Preprints of the 19th World Congress
The International Federation of Automatic Control
Cape Town, South Africa. August 24-29, 2014

Controllability of Discrete-time Networked Control Systems with Try Once Discard Protocol

Merid Lješnjanin * **Daniel E. Quevedo** ** **Dragan Nešić** *

* *Department of Electrical and Electronic Engineering, The University of Melbourne, Parkville, VIC 3052(e-mail: meridl@student.unimelb.edu.au, dnesic@unimelb.edu.au).*
** *School of Electrical Engineering & Computer Science, The University of Newcastle, NSW 2308, Australia (e-mail: dquevedo@ieee.org)*

Abstract: This paper investigates controllability of discrete-time Networked Control Systems. The distinguishing feature is that the network imposes scheduling. The network is characterized by a dynamic protocol and different types of *additional processing capabilities*, as determined by available technology. For NCS with general nonlinear plants we present general controllability results. Finally, for NCS with linear plants we extend ideas motivated by NCS architectures with static protocols to state corresponding controllability results.

1. INTRODUCTION

A Networked Control System (NCS) is a control system that uses a network in *at least one* of its links as a communication medium. This class of control systems is important due to its positive impacts on system's cost reduction, flexibility, reliability, interoperability and maintenance, e.g., see Moyne and Tilbury (2007). There are many challenges that lie on the path of obtaining these positive impacts, some of them being delays, packet dropouts, sampling, quantization and scheduling. A considerable portion of control research output in recent years has been generated by addressing these challenges, e.g., see Hespanha et al. (2007).

In this manuscript we consider a NCS whose network resources are shared and we investigate the corresponding *scheduling* effects on the controllability of the plant. More precisely, we investigate whether a controllable plant preserves its controllability once the network, which imposes (*only*) scheduling, is introduced. One way to address network imposed issues is to design appropriate NCS architecture. For this, one often uses additional devices, such as smart actuators or buffers, e.g., see Lješnjanin et al. (2014); Polushin et al. (2008); Findeisen et al. (2011); Greco et al. (2012). In this manuscript, we consider devices located between a network and a controller in feedback link and a network and a plant in feed forward link; see Fig. 1. We model the network with a (dynamic) protocol, e.g., see Nešić and Teel (2004b,a), and we assume that the network possesses different types of *additional processing capabilities* which are determined by the choice of devices used (see Definitions 3, 4, 5, 7).

First, we provide controllability results for the case where a NCS encapsulates a general nonlinear plant. Then, controllability results for a case with a linear plant are presented. For the latter case, we first extend the controllability result from Suzuki et al. (2011) by showing the existence of a *admissible communication sequence* (see Definition 6) so that the corresponding controllability result holds. Then we use this extension and the resulting NCS architecture to state the controllability result. Recently, we have become aware of linear-case results related to ours, namely Yu and Andersson (2013), Smarra et al. (2012) and D'Innocenzo et al. (2013). The first reference considers a NCS with a SISO linear plant and investigates the effects of the so called *blind* periods in communication on controllability (authors do not consider scheduling issues but the corresponding result can be viewed as a special case of our result). The other two references, respectively, consider Multi-Hop Control Networks with MIMO and SISO linear plants and present conditions for controllability.

Notation and preliminaries Throughout this manuscript, $\mathbb{C}$ stands for the set of complex numbers, $\mathbb{R}$ stands for the set of real numbers, $\mathbb{Z}$ denotes the set of integers and $|\cdot|$ refers to Euclidean norm. Further, for any $p \in \mathbb{F}$, we use the notation $\mathbb{F}_{\Diamond p}$ to refer to a set $\{v \in \mathbb{F} : v \Diamond p\}$ where $\mathbb{F} \in \{\mathbb{R}, \mathbb{Z}\}$ and $\Diamond \in \{\geq, >\}$. Often, we use a tuple notation to represent a column vector. A function $\alpha \colon \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is said to be a class $\mathcal{K}$ function ($\alpha \in \mathcal{K}$) if it is continuous, zero at zero and strictly increasing. A function $\beta \colon \mathbb{R}_{\geq 0} \times \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is said to be a class $\mathcal{KL}$ function ($\beta \in \mathcal{KL}$) if $\beta(\cdot, t) \in \mathcal{K}$ for each fixed t and $\beta(v, \cdot)$ is decreasing to zero for each $v > 0$. Finally, we define the nth root of unity, where $n \in \mathbb{Z}_{\geq 1}$, as a $z \in \mathbb{C}$ such that $z^n = 1$; which is *primitive* if it is not kth root of unity for any $k \in \{1, \dots, n-1\}$.

2. SYSTEM MODEL

The considered NCS architecture is depicted in Fig. 1. Transmission of the data through the network is assumed *instantaneous* as well as the processing in devices μ_{p} and μ_{c}. The corresponding effects of *instantaneous* transmission and processing can be described as jumps in the dynamical model of the NCS. This motivates us to adopt a sampled-data approach as documented in Nešić and Teel (2004b,a) and use two indices to capture time evolution in system variables. More precisely, the first index refers to discrete time while the second index is a counter which refers to transmission *plus* processing instants. The indices do not evolve independently. Both are incremented alternatively and can only be incremented by 1. For example, consider a sequence of index pairs $\{(0,0), (0,1), (1,1), (1,2), \dots\}$; whenever the second index is

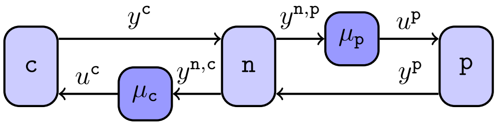

Fig. 1. NCS; $\mathtt{c}$ - controller, $\mathtt{n}$ - network, $\mathtt{p}$ - plant, $\mu_{\cdot}$ - processing device, symbol y denotes the corresponding outputs while u denotes inputs.

larger than the first, it indicates that transmission and processing has just occurred. Using this, the plant model can be represented as

$$\begin{aligned} x^{\mathtt{p}}(i+1,i+2) &= f^{\mathtt{p}}\big(x^{\mathtt{p}}(i,i+1), u^{\mathtt{p}}(i,i+1)\big),\\ y^{\mathtt{p}}(i,i+1) &= h^{\mathtt{p}}\big(x^{\mathtt{p}}(i,i+1)\big) \end{aligned} \tag{1}$$

where $x^{\mathtt{p}}(i,i+1) \in \mathbb{R}^{d_{x^{\mathtt{p}}}}$ is the plant state, $u^{\mathtt{p}}(i,i+1) \in \mathbb{R}^{d_{u^{\mathtt{p}}}}$ is the plant input and $y^{\mathtt{p}}(i,i+1) \in \mathbb{R}^{d_{y^{\mathtt{p}}}}$ is the plant output at discrete time instant $i \in \mathbb{Z}_{\geq 0}$ after transmission and processing have occurred. The mappings $f^{\mathtt{p}} : \mathbb{R}^{d_{x^{\mathtt{p}}}} \times \mathbb{R}^{d_{u^{\mathtt{p}}}} \to \mathbb{R}^{d_{x^{\mathtt{p}}}}$ and $h^{\mathtt{p}} : \mathbb{R}^{d_{x^{\mathtt{p}}}} \to \mathbb{R}^{d_{y^{\mathtt{p}}}}$ are assumed nonlinear where $d_{x^{\mathtt{p}}}, d_{u^{\mathtt{p}}}$ and $d_{y^{\mathtt{p}}}$ are positive integers.

We assume that when $(i,i) \to (i,i+1)$ the plant state does not change, i.e., $x^{\mathtt{p}}(i,i+1) = x^{\mathtt{p}}(i,i)$, $\forall i \in \mathbb{Z}_{\geq 1}$.

2.1 Network Protocols

The considered network is assumed to be error-free, i.e., packet dropouts do not occur. However, the communication link is shared among actuator nodes, which imposes scheduling. In order to address scheduling issues we characterize a network with a protocol which governs the medium access of each node; e.g., see Nešić and Teel (2004b,a); Lian et al. (2005). We focus on dynamic protocols which compare the data addressed to a node with network internal data, e.g., the corresponding buffer contents before transmission. Focusing on feed-forward link (see Fig. 1), data to be transmitted are controller outputs and the quantity used by the protocol is

$$e(i,i) \triangleq b^{\mathtt{n}}(i,i) - y^{\mathtt{c}}(i,i) \tag{2}$$

where $y^{\mathtt{c}}(i,i)$ is the controller output and $b^{\mathtt{n}}(i,i) = g^{\mathtt{n}}\big(y^{\mathtt{c}}(i,i), y^{\mathtt{p}}(i,i)\big)$ where $g^{\mathtt{n}} : \mathbb{R}^{d_{u^{\mathtt{p}}}} \times \mathbb{R}^{d_{y^{\mathtt{p}}}} \to \mathbb{R}^{d_{u^{\mathtt{p}}}}$. Introduction of $g^{\mathtt{n}}$ captures the fact that some networks can have *additional processing capabilities*. For instance, via appropriate $g^{\mathtt{n}}$ one can manipulate which node will be picked; e.g., see Section 3.

Further, a protocol can be described by diagonal matrices $\Psi(\cdot,\cdot)$ which contain zeros and ones on its diagonal; see equations (14), (16) and (17) in Nešić and Teel (2004b) for different types of protocols. In this manuscript we will focus on the so-called Try Once Discard protocol (TOD).

Definition 1. (TOD protocol, Nešić and Teel (2004a)). Suppose that there are $r \in \mathbb{Z}_{\geq 2}$ nodes competing for access to the network. Correspondingly, the error vector is partitioned as $e = (e_1, \ldots, e_r)$. The node $j \in \{1, \ldots, r\}$ with the greatest weighted error at instant $(i,i), i \in \mathbb{Z}_{\geq 0}$ will be granted access. (It is assumed that the weights are already incorporated into the model.) If a data packet fails to win access to the network, it is discarded and new data is used at the next transmission time. If two or more nodes have equal priority, a pre-specified ordering of the nodes is used to resolve the collision. More precisely, the diagonal matrix $\Psi(\cdot,\cdot)$ is given as

$$\Psi\big((i,i), e(i,i)\big) = \mathtt{diag}\Big(\psi_1\big(e(i,i)\big) I_{n_1}, \ldots, \psi_r\big(e(i,i)\big) I_{n_r}\Big) \tag{3}$$

where $i \in \mathbb{Z}_{\geq 0}$ and I_{n_j} are identity matrices of dimension $n_j \in \mathbb{Z}_{\geq 1}$ for every $j \in \{1, \ldots, r\}$ with $\sum_{j=1}^{r} n_j = d_{u^{\mathtt{p}}}$ and where

$$\psi_j\big(e(i,i)\big) = \begin{cases} 1, & \text{if } j = \min\Big(\arg\max_{j \in \{1,\ldots,l\}} |e_j(i,i)|\Big),\\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

for all $j \in \{1, \ldots, l\}$. □

2.2 Processing devices

As mentioned above, in order to design appropriate NCS architecture one can resort to introduction of extra devices. In this document we focus on two kinds.

One type of a device will just apply the received value for the addressed node and zeros to the remaining nodes:

$$y^{\mu_*^{\mathtt{s}}}(i,i+1) = h^{\mu_*^{\mathtt{s}}}\big(y^{\mathtt{n},*}(i,i+1)\big) \tag{5}$$

for all $i \in \mathbb{Z}_{\geq 0}$, where $* \in \{\mathtt{c}, \mathtt{p}\}$ and $\mathtt{s}$ alludes to static. The other type of device will be dynamic. Its output will depend on the networks output and buffer contents:

$$y^{\mu_*^{\mathtt{d}}}(i,i+1) = h^{\mu_*^{\mathtt{d}}}\big(y^{\mathtt{n},*}(i,i+1), b^{\mu_*^{\mathtt{d}}}(i-1,i)\big) \tag{6}$$

for all $i \in \mathbb{Z}_{\geq 1}$, where $\mathtt{d}$ alludes to dynamic.

Next, we proceed with the necessary preliminaries needed for stating the NCS architectures determined by the devices used. First, we assume that buffers in $\mu_*^{\mathtt{d}}$ hold their values until new data arrives, yielding

$$b^{\mu_*^{\mathtt{d}}}(i,i+1) = b^{\mu_*^{\mathtt{d}}}(i+1,i+1) \tag{7}$$

for all $i \in \mathbb{Z}_{\geq 0}$. Similarly, we assume that the value of controller's output $y^{\mathtt{c}}$ does not change during transmission and processing, thus, $y^{\mathtt{c}}(i,i) = y^{\mathtt{c}}(i,i+1)$, $\forall i \in \mathbb{Z}_{\geq 0}$.

Recall that our interest lies in the controllability of the plant, hence, the focus is on the feed-forward link.

Remark 1. Note that (7) could be replaced by more complex processing; e.g., see Quevedo and Nešić (2011); Pin and Parisini (2011); Findeisen and Varutti (2009); Munoz de la Pena and Christofides (2008); Montestruque and Antsaklis (2004). □

Next, the network output is defined as

$$y^{\mathtt{n},\mathtt{p}}(i,i+1) = \Psi\big(i, e(i,i)\big) y^{\mathtt{c}}(i,i) \tag{8}$$

for all $i \in \mathbb{Z}_{\geq 0}$ while the output of processing unit $\mu_{\mathtt{p}}^{\mathtt{d}}$, is given as

$$y^{\mu_{\mathtt{p}}^{\mathtt{d}}}(i,i+1) = \Psi\big(i, e(i,i)\big) y^{\mathtt{c}}(i,i) + \Big(I - \Psi\big(i, e(i,i)\big)\Big) b^{\mu_{\mathtt{p}}^{\mathtt{d}}}(i,i) \tag{9}$$

for all $i \in \mathbb{Z}_{\geq 0}$. Note also that according to (5) we have

$$y^{\mu_{\mathtt{p}}^{\mathtt{s}}}(i,i+1) = h^{\mu_{\mathtt{p}}^{\mathtt{s}}}\big(y^{\mathtt{n},\mathtt{p}}(i,i+1)\big). \tag{10}$$

As depicted in Fig. 1, we have that $u^{\mathtt{p}}(\cdot,\cdot) = y^{\mu_{\mathtt{p}}^*}(\cdot,\cdot)$. Moreover, for the case when $\mu_{\mathtt{p}}^{\mathtt{d}}$ is used $b^{\mu_{\mathtt{p}}^{\mathtt{d}}}(\cdot,\cdot) = y^{\mu_{\mathtt{p}}^{\mathtt{d}}}(\cdot,\cdot) = u^{\mathtt{p}}(\cdot,\cdot)$. Now, given the above, equation (9) becomes

$$\begin{aligned} &u^{\mathtt{p}}(i,i+1) = \\ &\quad \Psi\big(i, e(i,i)\big) y^{\mathtt{c}}(i,i+1) + \Big(I - \Psi\big(i, e(i,i)\big)\Big) b^{\mu_{\mathtt{p}}^{\mathtt{d}}}(i-1,i) \end{aligned} \tag{11}$$

whereas, assuming $b^{\mathtt{n}}(\cdot) = b^{\mu_{\mathtt{p}}^{\mathtt{d}}}(\cdot)$, $e(i,i)$ in (2) satisfies

$$e(i,i) = b^{\mathtt{n}}(i-1,i) - y^{\mathtt{c}}(i,i+1). \tag{12}$$

2.3 NCS architecture with device $\mu_{\mathtt{p}}^{\mathtt{d}}$

Equations (1)–(2), (6)–(9), (11)–(12), give

$$\begin{aligned} x^{\mathrm{p}}(i+1,i+2) &= f^{\mathrm{p}}\big(x^{\mathrm{p}}(i,i+1), \big(I-\Psi\big(i,e(i,i)\big)\big)e(i,i)+y^{\mathrm{c}}(i,i)\big), \\ e(i,i+1) &= \big(I-\Psi\big(i,e(i,i)\big)\big)e(i,i), \\ e(i,i) &= e(i-1,i)+y^{\mathrm{c}}(i-1,i)-y^{\mathrm{c}}(i,i+1). \end{aligned} \tag{13}$$

If we now write k for (i,i), then the above yields

$$\begin{aligned} x^{\mathrm{p}}(k+1) &= f^{\mathrm{p}}\big(x^{\mathrm{p}}(k), \big(I-\Psi\big(k,e(k)\big)\big)e(k)+y^{\mathrm{c}}(k)\big), \\ e(k+1) &= \big(I-\Psi\big(k,e(k)\big)\big)e(k)+y^{\mathrm{c}}(k)-y^{\mathrm{c}}(k+1). \end{aligned} \tag{14}$$

2.4 NCS architecture with device $\mu_{\mathrm{p}}^{\mathrm{s}}$

Simple manipulations of equations (1)–(2), (10) and (12) with assumptions that $b^{\mathrm{n}}(\cdot,\cdot) = y^{\mu_{\mathrm{p}}^{\mathrm{s}}}(\cdot,\cdot)$ and $b^{\mathrm{n}}(i,i+1) = b^{\mathrm{n}}(i+1,i+1)$ yields

$$\begin{aligned} x^{\mathrm{p}}(i+1,i+2) &= f^{\mathrm{p}}\big(x^{\mathrm{p}}(i,i+1), y^{\mu_{\mathrm{p}}^{\mathrm{s}}}(i,i+1)\big), \\ e(i,i+1) &= \big(I-\Psi\big(i,e(i,i)\big)\big)e(i,i), \\ e(i+1,i+1) &= y^{\mu_{\mathrm{p}}^{\mathrm{s}}}(i-1,i)-y^{\mathrm{c}}(i+1,i+2). \end{aligned} \tag{15}$$

Thus, if we write k for (i,i), then

$$\begin{aligned} x^{\mathrm{p}}(k+1) &= f^{\mathrm{p}}\big(x^{\mathrm{p}}(k), y^{\mu_{\mathrm{p}}^{\mathrm{s}}}(k)\big), \\ e(k+1) &= y^{\mu_{\mathrm{p}}^{\mathrm{s}}}(k)-y^{\mathrm{c}}(k+1). \end{aligned} \tag{16}$$

3. CONTROLLABILITY: NONLINEAR PLANTS

Our aim is to investigate the effects of the network on the controllability of the plant model. We begin by adopting the following notion of controllability.

Definition 2. (Controllability). The system

$$x(k+1) = f\big(x(k),u(k)\big),\ k\in\mathbb{Z}_{\geq 0} \tag{17}$$

where $x\in\mathbb{R}^{d_x}$, $u\in\mathbb{R}^{d_u}$, $d_x\in\mathbb{Z}_{\geq 1}$, $d_u\in\mathbb{Z}_{\geq 1}$ is said to be *asymptotically controllable to the origin*, if there exists $\beta\in\mathcal{KL}$, such that for any initial condition x, there exists a nonempty *set* of semi-infinite length control sequences $\mathbf{U}(x)$ such that for all $\mathbf{u}_\infty = \{u(0),u(1),\dots\}\in\mathbf{U}(x)$ the following inequality holds

$$\big|\phi(k,x,\mathbf{u}_\infty)\big| \leq \beta\big(|x|,k\big),\ \forall k\in\mathbb{Z}_{\geq 0}. \tag{18}$$

In (18), $\phi(k,x,\mathbf{u}_\infty)$ refers to solutions of (17) k steps into the future, starting at initial condition x under the influence of inputs from $\mathbf{u}_\infty$. If, furthermore, $\beta\big(|x|,k\big)$ in (18) can be chosen as $\beta\big(|x|,k\big) = Me^{-k\lambda}|x|$ for some $(M,\lambda)\in\mathbb{R}_{>0}\times\mathbb{R}_{>0}$, then (17) is said to be *exponentially controllable to the origin.* □

Focusing on the plant model, the above definition makes explicit the fact that for a given x^{p}, there may exist more than one control sequence which drives the plant state to the origin satisfying the desired bound. This allows us to study the controllability property of the same plant when the corresponding inputs and outputs are accessed through a network. The study is done by examining whether for every $x^{\mathrm{p}}\in\mathbb{R}^{d_{x^{\mathrm{p}}}}$, the network allows for *realization* of at least one sequence in $\mathbf{U}(x^{\mathrm{p}})$.

For the purpose of the forthcoming analysis we assume that the control vector partition corresponds to network nodes, namely

$$u(k) = \big(u_1(k),u_2(k),\dots,u_r(k)\big) \tag{19}$$

for all $k\in\mathbb{Z}_{\geq 0}$, $r\in\mathbb{Z}_{\geq 1}$, where for all $j\in\{1,2,\dots,r\}$, $u_j(k)\in\mathbb{R}^{m_j}$, $m_j\in\mathbb{Z}_{\geq 1}$ and $\sum_{j=1}^{r} m_j = d_{u^{\mathrm{p}}}$.

We proceed by employing Definition 2 for the investigation of controllability preservation of general nonlinear plants.

With $\mathbf{U}_{\mathrm{S}}(x^{\mathrm{p}})\subset\mathbf{U}(x^{\mathrm{p}})$ we will denote a set that consists of semi-infinite length control sequences from $\mathbf{U}(x^{\mathrm{p}})$ which can be realized by exploiting network processing capabilities, network protocol and an appropriate processing unit μ_{p}^{*}; the notation $\mathbf{U}_{\mathrm{S}}(x^{\mathrm{p}})$ alludes to a subset due to the scheduling. Unlike static protocols, dynamic protocols need not have a predefined schedule. In fact, values to be sent through a network have to satisfy a certain criterion defined by the protocol which is usually fixed, see, e.g., (4).

Now, recall that above we mention *additional processing capabilities* (*apc*); see paragraph after (2).

Definition 3. (*apc*). Suppose that there are $r\in\mathbb{Z}_{\geq 2}$ nodes competing for access to the network. Correspondingly, the network buffer vector is partitioned as $b^{\mathrm{n}} = (b_1^{\mathrm{n}},\dots,b_r^{\mathrm{n}})$. Then, there exist $\delta>0$ such that provided $j\in\{1,\dots,r\}$, $b_i^{\mathrm{n}}(k) := u_i(k)+\delta, \forall i\in\{1,\dots,r\},\ i\neq j$. □

Note that in Definition 3 we do not explain how j is provided; this is done in the sequel. The reason we introduce this *apc* is that, effectively, it enables us to manipulate the protocol criterion so that a specific element is picked. We capture this in the following lemma.

Lemma 2. (Tricking TOD). Consider any $u(k)\in\mathbf{u}_\infty\in\mathbf{U}(x^{\mathrm{p}})$, $k\in\mathbb{Z}_{\geq 0}$ for some $x^{\mathrm{p}}\in\mathbb{R}^{d_{x^{\mathrm{p}}}}$. Let the network be governed by the TOD protocol and let it have *apc*. Then, provided $j\in\{1,\dots,r\}$, $u_j(k)$ is chosen for transmission. □

Proof. Let $j\in\{1,\dots,r\}$ be given. Then, according to *apc*, for all $i\in\{1,\dots,r\}, i\neq j$, $b_i^{\mathrm{n}}(k) := u_i(k)+\delta$. It follows that the corresponding error vector (see (2)) $e(k) = (\delta,\dots,\delta,e_j(k),\delta,\dots,\delta)$ with $e_j(k)=0$. Hence, according to TOD protocol $u_j(k)$ is chosen for transmission. □

As promised above, next, we discuss how $j\in\{1,\dots,r\}$ is provided. First, we concentrate on a special subset of $\mathbf{U}_{\mathrm{S}}(x^{\mathrm{p}})$ which is additionally accompanied with appropriate processing unit μ_{p}^{*}. More precisely, let us consider set $\mathbf{U}_{\mathrm{S}}^{0}(x^{\mathrm{p}})\subset\mathbf{U}_{\mathrm{S}}(x^{\mathrm{p}})$ which consists of control sequences where at each time instant a member of the corresponding sequence has at most one nonzero element. Namely, let $\mathbf{u}_\infty^{0} = \{u^{0}(0),u^{0}(1),\dots,u^{0}(k),\dots\}\in\mathbf{U}_{\mathrm{S}}^{0}(x)$, $k\in\mathbb{Z}_{\geq 0}$. Then, for each $k\in\mathbb{Z}_{\geq 0}$ it follows $u^{0}(k) = \big(0,\dots,0,u_i^{0}(k),0,\dots,0\big)$, $u_i^{0}(k)\in\mathbb{R}^{m_i}$, where $i\in\{1,\dots,r\}$ for some $r\in\mathbb{Z}_{\geq 1}$ (see equation (19)). Furthermore, let the processing unit be $\mu_{\mathrm{p}}^{\mathrm{s}}$ (see (5)).

Now, we modify Definition 3 with respect to set $\mathbf{U}_{\mathrm{S}}^{0}(x^{\mathrm{p}})$. Namely, we extend the corresponding definition to additionally extract the index of nonzero element or to provide any index if all elements are equal to zero.

Definition 4. (apc^{0}). Consider $\mathbf{U}_{\mathrm{S}}^{0}(x^{\mathrm{p}})\subset\mathbf{U}_{\mathrm{S}}(x^{\mathrm{p}})$ and suppose that there are $r\in\mathbb{Z}_{\geq 2}$ nodes competing for network access. Correspondingly, the network buffer vector is partitioned as $b^{\mathrm{n}} = (b_1^{\mathrm{n}},\dots,b_r^{\mathrm{n}})$. Then, there exist $\delta>0$ and $j\in\{1,\dots,r\}$ is the index of the nonzero element of the corresponding control vector at time k or it is any index otherwise, and $b_i^{\mathrm{n}}(k) := u_i^{0}(k)+\delta, \forall i\in\{1,\dots,r\},\ i\neq j$. □

Finally, equipped with Definition 4 and processing unit $\mu_{\mathrm{p}}^{\mathrm{s}}$ we can state the following lemma related to realization of sequences from set $\mathbf{U}_{\mathrm{S}}^{0}(x^{\mathrm{p}})$, i.e., $u^{\mathrm{p}}(k) = u^{0}(k), \forall k\in\mathbb{Z}_{\geq 0}$.

Lemma 3. (Realizing sequences from $\mathbf{U}_{\mathrm{S}}^{0}(x^{\mathrm{p}})$). Consider set $\mathbf{U}_{\mathrm{S}}^{0}(x^{\mathrm{p}})\subset\mathbf{U}_{\mathrm{S}}(x^{\mathrm{p}})$ and the processing unit $\mu_{\mathrm{p}}^{\mathrm{s}}$. Let the network be governed by TOD protocol and let it have *apc*^{0}. Then, the sequences from $\mathbf{U}^0_{\mathtt{S}}(x^{\mathtt{P}})$ are realizable . □

Proof. Consider any sequence from $\mathbf{U}^0_{\mathtt{S}}(x) \subset \mathbf{U}_{\mathtt{S}}(x^{\mathtt{P}})$ and an element of the corresponding sequence at time instant $k \in \mathbb{Z}_{\geq 0}$. According to *apc*^{0}, an index $j \in \{1, \dots, r\}$ of the nonzero element from the corresponding control vector $u^0(k)$ or any index otherwise is picked and $b^{\mathtt{n}}_i(k) := u^0_i(k) + \delta, \delta > 0, \forall i \in \{1, \dots, r\},\ i \neq j$. Correspondingly, the error vector (see (2)) $e(k) = (\delta, \dots, \delta, e_j(k), \delta, \dots, \delta)$ with $e_j(k) = 0$; for simplicity let j be also the index if all elements of the $u^0(k)$ are equal to zero. Hence, according to TOD $u^0_j(k)$ is chosen. Effectively, processing unit $\mu^{\mathtt{s}}_{\mathtt{p}}$ receives $u^0(k)$ and according to Fig. 1 and (5) $u^{\mathtt{P}}(k) = y^{\mu^{\mathtt{s}}_{\mathtt{P}}}(k) = u^0(k)$, as desired. □

Next, we focus on the $\mathbf{U}^\delta_{\mathtt{S}}(x^{\mathtt{P}}) \subset \mathbf{U}_{\mathtt{S}}(x^{\mathtt{P}})$ which consists of control sequences where at each two consecutive time instances the corresponding members of the corresponding sequence differ in at most one element. Namely, let $\mathbf{u}^\delta_\infty = \{u^\delta(0), u^\delta(1), \dots, u^\delta(k), \dots\} \in \mathbf{U}^\delta_{\mathtt{S}}(x^{\mathtt{P}}),\ k \in \mathbb{Z}_{\geq 0}$, then for each $k \in \mathbb{Z}_{\geq 0}$ it follows $\big|u^\delta(k) - u^\delta(k+1)\big| = \big(0, \dots, 0, |u^\delta_i(k) - u^\delta_i(k+1)|, 0, \dots, 0\big) = \big(0, \dots, 0, \delta_i(k), 0, \dots, 0\big),\ i \in \{1, \dots, r\},\ r \in \mathbb{Z}_{\geq 1}, \delta_i(k) \geq 0$. Furthermore, let the processing unit be $\mu^{\mathtt{d}}_{\mathtt{p}}$ (see (6)).

We modify Definition 3 but now with respect to $\mathbf{U}^\delta_{\mathtt{S}}(x^{\mathtt{P}})$. Similarly as above, we extend the corresponding definition to additionally extract the index of differing element or to provide any index if all elements are the same.

Definition 5. (*apc*$^\delta$). Let the processing unit $\mu^{\mathtt{d}}_{\mathtt{p}}$ be used. Consider set $\mathbf{U}^\delta_{\mathtt{S}}(x^{\mathtt{P}})$ and suppose that there are $r \in \mathbb{Z}_{\geq 2}$ nodes competing for access to the network. The network buffer vector is partitioned as $b^{\mathtt{n}} = (b^{\mathtt{n}}_1, \dots, b^{\mathtt{n}}_r)$. Then, there exist $\delta > 0$ and by comparing $u^\delta(k)$ and $b^{\mathtt{n}}(k-1) = b^{\mu^{\mathtt{d}}_{\mathtt{P}}}(k-1), k \in \mathbb{Z}_{\geq 1}$, index $j \in \{1, \dots, r\}$ is the index of the differing element or it is any index if $u^\delta(k) = b^{\mathtt{n}}(k)$. Finally, $b^{\mathtt{n}}_i(k) := u^\delta_i(k) + \delta, \forall i \in \{1, \dots, r\},\ i \neq j$. □

Due to the fact that it is possible that $y^{\mathtt{c}}(0)$ and $b^{\mu^{\mathtt{d}}_{\mathtt{P}}}(0)$ differ in more than one element, we take a short detour to discuss how realizations of sequences from the set $\mathbf{U}^\delta_{\mathtt{S}}(x^{\mathtt{P}})$ impose restrictions: If $b^{\mu^{\mathtt{d}}_{\mathtt{P}}}(0)$ and $u^\delta(0)$ differ in more than one element, then determination of an index for element to be sent would have to be specified by some rule. However, by the time $u^{\mathtt{P}}(k) = u^\delta(k)$, the plant state might diverge from the trajectory that leads to the origin. Hence, without imposing constraint that at time when $u^{\mathtt{P}}(k) = u^\delta(k)$ we have $\phi(k, x^{\mathtt{P}}, \mathbf{u}^\delta_\infty) = \phi(k, x^{\mathtt{P}}, \tilde{\mathbf{u}}^\delta_\infty)$,[1] where $k \geq m$, we cannot guarantee controllability. Now we are ready to state the following lemma which is related to realization of sequences from $\mathbf{U}^\delta_{\mathtt{S}}(x^{\mathtt{P}})$, i.e., $u^{\mathtt{P}}(k) = u^\delta(k), \forall k \in \mathbb{Z}_{\geq 0}$.

Lemma 4. (Realizing sequences from $\mathbf{U}^\delta_{\mathtt{S}}(x^{\mathtt{P}})$). Consider set $\mathbf{U}^\delta_{\mathtt{S}}(x^{\mathtt{P}}) \subset \mathbf{U}_{\mathtt{S}}(x^{\mathtt{P}})$ and the processing unit $\mu^{\mathtt{d}}_{\mathtt{p}}$. Let the corresponding network be governed by TOD protocol and let it have *apc*$^\delta$. If $b^{\mu^\delta_{\mathtt{P}}}(0)$ and $u^\delta(0) \in \mathbf{u}^\delta_\infty \in \mathbf{U}^\delta_{\mathtt{S}}(x^{\mathtt{P}})$ differ in at most one element, then $\mathbf{u}^\delta_\infty$ is realizable. □

Proof. Consider $\mathbf{u}^\delta_\infty \in \mathbf{U}^\delta_{\mathtt{S}}(x^{\mathtt{P}})$ with $b^{\mu^{\mathtt{d}}_{\mathtt{P}}}(0) = u^\delta(0)$. Next, consider $u^\delta(k), k \in \mathbb{Z}_{\geq 1}$. According to *apc*$^\delta$ an index $j \in \{1, \dots, r\}$ of differing element between $u^\delta(k)$ and $u^\delta(k-1)$ or any index otherwise is picked and $b^{\mathtt{n}}_i(k) := u^\delta_i(k) + \delta, \delta > 0, \forall i \in \{1, \dots, r\},\ i \neq j$. Correspondingly, the error vector (see (2)) $e(k) = (\delta, \dots, \delta, e_j(k), \delta, \dots, \delta)$ with $e_j(k) = 0$; for simplicity let j be also the index if $u^\delta(k) = u^\delta(k-1)$. Hence, according to TOD protocol $u^\delta_j(k)$ is chosen for transmission. Finally, according to Fig. 1 and (11) $u^{\mathtt{P}}(k) = y^{\mu^{\mathtt{d}}_{\mathtt{P}}}(k) = u^\delta(k)$ as desired. □

[1] $\tilde{\mathbf{u}}^\delta_\infty = \big\{b^{\mu^{\mathtt{d}}_{\mathtt{P}}}(0), b^{\mu^{\mathtt{d}}_{\mathtt{P}}}(1), \dots, b^{\mu^{\mathtt{d}}_{\mathtt{P}}}(k-1), u^\delta(k), u^\delta(k+1), \dots\big\}$

The results above are rather general. Moreover, due to its requirement, Lemma 4 is more restrictive than Lemma 3. In the sequel we will focus on linear plant models.

4. CONTROLLABILITY: LINEAR PLANTS

In this section we consider a special case of (1), namely

$$\begin{aligned} x^{\mathtt{P}}(k+1) &= Ax^{\mathtt{P}}(k) + Bu^{\mathtt{P}}(k), \\ y^{\mathtt{P}}(k) &= Cx^{\mathtt{P}}(k) \end{aligned} \tag{20}$$

where $A \in \mathbb{R}^{d_{x^{\mathtt{P}}} \times d_{x^{\mathtt{P}}}}$, $B \in \mathbb{R}^{d_{u^{\mathtt{P}}} \times d_{x^{\mathtt{P}}}}$ and $C \in \mathbb{R}^{d_{y^{\mathtt{P}}} \times d_{x^{\mathtt{P}}}}$.

4.1 NCS with a linear plant and $\mu^{\mathtt{d}}_{\mathtt{p}}$.

Sufficient conditions for controllability of a NCS with a linear plant (20), a processing unit $\mu^{\mathtt{d}}_{\mathtt{p}}$ and a network which imposes *periodic* scheduling are documented in Suzuki et al. (2011). We use this result in the sequel. We start by providing some additional notation. A periodic transmission sequences is denoted by $\sigma_w = \big\{\sigma(0), \sigma(1), \dots, \sigma(w-1)\big\}$ where $w \in \mathbb{Z}_{\geq 1}$ denotes the period. Further, for each $i \in \{0, \dots, w-1\}$ we introduce a vector $\sigma(i) \in \{0, 1\}^{d_{u^{\mathtt{P}}}}$ with 0 denoting corresponding node not to be updated and 1 denoting corresponding node to be updated. For simplicity we refer to periodic transmission sequences as communication sequences. Next, we define admissible communication sequences, which means that during period w every node will be updated at least once.

Definition 6. (Suzuki et al. (2011)). Let the maximum number of nodes which can be addressed be a $b < d_{u^{\mathtt{P}}}$. If for any period $w \in \mathbb{Z}_{\geq 1}$ the following is satisfied

(1) for each $i \in \{0, \dots, w-1\}$, $\big|\sigma_w(i)\big| \leq b$;
(2) $\mathtt{span}\big(\sigma(0), \dots, \sigma(w-1)\big) = \mathbb{R}^{d_{u^{\mathtt{P}}}}$,

then the communication sequence σ_w is *admissible*. □

Notice that an admissible sequence remains admissible if an element that already exists in the sequence or the element consisting only of zero values is added to it.

We introduce the communication sequence matrix

$$\mathcal{E}(k, i) = \prod_{j=i}^{k} \big(I - \mathtt{diag}\big(\sigma(j)\big)\big),\ k \geq i. \tag{21}$$

The communication sequence matrix polynomial is defined as

$$\mathcal{G}(\mu) = \sum_{l=0}^{w-1} \big(\big(\mu^{w-1}I + \mu^{w-2}\mathcal{E}(l+1, l+1) + \cdots \\ \cdots + \mathcal{E}(l+w-1, l+1)\big)\mathtt{diag}\big(\sigma(l)\big)\big) \tag{22}$$

with the indeterminate μ. The communication sequence characteristic polynomial is defined as

$$g(\mu) = \mathtt{det}\big(\mathcal{G}(\mu)\big). \tag{23}$$

Theorem 5. (Suzuki et al. (2011)-Theorem 1). Consider a NCS with (20) and a processing unit $\mu^{\mathtt{d}}_{\mathtt{p}}$. Let the corresponding network impose periodic scheduling with period $w \in \mathbb{Z}_{\geq 1}$. If

(1) a communication sequence σ_w is admissible;
(2) the nonzero eigenvalues of matrix A do not coincide with the zeros of the communication sequence characteristic polynomial $g(\mu)$;
(3) the pair (A, B) is controllable,

then the NCS is controllable. □

Theorem 5 establishes that if we have a controllable plant, a processing unit $\mu_{\mathtt{p}}^{\mathtt{d}}$ and a suitable admissible *periodic* communication sequence, then the resulting NCS will be controllable as well. However, the theorem does not address the existence of such a sequence. Answering this question is one of the contributions of the present manuscript.

We start by noticing that if an admissible sequence is extended so that it remains admissible, then the order of the resulting polynomials in (23) increases accordingly. Next, we concentrate on the effects to the roots of the corresponding polynomials. However, before we provide some insight, notice that simple calculations yield

$$\mathcal{G}(\mu) = \mathtt{diag}\big(\mathcal{P}_1(w,\mu), \ldots, \mathcal{P}_{d_{u^{\mathtt{P}}}}(w,\mu)\big)$$

where

$$\mathcal{P}_i(w,\mu) = \mu^{w-1}\sum_{l=0}^{w-1}\sigma_i(l) + \mu^{w-2}\sum_{l=0}^{w-1}\sigma_i(l)\prod_{j=l+1}^{l+1}(1-\sigma_i(j)) + \cdots$$
$$+\,\mu\sum_{l=0}^{w-1}\sigma_i(l)\prod_{j=l+1}^{l+w-2}(1-\sigma_i(j)) + \sum_{l=0}^{w-1}\sigma_i(l)\prod_{j=l+1}^{l+w-1}(1-\sigma_i(j))$$

$\forall i \in \{1, \ldots, d_{u^{\mathtt{P}}}\}$. Note that the "minimum-length" communication sequence is the standard basis for $\mathbb{R}^{d_{u^{\mathtt{P}}}}$.

To gain some insight into the effects of enlarging the length of an admissible communication sequence in the way described above, we provide Table 1. One should notice that the "minimum-length" communication sequence (first row) and communication sequences formed by adding elements consisting only of zero values to the "minimum-length" communication sequence (second, fourth and seventh row) have roots on a unit circle. In fact, by adding $(0, \ldots, 0)$'s to the "minimum-length" communication sequence, for each $i \in \{1, \ldots, d_{u^{\mathtt{P}}}\}$, we generate polynomials $\mathcal{P}_i(w,\mu) = \sum_{j=0}^{w-1}\mu^j$. Such polynomials have *all* roots on a unit circle. However, if n is an odd number, then the corresponding polynomial will always have one root at -1. On the other hand, in the theory on cyclotomic polynomials, it is a well known fact that if $n = p - 1$ where p is an *odd* prime number (any prime number other than 2 which is the unique *even* prime), then the roots of the corresponding polynomial will correspond to the pth *primitive* roots of the *unity* (see Riesel (1994), page 306). More precisely $\sum_{j=0}^{p-1}\mu^j = \frac{\mu^p-1}{\mu-1}$ with all roots being distinct for each such p.

Now, before stating our first extension of Theorem 5 we modify Definition 3 so that it periodically provides indices provided in admissible communication sequence.

Definition 7. (apc^{σ_w}). Suppose that there are $r \in \mathbb{Z}_{\geq 2}$ nodes competing for access to the network. The network buffer vector is partitioned as $b^{\mathtt{n}} = (b_1^{\mathtt{n}}, \ldots, b_r^{\mathtt{n}})$. Let an admissible communication sequence σ_w be provided. Then, there exist $\delta > 0$ and at time instant lk where $l \in \{0, \ldots, w\}$ and $k \in \mathbb{Z}_{\geq 0}$, the index $j \in \{1, \ldots, r\}$ is an index of a nonzero element from σ_w, and $b_i^{\mathtt{n}}(k) := y_i^{\mathtt{c}}(k) + \delta, \forall i \in \{1, \ldots, r\},\ i \neq j$. □

Theorem 6. Consider a NCS with a linear plant (20) and a processing unit $\mu_{\mathtt{p}}^{\mathtt{d}}$. Let the network be governed by the TOD protocol and let it have apc^{σ_w}. If the pair (A, B) is controllable, then the NCS is controllable.

Proof. Consider $A \in \mathbb{R}^{d_{x^{\mathtt{P}}} \times d_{x^{\mathtt{P}}}}$ and recall that for a given $w \in \mathbb{Z}_{\geq 1}$, the polynomial $\sum_{j=0}^{w-1}\mu^j$, indeterminate μ, has *all* roots on the unit circle. The corresponding matrix either has no eigenvalues on the unit circle or finitely many. If A has no eigenvalues on the unit circle, then $w \geq d_{x^{\mathtt{P}}}$. Otherwise, there exists a finite number of *odd* prime numbers for which roots of the corresponding polynomial coincide with the eigenvalues of A. However, since there are infinitely many *odd* prime numbers, there exists an *odd* prime number w for which roots of the corresponding polynomial do not coincide with the eigenvalues of A. We proceed with adding $(0, \ldots, 0)$ elements to the "minimum-length" communication sequence so that the resulting length of the new admissible communication sequence equals to w. We denote this new sequence with σ_w. It follows that all conditions of Theorem 5 are satisfied. Hence, the NCS is controllable. Now, according to apc^{σ_w}, at time instant lk where $l \in \{0, \ldots, w\}$ and $k \in \mathbb{Z}_{\geq 0}$, the index $j \in \{1, \ldots, r\}$ is an index of a nonzero element from σ_w, and $b_i^{\mathtt{n}}(lk) := y_i^{\mathtt{c}}(lk) + \delta, \delta > 0, \forall i \in \{1, \ldots, r\},\ i \neq j$. Correspondingly, the error vector (see (2)) $e(lk) = (\delta, \ldots, \delta, e_j(lk), \delta, \ldots, \delta)$ with $e_j(lk) = 0$. Hence, according to TOD, $y_j^{\mathtt{c}}(lk)$ is chosen for transmission. Finally, according to Fig. 1 and (11), $u^{\mathtt{p}}(lk) = y^{\mu_{\mathtt{p}}^{\mathtt{d}}}(lk) = y^{\mathtt{c}}(lk)$, as desired. □

Remark 7. Results in Suzuki et al. (2011) implicitly require that condition from Lemma 4 is satisfied (see (4) in Suzuki et al. (2011)). This appears very restrictive unless special control sequences are considered for which appropriate devices exist; see Section 3. □

4.2 NCS with a linear plant and $\mu_{\mathtt{p}}^{\mathtt{s}}$.

We recall that, at each time instant, processing unit $\mu_{\mathtt{p}}^{\mathtt{s}}$ applies the received value to the addressed node and zero values to the remaining nodes. Thus, we are considering control sequences from the set $\mathbf{U}_{\mathtt{S}}^{0}(x^{\mathtt{P}}), x^{\mathtt{P}} \in \mathbb{R}^{d_{x^{\mathtt{P}}}}$; see Section 3. This means that the requirement as in Lemma 4 is not needed, making the forthcoming results less restrictive than Theorems 5 and 6.

Corollary 1. Consider a NCS with (20) and a processing unit $\mu_{\mathtt{p}}^{\mathtt{s}}$. Let the corresponding network impose periodic scheduling with a given period w. If

(1) a communication sequence σ_w is admissible;
(2) the nonzero eigenvalues of matrix A do not coincide with the zeros of the communication sequence characteristic polynomial $g(\mu)$;
(3) the pair (A, B) is controllable,

then the NCS remains controllable. □

Proof. The proof follows the same lines of the proof of Theorem 5 in Suzuki et al. (2011). Note that $\mathbf{u}(t) = \mathtt{diag}(\sigma_c(t))\mathbf{u}_\ell(t)$ which impacts equations from (9) to (40) in the following way $D_c(\cdot,\star) := \mathtt{diag}\big(\sigma_c(\cdot)\big)$. □

Using the same ideas as in Theorem 6 we state

Corollary 2. Consider a NCS with (20) and a processing unit $\mu_{\mathtt{p}}^{\mathtt{s}}$. Let the corresponding network be governed by the TOD protocol and let it have apc^{σ_w} If the pair (A, B) is controllable, then the NCS is controllable. □

w	σ_w	$\mathcal{G}(\mu)$	$\{\mu \vert g(\mu)=0\}$
2	$\{(0,1),(1,0)\}$	$\begin{bmatrix} \mu+1 & 0 \\ 0 & \mu+1 \end{bmatrix}$	$\{-1\}$
3	$\{(0,1),(0,0),(1,0)\}$	$\begin{bmatrix} \mu^2+\mu+1 & 0 \\ 0 & \mu^2+\mu+1 \end{bmatrix}$	$\{-0.5 \pm i0.86\}$
3	$\{(0,1),(0,1),(1,0)\}$	$\begin{bmatrix} \mu^2+\mu+1 & 0 \\ 0 & 2\mu^2+\mu \end{bmatrix}$	$\{-0.5, 0, -0.5 \pm i0.86\}$
4	$\{(0,1),(0,0),(0,0),(1,0)\}$	$\begin{bmatrix} \mu^3+\mu^2+\mu+1 & 0 \\ 0 & \mu^3+\mu^2+\mu+1 \end{bmatrix}$	$\{-1, \pm i\}$
4	$\{(0,1),(0,1),(0,0),(1,0)\}$	$\begin{bmatrix} \mu^3+\mu^2+\mu+1 & 0 \\ 0 & 2\mu^3+\mu^2+\mu \end{bmatrix}$	$\{-1, 0, \pm i, -0.25 \pm i0.66\}$
4	$\{(0,1),(0,1),(0,1),(1,0)\}$	$\begin{bmatrix} \mu^3+\mu^2+\mu+1 & 0 \\ 0 & 3\mu^3+\mu^2 \end{bmatrix}$	$\{-1, -0.33, 0, \pm i\}$
5	$\{(0,1),(0,0),(0,0),(0,0),(1,0)\}$	$\begin{bmatrix} \mu^4+\mu^3+\mu^2+\mu+1 & 0 \\ 0 & \mu^4+\mu^3+\mu^2+\mu+1 \end{bmatrix}$	$\{-0.81 \pm i0.56, 0.31 \pm i0.95, -0.62 \pm i0.51, 0.37 \pm i0.8\}$
5	$\{(0,1),(0,1),(0,0),(0,0),(1,0)\}$	$\begin{bmatrix} \mu^4+\mu^3+\mu^2+\mu+1 & 0 \\ 0 & 2\mu^4+\mu^3+\mu^2+\mu \end{bmatrix}$	$\{0, -0.74, 0.12 \pm i0.81, -0.62 \pm i0.5, 0.37 \pm i0.8\}$
5	$\{(0,1),(0,1),(0,1),(0,0),(1,0)\}$	$\begin{bmatrix} \mu^4+\mu^3+\mu^2+\mu+1 & 0 \\ 0 & 3\mu^4+\mu^3+\mu^2 \end{bmatrix}$	$\{0, -0.81 \pm i0.59, 0.31 \pm i0.95, -0.16 \pm i0.55\}$
5	$\{(0,1),(0,1),(0,1),(0,1),(1,0)\}$	$\begin{bmatrix} \mu^4+\mu^3+\mu^2+\mu+1 & 0 \\ 0 & 4\mu^4+\mu^3 \end{bmatrix}$	$\{0, -0.25, -0.81 \pm i0.59, 0.31 \pm i0.96\}$

Table 1. Communication sequence matrix polynomials for different periodic sequences and roots of the corresponding polynomials for a second ordered system; permutation of added elements, adding $(1,0)$ instead of $(0,1)$, or adding both where possible, does not generate new polynomials.

Proof. The proof differs from the proof of Theorem 6 only in the fact that instead of Theorem 5, Corollary 1 is used and in the last line instead of $y^{\mu_{\mathrm{P}}^{\mathrm{d}}}$, $y^{\mu_{\mathrm{P}}^{\mathrm{s}}}$ is used. □

5. CONCLUSIONS

This manuscript considers a NCS where the network is characterized by a dynamic protocol and different types of *additional processing capabilities*. First, NCS with general nonlinear plants were considered and it was shown that the corresponding plant stays controllable if an appropriate processing device is chosen and the network has appropriate *additional processing capability* and dynamic protocol; e.g., TOD protocol. Next, NCS with linear plants were considered and the corresponding controllability result from Suzuki et al. (2011) is extended.

REFERENCES

D'Innocenzo, A., Di Benedetto, M., and Serra, E. (2013). Fault tolerant control of multi-hop control networks. *IEEE TAC*, 58(6), 1377–1389.

Findeisen, R. and Varutti, P. (2009). Stabilizing nonlinear predictive control over nondeterministic communication networks. In *Nonlinear Model Predictive Control*, volume 384 of *Lecture Notes in Control and Information Sciences*, 167–179. Springer Berlin Heidelberg.

Findeisen, R., Grüene, L., Pannek, J., and Varutti, P. (2011). Robustness of Prediction Based Delay Compensation for Nonlinear Systems. In *Proceedings of the 18th IFAC World Congress*, 203–208.

Greco, L., Chaillet, A., and Bicchi, A. (2012). Exploiting packet size in uncertain nonlinear networked control systems. *Automatica*, 48(11), 2801 – 2811.

Hespanha, J., Naghshtabrizi, P., and Xu, Y. (2007). A survey of recent results in networked control systems. *Proceedings of the IEEE*, 95(1), 138–162.

Lian, F.L., Moyne, J., and Tilbury, D. (2005). Network protocols for networked control systems. In *Handbook of Networked and Embedded Control Systems*, Control Engineering, 651–675. Birkhäuser Boston.

Lješnjanin, M., Quevedo, D.E., and Nešić, D. (2014). Packetized MPC with Dynamic Scheduling Constrints and Bounded Packet Dropouts. *Automatica*, 50(3).

Montestruque, L. and Antsaklis, P. (2004). Stability of model-based networked control systems with time-varying transmission times. *IEEE TAC*, 49(9), 1562–1572.

Moyne, J.R. and Tilbury, D. (2007). The emergence of industrial control networks for manufacturing control, diagnostics, and safety data. *Proceedings of the IEEE*, 95(1), 29–47.

Munoz de la Pena, D. and Christofides, P. (2008). Lyapunov-based model predictive control of nonlinear systems subject to data losses. *IEEE TAC*, 53(9), 2076–2089.

Nešić, D. and Teel, A.R. (2004a). Input-to-state stability of networked control systems. *Automatica*, 40(12), 2121 – 2128.

Nešić, D. and Teel, A. (2004b). Input-output stability properties of networked control systems. *IEEE TAC*, 49(10), 1650–1667.

Pin, G. and Parisini, T. (2011). Networked predictive control of uncertain constrained nonlinear systems: Recursive feasibility and input-to-state stability analysis. *IEEE TAC*, 56(1), 72–87.

Polushin, I.G., Liu, P.X., and Lung, C.H. (2008). On the model-based approach to nonlinear networked control systems. *Automatica*, 44(9), 2409 – 2414.

Quevedo, D. and Nešić, D. (2011). Input-to-state stability of packetized predictive control over unreliable networks affected by packet-dropouts. *IEEE TAC*, 56(2), 370–375.

Riesel, H. (1994). *Prime numbers and computer methods for factorization*, volume 126. Springer.

Smarra, F., D'Innocenzo, A., and Benedetto, M.D.D. (2012). Fault tolerant stabilizability of mimo multi-hop cotrol networks. In *3rd IFAC Workshop on Distributed Estimiation and Control in Networked Systems*.

Suzuki, T., Kono, M., Takahashi, N., and Sato, O. (2011). Controllability and stabilizability of a networked control system with periodic communication constraints. *Systems & Control Letters*, 60(12), 977 – 984.

Yu, X. and Andersson, S.B. (2013). Effect of switching delay on a network control system. In *52nd IEEE Conference on Decision and Control*.