2011 American Control Conference
on O'Farrell Street, San Francisco, CA, USA
June 29 - July 01, 2011

Adaptive Control of a Networked Control System with Hierarchical Scheduling

Harald Voit and Anuradha Annaswamy

***Abstract*— In this paper, we consider a special class of Networked Control Systems (NCS) that occur in embedded systems where a hierarchical schedule is employed and uncertainties may be present either in the plant or in the network. An adaptive controller is proposed to accommodate the effect of uncertainties. It is shown that this adaptive controller can accommodate the uncertainties, stabilize the system, make use of the structure of the hierarchical scheduler in its design, and result in improved performance compared to non-adaptive NCS.**

I. INTRODUCTION

Embedded control systems are ubiquitous and can be found in several applications including aircraft, automobiles, process control, and buildings. An embedded control system is one in which the computer system is designed to perform dedicated functions with real-time computational constraints [1]. Typical features of such embedded control systems are shared networks used by different components of the systems to communicate with each other, a large number of sensors as well as actuators, and their distributed presence in the overall system. The design of embedded controllers and the communication and computational networks that support them poses a number of challenges in their analysis and synthesis including network protocols, compatibility of operating systems, and methods for optimizing the combined performance of the control and real-time computing systems.

A common feature in networked control systems is the need for shared resources. Constrained by space, speed, and cost, often information has to be transmitted using a shared communication network. In order to manage the flow of information, protocols that are time-triggered [2] and event-triggered [3, 4] have been suggested over the years. In order to maintain flexibility in scheduling while minimizing delays for critical functions and conserving resources, a hierarchical schedule such as FlexRay has been proposed in [5] and has been used in several applications [6].

The digital implementation of the communication network introduces a delay. Therefore, for accurate performance, the control system has to explicitly incorporate the delay in its design. A good deal of the research on networked control systems (NCS) is focused on this effort (for example, [7, 8] and references therein). A specific aspect of NCS is the combined design of its two major components of control and communication. This co-design is being explored more recently in [9–24]. In [12], a Control Server was used to reduce the input-output latency. In addition, different strategies such as a feedback loop for sampling times were used to minimize jitter. In [22], feedback-feedforward loop was used to adjust the sampling period of the control tasks in order to optimize a performance criterion. In [14], the problem of optimal control and scheduling was solved by transforming the system into a mixed integer quadratic programming problem. Control theoretic principles based on linear systems, feedback control, and optimization are used to determine parameters such as sampling period and resource allocation so as to maintain both an efficient control performance and CPS utilization. In [18] and [23], the schedulability analysis of real-time tasks with respect to the stability of control functions is discussed. However, the platforms are simple platforms and the focus of our paper is the use of more advanced platforms. In [20], modeling the real-time scheduling process as a dynamic system, an adaptive self-tuning regulator is proposed to adjust the bandwidth of each single task in order to achieve an efficient CPS utilization. But the focus, however, is on a single processor. In [24], for subsets of the set of applications a schedule is derived with an iterative procedure. Theses schedules are stored on the processor if they are necessary and fit into the memory. A good survey paper on co-design can be found in [13].

In both [16] and [25], the implementation of multiple control applications on multiple computational nodes is considered. In [25], an empirical co-design is proposed by which a controller is first derived that is robust to specified sampling period and delays, and then a scheduling procedure is proposed that is shown in simulation results to lead to a desired QoC and meet the scheduling constraints. No analytical guarantees are provided for this co-design. Also, no hybrid protocols are considered. If any changes are made, this entire process of control and scheduling designs is repeated. A similar sequential co-design is proposed in [16], where a control system is designed using a delay-impulsive modeling approach first, and then a physical architecture with an EDF scheduling policy is chosen. A schedulability analysis is carried out that guarantees that all control-related messages can be transmitted in finite time. If schedulability is not feasible, the design procedure starts over.

While scheduling in general introduces a delay, the use of a hierarchical scheduling policy introduces a more specific

This work was supported by the Technische Universität München - Institute for Advanced Study, funded by the German Excellence Initiative and by Deutsche Forschungsgemeinschaft (DFG) through the TUM International Graduate School of Science and Engineering (IGSSE).

Harald Voit is with the Institute of Automatic Control Engineering, Technische Universität München, D-80290 Munich, Germany. harald.voit@tum.de

Anuradha Annaswamy is with the Department of Mechanical Engineering, Massachussets Institute of Technology, Cambridge, MA 02319, USA. aanna@mit.edu

structure into the problem. In particular, the hierarchical scheduling services different parts of the subsystem at specific time-instances thereby providing more information about the inherent delay. That is, while latency is indeed present between inputs and outputs due to the presence of a network, the network together with a hierarchical schedule allows a part of this delay to be known to the control designer. In this paper, we propose a control design that makes use of such a structure.

In an embedded control system, as in any control system, uncertainties are present. Uncertainties can be present in the plant to be controlled due to changes in the operating conditions. Network parameters such as sampling period, slot-size, and the network latency may change as well due to varying needs of power consumption, environmental conditions, and emergence of critical sub-systems. They can also be present in the network. In order to accommodate these uncertainties, the underlying controller may need to be adaptive. This is our focus in this paper. We design this adaptive controller so that all prior information that is available is explicitly made use of. In particular, in the hierarchical scheduler, if a part of the delay is known, we make use of the delay in selecting the control structure. Adaptive control in NCS is a topic that is being explored relatively recently. While a gain-scheduled approach has been used more commonly [26, 27], an adaptive approach which combines elements of parameter estimation followed by an update in the control algorithm has been investigated minimally (see [28, 29] for example). In [28], a least-squares error is used to adjust the estimate of the unknown time-delay, while in [29], the controller gains are recalculated every so often, assuming that the delay has changed and is known, so that a desired performance metric is optimized. No theoretical discussion of the underlying stability is provided in either of these papers. In contrast, we carry out a detailed stability discussion, adapt to unknown delays, and suitably utilize any partial information.

In Section II, we state the underlying problem of control using an embedded system with a hierarchical schedule. The relevant stability result from NCS is stated and discussed. In Section III, we propose an adaptive controller which accommodates the uncertainties, stabilizes the system, and makes use of the structure of the hierarchical scheduler. Finally, we present numerical simulations in section IV which shows the advantage of the proposed adaptive controller over non-adaptive ones proposed in the literature.

II. THE UNDERLYING NCS MODEL

In this section we introduce the NCS model we use in the rest of the paper. The communication bus in the embedded control system is digital as a result of which it is necessary to sample the plant to get digital signals and to translate the control inputs into piecewise-continuous signals. While the first is done by periodically sending sensor measurements of the plant output (or states) to the controller, the latter is done by a zero-order hold (ZOH) device or any higher order hold device. The overall structure of our NCS model is shown in Figure 1. Each device or processing unit that has access to the network is called a node.

The second aspect of an embedded control system is the presence of a shared communication medium. All communication between the sensors, actuators, and controllers is processed over the shared medium. Since at a time only one message can be transmitted, a schedule is needed that guarantees that all subsystems can exchange enough information in order to achieve their goals. However, the presence of a schedule introduces a communication delay.

As denoted in Figure 1, the network introduces two time delays, τ_{sc} and τ_{ca}. These two time delays are communication induced. The third time delay, τ_c represents the computation time in the controller node. We assume that the sensor is clock-driven, while the controller and the actuator are event-driven. It is also possible to consider a clock-driven controller as in [9] but we do not consider this case here. The total time-delay of the NCS is given by $\tau = \tau_{sc} + \tau_{ca} + \tau_c$. Usually, the computational delay τ_c is small compared to τ_{sc} and τ_{ca} and is therefore neglected in the following.

To coordinate the message exchange between nodes, the underlying schedule needs to be arbitrated in a sophisticated manner. The arbitration may be fixed, dynamic, or hierarchical. A fixed schedule assigns one time slot to each message which will be repeated periodically, and is referred to as a Time Division Multiple Access (TDMA) strategy. Dynamic schedules assign priorities to messages which are ready for transmission and grants the message with the highest priority access to the network, e.g. Earliest Deadline First (EDF). A hierarchical scheduler has multiple fixed and dynamic levels embedded in a hierarchical structure [17]. For example, in a first level of the hierarchy, a subset of messages is assigned a time slot. In the second level, each of these messages has a fixed priority. This is referred to as a hierarchical TDMA/FPS (Figure 3) [17]. This may be needed in problems where multiple control loops have to be designed at multiple time-scales. In [17], a co-design of control and hierarchical scheduling is carried out and is shown to result in optimal performance.

We now state the problem that we address in this paper. Our focus is on an embedded control system, and to control a plant whose model is linear and is given by

$$\dot{x}(t) = Ax(t) + Bu(t). \qquad (1)$$

Since the controller is implemented in a digital manner, we assume that a discrete state feedback controller of the following form is implemented:

$$u(k) = -Kx(k), \qquad k = 0, 1, 2, 3, \ldots \qquad (2)$$

This controller is implemented using a shared communication network as described above. Since the communication network may have parameters that may be subject to uncertainties, we may design the controller in (2) to be adaptive. The goal is to control the continuous plant in (1) using a discrete, possibly adaptive, controller while the operating conditions of the communication network or of the plant change.

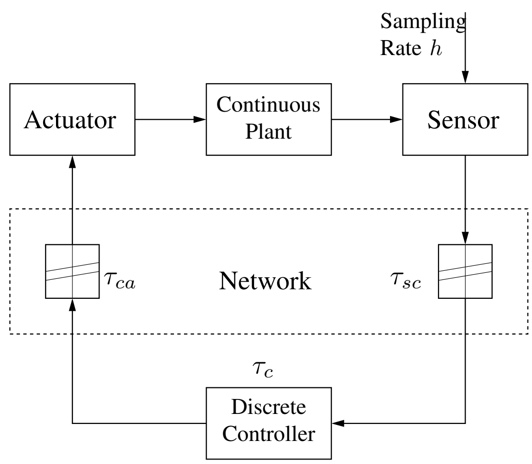

Fig. 1. The underlying NCS Model

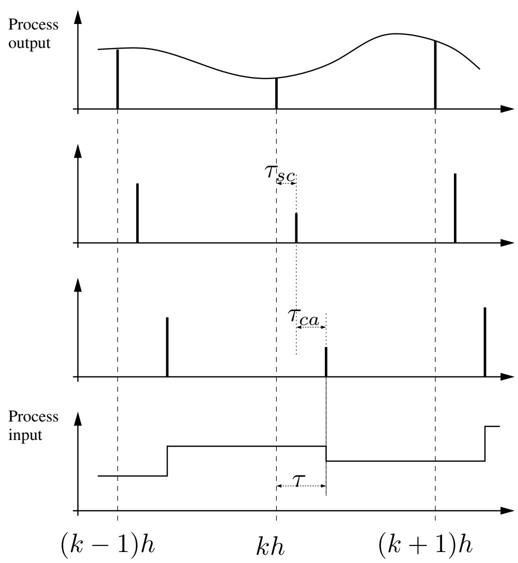

Fig. 2. Timing of an event-driven control for a single plant

We assume that the total time delay in the system τ is less than one sampling period h. Then the system can be written as

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ u(t^+) &= -Kx(t-\tau). \end{aligned} \tag{3}$$

We refer to this controller as a static controller. We can then compute the state of the plant at the sampling time $(k+1)h$ as [30]

$$\begin{aligned} &x((k+1)h) = \\ &\quad e^{Ah}x(kh) + \int_{kh}^{kh+h} e^{A(kh+h-s')}Bu(s'-\tau)ds' \\ &\quad = \int_{kh}^{kh+\tau} e^{A(kh+h-s')}Bds'u(kh-h) + \\ &\quad \int_{kh+\tau}^{kh+h} e^{A(kh+h-s')}Bds'u(kh) + e^{Ah}x(kh) \end{aligned} \tag{4}$$

where the last step follows from the fact that the control input is held constant over one sampling period but changes due to the time delay (see Figure 2). With

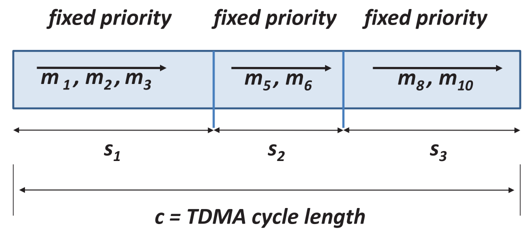

Fig. 3. Hierachical TDMA/FPS

$$\begin{aligned} \Phi &= e^{Ah} \\ \Gamma_0 &= \int_0^{h-\tau} e^{As}dsB \\ \Gamma_1 &= e^{A(h-\tau)}\int_0^{\tau} e^{As}dsB \end{aligned} \tag{5}$$

we can write the sampled version of (3) as

$$x(kh+h) = \Phi x(kh) + \Gamma_1 u(kh-h) + \Gamma_0 u(kh). \tag{6}$$

Note that Φ, Γ_0, and Γ_1 depend on the network parameters h and τ. If any one of these parameters changes, the whole system changes.

A. Stability of a NCS

The stability of the NCS described by Equation (6) is established in [31] where methods based on stability of hybrid systems are used to establish the following result:

Theorem 1: If all eigenvalues of

$$H = \begin{bmatrix} e^{Ah} & -E(h)BK \\ e^{A(h-\tau)} & -e^{A\tau}(E(h)-E(\tau))BK \end{bmatrix}, \tag{7}$$

where $E(x) = \int_0^x e^{A(x-s)}ds$, have magnitude less than one, then the NCS is stable.

Proof: See [31]. ■

Theorem 1 implies that for a given A and B and a sampling period h, a controller parameter K and time delay τ can be determined so that stability is guaranteed using Eq. (7). This also implies that instability may occur if either the plant parameters or the network parameters change.

B. Arbitrated Networked Control Systems

In this section we describe how the knowledge about the network and its associated parameters can be used to improve the performance of the NCS. As mentioned before, a schedule is needed if more than one plant uses the network to communicate with its controller node (or if there is only one controller node which controls several plants connected to the same bus). Suppose that there are two plants each equipped with a sensor, actuator, and a controller node. One of the simplest protocols for such a case is a time triggered protocol such as the TDMA protocol. The communication bandwidth is divided into equal cycles of length c. Each cycle is further divided into m slots which do not have to have the same size. Let us assume that the size of each slot is an integer multiple of a minimum slot size s called base slot size and that the sampling periods of the plants can be written as

$$h = l \cdot s, \tag{8}$$

i.e. each plant is sampled after l basis slots.

Let σ_x (and σ_u) denote the number of the slot when the sensor (and the controller) has access to the network, respectively. Then we can write the sampled system in (4) as

$$\begin{aligned} x(k+1) &= e^{Als}x(k) \\ &+ \int_{\sigma_x+kls}^{\sigma_u+kls+\tau_{ca}} e^{A(\sigma_x+(k+1)ls-\xi)}d\xi Bu(\sigma_u+(k-1)ls) \\ &+ \int_{\sigma_u+kls+\tau_{ca}}^{\sigma_x+(k+1)ls} e^{A(\sigma_x+(k+1)ls-\xi)}d\xi Bu(\sigma_u+kls) \end{aligned} \tag{9}$$

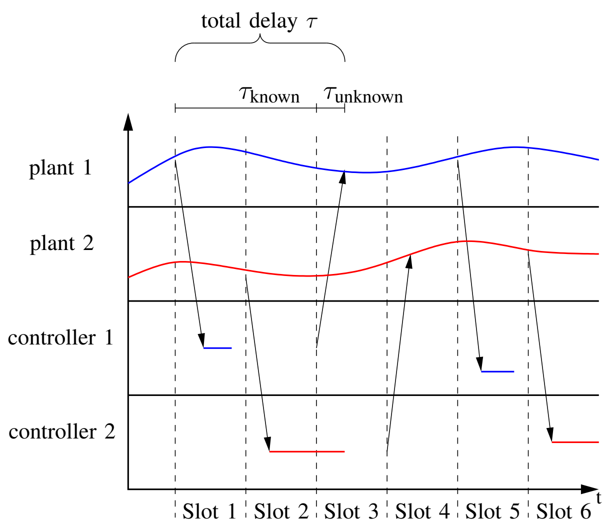

Fig. 4. Effect of Scheduling on the Control of Multiple Plants

An example of the resulting network control system is shown in Figure 4. This case corresponds to a scenario where two plants communicate with their controllers using a single communication bus. This bus therefore has to arbitrate the scheduling of messages sent by both plants to the controllers and vice versa. Typically a TDMA protocol is used to carry out such arbitration. The class of such network control systems is therefore denoted as *Arbitrated* Networked Control System (ANCS). When Plant 1 is sampled and the sensor measurement is sent through the network, Controller 1 can begin its computation (Slot 1). The schedule does not grant Controller 1 access to the network in Slot 2 when its message would be ready for transmission. Therefore, Controller 1 has to wait until Slot 3. This results in a known amount of delay of twice the base slot size, i.e. $\tau_{\text{known}} = 2 \cdot s$. In Slot 3 Controller 1 is allowed to transmit its message. This message will arrive after an unknown transmission time. Hence, an unknown time delay τ_{unknown} related to the communication is added to the total delay experienced by the system. Therefore the total delay τ that the system experiences is composed of two parts, (i) a known part τ_{known} due to the schedule, (ii) an unknown part τ_{unknown} due to the finite communication speed of the network, with $\tau = \tau_{\text{known}} + \tau_{\text{unknown}}$.

The advantage of this model of the overall ANCS is the partial knowledge of the time delay. This can be used to design controllers which make use of the known delay in a suitable manner and are robust with respect to the unknown delay. Such a controller is described in the next section.

III. Adaptive Controller Design

The network parameters can have an important influence on the stability of the NCS. In reality, the delay experienced by the plant is distributed in a random manner and depends, among other things, on the operating conditions of the system. The operating conditions may, for example, depend on power consumption considerations. Furthermore, the plant may include uncertainties due to changes in the operating environment of the plant or modeling errors. In this section we develop an adaptive controller that ensures stability and tracking of a reference signal under changes in the network parameters and uncertainties and modeling errors in the plant to be controlled.

A. The Adaptive Controller

We rewrite (6) as

$$\begin{aligned} x(k+1) &= \Phi x(k) + \Gamma_0 u(k-1) + \Gamma_1 u(k) \\ &= p^T w(k) + \Gamma_1 u(k) \end{aligned} \tag{10}$$

where $p^T = [\Phi \;\; \Gamma_0]$ and $w^T(k) = [x(k) \;\; u(k-1)]$. The goal is to design a controller such that the closed-loop system mimics the behavior of a reference model. This model is chosen as

$$x_m(k+1) = r(k) \tag{11}$$

where $r(k)$ is a bounded reference signal which should be tracked by the NCS with a unit delay. The reference model is located at the controller and is therefore not affected by network induced delays. The structure of the plant in (10) implies that the tracking objective can be met by using the controller

$$u(k) = -p_1^{*T} w(k) + p_2^* r(k), \tag{12}$$

where $p_1^{*T} = \Gamma_1^{-1} p^T$ and $p_2^* = \Gamma_1^{-1}$. We note that p_1^* and p_2^* are functions of τ, A, and B and are hence unknown. The controller in (12) cannot be implemented since the parameters of (12) are dependent on the network as well as the plant to be controlled and are assumed to be unknown. We therefore implement an adaptive controller where the control parameters are estimated as $\hat{p}_1(k)$ and $\widehat{p}_2(k)$, and is given by the adaptive version of (12),

$$u(k) = -\hat{p}_1^T(k) w(k) + \hat{p}_2(k) r(k), \tag{13}$$

where $\hat{p}_1$ and $\hat{p}_2$ are estimates of the unknown parameters $\Gamma_1^{-1} p^T$ and Γ_1^{-1}, respectively. The closed loop system is thus given by

$$x(k+1) = \Gamma_1(\Gamma_1^{-1} p^T - \hat{p}_1^T(k)) w(k) + \Gamma_1 \hat{p}_2(k) r(k). \tag{14}$$

The problem is to determine adaptive laws for the adjustment of $\hat{p}_1(k)$ and $\hat{p}_2(k)$ such that the closed-loop system behaves like the reference model in (11). Defining $e(k) = x(k) - x_m(k)$ we obtain from (11) and (14) that

$$e(k+1) = \Gamma_1 \tilde{p}_1^T(k) w(k) + \Gamma_1 \tilde{p}_2(k) r(k), \tag{15}$$

where $\tilde{p}_1(k) = \Gamma_1^{-1} p - \hat{p}_1(k)$ and $\tilde{p}_2(k) = \hat{p}_2(k) - \Gamma_1^{-1}$ are the parameter estimation errors. We use a gradient design to update the estimation errors as follows [32]:

$$\begin{aligned} \tilde{p}_1(k+1) &= \tilde{p}_1(k) - \frac{\gamma_1 \text{sgn}(\Gamma_1) e(k+1) w(k)}{1 + w^T(k) w(k) + r^2(k)} \\ \tilde{p}_2(k+1) &= \tilde{p}_2(k) - \frac{\gamma_2 \text{sgn}(\Gamma_1) e(k+1) r(k)}{1 + w^T(k) w(k) + r^2(k)} \end{aligned} \tag{16}$$

with the adaptation gains $\gamma_1$and γ_2 which satisfy $0 < \gamma_1 + \gamma_2 < 2/|\Gamma_1|$.

B. Proof of Stability

Theorem 2: The closed-loop adaptive system defined by the plant in (6), the adaptive controller in (13), and the adaptive laws in (16) is globally stable.

Proof: In order to show that the adaptive controller (13) results in a stable system, we consider the positive function

$$V(k) = |\Gamma_1| \gamma_1^{-1} \tilde{p}_1^T(k) \tilde{p}_1(k) + |\Gamma_1| \gamma_2^{-1} \tilde{p}_2^2. \tag{17}$$

The time increment of $V(k)$ along (15) and (16) is

$$\begin{aligned} \Delta V(k) &= V(k+1) - V(k) \\ &= |\Gamma_1| \gamma_1^{-1} \tilde{p}_1^T(k+1) \tilde{p}_1(k+1) + |\Gamma_1| \gamma_2^{-1} \tilde{p}_2^2(k+1) \\ &\quad - |\Gamma_1| \gamma_1^{-1} \tilde{p}_1^T(k) \gamma_1^{-1} \tilde{p}_1(k) - |\Gamma_1| \gamma_2^{-1} \tilde{p}_2^2(k) \\ &= \frac{e(k+1)}{m^2(k)} \Gamma_1 (-\tilde{p}_2^T w - w^T \tilde{p}_1 - 2 \tilde{p}_2 r(k)) \\ &\quad + \frac{|\Gamma_1| e^2(k+1)}{m^2(k)} \frac{\gamma_1 w^T(k) w(k) + \gamma_2 r^2(k)}{m^2(k)} \\ &= \frac{e^2(k+1)}{m^2(k)} \cdot \\ &\quad \left(\frac{|\Gamma_1| \gamma_1 w^T(k) w(k) + |\Gamma_1| \gamma_2 r^2(k)}{m^2(k)} - 2 \right) \\ &< 0 \end{aligned} \tag{18}$$

Hence, $V(k)$ is a Lyapunov function and the parameter errors $\tilde{p}_1$ and $\tilde{p}_2$ are bounded. Thus, using standard techniques stated for example in [32], it follows that the system defined by the plant in (6), the adaptive controller in (13), and the adaptive laws in (16) is globally stable. ∎

C. Choice of Initial Conditions

As described in Section II-B, the partial knowledge of the time delay experienced by the system can be used to design more efficient controllers. Here we use this knowledge to choose the initial values of the parameter estimates $\hat{p}_1$ and $\hat{p}_2$. From Equation (5) it is clear that Φ and Γ_1 are functions of τ. Denoting the control parameters in (12) as $p_1^*(A, B, \tau)$ and $p_2^*(A, B, \tau)$, and the nominal values of A and B as $A_{nominal}$ and $B_{nominal}$, we choose

$$\begin{aligned} \hat{p}_1(0) &= p_1^*(A_{nominal}, B_{nominal}, \tau_{known}) \\ \hat{p}_2(0) &= p_2^*(A_{nominal}, B_{nominal}, \tau_{known}) \end{aligned} \tag{19}$$

It should be pointed out that the adaptive control design proposed here explicitly makes use of the prior knowledge of τ, by incorporating the knowledge of τ_{known} in the initial parameter estimates of the controller, and adapting to the uncertainty due to τ_{unknown}.

IV. Simulation Results

In this section we provide numerical simulations to illustrate the results of the previous sections. To this end we consider the scalar system

$$\dot{x}(t) = 2x(t) + u(t). \tag{20}$$

The goal is to stabilize (20) and have its output follow that of a reference model given by

$$x_m(k) \equiv 1 \tag{21}$$

It is assumed that the sampling rate $h = 0.001s$. The network induced delay is assumed to be fixed with $\tau = 0.0005s$. First, we design a static feedback control gain using the stability criterion given in Section II-A. For $K = -2.5$ the matrix H in (7) is given as

$$H = \begin{bmatrix} 1.0020 & -0.0025 \\ 1.0010 & -0.0013 \end{bmatrix} \tag{22}$$

with eigenvalues

$$\text{eig}(H) = [0.9995 \quad 0.0013]. \tag{23}$$

Hence, the system is stabilized by K. The resulting closed-loop system response is shown in Figure 6.

Next, we choose the adaptive controller specified by Eqs. (6), (13), and (16), with initial conditions $\hat{p}_1(0) = [1.002 \quad 5.0025 \cdot 10^{-4}]$ and $\hat{p}_2(0) = 5.0025 \cdot 10^{-4}$. The response of the resulting adaptive system is shown in Figure 5. It should be noted that both the controller in Section II and the adaptive controller have a similar, state-feedback based structure.

Figures 5 and 6 show the closed-loop responses with the adaptive and fixed controller. At $t = 9s$, a change in the operating conditions is assumed to occur, which may be due to considerations of power consumption, causing the sampling rate to reduce from 1ms to 10ms. The results in Figure 6 shows that the fixed controller is unable to track the reference signal. Figure 5, on the other hand, shows that the adaptive controller can easily accommodate this change.

V. Summary

We considered a special class of Networked Control Systems that occur in embedded systems where a hierarchical schedule is employed, denoted as Arbitrated Networked Control System (ANCS) [33]. A new model is proposed to represent its features, and is shown to be in the form of an NCS with a partial knowledge of the delay. We addressed the effect of uncertainties that may be present either in the plant to be controlled or in the network by introducing an adaptive controller. This adaptive controller is shown to accommodate the uncertainties, stabilize the system, and make use of the structure of the hierarchical scheduler. Numerical simulations were carried out and shown to support the theoretical derivations.

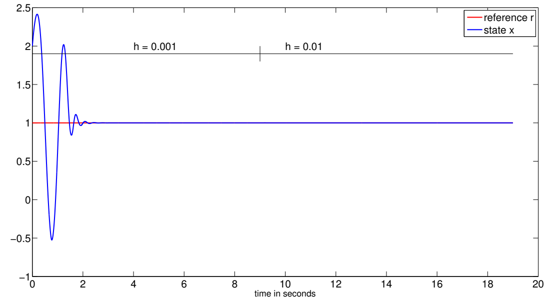

Fig. 5. Closed-loop Response of (20) with the adaptive controller

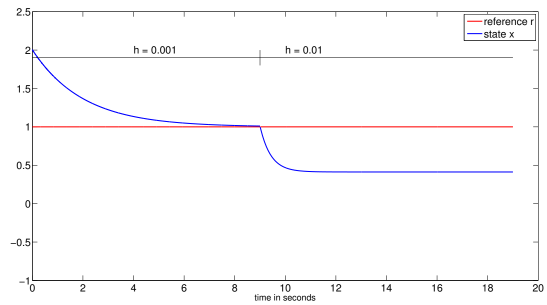

Fig. 6. Closed-loop response of (20) with the static controller $K = -2.5$

VI. ACKNOWLEDGMENTS

The authors would like to thank Dr. Samarjit Chakraborty, TUM, for several valuable discussions on real-time computing and scheduling.

REFERENCES

[1] J. W. S. Liu, *Real-Time Systems*. Prentice Hall, 2000.

[2] T. Nghiem, G. J. Pappas, R. Alur, and A. Girard, "Time-triggered implementations of dynamic controllers," in *Proceedings of the 6th ACM & IEEE International conference on Embedded software*, ser. EMSOFT '06, 2006, pp. 2–11. [Online]. Available: http://doi.acm.org/10.1145/1176887.1176890

[3] A. Cervin and T. Henningsson, "Scheduling of event-triggered controllers on a shared network," in *47th IEEE Conference on Decision and Control*, 2008, pp. 3601–3606.

[4] X. Wang and M. Lemmon, "Event-triggering in distributed networked control systems," *IEEE Transactions on Automatic Control*, vol. 56, no. 3, pp. 586 –601, 2011.

[5] AUTOSTAR, "Specification of flexray interface, ver. 3.0.3," www.autostar.org, Tech. Rep.

[6] "Bmw brake system relies on flexray," http://www.automotivedesignline.com/news/218501196, July 2009.

[7] J. Hespanha, P. Naghshtabrizi, and Y. Xu, "A survey of recent results in networked control systems," *Proceedings of the IEEE*, vol. 95, pp. 138 –162, 2007.

[8] Y. Tipsuwan and M.-Y. Chow, "Control methodologies in networked control system," *Control Engineering Practice*, vol. 11, pp. 1099–1111, 2003.

[9] Y. Halevi and A. Ray, "Integrated communication and control systems: Part i - analysis," *Journal of Dynamic Systems, Measurement, and Control*, vol. 110, pp. 367–373, 1988.

[10] K.-E. Årzén, A. Cervin, J. Eker, and L. Sha, "An introduction to control and scheduling co-design," in *Proceedings of the 39th IEEE Conference on Decision and Control, 2000*, vol. 5, 2000, pp. 4865–4870.

[11] G. Walsh and H. Ye, "Scheduling of networked control systems," *IEEE Control Systems Magazine*, vol. 21, no. 1, pp. 57 –65, 2001.

[12] A. Cervin, "Integrated control and real-time scheduling," Ph.D. dissertation, Department of Automatic Control, Lund University, Sweden, 2003.

[13] F. Xia and Y. Sun, "Control-scheduling codesign: A perspective on integrating control and computing," *Dynamics of Continuous, Discrete and Impulsive Systems*, vol. 13, no. S1, pp. 1352–1358, 2006. [Online]. Available: http://arxiv.org/pdf/0806.1385

[14] M. Gaid, A. Cela, and Y. Hamam, "Optimal integrated control and scheduling of networked control systems with communication constraints: application to a car suspension system," *IEEE Transactions on Control Systems Technology*, vol. 14, no. 4, pp. 776 – 787, july 2006.

[15] G. Che and Y. Jin, "Online co-design of feedback control and real-time scheduling for embedded systems with commnuication delays," in *4th International Conference on Computer Science Education*, 2009, pp. 618–623.

[16] P. Naghshtabrizi and J. Hespanha, "Analysis of distributed control systems with shared communication and computation resource," in *Proc. of the 2009 Amer. Contr. Conf.*, 2009.

[17] H. Voit, R. Schneider, D. Goswami, A. Annaswamy, and S. Chakraborty, "Optimizing hierarchical schedules for improved control performance," in *Proceedings of the IEEE Symposium on Industrial Embedded Systems (SIES)*, 2010.

[18] D. Seto, J. Lehoczky, L. Sha, and K. Shin, "On task schedulability in real-time control systems," in *17th IEEE Real-Time Systems Symposium*, 1996, pp. 13–21.

[19] J. Stankovic, C. Lu, S. Son, and G. Tao, "The case for feedback control real-time scheduling," in *Proceedings of the 11th Euromicro Conference on Real-Time Systems*, 1999, pp. 11–20.

[20] L. Abeni, L. Palopoli, and G. Buttazzo, "On adaptive control techniques in real-time resource allocation," in *12th Euromicro Conference on Real-Time Systems*. Los Alamitos, CA, USA: IEEE Computer Society, 2000, pp. 129–136.

[21] J. Eker, P. Hagander, and K.-E. Årzén, "A feedback scheduler for real-time controller tasks," *Control Engineering Practice*, vol. 8, pp. 1369–1378, 2000.

[22] A. Cervin, J. Eker, B. Bernhardsson, and K.-E. Årzén, "Feedback-feedforward scheduling of control tasks," *Real-Time Systems*, vol. 23, pp. 25–53, 2002.

[23] E. Bini and A. Cervin, "Delay-aware period assignment in control systems," in *Proc. 29th IEEE Real-Time Systems Symposium*, 2008.

[24] S. Samii, P. Eles, Z. Peng, and A. Cervin, "Quality-driven synthesis of embedded multi-mode control systems," in *Proc. 46th Design Automation Conference (DAC)*, 2009, pp. 864–869.

[25] S. Samii, A. Cervin, P. Eles, and Z. Peng, "Integrated scheduling and synthesis of control applications on distributed embedded systems," in *Proc. Design, Automation & Test in Europe (DATE'09)*, 2009.

[26] E. P. Godoy, A. J. V. Porto, and R. Y. Inamasu, "Sampling time adaptive control methodology for CAN-based Networked Control Systems," in *2010 9th IEEE/IAS International Conference on Industry Applications (INDUSCON)*, 2010.

[27] S. Piltz, M. Bjorkbom, L. Eriksson, and H. Koivo, "Step adaptive controller for networked mimo control systems," in *International Conference on Networking, Sensing and Control (ICNSC)*, 2010.

[28] L. Chunmao and X. Jian, "Adaptive delay estimation and control of networked control systems," in *International Symposium on Communications and Information Technologies (ISCIT)*, 2006.

[29] P. Marti, J. Yepez, M. Velasco, R. Villa, and J. Fuertes, "Managing quality-of-control in network-based control systems by controller and message scheduling co-design," *IEEE Transactions on Industrial Electronics*, vol. 51, no. 6, pp. 1159 – 1167, dec 2004.

[30] K. J. Åström and B. Wittenmark, *Computer-Controlled Systems: Theory and Design*. Prentice Hall, 1990.

[31] M. S. Branicky, S. M. Phillips, and W. Zhang, "Stability of networked control systems: Explicit analysis of delay," in *Proceedings of the American Control Conference*, 2000.

[32] G. C. Goodwin and K. S. Sin, *Adaptive Filtering Prediction and Control*, ser. Information and Systems Sciences Series. Prentice Hall, 1984.

[33] A. Annaswamy, D. Goswami, and S. Chakraborty. *Tutorial on the Arbitrated Networked Control Systems Approach to Cyber-Physical Systems*. CPSWeek 2011.