Available online at www.sciencedirect.com

ScienceDirect

IFAC-PapersOnLine 48-10 (2015) 147–152

Networked Control Systems with Application in the Industrial Tele-Robotics

A. Kheirkhah * **D. Aschenbrenner** * **M. Fritscher** * **F. Sittner** * **K. Schilling** **

* *Zentrum fuer Telematik, Allesgrundweg 12, DE-97218 Gerbrunn (e-mail: author@telematik-zentrum.de).*
** *Department of Computer Science VII: Robotics and Telematics, University of Wuerzburg, Am Hubland, DE-97074 Wuerzburg (e-mail: schi@informatik.uni-wuerzburg.de).*

Abstract: Networked Control Systems (NCS) are used for remote control of distributed or non-co-located systems where the control loop is closed via a communication link. Remote access to industrial robotic systems is one of the application fields where robustness and reliability of the closed loop plays a significant role.
Caused by the communication chain in the control loop, there are challenging constraints that can affect the stability and performance of a closed loop. One can consider a large number of difficulties in this type of control systems, for instance unknown delay and packet drop-out.
There are already a large number of methods and approaches to handle issues in the NCS and because of increasing interest of the industry, many are still being developed to improve the features of the closed loops over communication networks.
This paper summarizes the existing theory of the networked control systems under communication constraints and presents an H_∞ synthesis for a simple plant. Finally the results are illustrated, discussed and validated using real measurements from a robotic system.

Keywords: Network Control; Computer-based control systems; Training and adaptation algorithms, Constructive algorithms; Tele-robotics

1. INTRODUCTION

In the last two decades conventional industrial robots are heavily applied in industrial applications. In the recent years, teleoperation of robots has been paid much attention by researchers, especially remote controlled robots in hazardous operations, like nuclear or high radioactive environments, or space or planetary exploration operations. But also in the context of a working facility, these achievements in teleoperation provide a benefit: If robots can be controlled remotely, maintenance and optimization tasks can be carried out with external support of robotics experts, who would be too expensive if they had to travel to the facility. If teleoperation is carried out via the Internet, the quality of the connection can decrease spontaneously. A control algorithm applied to a teleoperation task needs to cope with these circumstances and ensure a safe teleoperation.

2. APPLICATION

The algorithms described in this article have been developed to support a telemaintenance system which has been established in an active production line for research purposes. The local system consists of a six-axis cartesian industrial Reis-Robot, a two-component injection molding system and a handling unit which altogether produce plastic parts for electric toothbrushes. The system has been set up during the ongoing project MainTelRob which is

Fig. 1. The robotic system to be controlled remotely

performed by a cooperation of the Zentrum fuer Telematik e.V. (ZfT) and its project partners, Kuka Industries and Braun Marktheidenfeld, which belongs to Procter & Gamble. Both companies believe in a positive effect of the telemaintenance concept. The local service technician might discover problems with the robot, which he cannot solve for himself in a short time. If external experts are called at the moment, they rely on telephone support or the travel to the facility, which is both expensive and time

Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2015.08.123

consuming. In the teleoperation scenario, the challenge lies in providing a good situation awareness for the expert on the one hand and ensuring a safe operation process on the other hand, so that the robot doesnt damage anything, because to packet loss or other network issues.

3. NCS WITH PACKET DROPOUT

Packet dropouts are one of the most common problems in networked control systems. Especially because of the nature of network communications, dropouts occur randomly and cannot be defined deterministically. In most approaches, a probabilistic process describes the dropout and the controller is designed to achieve stochastic stability of the closed loop. Here, the network is assumed to be modelled as a simple packet dropout generator with a given probability p. The states of the network are modelled by a Markovian chain. For more detail, refer to Seiler (2001), Nilsson (1998) or Naghshtabrizi et al. (2007).

3.1 Markovian Jump Linaear Systems (MJLS)

Assume the linear time discrete system

$$\begin{cases} \mathbf{x}(k+1) = \mathbf{A}\,\mathbf{x}(k) + \mathbf{B}\,\mathbf{u}(k) \\ \mathbf{y}(k) = \mathbf{C}\,\mathbf{x}(k) \end{cases} \tag{1}$$

With a network in the closed loop modelled by a simple dropout as shown in Figure 2.

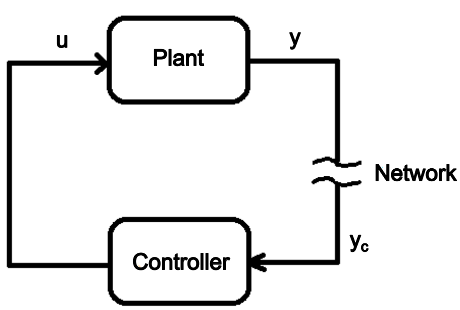

Fig. 2. Schematic closed loop including the network

The output of the network can be modelled by the following expression:

$$\mathbf{y}_c(k) = \theta(k)\,\mathbf{y}(k) + (1-\theta(k))\,\mathbf{y}(k-1) \tag{2}$$

$\theta \in 0, 1$ defines the state of the network, where 0 indicates packet dropout and 1 the regular transmission of data. Assuming p as the probability of a packet being dropped:

$$\begin{array}{c} P[\theta = 0] = p \\ P[\theta = 1] = 1 - p \end{array} \tag{3}$$

and $\theta(k)$ is defined by a two-state Markovian chain.

Using the following augmentation of the state vector

$$\tilde{\mathbf{x}}(k) = \left[\, \mathbf{x}(k)^T \;\; \mathbf{y}_c^T(k-1) \,\right]^T \tag{4}$$

The plant including the network can be described by:

$$\begin{cases} \tilde{\mathbf{x}}(k+1) = \tilde{\mathbf{A}}_{\theta(k)}\,\tilde{\mathbf{x}}(k) + \tilde{\mathbf{B}}\,\mathbf{u}(k) \\ \mathbf{y}_c(k) = \tilde{\mathbf{C}}_{\theta(k)}\,\tilde{\mathbf{x}}(k) \end{cases} \tag{5}$$

where

$$\begin{aligned} \tilde{\mathbf{A}}_{\theta(k)} &= \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ \theta(k)\,\mathbf{C} & (1-\theta(k))\,\mathbf{I} \end{bmatrix} \\ \tilde{\mathbf{B}} &= \begin{bmatrix} \mathbf{B} \\ \mathbf{0} \end{bmatrix}^T \\ \tilde{\mathbf{C}}_{\theta(k)} &= [\,\theta(k)\,\mathbf{C} \;\; (1-\theta(k))\,\mathbf{I}\,] \end{aligned} \tag{6}$$

The equations 5 and 6 introduce a time varying state space description where the dependency of the network is included in the system equations.

3.2 Stability of stochastic systems

Assume a general stochastic system in the form

$$\begin{cases} \mathbf{x}(k+1) = \mathbf{A}_{\theta(k)}\,\mathbf{x}(k) + \mathbf{B}_{\theta(k)}\,\mathbf{u}(k) \\ \mathbf{y}(k) = \mathbf{C}_{\theta(k)}\,\mathbf{x}(k) \\ \mathbf{x}(0) = \mathbf{x}_0\,;\ \theta(0) = \theta_0 \end{cases} \tag{7}$$

Assume also that the transition matrix of the Markovian chain is given by $\mathbf{P} = [p_{i,j}]$ where

$$p_{i,j} := Pr(\theta(k+1) = j \,|\, \theta(k) = i)$$

There are several definitions of stability for stochastic systems in form of (7). One of these definitions is described below, referring to Seiler (2001):

Definition 1. The autonomous system from (7) ($\mathbf{u} = \mathbf{0}$) is called **mean-square stable** (MSS) if for every initial state $(\tilde{\mathbf{x}}_0, \theta_0)$, $\lim_{k\to\infty} E[\|\mathbf{x}(k)\|^2] = 0$.

To prove the mean-squared stability, the following theorem can be used:

Theorem 1. If $p_{i,j} = p_j$ for all $i, j \in 1, \cdots, N$ then the system (7) is MSS if and only if there exists a matrix $\mathbf{G} > 0$ such that

$$\mathbf{G} - \Sigma_{j=1}^{N}\, p_j \mathbf{A}_j^T\,\mathbf{G}\,\mathbf{A}_j > 0 \tag{8}$$

Theorem 1 uses a Lyapunov based method and searching for a positive definite matrix which satisfies a Lyapunov equation. A more general theorem can be found in Montestruque and Antsaklis (2004).

3.3 Closed loop and controller design

To achieve stability of the closed loop using theorem 1, the controller is taken to be also a dynamic system in the form

$$\begin{cases} {}^c\mathbf{x}(k+1) = {}^c\mathbf{A}_i\,{}^c\mathbf{x}(k) + {}^c\mathbf{B}_i\,\mathbf{y}_c(k) \\ \mathbf{u}(k) = {}^c\mathbf{C}_i\,{}^c\mathbf{x}(k) \end{cases} \tag{9}$$

where X_i simplifies $[X_{\theta(k)} \,|\, \theta(k) = i]$. Assuming the Markovian chain with two states $\theta \in 0, 1$, the closed loop is also a dynamic system in the form of (7) with the system matrix

$${}^{cl}\mathbf{A}_i = \begin{bmatrix} \tilde{\mathbf{A}}_i & \tilde{\mathbf{B}}_i\,{}^c\mathbf{C}_i \\ {}^c\mathbf{B}_i\,\tilde{\mathbf{C}}_i & {}^c\mathbf{A}_i \end{bmatrix} \quad \text{for } i \in 0, 1 \tag{10}$$

For this matrix using the Schur complements, a more useful form of the condition (8) can be obtained.

$$\begin{bmatrix} \mathbf{G}^{-1} & \star^T & \star^T \\ \sqrt{p}\,{}^{cl}\mathbf{A}_0\,\mathbf{G}^{-1} & \mathbf{G}^{-1} & \mathbf{0} \\ \sqrt{1-p}\,{}^{cl}\mathbf{A}_1\,\mathbf{G}^{-1} & \mathbf{0} & \mathbf{G}^{-1} \end{bmatrix} > 0 \tag{11}$$