainties, unknown control actions, unmodeled system dynamics or even actuator faults.

The linearized plant dynamics can be written as:

$$\dot{\boldsymbol{x}}_p = \boldsymbol{A}_p\boldsymbol{x}_p + \boldsymbol{B}_p\boldsymbol{u} = \boldsymbol{A}_p\boldsymbol{x}_p + \boldsymbol{B}_p^{(1)}\boldsymbol{u}_1 + \boldsymbol{B}_p^{(2)}\boldsymbol{u}_2. \quad (1)$$

where $\boldsymbol{A} \in \mathbb{R}^{n\times n}$, $\boldsymbol{B}_p^{(1)} \in \mathbb{R}^{n\times m_1}, \boldsymbol{B}_p^{(2)} \in \mathbb{R}^{n\times m_2}$. $\boldsymbol{x}_p, \boldsymbol{u}_1, \boldsymbol{u}_2, \boldsymbol{y}$ are the plant state, known input, unknown input and output, respectively. . We assume that $\boldsymbol{A}_p, \boldsymbol{B}_p^{(1)}, \boldsymbol{B}_p^{(2)}$ and $\boldsymbol{C}_p$ are all known system parameters. The dynamics of the UIO presented in [3] for non-networked systems are

$$\dot{\boldsymbol{q}} = (\boldsymbol{I} - \boldsymbol{M}\boldsymbol{C}_p)\bigg(\boldsymbol{A}_p\boldsymbol{q} + \boldsymbol{A}_p\boldsymbol{M}\boldsymbol{y} + \boldsymbol{B}_p^{(1)}\boldsymbol{u}_1 \quad (2)$$

$$+ \boldsymbol{L}(\boldsymbol{y} - \boldsymbol{C}_p\boldsymbol{q} - \boldsymbol{C}_p\boldsymbol{M}\boldsymbol{y})\bigg)$$

$$\hat{\boldsymbol{x}}_p = \boldsymbol{q} + \boldsymbol{M}\boldsymbol{y}, \quad (3)$$

where $\boldsymbol{M} \in \mathbb{R}^{n\times p}$ is chosen such that $(\boldsymbol{I} - \boldsymbol{M}\boldsymbol{C}_p)\boldsymbol{B}_p^{(2)} = \boldsymbol{O}$ and $\boldsymbol{L}$ is an added gain to improve the convergence of the estimated state ($\hat{\boldsymbol{x}}_p$). The initial conditions for the observer are $\boldsymbol{q}(0) = (\boldsymbol{I} - \boldsymbol{M}\boldsymbol{C}_p)\hat{\boldsymbol{x}}(0)$, where $\hat{\boldsymbol{x}}(0)$ is an estimate of the initial plant state. Under the assumption that the pair $(\boldsymbol{C}_p, \boldsymbol{A}_p)$ is detectable, this observer for non-networked control systems guarantees that the estimation error ($\boldsymbol{e}(t) = \boldsymbol{x}_p(t) - \hat{\boldsymbol{x}}_p(t)$) converges to zero as $t \to \infty$ under mild conditions [3].

As mentioned in the introduction, the objective of the paper is to analyze the effect of the communication network on the state and unknown input estimation for an UIO architecture. To do so, we first rewrite the dynamics of the observer so that it matches the typical setup of controllers/observers from the NCS literature. Letting $\boldsymbol{x}_c = \boldsymbol{q}$, the dynamics of the UIO can be rewritten as follows:

$$\dot{\boldsymbol{x}}_c = (\boldsymbol{I} - \boldsymbol{M}\boldsymbol{C}_p)\bigg(\boldsymbol{A}_p\boldsymbol{x}_c + \boldsymbol{A}_p\boldsymbol{M}\boldsymbol{y} + \boldsymbol{B}_p^{(1)}\boldsymbol{u}_1$$

$$+ \boldsymbol{L}(\boldsymbol{y} - \boldsymbol{C}_p\boldsymbol{x}_c - \boldsymbol{C}_p\boldsymbol{M}\boldsymbol{y})\bigg)$$

$$\dot{\boldsymbol{x}}_c = \boldsymbol{A}_c\boldsymbol{x}_c + \boldsymbol{B}_c^{(1)}\boldsymbol{y} + \boldsymbol{B}_c^{(2)}\boldsymbol{u}_1,$$

where

$$\boldsymbol{A}_c = (\boldsymbol{I} - \boldsymbol{M}\boldsymbol{C}_p)(\boldsymbol{A}_p - \boldsymbol{L}\boldsymbol{C}_p), \boldsymbol{B}_c^{(2)} = (\boldsymbol{I} - \boldsymbol{M}\boldsymbol{C}_p)\boldsymbol{B}_p^{(1)}$$

$$\boldsymbol{B}_c^{(1)} = (\boldsymbol{I} - \boldsymbol{M}\boldsymbol{C}_p)(\boldsymbol{A}_p\boldsymbol{M} + \boldsymbol{L} - \boldsymbol{L}\boldsymbol{C}_p\boldsymbol{M}).$$

The addition of the communication network perturbs the UIO's inputs (which are $\boldsymbol{y}$ and $\boldsymbol{u}_1$), as the observer uses the plant's input and output to estimate the state of the plant. Hence, the dynamics of the UIO are as follows:

$$\dot{\boldsymbol{x}}_c = \boldsymbol{A}_c\boldsymbol{x}_c + \boldsymbol{B}_c^{(1)}\hat{\boldsymbol{y}} + \boldsymbol{B}_c^{(2)}\hat{\boldsymbol{u}}_1, \quad (4)$$

$$\hat{\boldsymbol{x}}_p = \boldsymbol{x}_c + \boldsymbol{M}\hat{\boldsymbol{y}}. \quad (5)$$

We assume that state-feedback control is used:

$$\boldsymbol{u}_1 = -\boldsymbol{K}\hat{\boldsymbol{x}}_p = -\boldsymbol{K}\boldsymbol{x}_c - \boldsymbol{K}\boldsymbol{M}\hat{\boldsymbol{y}}. \quad (6)$$

III. Network Effect as Pure Time Delay

A. Closed Loop Dynamics

In this section, we model the communication network by a pure-time delay. Precisely, we assume that $\hat{\boldsymbol{y}} = \boldsymbol{y}(t-\tau)$ and $\hat{\boldsymbol{u}}_1 = \boldsymbol{u}_1(t-\tau)$, where τ is the time delay due to the presence of the network in the feedback loops. To simplify the derivations, we assume that the communication network is only inserted between UIO and its inputs.

Assuming that the innovation function of the observer is embedded in the UIO dynamics, we can rewrite the dynamics of the observer and the controller as in the typical form of an NCS controller/observer:

$$\begin{cases} \dot{\boldsymbol{x}}_c(t) = \boldsymbol{A}_c\boldsymbol{x}_c(t) + \boldsymbol{B}_c^{(1)}\boldsymbol{y}(t-\tau) + \boldsymbol{B}_c^{(2)}\boldsymbol{u}_1(t-\tau) \\ \boldsymbol{u}_1(t) = \boldsymbol{C}_c\boldsymbol{x}_c(t) + \boldsymbol{D}_c\boldsymbol{y}(t-\tau), \end{cases} \quad (7)$$

where $\boldsymbol{C}_c = -\boldsymbol{K}$ and $\boldsymbol{D}_c = -\boldsymbol{K}\boldsymbol{M}$. The plant and the controller state dynamics can be written as:

$$\begin{aligned} \dot{\boldsymbol{x}}_p(t) &= \boldsymbol{A}_p\boldsymbol{x}_p(t) + \boldsymbol{B}_p^{(1)}\boldsymbol{C}_c\boldsymbol{x}_c(t) \\ &\quad + \boldsymbol{B}_p^{(1)}\boldsymbol{D}_c\boldsymbol{C}_p\boldsymbol{x}_p(t-\tau) + \boldsymbol{B}_p^{(2)}\boldsymbol{u}_2(t) \\ \dot{\boldsymbol{x}}_c(t) &= \boldsymbol{A}_c\boldsymbol{x}_c(t) + \boldsymbol{B}_c^{(1)}\boldsymbol{C}_p\boldsymbol{x}_p(t-\tau) + \boldsymbol{B}_c^{(2)}\boldsymbol{u}_1(t-\tau) \\ &= \boldsymbol{A}_c\boldsymbol{x}_c(t) + \boldsymbol{B}_c^{(1)}\boldsymbol{C}_p\boldsymbol{x}_p(t-\tau) \\ &\quad + \boldsymbol{B}_c^{(2)}\boldsymbol{C}_c\boldsymbol{x}_c(t-\tau) + \boldsymbol{B}_c^{(2)}\boldsymbol{D}_c\boldsymbol{C}_p\boldsymbol{x}_p(t-2\tau) \end{aligned}$$

Combining $\dot{\boldsymbol{x}}_p(t)$ and $\dot{\boldsymbol{x}}_c(t)$ to find

$$\dot{\boldsymbol{x}}(t) = \begin{bmatrix}\dot{\boldsymbol{x}}_p^\top(t) & \dot{\boldsymbol{x}}_c^\top(t)\end{bmatrix}^\top,$$

we get:

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{\Gamma}_0\boldsymbol{x}(t) + \boldsymbol{\Gamma}_1\boldsymbol{x}(t-\tau) + \boldsymbol{\Gamma}_2\boldsymbol{x}(t-2\tau) + \boldsymbol{\Gamma}_3\,\boldsymbol{u}_2(t). \quad (8)$$

where $\boldsymbol{\Gamma}_3 = \begin{bmatrix}\boldsymbol{B}_p^{(2)} \\ \boldsymbol{O}\end{bmatrix}$ and

$$\boldsymbol{\Gamma}_0 = \begin{bmatrix}\boldsymbol{A}_p & \boldsymbol{B}_p^{(1)}\boldsymbol{C}_c \\ \boldsymbol{O} & \boldsymbol{A}_c\end{bmatrix}, \boldsymbol{\Gamma}_1 = \begin{bmatrix}\boldsymbol{B}_p^{(1)}\boldsymbol{D}_c\boldsymbol{C}_p & \boldsymbol{O} \\ \boldsymbol{B}_c^{(1)}\boldsymbol{C}_p & \boldsymbol{O}\end{bmatrix}, \boldsymbol{\Gamma}_2 = \begin{bmatrix}\boldsymbol{O} & \boldsymbol{O} \\ \boldsymbol{B}_c^{(2)}\boldsymbol{D}_c\boldsymbol{C}_p & \boldsymbol{O}\end{bmatrix}$$

B. NCS and Time-Delay Approximations

Between any two transmission instances, the outputs of the communication network ($\hat{\boldsymbol{y}}$ and $\hat{\boldsymbol{u}}$) can be held constant, in analogy with zero-order hold sampling [18]. Precisely, and as in [18] and [19, p. 62], we assume that $\dot{\hat{\boldsymbol{y}}} = \dot{\hat{\boldsymbol{u}}}_1 = 0$. Recall that, $\hat{\boldsymbol{y}} = \boldsymbol{C}_p\boldsymbol{x}(t-\tau)$, and since $\hat{\boldsymbol{y}}$ can be held constant, then $\dot{\hat{\boldsymbol{y}}} = \boldsymbol{C}_p\dot{\boldsymbol{x}}_p(t-\tau) = 0$ and thus $\dot{\boldsymbol{x}}_p(t-\tau) = 0$. Similarly, $\dot{\boldsymbol{x}}_c(t-\tau) = 0$. Taking the second derivative of $\boldsymbol{x}(t)$ and substituting the above approximation, we have,

$$\ddot{\boldsymbol{x}}(t) = \boldsymbol{\Gamma}_0\dot{\boldsymbol{x}}(t) + \boldsymbol{\Gamma}_3\,\dot{\boldsymbol{u}}_2(t). \quad (9)$$

We use the following Taylor series expansion for $x(t-\tau)$:

$$x(t-\tau) = \sum_{n=0}^{\infty} (-1)^n \frac{\tau^n}{n!} x^{(n)}(t).$$

Neglecting the higher order terms, we get an approximated expression of $\dot{x}(t)$ in terms of only $x(t)$ and τ as follows:

$$x(t-\tau) = x(t) - \tau\dot{x}(t) + \frac{\tau^2}{2}\ddot{x}(t) \quad (10)$$

$$x(t-2\tau) = x(t) - 2\tau\dot{x}(t) + 2\tau^2\ddot{x}(t) \quad (11)$$

Substituting (9) into (10) and (11):

$$x(t-\tau) = x(t) + (\frac{\tau^2}{2}\Gamma_0 - \tau)\dot{x}(t) + \frac{\tau^2}{2}\Gamma_3\dot{u}_2(t) \quad (12)$$

$$x(t-2\tau) = x(t) + (2\tau^2\Gamma_0 - 2\tau)\dot{x}(t) + 2\tau^2\Gamma_3\dot{u}_2(t) \quad (13)$$

By substituting (12) and (13) into (8), we can write $\dot{x}(t)$ as:

$$\dot{x} = (\Gamma_0 + \Gamma_1 + \Gamma_2)\,x + \left(\frac{\tau^2}{2}\Gamma_1\Gamma_3 + 2\tau^2\Gamma_2\Gamma_3\right)\dot{u}_2 + \left(-\tau\Gamma_1 + \frac{\tau^2}{2}\Gamma_1\Gamma_0 + 2\tau^2\Gamma_2\Gamma_0 - 2\tau\Gamma_2\right)\dot{x} + \Gamma_3\,u_2.$$

Rearranging, we get

$$\boxed{\dot{x}(t) = \Psi_0 x(t) + \Psi_1\dot{u}_2(t) + \Psi_2 u_2(t)} \quad (14)$$

where

$$\Psi_0 = \Theta\,(\Gamma_0 + \Gamma_1 + \Gamma_2),$$

$$\Psi_1 = \Theta\left(\frac{\tau^2}{2}\Gamma_1\Gamma_3 + 2\tau^2\Gamma_2\Gamma_3\right), \Psi_2 = \Theta\Gamma_3$$

$$\Theta = \left(I - \left(-\tau\Gamma_1 + \tfrac{\tau^2}{2}\Gamma_1\Gamma_0 + 2\tau^2\Gamma_2\Gamma_0 - 2\tau\Gamma_2\right)\right)^{-1}.$$

C. Time-Delay Based Networked UIO Design

In this section, we design the controller and the observer to minimize the effect of the unknown input from the global dynamics of the closed loop system as well as the higher order time-delay terms.

Recall that

$$\Gamma_2 = \begin{bmatrix} O & O \\ B_c^{(2)}D_cC_p & O \end{bmatrix}, \Gamma_3 = \begin{bmatrix} B_p^{(2)} \\ O \end{bmatrix} \;\&\; \Gamma_2\Gamma_3 = \begin{bmatrix} O \\ B_c^{(2)}D_cC_pB_p^{(2)} \end{bmatrix}.$$

The matrices $B_p^{(1)}, B_p^{(2)}$ and C_p are given matrices. Also, $B_c^{(2)} = (I - MC_p)B_p^{(1)}$ and $D_c = -KM$. In addition, the UIO for the non-networked system is designed with the following restriction on $B_p^{(2)}$ as in [3]:

$$(I - MC_p)B_p^{(2)} = O.$$

We can design the observer such that the higher order delay terms are nullified and the effect of unknown input is minimized. Setting $\Gamma_2\Gamma_3 = O$, then we must have $B_c^{(2)}D_cC_pB_p^{(2)} = O$, or

$$B_c^{(2)}D_c = -(I - MC_p)B_p^{(1)}KM = O.$$

Hence, we have the following matrix equations to solve:

$$(I - MC_p)B_p^{(1)}KM = O \quad (15)$$

$$(I - MC_p)B_p^{(2)} = O. \quad (16)$$

We do not have much control over K as this depends on what needs to be achieved through the linear state feedback. Since the controller gain matrix is already designed, the only Networked UIO design variable to be solved for is M. All other matrices involved in the above system of equations is assumed to be given. Letting $R = B_p^{(1)}K \in \mathbb{R}^{n\times n}$, we can rewrite the above system as:

$$RM - MC_pRM = O_{n\times p} \quad (17)$$

$$B_p^{(2)} - MC_pB_p^{(2)} = O_{n\times m_2}, \quad (18)$$

where $0_{n\times m_2}$ is the zero-matrix $\in \mathbb{R}^{n\times m_2}$. Multiplying (18) by any nonsingular matrix $T \in \mathbb{R}^{m_2\times p}$, we get:

$$RM - MC_pRM = O_{n\times p} \quad (19)$$

$$B_p^{(2)}T - MC_pB_p^{(2)}T = O_{n\times p}. \quad (20)$$

Adding (19) and (20), we get a bilateral matrix quadratic equation (BMQE) in terms of the Networked UIO design parameter M:

$$MC_pRM - RM - MC_pB_p^{(2)}T + B_p^{(2)}T = O_{n\times p}. \quad (21)$$

The above equation can be simplified as follows:

$$\boxed{M\mathcal{A}M + M\mathcal{B} + \mathcal{C}M + \mathcal{D} = O_{n\times p}} \quad (22)$$

where $\mathcal{A} \in \mathbb{R}^{p\times n}, \mathcal{B} \in \mathbb{R}^{p\times p}, \mathcal{C} \in \mathbb{R}^{n\times n}$ and $\mathcal{D} \in \mathbb{R}^{n\times p}$ are all constant matrices. Hence, the design of the $\mathrm{N_{et}}$UIO can be simplified by solving (22). In Appendix A, we present a discussion on existence of a solution for (22) and a solution algorithm for the design matrix M.

IV. Stability Analysis of The $\mathrm{N_{et}}$UIO

After designing the $\mathrm{N_{et}}$UIO, the simplified dynamics of the closed loop system with a feedback controller can be written as:

$$\dot{x}(t) = \Psi_0 x(t) + \Psi_1\dot{u}_2(t) + \Psi_2 u_2(t), \quad (23)$$

where

$$\Psi_0 = \Theta\,(\Gamma_0 + \Gamma_1), \Psi_1 = \Theta\left(\frac{\tau^2}{2}\Gamma_1\Gamma_3\right),$$

$$\Psi_2 = \Theta\Gamma_3,\; \Theta = \left(I - \left(-\tau\Gamma_1 + \frac{\tau^2}{2}\Gamma_1\Gamma_0\right)\right)^{-1}.$$

In this section, we analyze the stability of the $\mathrm{N_{et}}$UIO with the feedback controller. First, recall that the non-networked UIO is designed such that the closed loop system is stable. Hence, the non-networked system (i.e., with $\tau = 0$) is asymptotically stable for any bounded unknown input. The dynamics of the non-networked system for the UIO can be written as:

$$\dot{x}(t) = (\Gamma_0 + \Gamma_1)\,x(t) + \Gamma_3 u_2(t) = \Gamma x(t) + \Gamma_3 u_2(t). \quad (24)$$

where Γ is Hurwitz by the design assumption of the non-networked UIO.

Theorem 1. *For the $\mathrm{N_{et}}$UIO in (23) and for a Hurwitz Γ, we have $P = P^\top \succ O$, is the solution to the Lyapunov matrix equation*

$$\Gamma^\top P + P\Gamma = -2Q,$$

for a given $Q = Q^\top \succ O$ and if the UIO design parameter (M) satisfies (22), and if $\|\dot{u}_2\| \le \rho\|x\|$ and $\|u_2\| < \mu\|x\|$, where $\rho, \mu > 0$, then if the network induced delay satisfies the following inequality,

$$\Big(\mu\|P\Gamma_1\Gamma_0\Gamma_3\| + \mu\|P\Gamma_1^2\Gamma_3\| + \|P\Gamma_1\Gamma_0\Gamma\| + 2\|P\Gamma_1^2\Gamma\|$$

$$+\rho\|P\Gamma_1\Gamma_3\|\Big)\tau^2+\Big(-2\mu\|P\Gamma_1\Gamma_3\|-2\|P\Gamma_1\Gamma\|\Big)\tau$$

$$+\Big(2\mu\|P\Gamma_3\|-2\lambda_{min}(Q)\Big)<0$$

then the origin is a globally exponentially stable equilibrium point of the N_{et}UIO.

Proof: Since $\boldsymbol{\Gamma}$ is asymptotically stable, the solution $\boldsymbol{P}$ to the Lyapunov matrix equation

$$\boldsymbol{\Gamma}^\top \boldsymbol{P}+\boldsymbol{P}\boldsymbol{\Gamma}=-2\boldsymbol{Q},$$

is symmetric positive definite (i.e., $\boldsymbol{P}=\boldsymbol{P}^\top \succ \boldsymbol{O}$.) for a given $\boldsymbol{Q}=\boldsymbol{Q}^\top \succ \boldsymbol{O}$.

Let $V(\boldsymbol{x})=\boldsymbol{x}^\top \boldsymbol{P}\boldsymbol{x}$ be a Lyapunov candidate function, used to verify stability and derive bounds on the network-induced time-delay. Particularly, we need to find a bound on the time-delay such that $\dot{V}(\boldsymbol{x})<0$. Taking the derivative of the Lyapunov candidate function, we get,

$$\dot{V}(\boldsymbol{x})=2\boldsymbol{x}^\top \boldsymbol{P}\dot{\boldsymbol{x}}=\dot{\boldsymbol{x}}^\top \boldsymbol{P}\boldsymbol{x}+\boldsymbol{x}^\top \boldsymbol{P}\dot{\boldsymbol{x}}. \tag{25}$$

Substituting (23) into (25), we get:

$$\begin{aligned}
\dot{V}(\boldsymbol{x}) &= 2\boldsymbol{x}^\top \boldsymbol{P}\dot{\boldsymbol{x}}\\
&= \boldsymbol{x}^\top\boldsymbol{\Gamma}^\top\boldsymbol{\Theta}^\top\boldsymbol{P}\boldsymbol{x}+\boldsymbol{x}^\top\boldsymbol{P}\boldsymbol{\Theta}\boldsymbol{x}+2\boldsymbol{x}^\top\boldsymbol{P}\boldsymbol{\Theta}\boldsymbol{\Psi}_1\dot{\boldsymbol{u}}_2\\
&\quad +2\boldsymbol{x}^\top\boldsymbol{P}\boldsymbol{\Theta}\boldsymbol{\Gamma}_3\boldsymbol{u}_2\\
&= \boldsymbol{x}^\top\boldsymbol{\Gamma}^\top\boldsymbol{P}\boldsymbol{P}^{-1}\boldsymbol{\Theta}^\top\boldsymbol{P}\boldsymbol{x}+\boldsymbol{x}^\top\boldsymbol{P}\boldsymbol{\Theta}\boldsymbol{P}^{-1}\boldsymbol{P}\boldsymbol{\Gamma}\boldsymbol{x}\\
&\quad -\boldsymbol{x}^\top(\boldsymbol{\Gamma}^\top\boldsymbol{P}+\boldsymbol{P}\boldsymbol{\Gamma})\boldsymbol{x}+\boldsymbol{x}^\top(\boldsymbol{\Gamma}^\top\boldsymbol{P}+\boldsymbol{P}\boldsymbol{\Gamma})\boldsymbol{x}\\
&\quad +\tau^2\boldsymbol{x}^\top\boldsymbol{P}\boldsymbol{\Theta}\boldsymbol{\Gamma}_1\boldsymbol{\Gamma}_3\dot{\boldsymbol{u}}_2+2\boldsymbol{x}^\top\boldsymbol{P}\boldsymbol{\Theta}\boldsymbol{\Gamma}_3\boldsymbol{u}_2\\
&= \boldsymbol{x}^\top\boldsymbol{\Gamma}^\top(\boldsymbol{P}\boldsymbol{P}^{-1}\boldsymbol{\Theta})^\top\boldsymbol{P}\boldsymbol{x}+\boldsymbol{x}^\top\boldsymbol{P}\boldsymbol{\Theta}\boldsymbol{P}^{-1}\boldsymbol{P}\boldsymbol{\Gamma}\boldsymbol{x}\\
&\quad -\boldsymbol{x}^\top\boldsymbol{\Gamma}^\top\boldsymbol{P}\boldsymbol{x}-\boldsymbol{x}^\top\boldsymbol{P}\boldsymbol{\Gamma}\boldsymbol{x}-2\boldsymbol{x}^\top\boldsymbol{Q}\boldsymbol{x}\\
&\quad +\tau^2\boldsymbol{x}^\top\boldsymbol{P}\boldsymbol{\Theta}\boldsymbol{\Gamma}_1\boldsymbol{\Gamma}_3\dot{\boldsymbol{u}}_2\\
&= \boldsymbol{x}^\top\boldsymbol{\Gamma}^\top(\boldsymbol{P}\boldsymbol{P}^{-1}\boldsymbol{\Theta}^\top\boldsymbol{P}-\boldsymbol{I})\boldsymbol{x}-2\boldsymbol{x}^\top\boldsymbol{Q}\boldsymbol{x}\\
&\quad +\boldsymbol{x}^\top(\boldsymbol{P}\boldsymbol{\Theta}\boldsymbol{P}^{-1}-\boldsymbol{I})\boldsymbol{P}\boldsymbol{\Gamma}\boldsymbol{x}+\tau^2\boldsymbol{x}^\top\boldsymbol{P}\boldsymbol{\Theta}\boldsymbol{\Gamma}_1\boldsymbol{\Gamma}_3\dot{\boldsymbol{u}}_2\\
&\quad +2\boldsymbol{x}^\top\boldsymbol{P}\boldsymbol{\Theta}\boldsymbol{\Gamma}_3\boldsymbol{u}_2\\
&= 2\boldsymbol{x}^\top\boldsymbol{P}(\boldsymbol{\Theta}-\boldsymbol{I})\boldsymbol{\Gamma}\boldsymbol{x}-2\boldsymbol{x}^\top\boldsymbol{Q}\boldsymbol{x}\\
&\quad +\tau^2\boldsymbol{x}^\top\boldsymbol{P}\boldsymbol{\Theta}\boldsymbol{\Gamma}_1\boldsymbol{\Gamma}_3\dot{\boldsymbol{u}}_2+2\boldsymbol{x}^\top\boldsymbol{P}\boldsymbol{\Theta}\boldsymbol{\Gamma}_3\boldsymbol{u}_2
\end{aligned} \tag{26}$$

We can write $\boldsymbol{\Theta}$ as

$$\boldsymbol{\Theta}=\left(\boldsymbol{I}+\tau\boldsymbol{\Gamma}_1-\frac{\tau^2}{2}\boldsymbol{\Gamma}_1\boldsymbol{\Gamma}_0\right)^{-1}=\left(\boldsymbol{I}+\tau\underbrace{\left(\boldsymbol{\Gamma}_1-\frac{\tau}{2}\boldsymbol{\Gamma}_1\boldsymbol{\Gamma}_0\right)}_{A_1}\right)^{-1}$$

Using the Neumann series formula for the inverse of the sum of two matrices,

$$\begin{aligned}
\boldsymbol{\Theta} &= (\boldsymbol{I}+\tau\boldsymbol{A}_1)^{-1}=\boldsymbol{I}-\tau\boldsymbol{A}_1+\tau^2\boldsymbol{A}_1^2-\tau^3\boldsymbol{A}_1^3+\dots\\
&\approx \boldsymbol{I}-\tau\boldsymbol{A}_1+\tau^2\boldsymbol{A}_1^2\\
&= \boldsymbol{I}-\tau\left(\boldsymbol{\Gamma}_1-\frac{\tau}{2}\boldsymbol{\Gamma}_1\boldsymbol{\Gamma}_0\right)+\tau^2\left(\boldsymbol{\Gamma}_1-\frac{\tau}{2}\boldsymbol{\Gamma}_1\boldsymbol{\Gamma}_0\right)^2\\
&= \boldsymbol{I}-\tau\boldsymbol{\Gamma}_1+\frac{\tau^2}{2}\boldsymbol{\Gamma}_1\boldsymbol{\Gamma}_0+\tau^2\boldsymbol{\Gamma}_1^2+\frac{\tau^4}{4}\boldsymbol{\Gamma}_1^2\boldsymbol{\Gamma}_0^2-\tau^3\boldsymbol{\Gamma}_1^2\boldsymbol{\Gamma}_0\\
\boldsymbol{\Theta} &\approx \boldsymbol{I}-\tau\boldsymbol{\Gamma}_1+\frac{\tau^2}{2}\boldsymbol{\Gamma}_1\boldsymbol{\Gamma}_0+\tau^2\boldsymbol{\Gamma}_1^2
\end{aligned} \tag{27}$$

Since the networked induced delay (τ) is generally estimated to be within the range of 10^{-3}, we eliminate the terms involving higher powers (≥ 3) of τ. Substituting (27) into (26),

$$\begin{aligned}
\dot{V} &= 2\boldsymbol{x}^\top\boldsymbol{P}\left(-\tau\boldsymbol{\Gamma}_1+\frac{\tau^2}{2}\boldsymbol{\Gamma}_1\boldsymbol{\Gamma}_0+\tau^2\boldsymbol{\Gamma}_1^2\right)\boldsymbol{\Gamma}\boldsymbol{x}-2\boldsymbol{x}^\top\boldsymbol{Q}\boldsymbol{x}\\
&\quad +\tau^2\boldsymbol{x}^\top\boldsymbol{P}\left(\boldsymbol{I}-\tau\boldsymbol{\Gamma}_1+\frac{\tau^2}{2}\boldsymbol{\Gamma}_1\boldsymbol{\Gamma}_0+\tau^2\boldsymbol{\Gamma}_1^2\right)\boldsymbol{\Gamma}_1\boldsymbol{\Gamma}_3\dot{\boldsymbol{u}}_2\\
&\quad +2\boldsymbol{x}^\top\boldsymbol{P}\left(\boldsymbol{I}-\tau\boldsymbol{\Gamma}_1+\frac{\tau^2}{2}\boldsymbol{\Gamma}_1\boldsymbol{\Gamma}_0+\tau^2\boldsymbol{\Gamma}_1^2\right)\boldsymbol{\Gamma}_3\boldsymbol{u}_2\\
&\approx 2\boldsymbol{x}^\top\boldsymbol{P}\left(-\tau\boldsymbol{\Gamma}_1+\frac{\tau^2}{2}\boldsymbol{\Gamma}_1\boldsymbol{\Gamma}_0+\tau^2\boldsymbol{\Gamma}_1^2\right)\boldsymbol{\Gamma}\boldsymbol{x}\\
&\quad -2\boldsymbol{x}^\top\boldsymbol{Q}\boldsymbol{x}+\tau^2\boldsymbol{x}^\top\boldsymbol{P}\boldsymbol{\Gamma}_1\boldsymbol{\Gamma}_3\dot{\boldsymbol{u}}_2\\
&\quad +2\boldsymbol{x}^\top\boldsymbol{P}\left(\boldsymbol{I}-\tau\boldsymbol{\Gamma}_1+\frac{\tau^2}{2}\boldsymbol{\Gamma}_1\boldsymbol{\Gamma}_0+\tau^2\boldsymbol{\Gamma}_1^2\right)\boldsymbol{\Gamma}_3\boldsymbol{u}_2\\
&= -2\tau\boldsymbol{x}^\top\boldsymbol{P}\boldsymbol{\Gamma}_1\boldsymbol{\Gamma}\boldsymbol{x}+\tau^2\boldsymbol{x}^\top\boldsymbol{P}\boldsymbol{\Gamma}_1\boldsymbol{\Gamma}_0\boldsymbol{\Gamma}\boldsymbol{x}\\
&\quad +2\tau^2\boldsymbol{x}^\top\boldsymbol{P}\boldsymbol{\Gamma}_1^2\boldsymbol{\Gamma}\boldsymbol{x}-2\boldsymbol{x}^\top\boldsymbol{Q}\boldsymbol{x}+\tau^2\boldsymbol{x}^\top\boldsymbol{P}\boldsymbol{\Gamma}_1\boldsymbol{\Gamma}_3\dot{\boldsymbol{u}}_2\\
&\quad +2\boldsymbol{x}^\top\boldsymbol{P}\boldsymbol{\Gamma}_3\boldsymbol{u}_2-2\tau\boldsymbol{x}^\top\boldsymbol{P}\boldsymbol{\Gamma}_1\boldsymbol{\Gamma}_3\boldsymbol{u}_2\\
&\quad +\tau^2\boldsymbol{x}^\top\boldsymbol{P}\boldsymbol{\Gamma}_1\boldsymbol{\Gamma}_0\boldsymbol{\Gamma}_3\boldsymbol{u}_2+2\tau^2\boldsymbol{x}^\top\boldsymbol{P}\boldsymbol{\Gamma}_1^2\boldsymbol{\Gamma}_3\boldsymbol{u}_2
\end{aligned}$$

In addition, for any symmetric matrix $\boldsymbol{Q}$ we have:

$$\lambda_{min}(\boldsymbol{Q})\|\boldsymbol{x}\|_2^2\leq\boldsymbol{x}^\top\boldsymbol{Q}\boldsymbol{x}\leq\lambda_{max}(\boldsymbol{Q})\|\boldsymbol{x}\|_2^2.$$

The above is known as Raleigh's inequality. Also, since $\|\dot{\boldsymbol{u}}_2\|\leq\rho\|\boldsymbol{x}\|$ (i.e., the variations of the unmatched uncertainties, disturbances and the effect of unknown inputs are bounded by an upper bound directly proportional to the state norm, where $\rho>0$), similar to the assumption in [20]. We also similarly have as in [21, p. 168] that $\|\boldsymbol{u}_2\|\leq\mu\|\boldsymbol{x}\|$, where $\mu>0$. We can now upper bound the derivative of the Lyapunov candidate function $\dot{V}(\boldsymbol{x})$ as follows:

$$\begin{aligned}
\dot{V}(\boldsymbol{x}) &\leq -2\tau\|\boldsymbol{P}\boldsymbol{\Gamma}_1\boldsymbol{\Gamma}\|\|\boldsymbol{x}\|^2+\tau^2\|\boldsymbol{P}\boldsymbol{\Gamma}_1\boldsymbol{\Gamma}_0\boldsymbol{\Gamma}\|\|\boldsymbol{x}\|^2\\
&\quad +2\tau^2\|\boldsymbol{P}\boldsymbol{\Gamma}_1^2\boldsymbol{\Gamma}\|\|\boldsymbol{x}\|^2\\
&\quad -2\lambda_{min}(\boldsymbol{Q})\|\boldsymbol{x}\|^2+\rho\tau^2\|\boldsymbol{x}^\top\boldsymbol{P}\boldsymbol{\Gamma}_1\boldsymbol{\Gamma}_3\|\|\boldsymbol{x}\|\\
&\quad +\|\boldsymbol{x}\|^2\Big(2\mu\|\boldsymbol{P}\boldsymbol{\Gamma}_3\|-2\mu\tau\|\boldsymbol{P}\boldsymbol{\Gamma}1\boldsymbol{\Gamma}_3\|\\
&\quad +\mu\tau^2\left(\|\boldsymbol{P}\boldsymbol{\Gamma}_1\boldsymbol{\Gamma}_0\boldsymbol{\Gamma}_3\|+\|\boldsymbol{P}\boldsymbol{\Gamma}_1^2\boldsymbol{\Gamma}_3\|\right)\Big)\\
&\leq \Big(-2\tau\|\boldsymbol{P}\boldsymbol{\Gamma}_1\boldsymbol{\Gamma}\|+\tau^2\|\boldsymbol{P}\boldsymbol{\Gamma}_1\boldsymbol{\Gamma}_0\boldsymbol{\Gamma}\|+2\tau^2\|\boldsymbol{P}\boldsymbol{\Gamma}_1^2\boldsymbol{\Gamma}\|\\
&\quad -2\lambda_{min}(\boldsymbol{Q})+\rho\tau^2\|\boldsymbol{P}\boldsymbol{\Gamma}_1\boldsymbol{\Gamma}_3\|\Big)\|\boldsymbol{x}\|^2\\
&\quad +\Big(2\mu\|\boldsymbol{P}\boldsymbol{\Gamma}_3\|-2\mu\tau\|\boldsymbol{P}\boldsymbol{\Gamma}1\boldsymbol{\Gamma}_3\|\\
&\quad +\mu\tau^2\left(\|\boldsymbol{P}\boldsymbol{\Gamma}_1\boldsymbol{\Gamma}_0\boldsymbol{\Gamma}_3\|+\|\boldsymbol{P}\boldsymbol{\Gamma}_1^2\boldsymbol{\Gamma}_3\|\right)\Big)\|\boldsymbol{x}\|^2\\
&\leq \mathcal{V}\|\boldsymbol{x}\|^2
\end{aligned}$$

where

$$\mathcal{V}=\Big(-2\tau\|\boldsymbol{P}\boldsymbol{\Gamma}_1\boldsymbol{\Gamma}\|+\tau^2\|\boldsymbol{P}\boldsymbol{\Gamma}_1\boldsymbol{\Gamma}_0\boldsymbol{\Gamma}\|+2\tau^2\|\boldsymbol{P}\boldsymbol{\Gamma}_1^2\boldsymbol{\Gamma}\|$$

$$-2\lambda_{min}(\boldsymbol{Q})+\rho\tau^2\|\boldsymbol{P}\boldsymbol{\Gamma}_1\boldsymbol{\Gamma}_3\|+2\mu\|\boldsymbol{P}\boldsymbol{\Gamma}_3\|-2\mu\tau\|\boldsymbol{P}\boldsymbol{\Gamma}1\boldsymbol{\Gamma}_3\|$$

$$+\mu\tau^2\left(\|\boldsymbol{P}\boldsymbol{\Gamma}_1\boldsymbol{\Gamma}_0\boldsymbol{\Gamma}_3\|+\|\boldsymbol{P}\boldsymbol{\Gamma}_1^2\boldsymbol{\Gamma}_3\|\right)\Big)$$

$$= \Big(\mu\|P\Gamma_1\Gamma_0\Gamma_3\| + \mu\|P\Gamma_1^2\Gamma_3\| + \|P\Gamma_1\Gamma_0\Gamma\| + 2\|P\Gamma_1^2\Gamma\| + \rho\|P\Gamma_1\Gamma_3\|\Big)\tau^2 + \Big(-2\mu\|P\Gamma_1\Gamma_3\| - 2\|P\Gamma_1\Gamma\|\Big)\tau + \Big(2\mu\|P\Gamma_3\| - 2\lambda_{min}(Q)\Big).$$

Hence, if the networked induced delay satisfies the following inequality,

$$\boxed{\Big(\mu\|P\Gamma_1\Gamma_0\Gamma_3\| + \mu\|P\Gamma_1^2\Gamma_3\| + \|P\Gamma_1\Gamma_0\Gamma\| + 2\|P\Gamma_1^2\Gamma\| + \rho\|P\Gamma_1\Gamma_3\|\Big)\tau^2 + \Big(-2\mu\|P\Gamma_1\Gamma_3\| - 2\|P\Gamma_1\Gamma\|\Big)\tau + \Big(2\mu\|P\Gamma_3\| - 2\lambda_{min}(Q)\Big) < 0}$$

then the origin is a globally exponentially stable equilibrium point of the $\mathrm{N_{et}UIO}$. This ends the proof. ■

V. Network Effect As Perturbation

In the previous section, we assumed that the network effect is modeled as pure time-delay (i.e., $\hat{y} = y(t-\tau)$). In this section, we model the network effect as a perturbation,

$$\hat{y} = y - e_y, \;\; \hat{u}_1 = u_1 - e_{u_1},$$

where e is the networked induced error. The plant dynamics can be written as:

$$\begin{cases} \dot{x}_p = A_p x_p + B_p^{(1)}\hat{u}_1 + B_p^{(2)}u_2, \\ y = C_p x_p \end{cases}$$

whereas the combined controller/observer dynamics can be formulated as follows:

$$\begin{cases} \dot{x}_c = A_c x_c + B_c^{(1)}\hat{y} + B_c^{(2)}\hat{u}_1 \\ u_1 = C_c x_c + D_c\hat{y}, \end{cases}$$

A. Closed Loop Dynamics

The objective of this section is to formulate the global dynamics of the overall closed loop states of the system that augments the networked-induced error, namely x_p, x_c, e_y and e_{u_1} where the dynamics of the non-networked UIO are previously derived (i.e., we assume that the developments in this section assume that matrices $A_c, B_c^{(1)}, B_c^{(2)}, C_c$ and D_c are all computed for the non-networked system as in any UIO architecture from the literature). The plant dynamics can be written as follows,

$$\begin{aligned} \dot{x}_p &= A_p x_p + B_p^{(1)}\hat{u}_1 + B_p^{(2)}u_2 \\ &= A_p x_p + B_p^{(1)}(u_1 - e_{u_1}) + B_p^{(2)}u_2 \\ &= A_p x_p + B_p^{(1)}\left(C_c x_c + D_c\hat{y} - e_{u_1}\right) + B_p^{(2)}u_2 \\ &= A_p x_p + B_p^{(1)}\left(C_c x_c + D_c\left(y - e_y\right) - e_{u_1}\right) + B_p^{(2)}u_2 \\ &= A_p x_p + B_p^{(1)}\left(C_c x_c + D_c\left(C_p x_p - e_y\right) - e_{u_1}\right) \\ &\quad + B_p^{(2)}u_2 \\ \dot{x}_p &= \left(A_p + B_p^{(1)}D_c C_p\right)x_p + B_p^{(1)}C_c x_c - B_p^{(1)}D_c e_y \\ &\quad - B_p^{(1)}e_{u_1} + B_p^{(2)}u_2. \end{aligned} \tag{28}$$

The controller/observer's dynamics can be written as,

$$\begin{aligned} \dot{x}_c &= A_c x_c + B_c^{(1)}\hat{y} + B_c^{(2)}\hat{u}_1 \\ &= A_c x_c + B_c^{(1)}\left(y - e_y\right) + B_c^{(2)}\left(u_1 - e_{u_1}\right) \\ &= A_c x_c + B_c^{(1)}\left(C_p x_p - e_y\right) \\ &\quad + B_c^{(2)}\left(C_c x_c + D_c\hat{y} - e_{u_1}\right) \\ &= A_c x_c + B_c^{(1)}\left(C_p x_p - e_y\right) \\ &\quad + B_c^{(2)}\left(C_c x_c + D_c\left(y - e_y\right) - e_{u_1}\right) \\ &= \left(B_c^{(1)}C_p + B_c^{(2)}D_c C_p\right)x_p + \left(A_c + B_c^{(2)}C_c\right)x_c \\ &\quad - \left(B_c^{(1)} + B_c^{(2)}D_c\right)e_y - B_c^{(2)}e_{u_1} \end{aligned} \tag{29}$$

The network induced error of the plant output (e_y) for the plant can be written in terms of the states of the plant and the controller, as follows[1]:

$$\begin{aligned} \dot{e}_y &= \dot{y} - \dot{\hat{y}} \;(\dot{\hat{y}} \to \mathbf{0}) \\ \dot{e}_y &= C_p\left(A_p + B_p^{(1)}D_c C_p\right)x_p + C_p\, B_p^{(1)}C_c x_c \\ &\quad - C_p B_p^{(1)}D_c e_y - C_p B_p^{(1)}e_{u_1} + C_p B_p^{(2)}u_2. \end{aligned} \tag{30}$$

Also, the network induced error of the plant input (e_{u_1}) can be written as[1]:

$$\begin{aligned} \dot{e}_{u_1} &= \dot{u}_1 - \dot{\hat{u}}_1 \;(\dot{\hat{u}}_1 \to \mathbf{0}) \\ \dot{e}_{u_1} &= C_c\left(B_c^{(1)}C_p + B_c^{(2)}D_c C_p\right)x_p - C_c B_c^{(2)}e_{u_1} \\ &\quad + C_c\left(A_c + B_c^{(2)}C_c\right)x_c - C_c\left(B_c^{(1)} + B_c^{(2)}D_c\right)e_y \end{aligned} \tag{31}$$

Let $x = \left[x_p^\top \;\; x_c^\top\right]^\top$, $e = \left[e_y^\top \;\; e_{u_1}^\top\right]^\top$, & $r = \left[x^\top \;\; e^\top\right]^\top$. We can now write equations (28–31) in a matrix-vector form as in (32) where $\Lambda_{1,1} = A_p + B_p^{(1)}D_c C_p$ and $\Lambda_{2,1} = B_c^{(1)}C_p + B_c^{(2)}D_c C_p$. Recall that by the UIO design for the non-networked system, the matrix

$$\Lambda_1 = \begin{bmatrix} \Lambda_{1,1} & B_p^{(1)}C_c \\ \Lambda_{2,1} & \left(A_c + B_c^{(2)}C_c\right) \end{bmatrix}$$

is asymptotically stable. We can isolate the perturbations due to the network and due to the unknown input and rewrite (32) as follows,

$$\begin{aligned} \dot{r} &= \left(\begin{bmatrix} \Lambda_1 & O \\ O & O \end{bmatrix} - H_\epsilon\right)r + \left(\Lambda - \begin{bmatrix} \Lambda_1 & O \\ O & O \end{bmatrix} + H_\epsilon\right)r \\ &\quad + \Omega u_2 \end{aligned} \tag{33}$$

$$\dot{r} = \Lambda_r r + \delta\Lambda_r r + \Omega u_2, \tag{34}$$

where H_ϵ is a Hurwitz matrix (for example, a lower-block diagonal identity matrix, multiplied by a small positive constant ϵ) that guarantees the stability of Λ_r. $\delta\Lambda_r$ is the perturbation due to the addition of the network and Ω is the perturbation due to the unknown input.

B. Stability Analysis of the Perturbed Networked System

In this section, we analyze the stability of the closed loop dynamics of the perturbed system with the network and plant unknown inputs and disturbances. To do so, we establish a

[1]Note that $\hat{y}$ and $\hat{u}_1$ are both piece-wise constant functions, thus $\dot{\hat{y}} = \dot{\hat{u}}_1 = \mathbf{0}$ [18].

$$\dot{\boldsymbol{r}} = \begin{bmatrix} \dot{\boldsymbol{x}}_p \\ \dot{\boldsymbol{x}}_c \\ \dot{\boldsymbol{e}}_y \\ \dot{\boldsymbol{e}}_{u_1} \end{bmatrix} = \underbrace{\begin{bmatrix} \boldsymbol{\Lambda}_{1,1} & \boldsymbol{B}_p^{(1)}\boldsymbol{C}_c & -\boldsymbol{B}_p^{(1)}\boldsymbol{D}_c & -\boldsymbol{B}_p^{(1)} \\ \boldsymbol{\Lambda}_{2,1} & \left(\boldsymbol{A}_c + \boldsymbol{B}_c^{(2)}\boldsymbol{C}_c\right) & -\left(\boldsymbol{B}_c^{(1)} + \boldsymbol{B}_c^{(2)}\boldsymbol{D}_c\right) & -\boldsymbol{B}_c^{(2)} \\ \boldsymbol{C}_p\boldsymbol{\Lambda}_{1,1} & \boldsymbol{C}_p\boldsymbol{B}_p^{(1)}\boldsymbol{C}_c & -\boldsymbol{C}_p\boldsymbol{B}_p^{(1)}\boldsymbol{D}_c & -\boldsymbol{C}_p\boldsymbol{B}_p^{(1)} \\ \boldsymbol{C}_c\boldsymbol{\Lambda}_{2,1} & \boldsymbol{C}_c\left(\boldsymbol{A}_c + \boldsymbol{B}_c^{(2)}\boldsymbol{C}_c\right) & -\boldsymbol{C}_c\left(\boldsymbol{B}_c^{(1)} + \boldsymbol{B}_c^{(2)}\boldsymbol{D}_c\right) & -\boldsymbol{C}_c\boldsymbol{B}_c^{(2)} \end{bmatrix}}_{\boldsymbol{\Lambda}} \begin{bmatrix} \boldsymbol{x}_p \\ \boldsymbol{x}_c \\ \boldsymbol{e}_y \\ \boldsymbol{e}_{u_1} \end{bmatrix} + \underbrace{\begin{bmatrix} \boldsymbol{B}_p^{(2)} \\ \boldsymbol{O} \\ \boldsymbol{C}_p\boldsymbol{B}_p^{(2)} \\ \boldsymbol{O} \end{bmatrix}}_{\boldsymbol{\Omega}} \boldsymbol{u}_2. \tag{32}$$

bound on the maximum perturbation bound that results from the insertion of the network, in accordance with a bound on the unknown input.

Theorem 2. *For the* N_{et}*UIO (34) and for a Hurwitz* $\boldsymbol{\Lambda}_r$*, we have* $\boldsymbol{P} = \boldsymbol{P}^\top \succ \boldsymbol{O}$*, is the solution to the Lyapunov matrix equation*

$$\boldsymbol{\Lambda}_r^\top \boldsymbol{P} + \boldsymbol{P}\boldsymbol{\Lambda}_r = -2\boldsymbol{Q},$$

for a given $\boldsymbol{Q} = \boldsymbol{Q}^\top \succ \boldsymbol{O}$ *and if* $\|\boldsymbol{u}_2\| < \mu\|\boldsymbol{r}\|$*, where* $\mu > 0$*, and if* $(\lambda_{min}(\boldsymbol{Q}) - \mu\lambda_{max}(\boldsymbol{P})\|\boldsymbol{\Omega}\|) > 0$ *then if the norm of the network induced perturbation* $\delta\boldsymbol{\Lambda}_r$ *satisfies the following inequality,*

$$\|\delta\boldsymbol{\Lambda}_r\| \leq \frac{\lambda_{min}(\boldsymbol{Q}) - \mu\lambda_{max}(\boldsymbol{P})\|\boldsymbol{\Omega}\|}{\lambda_{max}(\boldsymbol{P})}$$

then the origin is a globally exponentially stable equilibrium point of the N_{et}*UIO.*

Proof: Since $\boldsymbol{\Lambda}_r$ is asymptotically stable, the solution $\boldsymbol{P}$ to the Lyapunov matrix equation

$$\boldsymbol{\Lambda}_r^\top \boldsymbol{P} + \boldsymbol{P}\boldsymbol{\Lambda}_r = -2\boldsymbol{Q},$$

is symmetric positive definite (i.e., $\boldsymbol{P} = \boldsymbol{P}^\top \succ \boldsymbol{O}$.) for a given $\boldsymbol{Q} = \boldsymbol{Q}^\top \succ \boldsymbol{O}$.

Let $V(\boldsymbol{r}) = \frac{1}{2}\boldsymbol{r}^\top\boldsymbol{P}\boldsymbol{r}$ be a Lyapunov candidate function, used to verify stability and establish bounds on the network-induced perturbation. Particularly, we need to find a bound on the perturbation $\delta\boldsymbol{\Lambda}_r$ such that $\dot{V}(\boldsymbol{x}) < 0$. Taking the derivative of the Lyapunov candidate function, we get,

$$\dot{V}(\boldsymbol{r}) = \boldsymbol{r}^\top\boldsymbol{P}\dot{\boldsymbol{r}} \tag{35}$$

Substituting (34) into (35), we get:

$$\begin{aligned} \dot{V}(\boldsymbol{r}) &= \boldsymbol{r}^\top\boldsymbol{P}\left(\boldsymbol{\Lambda}_r\boldsymbol{r} + \delta\boldsymbol{\Lambda}_r\boldsymbol{r} + \boldsymbol{\Omega}\boldsymbol{u}_2\right) \\ &= \boldsymbol{r}^\top\boldsymbol{P}\boldsymbol{\Lambda}_r\boldsymbol{r} + \boldsymbol{r}^\top\boldsymbol{P}\delta\boldsymbol{\Lambda}_r\boldsymbol{r} + \boldsymbol{r}^\top\boldsymbol{P}\boldsymbol{\Omega}\boldsymbol{u}_2 \\ &= \frac{1}{2}\boldsymbol{r}^\top\boldsymbol{P}\boldsymbol{\Lambda}_r\boldsymbol{r} + \frac{1}{2}\boldsymbol{r}^\top\boldsymbol{P}\boldsymbol{\Lambda}_r\boldsymbol{r} + \boldsymbol{r}^\top\boldsymbol{P}\delta\boldsymbol{\Lambda}_r\boldsymbol{r} \\ &\quad + \boldsymbol{r}^\top\boldsymbol{P}\boldsymbol{\Omega}\boldsymbol{u}_2 \\ &= \frac{1}{2}\boldsymbol{r}^\top\boldsymbol{P}\boldsymbol{\Lambda}_r\boldsymbol{r} + \frac{1}{2}\boldsymbol{r}^\top\boldsymbol{\Lambda}_r^\top\boldsymbol{P}\boldsymbol{r} + \boldsymbol{r}^\top\boldsymbol{P}\delta\boldsymbol{\Lambda}_r\boldsymbol{r} \\ &\quad + \boldsymbol{r}^\top\boldsymbol{P}\boldsymbol{\Omega}\boldsymbol{u}_2 \\ &= \frac{1}{2}\boldsymbol{r}^\top\left(\boldsymbol{P}\boldsymbol{\Lambda}_r + \boldsymbol{\Lambda}_r^\top\boldsymbol{P}\right)\boldsymbol{r} + \boldsymbol{r}^\top\boldsymbol{P}\delta\boldsymbol{\Lambda}_r\boldsymbol{r} + \boldsymbol{r}^\top\boldsymbol{P}\boldsymbol{\Omega}\boldsymbol{u}_2 \\ &= -\boldsymbol{r}^\top\boldsymbol{Q}\boldsymbol{r} + \boldsymbol{r}^\top\boldsymbol{P}\delta\boldsymbol{\Lambda}_r\boldsymbol{r} + \boldsymbol{r}^\top\boldsymbol{P}\boldsymbol{\Omega}\boldsymbol{u}_2 \end{aligned} \tag{36}$$

For a symmetric positive definite matrix $\boldsymbol{Q}$, we have

$$-\lambda_{\max}(\boldsymbol{Q})\|\boldsymbol{r}\|^2 \leq -\boldsymbol{r}^\top\boldsymbol{Q}\boldsymbol{r} \leq -\lambda_{\min}(\boldsymbol{Q})\|\boldsymbol{r}\|^2.$$

Also since $\boldsymbol{P}$ is a symmetric positive definite matrix, $\|\boldsymbol{P}\| = \lambda_{\max}(\boldsymbol{P})$. Also,

$$\boldsymbol{r}^\top\boldsymbol{P}\delta\boldsymbol{\Lambda}_r\boldsymbol{r} = (\boldsymbol{r}^\top)(\boldsymbol{P}\delta\boldsymbol{\Lambda}_r\boldsymbol{r}) \leq \|\boldsymbol{r}^\top\|\|\boldsymbol{P}\delta\boldsymbol{\Lambda}_r\boldsymbol{r}\|$$

$$\leq \|\boldsymbol{r}^\top\|\|\boldsymbol{P}\delta\boldsymbol{\Lambda}_r\|\|\boldsymbol{r}\| \leq \lambda_{\max}(\boldsymbol{P})\|\boldsymbol{r}\|^2\|\delta\boldsymbol{\Lambda}_r\|.$$

In addition, since $\|\boldsymbol{u}_2\| < \mu\|\boldsymbol{x}\|$ we also get,

$$\boldsymbol{r}^\top\boldsymbol{P}\boldsymbol{\Omega}\boldsymbol{u}_2 \leq \|\boldsymbol{r}^\top\|\|\boldsymbol{P}\|\|\boldsymbol{\Omega}\|\|\boldsymbol{u}_2\| \leq \mu\lambda_{\max}(\boldsymbol{P})\|\boldsymbol{r}\|^2\|\boldsymbol{\Omega}\|.$$

Applying the above inequalities in (36), we get:

$$\begin{aligned} \dot{V} &= -\boldsymbol{r}^\top\boldsymbol{Q}\boldsymbol{r} + \boldsymbol{r}^\top\boldsymbol{P}\delta\boldsymbol{\Lambda}_r\boldsymbol{r} + \boldsymbol{r}^\top\boldsymbol{P}\boldsymbol{\Omega}\boldsymbol{u}_2 \\ &\leq -\lambda_{\min}(\boldsymbol{Q})\|\boldsymbol{r}\|^2 + \lambda_{\max}(\boldsymbol{P})\|\boldsymbol{r}\|^2\|\delta\boldsymbol{\Lambda}_r\| \\ &\quad + \mu\lambda_{\max}(\boldsymbol{P})\|\boldsymbol{r}\|^2\|\boldsymbol{\Omega}\| \\ \dot{V} &\leq \left(-\lambda_{\min}(\boldsymbol{Q}) + \lambda_{\max}(\boldsymbol{P})\|\delta\boldsymbol{\Lambda}_r\| + \mu\lambda_{\max}(\boldsymbol{P})\|\boldsymbol{\Omega}\|\right)\|\boldsymbol{r}\|^2. \end{aligned}$$

Hence, if the networked induced perturbation satisfies the following inequality:

$$\|\delta\boldsymbol{\Lambda}_r\| \leq \frac{\lambda_{\min}(\boldsymbol{Q}) - \mu\lambda_{\max}(\boldsymbol{P})\|\boldsymbol{\Omega}\|}{\lambda_{\max}(\boldsymbol{P})},$$

then the origin is a globally exponentially stable equilibrium point of the $\mathrm{N_{et}}$UIO. This ends the proof. ■

VI. Network Modeling as Perturbation, Another Approach

Walsh *et al.* [22] proposed another approach to analyze the stability of the perturbed system for a general NCS with no unknown inputs. The approach is based on modeling the network induced error states $(\boldsymbol{e}_y, \boldsymbol{e}_{u_1})$ as a perturbation to the dynamics of the closed-loop dynamics of the plant and controller. In this section, we apply the network modeling approach in [22] for NCSs with unknown inputs. Precisely, the dynamics of the perturbed system with the network and unknown input can be written as follows:

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{\Lambda}_1\boldsymbol{x}(t) + \boldsymbol{\Lambda}_2\boldsymbol{e}(t) + \boldsymbol{\Omega}_1\boldsymbol{u}_2(t), \tag{37}$$

where

$$\boldsymbol{\Lambda}_2 = \begin{bmatrix} -\boldsymbol{B}_p^{(1)}\boldsymbol{D}_c & -\boldsymbol{B}_p^{(1)} \\ -\left(\boldsymbol{B}_c^{(1)} + \boldsymbol{B}_c^{(2)}\boldsymbol{D}_c\right) & -\boldsymbol{B}_c^{(2)} \end{bmatrix}, \ \boldsymbol{\Omega}_1 = \begin{bmatrix} \boldsymbol{B}_p^{(2)} \\ \boldsymbol{O} \end{bmatrix},$$

$\boldsymbol{\Lambda}_2\boldsymbol{e}(t)$ is the perturbation due to the network, and $\boldsymbol{\Omega}_1\boldsymbol{u}_2(t)$ is the perturbation due to the unknown input. Consider the time interval between transmissions: $t \in [t_i, t_{i+1}]$ where $i = 0, 1, 2, \ldots$, we get:

$$\hat{\boldsymbol{y}}(t) = \boldsymbol{y}(t_i) = \boldsymbol{C}_p\boldsymbol{x}_p(t_i)$$

and

$$\hat{\boldsymbol{u}}_1(t) = \boldsymbol{u}_1(t_i) = \boldsymbol{C}_c\boldsymbol{x}_c(t_i) + \boldsymbol{D}_c\boldsymbol{y}(t_i) = \boldsymbol{C}_c\boldsymbol{x}_c(t_i) + \boldsymbol{D}_c\boldsymbol{C}_p\boldsymbol{x}_p(t_i).$$

Recall that

$$\boldsymbol{e}_y(t) = \boldsymbol{y}(t) - \hat{\boldsymbol{y}}(t) = \boldsymbol{C}_p(\boldsymbol{x}(t) - \boldsymbol{x}(t_i))$$

and

$$\boldsymbol{e}_{u_1} = \boldsymbol{u}_1(t) - \hat{\boldsymbol{u}}_1(t) = \boldsymbol{C}_c\boldsymbol{x}_c(t) + \boldsymbol{D}_c\boldsymbol{C}_p\boldsymbol{x}_p(t) - \left(\boldsymbol{C}_c\boldsymbol{x}_c(t_i) + \boldsymbol{D}_c\boldsymbol{C}_p\boldsymbol{x}_p(t_i)\right)$$

$$= C_c(x_c(t) - x_c(t_i)) + D_c C_p(x_p(t) - x_p(t_i)).$$

Let $e_x(t) = x(t) - x(t_i)$, then we can write

$$\Lambda_2 e(t) = \Lambda_2 \begin{bmatrix} C_p & O \\ D_c C_p & C_c \end{bmatrix} e_x(t) = E e_x(t).$$

The dynamics of the perturbed system can now be written as,

$$\dot{x}(t) = \Lambda_1 x(t) + E e_x(t) + \Omega_1 u_2(t), \tag{38}$$

Theorem 3. *For the perturbed N_{et}UIO as represented in (38) and for a Hurwitz Λ_1, we have $P = P^\top \succ O$, is the solution to the Lyapunov matrix equation*

$$\Lambda_1^\top P + P\Lambda_1 = -2Q,$$

for a given $Q = Q^\top \succ O$. If $\|u_2\| < \mu_x\|x\|$, where $\mu_x > 0$, and if $\lambda_{min}(Q) - \mu_x\lambda_{max}(P)\|\Omega_1\| > 0$ then if the norm of the network induced perturbation $\|e_x\|$ satisfies $\|e_x\| < \zeta\|x\|$ where

$$\zeta \le \frac{\lambda_{min}(Q) - \mu_x\|\Omega_1\|\lambda_{max}(P)}{\lambda_{max}(P)\|E\|},$$

the origin is a globally exponentially stable equilibrium point of the perturbed N_{et}UIO (38).

Proof: The proof presented here is very similar to Since Λ_1 is asymptotically stable, the solution P to the Lyapunov matrix equation

$$\Lambda_1^\top P + P\Lambda_1 = -2Q,$$

is symmetric positive definite (i.e., $P = P^\top \succ O$.) for a given $Q = Q^\top \succ O$.

Let $V(x) = \frac{1}{2}x^\top P x$ be a Lyapunov candidate function, used to verify stability and establish bounds on the network-induced perturbation. Particularly, we need to find a bound on the perturbation $\|e_x\|$ such that $\dot{V}(x) < 0$. Taking the derivative of the Lyapunov candidate function, we get,

$$\dot{V}(x) = x^\top P\dot{x} \tag{39}$$

Substituting (38) into (39), we get:

$$\begin{aligned} \dot{V}(x) &= x^\top P(\Lambda_1 x + E e_x + \Omega_1 u_2) \\ &= x^\top P\Lambda_1 x + x^\top P E e_x + x^\top P\Omega_1 u_2 \\ &= \frac{1}{2}x^\top P\Lambda_1 x + \frac{1}{2}x^\top P\Lambda_1 x \\ &\quad + x^\top P E e_x + x^\top P\Omega_1 u_2 \\ &= \frac{1}{2}x^\top P\Lambda_1 x + \frac{1}{2}x^\top \Lambda_1^\top P x \\ &\quad + x^\top P E e_x + x^\top P\Omega_1 u_2 \\ &= \frac{1}{2}x^\top \left(P\Lambda_1 + \Lambda_1^\top P\right) x + x^\top P E e_x + x^\top P\Omega_1 u_2 \\ &= -x^\top Q x + x^\top P E e_x + x^\top P\Omega_1 u_2 \end{aligned} \tag{40}$$

For a symmetric positive definite matrix Q, we have

$$-\lambda_{\max}(Q)\|x\|^2 \le -x^\top Q x \le -\lambda_{\min}(Q)\|x\|^2.$$

Also since P is a symmetric positive definite matrix, $\|P\| = \lambda_{\max}(P)$, and since $\|e_x\| < \zeta\|x\|$, then

$$\begin{aligned} x^\top P E x = (x^\top)(P E x) &\le \|x^\top\|\|P E x\| \\ &\le \zeta\|x^\top\|\|P E\|\|x\| \le \zeta\lambda_{\max}(P)\|x\|^2\|E\|. \end{aligned}$$

In addition, since $\|u_2\| < \mu_x\|x\|$ we also get,

$$\begin{aligned} x^\top P\Omega_1 u_2 &\le \|x^\top\|\|P\|\|\Omega_1\|\|u_2\| \\ &\le \mu_x\lambda_{\max}(P)\|x\|^2\|\Omega_1\|. \end{aligned}$$

Applying the above inequalities in (40), we get:

$$\begin{aligned} \dot{V} &= -x^\top Q x + x^\top P E x + x^\top P\Omega_1 u_2 \\ &\le -\lambda_{\min}(Q)\|x\|^2 + \zeta\lambda_{\max}(P)\|x\|^2\|E\| \\ &\quad + \mu_x\lambda_{\max}(P)\|x\|^2\|\Omega_1\| \\ \dot{V} &\le (-\lambda_{\min}(Q) + \zeta\lambda_{\max}(P)\|E\| + \mu_x\lambda_{\max}(P)\|\Omega_1\|)\,\|x\|^2. \end{aligned}$$

Hence, if the networked induced perturbation satisfies the following inequality:

$$\zeta \le \frac{\lambda_{\min}(Q) - \mu_x\lambda_{\max}(P)\|\Omega_1\|}{\lambda_{\max}(P)\|E\|},$$

then the origin is a globally exponentially stable equilibrium point. This ends the proof. ∎

We can compare our derived bound on the network induced perturbation ζ with the one derived in [22]. This comparison is carried out in the numerical simulations section.

VII. Numerical Examples for Network Effect as Time-Delay

In this section, we illustrate the usefulness of our proposed N_{et}UIO design with a single-input single-output (SISO) numerical example. In Section VII-A, we show the UIO design for the non-networked systems, while the design and stability analysis are included in Section VII-B.

A. UIO Design Example for A Non-Networked System

The given system example is an LTI system with one known input, one unknown input, one output(i.e., $n = 3, m_1 = 1, m_2 = 1, p = 1$). The system is modeled by:

$$A_p = \begin{bmatrix} -5 & 3 & 0 \\ 4 & -10 & 4 \\ 0 & 0 & -4 \end{bmatrix}, B_p^{(1)} = \begin{bmatrix} 1 \\ 0 \\ 2 \end{bmatrix}, B_p^{(2)} = \begin{bmatrix} 2 \\ 1 \\ 2 \end{bmatrix}, C_p = \begin{bmatrix} 2 \\ 4 \\ -1 \end{bmatrix}^T.$$

Before analyzing the networked UIO, we first follow the design algorithm for the UIO in [3] and we map the non-networked UIO to a general NCS form. We follow the steps in Algorithm 2, that can be found in Appendix B. Applying Algorithm 2, we get the following results:

1) Non-networked UIO existence:

$$\text{rank}(C_p B_p^{(2)}) = 1, \text{rank}(B_p^{(2)}) = 1. \text{ UIO exists.}$$

2) Compute the design matrix M:

$$M = B_p^{(2)}(C_p B_p^{(2)})^\dagger = \begin{bmatrix} 0.33 & 0.16 & 0.33 \end{bmatrix}^\top.$$

3) Compute $\tilde{P}$:

$$\tilde{P} = I_n - M C_p = \begin{bmatrix} 0.33 & -1.33 & 0.33 \\ -0.33 & 0.33 & 0.16 \\ -0.66 & -1.33 & 1.33 \end{bmatrix}.$$

4) Find Q:

$$Q = \begin{bmatrix} 0.87 & 0.09 & -0.66 \\ -0.35 & 0.19 & -0.33 \\ 0.33 & 0.97 & -0.66 \end{bmatrix}.$$

5) Compute $\tilde{A}_p$ and $\tilde{C}_p$:

$$\tilde{A}_p = \begin{bmatrix} -14.87 & -4.20 & 4.61 \\ -4.43 & -7.12 & 3.19 \\ -11.97 & -6.70 & 3.00 \end{bmatrix}, \tilde{C}_p = \begin{bmatrix} 0 & 0 & -2 \end{bmatrix}.$$

6) Eigenvalues of $\tilde{\boldsymbol{A}}_p^{(11)}$:

$$\text{eig}(\tilde{\boldsymbol{A}}_p^{(11)}) = \{-16.8023, -5.1977\} < 0.$$

7) Since $\tilde{\boldsymbol{A}}_p^{(11)}$ is stable with negative eigenvalues, then $(\tilde{\boldsymbol{A}}_p^{(11)}, \tilde{\boldsymbol{C}}_p^{(1)})$ is detectable.
8) Design of $\boldsymbol{L}$:

$$\boldsymbol{L} = \texttt{place}(\boldsymbol{A}_p^\top, \boldsymbol{C}_p^\top, [-10-11-12])^\top = \begin{bmatrix} 1.4348 \\ 4.6087 \\ 7.3043 \end{bmatrix}$$

where `place` is a closed-loop pole assignment using state feedback Matlab function.
9) Compute NCS parameters $\boldsymbol{A}_c, \boldsymbol{B}_c^{(1)}, \boldsymbol{B}_c^{(2)}$:

$$\boldsymbol{A}_c = \begin{bmatrix} -0.5362 & 27.2609 & -9.8986 \\ -1.5507 & -13.4348 & 2.9420 \\ -7.2754 & 0.7826 & -8.0290 \end{bmatrix},$$

$$\boldsymbol{B}_c^{(1)} = \begin{bmatrix} -2.16 \\ 0.50 \\ -2.33 \end{bmatrix}, \boldsymbol{B}_c^{(2)} = \begin{bmatrix} 1 \\ 0 \\ 2 \end{bmatrix}.$$

The UIO observer for the non-networked system can be written as follows:

$$\begin{aligned} \dot{\boldsymbol{x}}_c &= \boldsymbol{A}_c \boldsymbol{x}_c + \boldsymbol{B}_c^{(1)} \boldsymbol{y} + \boldsymbol{B}_c^{(2)} \boldsymbol{u}_1, \\ \hat{\boldsymbol{x}}_p &= \boldsymbol{x}_c + \boldsymbol{M}\boldsymbol{y}. \end{aligned}$$

The initial plant state are randomly selected to be $\boldsymbol{x}_p(0) = \begin{bmatrix} 0.356 & -0.0422 & -0.719 \end{bmatrix}^\top$, the initial UIO state to

$$\boldsymbol{x}_c(0) = (\boldsymbol{I}_n - \boldsymbol{M}\boldsymbol{C}_p) \begin{bmatrix} 10 & 10 & 10 \end{bmatrix}^\top = \begin{bmatrix} -6.66 & -1.66 - 6.66 \end{bmatrix}^\top,$$

the unknown input $u_2(t) = 0.1(\sin(t) + \cos(t))$. Figure 3 shows a very good estimation for the plant states for the non-networked UIO, hence the estimation error converges rapidly to zero.

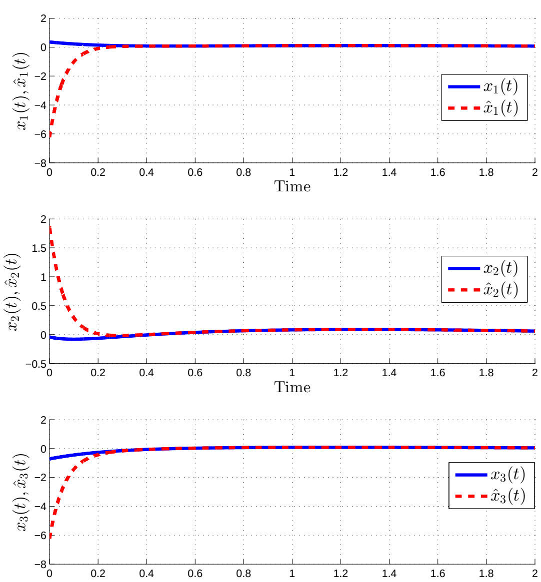

Fig. 3. UIO State Estimation For Non-Networked System ($\tau = 0$). The non-networked UIO perfectly tracks the plant state.

B. UIO Design Example for the Networked System

After finding the UIO parameters for the non-networked system as in Section VII-A, we follow the steps mentioned in Sections II and III to map the UIO to the typical NCS configuration and then follow Algorithm 2 (in Appendix C).

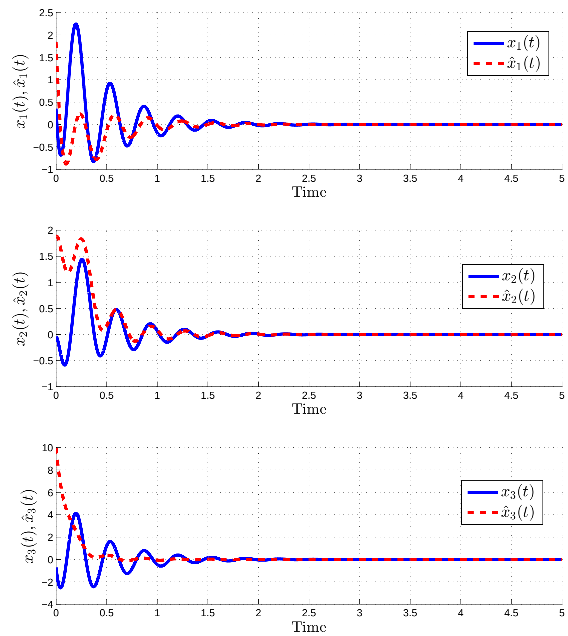

Fig. 4. UIO State Estimation for the N_{et}UIO ($\tau = 0.03$ sec). After solving for the $\boldsymbol{M}$ matrix, the designed N_{et}UIO succeeds in tracking the plant state. The transients in the plant states are due to the addition of the network.

Although the plant is already stable, we can design a state-feedback gain matrix $\boldsymbol{K}$ such that the closed-loop poles are further away in the left-half plane using the `place` Matlab function:

$$\boldsymbol{K} = \texttt{place}(\boldsymbol{A}_p, B_p^{(1)}, [-10-15-20]) = \begin{bmatrix} 36.0 & 16.5 & -5.0 \end{bmatrix}.$$

We can now solve for $\boldsymbol{M}$ in (22) by applying Algorithm 1 in Appendix A:

$$\boldsymbol{M} = \begin{bmatrix} 0.1673 & 0.1667 & 0.0012 \end{bmatrix}^\top.$$

Choosing the same unknown inputs and initial conditions for the networked observer and plant as in Section VII-A, we simulate and analyze the stability of the N_{et}UIO by following Algorithm 3 in Appendix C. Through varying τ, we find the experimental bound on τ that guarantees the stability of the N_{et}UIO as follows: $\tau^{\max}_{\text{exper.}} = 0.0485$ sec.

Recall that the bound on τ would guarantee the stability of the N_{et}UIO is given by Theorem 1. Evaluating the coefficients for the second degree polynomial (in terms of τ), we get the following theoretical bound on the time-delay:

$$0 < \tau < \tau^{\max}_{\text{theor.}} = 0.0327 \text{ sec.}$$

Hence, the derived upper bound ($\tau^{\max}_{\text{theor.}} = 0.0327$ sec.) for the time-delay that guarantees the convergence of the estimation error for the N_{et}UIO is close to the actual one ($\tau^{\max}_{\text{exper.}} = 0.0485$ sec) and not too conservative. This would allow for a better determination of a sampling period of the NCS. Figure 4 show the state trajectories of the networked system for $\tau = 0.03$ sec. The plots show that the networked system is stable for this value of the time-delay, albeit exhibiting more transient response than the non-networked case due to the network effect.

VIII. Numerical Example for Network Effect as Perturbation

In this section, we test the derived bounds from Sections V and VI on the perturbation for the N_{et}UIO. We consider the SISO example, previously discussed in Section VII-A.

After finding the design parameters for the non-networked UIO ($\boldsymbol{A}_c, \boldsymbol{B}_c^{(1)}, \boldsymbol{B}_c^{(2)}, \boldsymbol{C}_c$ and $\boldsymbol{D}_c$) and using the system parameters for the plant ($\boldsymbol{A}_p, \boldsymbol{B}_p^{(1)}, \boldsymbol{B}_p^{(2)}, \boldsymbol{C}_p$), we compute the the parameters of the closed loop dynamics ($\boldsymbol{\Lambda}_r, \delta\boldsymbol{\Lambda}_r, \boldsymbol{\Omega}$) as in (34). Then, we find a matrix $\boldsymbol{P} = \boldsymbol{P}^\top \succ \boldsymbol{O}$, a solution to the Lyapunov matrix equation

$$\boldsymbol{\Lambda}_r^\top \boldsymbol{P} + \boldsymbol{P}\boldsymbol{\Lambda}_r = -2\boldsymbol{Q},$$

for a given $\boldsymbol{Q} = \boldsymbol{Q}^\top \succ \boldsymbol{O}$ such that $(\lambda_{\min}(\boldsymbol{Q}) - \mu\lambda_{\max}(\boldsymbol{P})\|\boldsymbol{\Omega}\|) > 0$ using Linear Matrix Inequalities (LMI). We now apply Theorem 2 and find the maximum allowable perturbation due to the network that guarantees the stability of the closed-loop system (sufficiency condition). Precisely, if $\|\delta\boldsymbol{\Lambda}_r\| \le 0.0674$, then the system is stable. The determination of this bound would improve the design of the N_{et}UIO by setting constraints on some of the parameters of the controller and observer. For example, the design matrices $\boldsymbol{K}$, $\boldsymbol{L}$ and $\boldsymbol{M}$ (and subsequently $\boldsymbol{A}_c, \boldsymbol{B}_c, \boldsymbol{C}_c$ and $\boldsymbol{D}_c$) can be designed such that $\|\delta\boldsymbol{\Lambda}_r\| \le 0.0674$. A LMI-based method could then be used to find these design matrices with norm constraints. This bound is a sufficient condition for the stability of the N_{et}UIO.

In Section VI, we show that if the norm of the network induced perturbation satisfies $\|\boldsymbol{e}_x\| < \zeta\|\boldsymbol{x}\|$ where

$$\zeta \le \frac{\lambda_{\min}(\boldsymbol{Q}) - \mu_x\|\boldsymbol{\Omega_1}\|\lambda_{\max}(\boldsymbol{P})}{\lambda_{\max}(\boldsymbol{P})\|\boldsymbol{E}\|},$$

the origin is a globally exponentially stable equilibrium point of the perturbed N_{et}UIO of the following form,

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{\Lambda_1}\boldsymbol{x}(t) + \boldsymbol{E}\boldsymbol{e}_x(t) + \boldsymbol{\Omega_1}\boldsymbol{u}_2(t).$$

After computing $\boldsymbol{E}$ and $\boldsymbol{\Omega}_1$ and finding $\boldsymbol{P} = \boldsymbol{P}^\top \succ \boldsymbol{O}$, the bound that would render the system unstable is $\|\boldsymbol{e}_x\| < 0.0011 \cdot \|\boldsymbol{x}\|$. Walsh *et al.* [22] derived a similar bound for the perturbation for the NCS with no unknown inputs. Particularly, the authors is [22] claim that the no unknown inputs NCS is asymptotically stable if $\|\boldsymbol{e}_x(t)\| \le \gamma\|\boldsymbol{x}(t)\|$ where

$$\gamma = \frac{||\boldsymbol{\Lambda}_1||\ ||\boldsymbol{\Lambda}_1 + \boldsymbol{E}||^{-1}(e^{||\boldsymbol{\Lambda}_1+\boldsymbol{E}||\tau_m} - 1)e^{||\boldsymbol{\Lambda}_1+\boldsymbol{E}||\tau_m}}{1 - ||\boldsymbol{E}||\ ||\boldsymbol{\Lambda}_1 + \boldsymbol{E}||^{-1}(e^{||\boldsymbol{\Lambda}_1+\boldsymbol{E}||\tau_m} - 1)},$$

where

$$\tau_m < \frac{\lambda_{min}(\boldsymbol{Q})}{16\lambda_2\sqrt{\frac{\lambda_2}{\lambda_1}}\|\boldsymbol{\Lambda}\|^2\left(1 + \sqrt{\frac{\lambda_2}{\lambda_1}}\right)\sum_{i=1}^{p} i},$$

is the maximum allowable transfer interval (which is similar to the time-delay derived in the previous sections), $\lambda_1 = \lambda_{\min}(\boldsymbol{P})$, $\lambda_2 = \lambda_{\max}(\boldsymbol{P})$, p is the number of sensor nodes in the NCS, and

$$1 - ||\boldsymbol{E}||\ ||\boldsymbol{\Lambda}_1 + \boldsymbol{E}||^{-1}(e^{||\boldsymbol{\Lambda}_1+\boldsymbol{E}||\tau_m} - 1) > 0,$$

for a τ_m that satisfies the aforementioned inequality. Numerical results for the bound derived in [22] show that the SISO N_{et}UIO is unstable for $\|\boldsymbol{e}_x\| > 2.3496 \cdot 10^{-4}\|\boldsymbol{x}\|$, which is too conservative compared to the computed bound.

IX. Conclusions and Future Work

Albeit it provides many advantages such as the ease of use, flexibility, and utilization of more efficient control laws, the addition of a communication network in the feedback loops of control systems complicates their analysis and design. In this paper, we discuss an Unknown Input Observer-based design for Networked Control Systems. The network effect is analyzed and discussed. First, we review an UIO design for non-networked system from the UIO literature. Second, we derive the dynamics of the N_{et}UIO. Third, we design the N_{et}UIO such the effect of higher delay order terms are nullified, assuring that the effect of unknown inputs to the plant state estimation is minimized. Fourth, we derive a bound on the maximum allowable time-delay. Fifth, two perturbation models that represent the network effect are studied and network perturbation bounds are established. Finally, numerical examples are shown to illustrate the usefulness of the proposed model.

The determination of an upper bound on the network induced time-delay is significantly important in the design of an NCS so that a suitable sampling period is chosen. When the time-delay is greater than the sampling period in an NCS, then the global stability of the overall NCS can not be guaranteed as discussed by Kim *et al.* [23]. The results show that the derived bound for N_{et}UIO is accurate. In addition, finding perturbation bounds due to the unknown inputs and presence of the network would facilitate and improve the design of controllers and observers such that the overall system performs desirably.

In our future work, we plan to consider other UIO architectures and Sliding Mode Observers for networked systems as the ones developed in [24] and [25]. We also intend to apply the proposed model of the N_{et}UIO for power networks with variable time-delay, where fault detection and isolation techniques can be employed to better monitor the typically networked power systems.

Acknowledgments

The authors gratefully acknowledge the constructive comments by the reviewers and the financial support from the National Science Foundation through NSF CMMI Grants 1201114 and 1265622.

References

[1] J. P. Hespanha, P. Naghshtabrizi, and Y. Xu, "A survey of recent results in networked control systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 138–162, January 2007.

[2] J. Allemong, "State estimation fundamentals for successful deployment," in *Proceedings of the 2005 IEEE Power Engineering Society General Meeting*, June 2005.

[3] S. Hui and S. H. Żak, "Observer design for systems with unknown inputs," *Int. J. Appl. Math. Comput. Sci*, vol. 15, no. 4, pp. 431–446, 2005.

[4] D. G. Luenberger, "Observers for multivariable systems," *IEEE Trans. Automat. Contr*, vol. 11, no. 2, pp. 190–197, 1966.

[5] ——, *Introduction to Dynamic Systems: Theory, Models, and Applications*. New York: John Wiley & Sons, 1979.

[6] S. P. Bhattacharyya, "Observer design for linear systems with unknown inputs," *IEEE Transactions on Automatic Control*, vol. 23, no. 3, pp. 483–484, June 1978.

[7] S.-K. Chang, W.-T. You, and P.-L. Hsu, "Design of general structured observers for linear systems with unknown inputs," *Journal of the Franklin Institute*, vol. 334B, no. 2, pp. 213–232, March 1997.

[8] J. Chen, R. Patton, and H. Y. Zhang, "Design of unknown input observers and robust fault detection filters," *International Journal of Control*, vol. 63, no. 1, pp. 85–105, 1996.

[9] J. Chen and R. Patton, *Robust model-based fault diagnosis for dynamic systems*. Norwell, MS: Kluwer Academic Publishers, 1999.

[10] M. Corless and J. Tu, "State and input estimation for a class of uncertain systems," *Automatica*, vol. 34, no. 6, pp. 757–764, 1998.

[11] M. Darouach, M. Zasadzinski, and S. J. Xu, "Full-order observers for linear systems with unknown inputs," *IEEE Trans. Automat. Contr*, vol. 39, no. 3, pp. 606–609, 1994.

[12] S. Hui and S. H. Żak, "Low-order unknown input observers," in *Proceedings of the 2005 American Control Conference*, June 2005, pp. 4192–4197.

[13] A. Teixeira, H. Sandberg, and K. H. Johansson, "Networked control systems under cyber attacks with applications to power networks," in *In Proceedings of the American Control Conference*, July 2010, pp. 3690–3696.

[14] M. Darouach, "Existence and design of functional observers for linear systems," *Automatic Control, IEEE Transactions on*, vol. 45, no. 5, pp. 940–943, 2000.

[15] T. Fernando, H. Trinh, and L. Jennings, "Functional observability and the design of minimum order functional observers," *IEEE Transactions on Automatic Control*, vol. 55, no. 5, pp. 1268–1273, 2010.

[16] M. Cloosterman, N. V. de Wouw, W. P. M. H. Heemels, and H. Nijmeijer, "Stability of networked control systems with uncertain time-varying delays," *Automatic Control, IEEE Transactions on*, vol. 54, no. 7, pp. 1575–1580, July 2009.

[17] M. Cloosterman, N. van de Wouw, W. P. M. H. Heemels, and H. Nijmeijer, "Robust stability of networked control systems with time-varying network-induced delays," in *2006 45th IEEE Conference on Decision and Control*, December 2006, pp. 4980–4985.

[18] M. Tabbara, D. Nešić, and A. R. Teel, "Networked control systems: Emulation based design," *Networked Control Systems, ser. Series in Intelligent Control and Intelligent Automation*, 2007.

[19] F. Wang and D. Liu, *Networked Control Systems*. Springer, 2008.

[20] I. Kolmanovsky, I. Sivergina, and J. Sun, "Simultaneous input and parameter estimation with input observers and set-membership parameter bounding: theory and an automotive application," *Int. J. Adapt. Control Signal Process*, vol. 20, pp. 225–246, 2006.

[21] S. H. Żak, *Systems and Control*. New York: Oxford University Press, 2003.

[22] G. C. Walsh, H. Ye, and L. G. Bushnell, "Stability analysis of networked control systems," *IEEE Transactions on Control Systems Technology*, vol. 10, no. 3, pp. 438–446, May 2002.

[23] D. S. Kim, Y. S. Lee, W. H. Kwon, and H. S. Park, "Maximum allowable delay bounds of networked control systems," *Control Eng. Practice*, vol. 11, pp. 1301–1313, 2003.

[24] T. Fernando, S. MacDougall, V. Sreeram, and H. Trinh, "Existence conditions for unknown input functional observers," *International Journal of Control*, vol. 86, no. 1, pp. 22–28, 2013.

[25] T. L. Fernando and H. M. Trinh, "A system decomposition approach to the design of functional observers," *International Journal of Control*, vol. 87, no. 9, pp. 1846–1860, 2014.

[26] L. Rodman, *Topics in Quaternion Linear Algebra*. Princeton, NJ: Princeton University Press, 2014.

[27] V. V. Palin, "Solvability of quadratic matrix equations," *Moscow University Mathematics Bulletin*, vol. 63, no. 6, pp. 256–261, 2008.

[28] G. I. Shurbet, T. O. Lewis, and T. L. Boullion, "Quadratic matrix equations," *Ohio J. Sci*, vol. 74, no. 5, 1974.

[29] J. Sivaramakrishnan, "Bilateral matrix quadratic equation solution," MATLAB Centeral, Tech. Rep., 2010. [Online]. Available: http://www.mathworks.com/matlabcentral/fileexchange/26956-solve-bilateral-matrix-quadratic-equation

[30] R. H. Bartels and G. W. Stewart, "Solution of the matrix equation $ax + xb = c$," in *Communications of the ACM*, 1972, pp. 820–826.

Appendices

Appendix A: $\mathrm{N_{et}}$UIO Design Matrix $\boldsymbol{M}$

In this appendix, we provide the reader with a discussion on the existence of a solution for the Networked UIO design equation,

$$\boldsymbol{M}\mathcal{A}\boldsymbol{M} + \boldsymbol{M}\mathcal{B} + \mathcal{C}\boldsymbol{M} + \mathcal{D} = \boldsymbol{O}_{n\times p},$$

where $\mathcal{A} \in \mathbb{R}^{p\times n}, \mathcal{B} \in \mathbb{R}^{p\times p}, \mathcal{C} \in \mathbb{R}^{n\times n}$ and $\mathcal{D} \in \mathbb{R}^{n\times p}$ are all constant matrices and $\boldsymbol{M} \in \mathbb{R}^{n\times p}$ is the design variable. A solution algorithm is also presented. In [26], [27] and [28], the authors present conditions on the existence of a solution to the BMQE, with the *same exact* dimensions to our problem in (IX) (with of course different notations) in [26]. In fact, and under certain existence conditions presented in [26], the author shows that a BMQE of the same form in (22) has a finite number of solutions, and that the number of solutions does not exceed $\frac{(p+n)!}{p!n!}$.

The above BMQE can be solved through different methods. We use the algorithm summarized below to find $\boldsymbol{M}$. This algorithm is similar to the one developed in [29]. The basic idea is to utilize the `sylv` solver on MATLAB to find $\boldsymbol{M}$. The `sylv`$(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C})$ solves the Sylvester equation $\boldsymbol{A}\boldsymbol{X} + \boldsymbol{X}\boldsymbol{B} = \boldsymbol{C}$. Hence, and as shown in the algorithm, by doing a change of variables and by substituting the arguments of the `sylv` solver $(\mathcal{C} + \boldsymbol{M}\mathcal{A}, \mathcal{B} + \mathcal{A}\boldsymbol{M}, -\boldsymbol{Z})$, the solver eventually solves the same BMQE derived above. The main solution existence condition using this algorithm is the same as for the Sylvester's matrix equation [30].

Algorithm 1 $\mathrm{N_{et}}$UIO: Design Matrix $\boldsymbol{M}$

1: **input:** $\{\boldsymbol{C}_p, \boldsymbol{R}, \boldsymbol{B}_p^{(2)}, \boldsymbol{T}\}$
2: **compute** $\mathcal{A}, \mathcal{B}, \mathcal{C}$ and $\mathcal{D}$
3: **initialize variables:**

$$\boldsymbol{M} = \boldsymbol{O}, \text{NumIter} = 0, \text{MaxIter} = 1000, \text{tol} = 10^{-7}$$

4: **cocmpute** convergence matrix:

$$\boldsymbol{Z} = \boldsymbol{M}\mathcal{A}\boldsymbol{M} + \boldsymbol{M}\mathcal{B} + \mathcal{C}\boldsymbol{M} + \mathcal{D}$$

5: % Idealy, we want $\|\boldsymbol{Z}\| \approx 0$ to get convergence
6: **while** (NumIter $<$ MaxIter AND $\|\boldsymbol{Z}\| >$ tol)
7: $\mathcal{M} =$ `sylv`$(\mathcal{C} + \boldsymbol{M}\mathcal{A}, \mathcal{B} + \mathcal{A}\boldsymbol{M}, -\boldsymbol{Z})$
8: **IF** no $\mathcal{M}$ exists that solves `sylv(.)`
9: STOP. The $\mathrm{N_{et}}$UIO design does not exist
10: **ELSE**
11: $\boldsymbol{M} = \boldsymbol{M} + \mathcal{M}$
12: NumIter $=$ NumIter $+ 1$
13: $\boldsymbol{Z} = \boldsymbol{M}\mathcal{A}\boldsymbol{M} + \boldsymbol{M}\mathcal{B} + \mathcal{C}\boldsymbol{M} + \mathcal{D}$
14: **end**
15: **output:** $\{\boldsymbol{M}\}$

Appendix B: NCS Construction and Non-networked UIO Design

This appendix includes the UIO-design algorithm from [3] and the NCS state-space configuration. The design algorithm is as follows:

Algorithm 2 UIO Design & NCS Construction

1: **input:** $\{\boldsymbol{A}_p, \boldsymbol{B}_p^{(1)}, \boldsymbol{B}_p^{(2)}, \boldsymbol{C}_p\}$
2: Check rank$(\boldsymbol{C}_p\boldsymbol{B}_p^{(2)})$:
if rank$(\boldsymbol{C}_p\boldsymbol{B}_p^{(2)}) <$ rank$(\boldsymbol{B}_p^{(2)})$
then STOP. Observer does not exist.
3: Compute the design matrix $\boldsymbol{M}$:

$$\boldsymbol{M} = \boldsymbol{B}_p^{(2)}(\boldsymbol{C}_p\boldsymbol{B}_p^{(2)})^{\dagger}$$

4: Compute the projector matrix $\tilde{\boldsymbol{P}}$:

$$\tilde{\boldsymbol{P}} = \boldsymbol{I}_n - \boldsymbol{M}\boldsymbol{C}_p$$

5: Find $\boldsymbol{Q}$ such that

$$\tilde{\boldsymbol{P}} = \boldsymbol{Q}\boldsymbol{P}\boldsymbol{Q}^{-1} \text{ where } \boldsymbol{P} = \begin{bmatrix} \boldsymbol{I}_{n-m_2} & \boldsymbol{O} \\ \boldsymbol{O} & \boldsymbol{O} \end{bmatrix}$$

6: Compute $\tilde{\boldsymbol{A}}_p = \boldsymbol{Q}^{-1}\boldsymbol{A}_p\boldsymbol{Q} = \begin{bmatrix} \tilde{\boldsymbol{A}}_p^{(11)} & \tilde{\boldsymbol{A}}_p^{(12)} \\ \tilde{\boldsymbol{A}}_p^{(21)} & \tilde{\boldsymbol{A}}_p^{(22)} \end{bmatrix}$ and $\tilde{\boldsymbol{C}}_p = \boldsymbol{C}_p\boldsymbol{Q} = \begin{bmatrix} \tilde{\boldsymbol{C}}_p^{(1)} & \tilde{\boldsymbol{C}}_p^{(2)} \end{bmatrix}$
7: Check the eigenvalues of $\tilde{\boldsymbol{A}}_p^{(11)}$
8: Check the detectability of the pair $(\tilde{\boldsymbol{A}}_p^{(11)}, \tilde{\boldsymbol{C}}_p^{(1)})$:
if $(\tilde{\boldsymbol{A}}_p^{(11)}, \tilde{\boldsymbol{C}}_p^{(1)})$ is not detectable
then STOP. Observer does not exist.
9: Design $\boldsymbol{L}$ such that $\tilde{\boldsymbol{A}}_p$ is stable and observer convergence rate is improved
10: Compute $\boldsymbol{A}_c, \boldsymbol{B}_c^{(1)}, \boldsymbol{B}_c^{(2)}$ as follows:

$$\boldsymbol{A}_c = \tilde{\boldsymbol{P}}(\boldsymbol{A}_p - \boldsymbol{L}\boldsymbol{C}_p), \boldsymbol{B}_c^{(2)} = \tilde{\boldsymbol{P}}\boldsymbol{B}_p^{(1)}$$

$$\boldsymbol{B}_c^{(1)} = \tilde{\boldsymbol{P}}(\boldsymbol{A}_p\boldsymbol{M} + \boldsymbol{L} - \boldsymbol{L}\boldsymbol{C}_p\boldsymbol{M}).$$

11: **output:** $\{\boldsymbol{A}_c, \boldsymbol{B}_c^{(1)}, \boldsymbol{B}_c^{(2)}\}$

Appendix C: NCS Construction and Non-networked UIO Design

To analyze the stability of the $\mathrm{N_{et}}$UIO, we follow this algorithm:

Algorithm 3 $\mathrm{N_{et}}$UIO Design and Stability Analysis

---

1: **input:** $\{\boldsymbol{A}_p, \boldsymbol{A}_c, \boldsymbol{B}_p^{(1)}, \boldsymbol{B}_p^{(2)}, \boldsymbol{B}_c^{(1)}, \boldsymbol{B}_c^{(2)}, \boldsymbol{C}_p, \boldsymbol{C}_c, \boldsymbol{D}_c, \boldsymbol{K}, \boldsymbol{T}, \boldsymbol{Q}\}$
2: Solve for $\boldsymbol{M}$ in (22) by applying Algorithm 1:

$$\boldsymbol{M}\mathcal{A}\boldsymbol{M} + \boldsymbol{M}\mathcal{B} + \mathcal{C}\boldsymbol{M} + \mathcal{D} = \boldsymbol{O}_{n\times p},$$

3: Given $\boldsymbol{A}_p, \boldsymbol{A}_c, \boldsymbol{B}_p^{(1)}, \boldsymbol{B}_p^{(2)}, \boldsymbol{B}_c^{(1)}, \boldsymbol{B}_c^{(2)}, \boldsymbol{C}_p, \boldsymbol{C}_c$ and $\boldsymbol{D}_c$, compute $\boldsymbol{\Gamma}, \boldsymbol{\Gamma}_0, \boldsymbol{\Gamma}_1, \boldsymbol{\Gamma}_2$ and $\boldsymbol{\Gamma}_3$
4: Find a matrix $\boldsymbol{P} = \boldsymbol{P}^{\top} \succ \boldsymbol{O}$, a solution to the Lyapunov matrix equation

$$\boldsymbol{\Gamma}^{\top}\boldsymbol{P} + \boldsymbol{P}\boldsymbol{\Gamma} = -2\boldsymbol{Q}$$

5: Analyze the stability of the networked system:

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{\Psi}_0\boldsymbol{x}(t) + \boldsymbol{\Psi}_1\dot{\boldsymbol{u}}_2(t) + \boldsymbol{\Psi}_2\boldsymbol{u}_2(t)$$

by varying the time-delay (τ)
6: Establish an experimental bound on τ that guarantees the stability of the $\mathrm{N_{et}}$UIO
7: Compare the theoretical bound on τ given by the quadratic polynomial in Theorem 1 and the experimental one computed in Step 6
8: **output:** *qualitative analysis of the theoretical bound*
