European Journal of Control (2005)11:242–254

European Journal of Control

Discussion on: "Development and Experimental Verification of a Mobile Client-Centric Networked Controlled System"

Paul Otanez

Sibley School of Mechanical & Aerospace Engineering, Cornell University, USA

Tzes et al. present the experimental verification of a mobile client-centric networked control system (NCS) implemented over a General Purpose Radio Service (GPRS) communication channel. The system is set up by defining a client, the controller, which sends information over the network to the server, the plant and actuator. The characteristics of GPRS make the system different from other NCSs studied, which are often implemented over Ethernet or DeviceNet. For security reasons, mobile-phone service providers preclude a server from independently sending information back to the client. In this network, all actions must be initiated by the controller creating a client-centric NCS. The mobile NCS time delays are composed of both encoding/decoding delays and as well the transmission times of data through the mobile network. The transmission delays are highly uncertain and depend on a number of factors that include the number of users, loss of packets and the existence of higher priority transmissions. As explained in the paper, GSPR transmissions have a lower priority than GSM-based voice calls. A useful contribution of the paper is the characterization of transmission delays in both UDP and FTP connections through GSPR. The measurements show that the average bit rates are significantly lower than the theoretical limits of GSPR.

The stability of the closed-loop is guaranteed by implementing results from time-delayed systems and by solving a set of linear matrix inequalities. The authors state that the maximum time delay, $\tau_{\max}$, calculated is conservative. Walsh, et al. [14] present an analytical proof for the global stability of the closed loop of the NCS based on perturbation theory; however, their results also yield a conservative $\tau_{\max}$. In most cases, an NCS controller is designed so that it broadcasts updates to the plant every T seconds, where $T < \tau_{\max}$ to insure stability. However, the most effective way to improve the performance of an NCS is to minimize time delays by reducing network traffic. Finding a tighter bound on $\tau_{\max}$ is an important research question since it would relax the minimum controller transmission frequency and reduce network traffic. An investigation of how to systematically reduce the node transmission frequency in NCSs using variable deadbands is presented in Ref. [2].

Researchers have often modelled NCSs as a variant of time-delayed systems and designed controllers that withstand the worst possible time delay. Although this approach simplifies analysis, it does not consider the effects of the controller on the network. In order to improve NCS performance, there is a need to formulate controllers that utilize the packet structure of network communication as well as monitor traffic to adjust their operating mode. Depending on the protocol, each packet has a minimum size requirement that in control applications is often not efficiently used. Georgiev and Tilbury [1] exploit the relative large size of Ethernet packets by sending several control commands in a single packet and thus reducing network traffic. Tang and de Silva [13] propose using a modified model predictive control strategy to encode a sequence of commands in a packet to reduce communication. As the characteristics of the network traffic change, the length of the prediction time horizon is adjusted to minimize the effects of time

E-mail: potanez@cornell.edu

delays, vacant sampling, packet-reordering and data loss. An extension of this approach would formulate a controller that monitors network traffic, but also considers time delays to specific nodes. Time delays as well as information received from the plant could dictate a switch by the controller from a normal mode of operation to a more conservative operating mode. This NCS controller would react to network traffic by changing its operating mode, the size of packets and by reducing the number of packets broadcast to nodes that exhibit above average response times. However, in some protocols accurately estimating transmission time delays is not trivial as it requires clock synchronization between all nodes.

References

1. Georgiev D, Tilbury D. Packet-based control. In: Proceedings of the American control conference, Boston, MA. June, 2004
2. Otanez P, Moyne J, Tilbury D. Using deadbands to reduce communication in networked control systems. In: Proceedings of the American control conference, Anchorage, AK. June, 2002
3. Tang PL, de Silva CW. Ethernet-based predictive control of an industrial hydraulic machine. In: Proceedings of the conference on decision and control, Maui, HI. December, 2003
4. Walsh GC, Hong Y, Bushnell LG. Stability analysis of networked control systems. IEEE Trans Control Syst Technol 2002; 10(3): 436–446

Discussion on: "Development and Experimental Verification of a Mobile Client-Centric Networked Controlled System"

Nathan A. Wedge*, Ahmad Al-Hammouri**, Michael S. Branicky*** and Vincenzo Liberatore****

Department of Electrical Engineering and Computer Science, Case Western Reserve University, Cleveland, OH 44106, USA

1. Summary of Contributions

The paper under discussion [12] presents the analytical results underlying the authors' design of a networked control system (NCS) operating over a cellular network. Their formulation fits the common framework [7,14], in which there are transmission delays present in the sensor/controller and controller/actuator paths. One key difference between this work and existing literature relates to the use of a client-centric system, which is mandated by security issues. However, this seems to make little to no functional difference in the implementation or resulting analysis. The chief theoretical results of the paper are based upon linear matrix inequality (LMI) theory. In particular, the authors have provided an interesting, LMI-derived analytical result for ascertaining the stability of an NCS with delays. Some existing work [4] has pursued similar derivations for systems with time-varying delays without applying the results directly to NCSs; a complementary approach [1] provided LMIs that could be used to guarantee the stability of an undelayed NCS under packet losses. Another key point in the work here is that the authors assume that the application of control signals and delivery of state information happens on the boundaries of sampling intervals. This approach is advantageous to their analytical approach, which augments the system with delayed state measurements up to a certain bound. Moreover, it provides results for situations in which the delay is greater than the sampling period, a scenario that is often neglected in NCS research, but fairly common in wide-area networks. In summary, their different viewpoints plus the development of an end-to-end system under quite restrictive bandwidth and delay constraints is of obvious value to the progress toward a larger vision of networked control.

2. The Bigger Picture

Networked control is central to the larger vision of *tele-epistemology*, which posits that knowledge can be

*E-mail: nathan.wedge@case.edu
**E-mail: ahmad. al-harmmouri@case.edu
***E-mail: mb@case.edu
****E-mail: vincenzo. Liberatore@case.edu

acquired from a distance through network-mediated interaction [3]. The fundamental tele-epistemological tenets require that physical environments be controlled irrespectively of physical or communication distance. The tele-epistemological direction is substantially advanced by this paper [12] in that the (wireless) telephony network is arguably the most pervasive, widespread, and reliable network. Whereas most previous work in networked control had focused on local area networks (LANs), such as those found in a single factory plant or in a peripheral interconnection bus, the approach in this paper can scale to practically arbitrary geographical distances. As a consequence, it is a significant extension of the networked control field.

Geographical scalability does require bridging long distances, and in practice such a capability is necessary but not sufficient. Geographic closeness gives rise to one distance function, but other critical metrics are induced, for example, by the interconnection topology – which results in geographical and network distances that are often unrelated. For example, a telephone GPRS is but one single network in the Internet. This type of interconnectivity has effectively created a *network of networks*, with the additional benefits afforded by flexibility, evolvability, and interoperability [2]. This paper is an excellent first step toward interconnected, geographically scalable networked control, and the natural future steps should complement geographic distances with the inherently complex structure of the interconnection topology.

The network of networks architecture has a profound impact on networked control. First, an individual network can usually be configured to support Quality-of-Service (QoS) assurances. However, QoS cannot be guaranteed in general in the Internet (see, for example, the seminal, Turing-award winning paper by Cerf and Kahn [2]). Specifically, a telephony GPRS provides hard bounds on jitter, loss rates, and bandwidth, but no such guarantees exist in the best-effort model that is prevalent in the Internet. In turn, poor QoS impacts the real-time properties of the feedback loop. Furthermore, the lack of QoS also implies relatively poor clock synchronization [11]. A striking consequence is that, in an asynchronous distributed system, there is no deterministic and fault-tolerant algorithm for two end-points to reach agreement on even one bit [5].

As for networked control, this lack of synchronization does not permit certain schemes upon which much previous work relied (e.g. [6,7,13,14]). Furthermore, this previous work – primarily on LANs – indicates that *event-driven control* which takes immediate action upon receipt of the sensor data (by calculation and dispatch of a control signal) and similarly upon receipt of the control signal (by beginning actuation) would provide superior performance given an optimal control strategy [6]. Thus, much previous work has focused on the use of event-driven control. In contrast, the paper under discussion assumes that the application of control signals and delivery of state information happens on the boundaries of sampling intervals. In extensive *wide-area* network experiments, we have recently found that this more regular application of control signals can lead to better overall performance.

In summary, networked control in the Internet age is a rich and mostly unexplored research area. It is the natural future direction implied by this paper.

3. Some Implementation Issues

In addition to the issues above, the complexity of distributed control applications in general has increased substantially. Control software itself is becoming more complex, as measured by the number of its function points, for example. Moreover, control problems bring hard real-time constraints. Software Engineering has made impressive strides for such complex, real-time distributed applications (e.g. [10]); Computer Networks has seen the creation of real-time protocols (e.g. [8]). Although some of those parallel contributions address primarily the software development process and communication standards, many impact directly the communication timing, which in turn affects control stability and performance.

As one specific example, real-time distributed applications, such as networked control, favor connectionless protocols, such as UDP, owing to the low overhead and the potentially detrimental effect of the use of stale information (and therefore lack of usefulness for a mechanism that achieves reliability by retransmission). In this paper, though UDP is the mechanism for control signal delivery, the transmission of sensor data is accomplished over FTP (and hence, TCP). Justification for this selection is conspicuously absent, especially considering the packet loss scenarios that follow in the paper. Aside from choosing TCP as a transmission protocol for transferring sensor data, choosing FTP in particular – among other applications, for example, plain TCP sockets – exaggerates the problem even more. This is true because even though FTP maintains a single persistent connection to communicate commands, it starts a new TCP connection for every data transfer

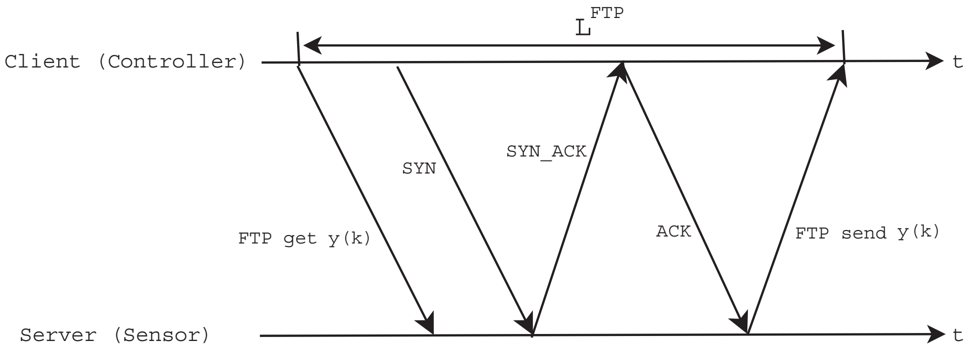

Fig. 1. FTP data transfer. FTP-get and FTP-send packets of the original paper decompose into FTP-get, SYN, SYN-ACK, ACK, and FTP-send packets.

request (i.e. every FTP get command) and closes the connection after the transmission completes [9]. Therefore, at least one round-trip time is wasted to set up such a connection before transferring data starts (see Fig. 1).

Consequently, the choice of FTP both exposes signals to more delays/losses and imposes restrictions on deploying this implementation in applications where sensors usually have very limited resource constraints. This questionable protocol choice probably contributes to the poor performance of the cellular network in the experimental study. Though the authors make the point that voice calls are given higher priority than data calls (i.e. their control system), their measured delays seem extreme, and it is doubtful that any modern cellular client would tolerate data delays that average 18 s. Furthermore, their quoted data rates are small compared with the attainable rates mentioned. Barring the possibility of extreme overhead (i.e. if packets contain only a few bytes of data, but there is significant overhead due to headers and connection establishment), this weakens the particular choice of cellular protocol from the bandwidth standpoint.

A related issue is the application of PCs in the control loop. One advantage of NCSs in general is the potential to save on resources by not allocating control components at the location of the plant. However, the scheme presented used a PC capable of FTP service in close proximity to the plant, both to deliver sensor data and to implement the control commands. Thus, at least in the case of the specific example system, the amount of resources devoted to FTP service and network data transmission by the PC (or, later, presumably by a network-enabled chipset) begs the question of whether they should be leveraged to accomplish the local control task directly, with the client providing remote supervision (e.g. sending setpoints).

4. Conclusions

Overall, the authors have presented the particular concerns of control design over a cellular network medium and presented an implementation of such an NCS. We pointed out a few issues with that implementation. However, their development of methodology for implementing control systems under quite restrictive bandwidth and delay constraints is of obvious value to the progress toward the larger tele-epistemological vision of globally distributed control over networks of networks.

References

1. Azimi-Sadjadi B. Stability of networked control systems in the presence of packet losses. In: *Proceedings of the conference on decision and control*, Maui, HI. December, 2003, pp 676–681
2. Cerf V, Kahn R. A protocol for packet network interconnection. *IEEE Trans Commun Technol* 1974; 22(5): 627–641
3. Goldberg K, ed. The Robot in the Garden: Telerobotics and Telepistemology in the Age of the Internet. MIT Press, Cambridge, MA, 2001
4. Jeung ET, Oh DC, Kim JH, Park HB. Robust controller design for uncertain systems with time delays: LMI approach. Automatica 1998; 32(9): 1229–1231
5. Lynch NA. Distributed Algorithms. Morgan Kaufmann, San Francisco, CA, 1996, Ch. 21
6. Nilsson J. Real-time control systems with delays. PhD thesis, Automatic Control, Lund Institute of Technology, Lund, Sweden, 1998
7. Nilsson J, Bernhardsson B, Wittenmark B. Stochastic analysis and control of real-time systems with random time delays. Automatica 1998; 34(1): 57–64
8. Peterson LL, Davie BS. Computer Networks. Morgan Kaufmann, San Francisco, CA, 2000, Ch. 9
9. Postel J, Reynolds JK. RFC 959: file transfer protocol, October, 1985
10. Shokri E, Shreu P. Real-time distributed object computing – an emerging field. Computer 2000; 33(6): 36–44

11. Tanenbaum AS, Van Steen M. Distributed Systems. Principles and Paradigms, Ch. 4. Prentice-Hall, Upper Saddle River, NJ, 2002
12. Tzes A, Nikolakopoulos G, Koutroulis I. Development and experimental verification of a mobile client-centric networked control system. Eur J Control 2005;
13. Zhang W. Stability Analysis of Networked Control Systems. PhD thesis, Electrical Engineering and Computer Science, Case Western Reserve University, Cleveland, OH, 2001
14. Zhang W, Branicky MS, Phillips SM. Stability of networked control systems. IEEE Control Syst Mag 2001; 21(1): 205–214

Discussion on: "Development and Experimental Verification of a Mobile Client-Centric Networked Controlled System

J. Daafouz*, P. Riedinger and C. Iung

Institut National Polytechnique de Lorraine, CRAN–CNRS UMR 7039, ENSEM, 2 av de la Forêt de Haye, 54516 Vandoeuvre-Les-Nancy cedex, France

The paper by A. Tzes, G. Nikolakopoulos and I. Koutroulis is an interesting and novel contribution to an interesting application. A detailed and well-written description of mobile networked communication issues is provided. This leads to a set of six situations covering possible problems that can be encountered in the data exchange procedure. Among these situations, the authors investigate the effects on the system's stability for three cases: normal data exchange (case 1), mutual loss of packets at the client and server side (case 5) and a mutual delay related to the data transmission (case 6). A comment giving the main reasons in not considering the three remaining cases [packet-reordering, unidirectional (UDP, or FTP) loss of packets] will be welcome.

Using recent developments in switched control design [2] and delay systems analysis [4], simulation and experimental studies have been carried out. The prototype is a single input–single output (SISO) continuous time system and the analysis is performed for constant delays and time-varying delays.

(1) *The case of constant delays*: Two methods are proposed for this case. The first one consists in analyzing the SISO continuous time model and computing a bound of the maximum allowable time delay that preserves stability using a linear matrix inequality (LMI) optimization problem from Ref. [11]. The second method consists in analyzing the eigenvalues of the corresponding SISO discrete time model for different values of the sampling period and different values of the communication delay. Both methods use an output feedback controller and the related constant gain is fixed *a priori*. The obtained bound of the maximum allowable time delay using the LMI optimization problem appears to be more conservative than the result obtained by the eigenvalues analysis.

(2) *The case of time-varying delays*: For this case, the procedure is based on switched systems stability analysis and the main steps are summarized hereafter. As mentioned earlier, the prototype is a SISO continuous time system with a transfer function:

$$G(s) = \frac{0.1^3}{(s+0.1)^3} \tag{1}$$

Assuming a sampling period T_s, a discrete time model (with a zoh) is derived

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k \\ y &= Cx_k \end{aligned} \tag{2}$$

A static output feedback controller

$$u(k) = K_{r_s} y(k - r_s(k)) \tag{3}$$

is considered. $r_s(k)$ denotes the time-varying overall delay at time k corresponding to the client-to-server transmission delay plus the reverse transmission delay. Moreover, $r_s \in [0, 1, \ldots, D]$ is

*E-mail: Jamal.Daafouz@ensem.inpl-nancy.fr a random sequence of integers and D its upper bound. A well-known result stating that any delayed linear discrete time system can be transformed into an equivalent linear discrete time system with no delay is used. For the application under study, this consists in considering the following augmented state:

$$\tilde{x} = [x_k^{\mathrm{T}}, x_{k-1}^{\mathrm{T}}, \ldots, x_{k-D}^{\mathrm{T}}]^{\mathrm{T}} \tag{4}$$

The augmented closed-loop system becomes

$$\tilde{x}_{k+1} = (\tilde{A} + \tilde{B}K_{r_s}\tilde{C}_{r_s})\tilde{x}_k \tag{5}$$

where $\tilde{A}$ and $\tilde{B}$ are constant matrices given by

$$\tilde{A} = \begin{bmatrix} A & \mathbf{0} & \ldots & \mathbf{0} \\ \mathbf{1} & \mathbf{0} & \ldots & \mathbf{0} \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{0} & \ldots & \mathbf{1} & \mathbf{0} \end{bmatrix}, \quad \tilde{B} = \begin{bmatrix} B \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix}$$

$\tilde{C}_{r_s}$ is a time-varying matrix which takes its values in a finite set of matrices

$$\mathcal{S}_C = \{\tilde{C}_0, \tilde{C}_1, \ldots, \tilde{C}_D\}$$

with

$$\tilde{C}_0 = C[\mathbf{1} \quad \mathbf{0} \quad \mathbf{0} \quad \cdots \quad \mathbf{0}]$$
$$\tilde{C}_1 = C[\mathbf{0} \quad \mathbf{1} \quad \mathbf{0} \quad \cdots \quad \mathbf{0}]$$
$$\vdots \; \vdots \; \vdots$$
$$\tilde{C}_D = C[\mathbf{0} \quad \mathbf{0} \quad \cdots \quad \mathbf{0} \quad \mathbf{1}]$$

The problem is now clearly reduced to the design of a switched static output feedback gains K_{r_s} such that the switched closed system (5) is asymptotically stable under arbitrary (or randomly) switching rule [3].

The aim is to find the largest sets $I = \{I_{\min}, I_{\min}+1, \ldots, I_{\max}\}$ where $0 \le I_{\min} < I_{\max} \le D$ such that the closed-loop switched system is asymptotically stable. Reading the paper, it is not clear whether the approach used for the selection of the controller gains consists in

(a) fixing *a priori* the gains values K_{r_s}, as done in the case of constant delays, and checking the stability of the closed-loop switched system using switched Lyapunov functions. This reduces to check the LMI condition:

$$\begin{pmatrix} P_i & A_i^{\mathrm{T}}P_j \\ P_jA_i & P_j \end{pmatrix} > \mathbf{0} \tag{6}$$

$\forall i = 0, \ldots, D, \forall j = 0, \ldots, D$ where the Lyapunov matrices P_i are the unknowns and $A_i = \tilde{A} + \tilde{B}K_i\tilde{C}_i$ are known matrices, the closed-loop matrices, or,

(b) computing the gain values K_{r_s} using the results proposed in Ref. [2]. The output feedback synthesis result proposed in Ref. [2] is also based on switched Lyapunov functions. The computation of the controller gains consists in solving the following LMIs

$$\begin{pmatrix} G_i + G_i^{\mathrm{T}} - S_i & (\tilde{A}G_i + \tilde{B}U_i\tilde{C}_i)^{\mathrm{T}} \\ \tilde{A}G_i + \tilde{B}U_i\tilde{C}_i & S_j \end{pmatrix} > \mathbf{0} \tag{7}$$

$\forall i = 0, \ldots, D, \forall j = 0, \ldots, D$ along with the following equality constraints

$$V_i\tilde{C}_i = \tilde{C}_iG_i, \quad \forall i = 0, \ldots, D \tag{8}$$

and where the unknowns are the symmetric matrices S_i related to the Lyapunov matrices $P_i = S_i^{-1}$ and the matrices G_i, U_i and V_i. The output feedback control gains are then given by

$$K_i = U_iV_i^{-1} \quad \forall i = 0, \ldots, D \tag{9}$$

The first possibility is quite complicated since one has to fix *a priori* $D+1$ gains arbitrarily and then check the stability condition. The second possibility avoids the problem of fixing *a priori* the gain values and is more appropriate for design problems. Most probably, the second possibility has been used as it is almost impossible to make a conclusion with the first one.

Our discussion concerns mainly two aspects.

Real-time measurement of the latency time $r_s(k)$: For practical reasons, corresponding to the implementation of the switched output feedback control, the switching rule must be known in real time which corresponds to the application investigated in the paper under discussion to the assumption that the latency time $r_s(k)$ can be measured at every time instant k. Otherwise, one cannot implement such a switched output feedback control. This assumption is not necessary to write Eq. (5) as it is mentioned in the paper but it is only motivated by practical considerations. In fact, as the control law is switched and the switching rule is directly related to the latency time, practical implementation of this control law imposes the real-time measurement of the latency time. As already done in Ref. [1], if the control law is designed with a constant gain, and not switched, the LMI stability condition (5) can be used without this assumption.

In the paper under discussion, a switched control law has been designed and experimental studies have been made over a private networks mobile service provider. This shows that the previous assumption can be practically satisfied. One may be interested by further details on possible difficulties (or not) in satisfying this assumption. A question of high interest is the following: in this application, can the latency time be always measured or is a special configuration needed for that?

Solving the LMIs: From the study performed in the paper under discussion, it follows that in the case of time-varyings delays, there is no switched controller that can tolerate delays up to 50 s. The procedure to obtain this conclusion can be discussed. In fact, the obtained results are relatively paradoxal. With $T_s = 5$ s, the LMI-related problem leads to a switched controller that can tolerate a maximum latency time of 30 s. With $T_s = 10$ s, a maximum latency time of 40 s can be guaranteed. What about $T_s = 20$ s? Perhaps we can reach the required 50 s!

We think that the problem is not so easy. If the maximum required latency time DT_s is fixed, the main reason in having difficulties to reach this time when T_s is small is that the LMI problem dimension (directly related to D) increases when T_s decreases. Recall that the number of vertices of the switched system is $D+1$. This implies that one has to solve $(D+1)^2$ LMIs with $D+1$ equality constraints. With $DT_s = 50$ s and $T_s = 5$ s we have $D = 10$ which means to solve $(D+1)^2 = 121$ LMIs with 11 equality constraints. Moreover, the dimensions of the unknowns are as follows:

- $n_u \times n_u$ for the $(D+1)$ Lyapunov symmetric matrices $S_i = S_i^{\mathrm{T}}$ with $n_u = n * (D+1)$ the dimension of augmented state $\tilde{x}$ and $n = 3$ the original system dimension,
- $n_u \times n_u$ for the $(D+1)$ matrices G_i,
- $1 \times n_u$ for the $(D+1)$ matrices U_i,
- $n_u \times 1$ for the $(D+1)$ matrices V_i.

Therefore, for $T_s = 5$ s and hence $D = 10$, one has to solve 121 LMIs with 11 equality constraints and

$$\begin{aligned}(D+1)&\left(\frac{n_u*(n_u+1)}{2}+n_u^2+n_u+n_u\right)\\ &= 11\left(\frac{33(33+1)}{2}+33^2+33+33\right)\\ &= 18876\end{aligned}$$

variables. The dimension of this LMI problem is very high. One may not get a solution not because it does not exist, but probably owing to numerical problems. Moreover, even if one may find a LMI solver which is able to handle such a huge problem, the computation time and the memory size will be certainly high. Notice that if we consider $T_s = 10$ s, one gets 49 LMIs with 4998 variables and for $T_s = 25$ s, only 9 LMIs with 432 variables have to be considered.

To sum up, we think that the way the original time-delay problem is transformed into a switched stability problem is the origin of this rapidly growing dimension. This nice application clearly shows the main drawback in transforming a discrete time-delay system into a discrete time system with no delay but with an augmented state and gives strong arguments for considering the problem of delayed discrete time systems as a difficult problem which has to be treated explicitly as it is. The problem is more critical when the delay is larger than the sampling period, which is the case in this application. A transformation of this problem into a non-delayed augmented problem is a first approach but certainly not the best solution.

Finally, concerning future research following this paper, the authors proposition to improve the results by taking into account the slow variation of the latency time and the overlapping property is interesting and will certainly help in reducing the number of LMIs and variables. This would probably help in covering the whole region. We believe that as an alternative to this important work, a reformulation of the problem under study without a state augmentation may lead to better results. In addition to stability considerations, taking into account performance specifications is another point of high importance that deserves investigation.

References

1. Daafouz J, Bernussou J. Poly-quadratic stability and H_∞ performance for discrete systems with time varying uncertainties. In: Proceedings of the IEEE conference on decision and control, Orlando, FL, USA. 4–7 December, 2001
2. Daafouz J, Riedinger P, Iung C. Stability analysis and control synthesis for switched systems: a switched Lyapunov function approach. IEEE Trans Autom Control 2002; 47(11):
3. Liberzon D. Switching in Systems and Control, Foundations & Applications. Birkhäuser, Boston, 2003
4. Mahmoud MS. Robust Control and Filtering for Time-Delay Systems, Control Engineering. Marcel Dekker, 2000

Discussion on: "Development and Experimental Verification of a Mobile Client-Centric Networked Controlled System"

Mohammad Tabbara* and Dragan Nešić**

Department of Electrical & Electronic Engineering, The University of Melbourne, Parkville, Victoria 3010, Australia

What has been presented in Ref. [1] is an implementation of a wireless single input–single output (SISO) networked control system (NCS) over a so-called 2.5G mobile network infrastructure. The mobile networks considered are GSM networks, essentially, those networks used for mobile voice communication with extensions to allow packet-based data transfer (GPRS). One of the primary implementation issues considered is the design of an application layer protocol to resolve limitations and restrictions that mobile carriers impose on GPRS-based communication. Specifically, mobile carriers typically block server-initiated data transfers, hence, requiring the use of client-centric (client-pull) architectures and protocols. See Ref. [2] for an excellent overview of issues in protocol design, GPRS and GSM.

Tzes et al. [1] present an interesting foray into the area of wireless NCSs and their implementation. Compared with the wireline NCSs, wireless NCS theory is relatively underdeveloped and there is much work need to be done. The implementations described in Ref. [1] show that wireless NCS can work in practice and underline a need for further research into NCS. As with any class of control problems, we stress the importance of modelling and the choice and development of theoretical frameworks that are appropriate for the applications being considered. The development of rich models for NCS with complementary stability and performance results remains an open and important area of research.

The differentiating feature of NCSs is the notion that data are exchanged over *shared* network links instead of dedicated point-to-point links found in traditional control systems. Quantization, data-rate constraints, transmission delays, frame collision and channel errors[1] can all affect NCS to varying degrees depending on the underlying physical channel, the protocols and transmission mechanisms used, the number of nodes (users) on the network and environmental conditions. This is particularly true for the GSM mobile networks used for the NCS described in Ref. [1] though the authors focus primarily on the effects that can be modelled as network-induced delays. Certain applications may allow some of these effects to be ignored in the analysis but Ref. [1] does not address any particular application and only incorporates the effects of sampling and delay, treated in a deterministic setting. Whilst there is much freedom in the choice of (deterministic or stochastic) analysis frameworks, network design decisions and physical realities often dictate whether a phenomenon must be treated stochastically or whether it can admit a deterministic formulation.

The GPRS (over GSM)-based NCS used in Ref. [1] allows for several nodes (mobile users and NCS nodes) to compete for certain fixed time-slots removed from the pool of time-slots used for voice communication. This contention for shared links leads naturally to the idea of arbitrating amongst nodes with a scheduling protocol (cf. medium access protocol). Arbitration occurs for both NCS nodes as well as "traditional" voice and data traffic on the same network so the notion of scheduling applies equally to SISO systems on mixed-traffic networks as well as multiple-node NCS.

Collision-free protocols ensure that only one node can transmit on the link at any one time[2] while collision protocols make no such guarantee of exclusivity – simultaneous transmission by more than one node results in a frame loss (a collision). Collision protocols rely on the bursty nature of *data* networks and various mechanisms to avoid collisions, for example, by having nodes "sense" if a transmission by another node is in progress and delaying transmission until the channel is free.

The assumption of bursty transfer patterns is unlikely to hold in an NCS. Indeed, the NCS model

*E-mail: m.tabbara@ee.unimelb.edu.au
**E-mail: d.nesic@ee.unimelb.edu.au

[1]We make the distinction between errors due to collisions during multiple-access and losses due to decoding errors as a result of channel noise, referring to the latter as *channel errors* and the former as *frame collisions*.

[2]The node transmits can be determined by an arbiter or through a distributed resolution process.

described [1] specifies the sampling of sensor values and regular computation of the control law with a corresponding need for regular transmission. ALOHA, Ethernet (CDMA/CD) and the contention period of GPRS reduce the likelihood of collisions occurring in lockstep (as would be the case when always transmitting sensor and control values) by having nodes wait ("backoff") for a random period of time before retransmitting after a collision. In that sense, collision protocols force stochasticity into models of NCS that use them.

Multiple nodes in GPRS communication compete for access to a network *queue* during a contention period, with simultaneous access resulting in a collision with a corresponding random backoff. Beyond this contention period "successful" nodes are placed in a first-come first-served queue to be eventually granted exclusive access. Following Daigle [3], the delay associated with contention and queuing can be modelled in a stochastic setting as, essentially, a slotted-time MMPP/G/1 queuing system [Markov modulated Poisson process arrivals, independently identically distributed service-times with a general probability distribution function $f_{\tau}(x)$, see Ref. [4], (Chapter 5), for instance]. Despite the use of a collision protocol with random backoff, Tzes et al. [1] treat delays deterministically, assuming that the per-node transmission delays can be drawn from a fixed, finite set of delays $\{1, 2, \ldots, d\}$. There is no reason to expect this (indeed, to expect that delays should be an integral multiple of the sampling period of the discrete-time system). While experimental evidence presented in Ref. [1] showed that the worst-case delay was in the order of 45 s, the tests were carried out on a dedicated, private GSM network with few users and low-traffic levels. The most probable scenario in a commercial GSM network is one where relatively few transmission slots are dedicated to GPRS, *vis-a-vis* voice slots, with varying and often high levels of contention in most network cells. Strictly speaking, multiple-access communication networks employing collision protocols demand a stochastic framework for their discussion.

An accurate model for GRPS communication and the NCS implementation discussed in Ref. [1] with non-conservative stability results may require the development of models of collision protocols that are the stochastic analogues of the models presented in Refs [5–9]. Using a collision-free protocol, such as Token Ring, can lead to purely deterministic models of NCS as described in Refs [5–7]. The models described in these papers treat the protocol as a dynamical system in its own right, leading to stability results intimately tied with the protocol behaviour. This is in contrast to the typical treatment of collision protocols as introducing a time-varying stochastic delay of transmissions with a certain probability distribution function. Data networks require frames to be eventually delivered to their destination and so it makes sense to only talk about delays and guaranteeing that such delays are bounded. Goodman et al. [10] provide a proof that this is indeed the case for Ethernet, under a set of conditions reasonable for access patterns in data networks. The eventual arrival of stale or delayed samples and control values is not as important as having (current) values transmitted frequently enough to maintain stability or a performance objective in NCS and models of collision protocols for NCS need to reflect this.

When taking a delay-centric approach to analysis and delays *are strictly upper bounded* then the use of LMIs and Lyapunov theory for stability analysis is well known. Niculsecu et al. [11] examine the case of constant delays while Kharitonov and Niculsecu [12] deal with time-varying delays in linear systems. The results in Ref. [1] do not address this scenario of continuously time-varying delays that might occur with varying buffer queue lengths, contention levels and varying distances between base-station and user.

Despite the generality of Ref. [12], in particular, delay-centric results only apply when delays are known to be *a priori* bounded. As alluded to earlier, this is the case in NCS when using a collision-free protocol[3] and the "delays" involved are due to nodes waiting for other nodes to transmit. Analysing system robustness with respect to these delays ignores the fact that there is generally something better to do when given network access than to transmit stale data – send fresh data. Within the scope of switched and hybrid systems literature, Hassibi et al. [13] examine the so-called asynchronous dynamical systems (ADSs) with an example of an NCS that does, indeed, send fresh data at each transmission instant with the potential for data loss (owing to collisions or otherwise). LMI-based techniques are used to construct controllers and Lyapunov functions for ADS but, more importantly, Ref. [13] capture the intuitive notion that the larger the fraction of time spent switched in a particular system Σ_i, the more the ADS behaviour is dominated by the nominal behaviour of Σ_i. For NCS, this translates into requiring a

[3]This may also be the case when using a collision protocol that guarantees "eventual" network access for some node after a bounded contention period. With assumptions on the number of nodes competing for access and the backoff process, this can be the case for GPRS and other CDMA/CD protocols.

constraint on collision and loss rates in order to infer stability. Requiring such a constraint is more intuitive than demanding stability with respect to arbitrary switching-signals of finite length and also opens up the possibility of subjecting the system to very long delays[4] and being able to conclude stabling if such delays are only "occasional". In summary, when the delays are (deterministically) bounded and switched systems results are applicable, always transmitting current data when a node is granted network access leads to NCS models where stability results can be less conservative (via the discussions in Refs [5–13]).

In conclusion, effective theoretical analysis requires accurate models that are amenable to analysis. In this case, the effects of contention in a multi-user environments with collision protocols motivate the need to revisit how these protocols are modelled with an emphasis on developing results tuned to control over networks as opposed to computer or data networks. Orthogonal to the effects of arbitration (scheduling) for multi-user access, there are also the issues of packet dropout and delay owing to channel imperfections that are especially true for wireless networks. Multi-path fading, inter-symbol interference and various additional sources of noise all add an additional layer of complexity that may need to be modelled to shape a coherent and complete picture of NCS. Owing to the complexity of the mechanisms behind these phenomena, they are always treated stochastically (see Ref. [14], for instance), underlining the need for the development of stochastic frameworks for the analysis and design of NCS.

Acknowledgement

This work was supported by the Australian Research Council under the Australian Professorial Fellow and Discovery Grants schemes.

[4]While the particular NCS example provided in Ref. [13] does treat the case of "true" delay (propagation delay or waiting in a queue), the framework discussed therein can treat delay in a completely analogous manner.

References

1. Tzes A, Nikolakopoulos G, Koutroulis I. Development and experimental verification of a mobile client-centric networked controlled system. In: Proceedings of the European Control Conference, no. 133, Cambridge, UK, 2003
2. Tanenbaum AS. Computer Networks. 4th edn. Prentice Hall, 2003
3. Daigle JN. Analysis of packet networks having contention-based reservation with application to GPRS. IEEE/ACM Trans Networking 2003; 11(4):
4. Kleinrock L. Queueing Systems, Volume 1: Theory. 1st edn. Wiley-Interscience, 1975
5. Tabbara M, Nešić D, Teel A. Input–output stability of wireless networked control systems. In: Proceedings of the 44th IEEE conference on decision and control, 2005, under review
6. Teel A, Nešić D. Input-to-state stability of networked control system. Automatica 2004; 40(12): 2121–2128
7. Teel A, Nešić D. Input–output stability properties of networked control systems. IEEE Trans Autom Control 2004; 49(10): 1650–1667
8. Walsh G, Ye H. Scheduling of networked control systems. IEEE Control Syst Mag 2001; 21(1): 57–65
9. Ye H, Walsh G, Bushnell L. Real-time mixed-traffic wireless networks. IEEE Trans Ind Electron 2001; 48(5): 883–890
10. Goodman J, Greenberg AG, Madras N, March P. On the stability analysis of the ethernet. In: Proceedings of the 17th ACM symposium on theory of computing, Providence, RI. 1985, pp 379–387
11. Niculsecu S-I, Nito AT, Dion J-M, Dugard L. Delay-dependent stability of linear systems with delayed state: an LMI approach. In: Proceedings of the conference on decision and control, No. 2, New Orleans, LA. December, 1995, pp 1495–1496
12. Kharitonov VL, Niculsecu S-I. On the stability of linear systems with uncertain delay. IEEE Trans Autom Control 2003; 48(1): 127–132
13. Hassibi A, Boyd SP, How JP. Control of asynchronous dynamical systems with rate constraints on events. In: Proceedings of the 38th IEEE conference on decision and control, Phoenix, AZ. 1999, pp 1345–1351
14. Proakis JG. Digital Communications. 4th edn. McGraw-Hill, 2001

Discussion on: "Development and Experimental Verification of a Mobile Client-Centric Networked Controlled System"

Paul A. Kawka and Andrew G. Alleyne*

Department of Mechanical and Industrial Engineering, University of Illinois at Urbana-Champaign, IL, USA

1. Background

The paper by A. Tzes, G. Nikolakopoulos and I. Koutroulis examines mobile client-centric network control motivated by the potential use of General Purpose Radio Service (GPRS) over mobile phone infrastructures for control applications. The authors in this work begin to address two key requirements for control utilizing GPRS, namely the need for client-centric communication and tolerance of variable and large periods of delay which is the primary contribution. Linear matrix inequality (LMI) techniques for closed-loop system stability analysis with delay are presented along with simulation, experimental tracking and network performance results showing their applicability. Additionally, the experiments demonstrate one of the first tests of a GPRS network for a simulated control application.

2. Discussion

Through the introduction and architecture description, the authors describe what is meant by client-centric remote control and why it is required. Owing primarily to concerns for security over mobile phone networks, server access privileges to remote clients are limited to prevent unauthorized access and intrusions from untrustworthy sites. As a result, control strategies intended to utilize such public links must have all commands initiated from the mobile client. In the authors' communication framework, the transmissions are executed by sending a UDP packet for control and issuing a FTP-get command to receive feedback. The inherent latencies associated with each of these commands drive the primary investigation of this work, treatment of delay in the closed-loop system.

These control requirements differ from those studied for other network architectures. In Ref. [1], a server centric communication method is developed for fast (250 Hz) closed-loop control over standard 802.11b wireless Ethernet hardware. In that work, a clock-driven server manages the sensing and actuation and communicates with an event-driven controller. Transmission latencies are assumed small due to the speed of the hardware, resulting in latencies that are almost always less than a sampling period. Delays that exceed the sample period or lost transmissions in either direction are treated identically as a loss in control signal. Other authors have focused on similar network architectures with enhancements to improve performance when packets are lost. For example, one strategy in Ref. [2] includes sending multiple control actions per packet based on a plant observer to be used in case a delay threshold is exceed or information is lost. These methods differ from this work where lost or delayed packets are all treated in a delay analysis framework.

In Figure 3, the authors clearly outline six possible scenarios for the closed-loop data exchange between the client and server for different cases of packet loss, information reordering or delay beyond the sampling period. The subsequent analysis presented in the paper treats three of the six possible conditions, all of which can be examined within a common delay framework.

For the cases of normal transmission, loss of transmission in both control and feedback directions, and round-trip delay exceeding the sampling period, the authors present an LMI framework for analyzing stability. The analysis treats output feedback control of a known discrete linear system with variable, but measurable delay. It also assumes that the control and feedback delays can be lumped together and applied equivalently as a single time varying delay ($r_s(k)$) on the output feedback control with a known upper bound (D). The framework treats delays in the discrete dynamics by augmenting the state vector to incorporate all of the possible delayed states. By placing a known upper bound on the total delay (D), the system remains finite dimensional. This upper bound D on the delay also indirectly imposes a limit on consecutive

*E-mail: alleyne@uiuc.edu

transmission losses since lost packets effectively increase the delay.

Although the proposed augmented problem formulation allows for time-varying control based on the combined delay through the control law

$$\tilde{u} = K_{r_s(k)}\tilde{C}_{r_s(k)}\tilde{x}(k),$$

by lumping the control and feedback latencies together, the actual delay from controller to actuator will not be known when the control is generated. Therefore, it will not be possible in practice to change this gain dynamically based on the delay. A constant value gain can be used as implemented in the simulation examples presented, or an alternative technique to choose a gain based on statistical measurement of past delays could be employed [3].

In the augmented framework, the delay system becomes a discretely switched system where the switching signal is effectively the total delay. An LMI technique recently developed in Ref. [4] is applied to find a common switched Lyapunov function that guarantees asymptotic system stability for varying delay values from 0 to D. This formulation and analysis technique provides a useful tool for checking stability of a known discrete system under variable delay.

To obtain a conservative estimate for the maximum delay D that preserves stability in the discrete analysis, the authors examine a continuous time LMI formulation from Ref. [5], which allows the calculation of a level of constant delay that can be tolerated for continuous dynamics. This result provides a method to obtain a first estimate for D for use in the discrete LMI analysis, or for determining whether a plant would be suitable for control over a particular network architecture with known transmission delay properties.

The simulation results demonstrate how the proposed techniques could be used to evaluate the effect of delay and controller gain on stability of a sample SISO plant. Figure 5 applies the continuous time LMI technique to show how the maximum tolerable constant delay varies as a function of static gain K. These values are intended to serve as possible conservative starting levels for D in the discrete analysis.

The next portion of the simulation section examines an eigenvalue-based stability analysis on a ZOH discretization of the continuous plant. System stability is checked as a function of fixed discrete time delay. This analysis provides an alternative and less conservative method for determining an upper level for the maximum tolerable delay. Figures 6 and 7 show the effects of gain and sample period on the delay tolerance. Higher magnitude gains decrease the system's delay tolerance while longer sample periods moderately increased the maximum constant delay for which stability could be guaranteed.

The remaining simulation portion demonstrated the key contribution of the paper, application of the switched Lyapunov function LMI technique for examining stability for varying delay. Figure 8 demonstrates an interesting idea that different overlapping delay subspaces for stability can be identified with the LMI technique using different switched Lyapunov functions. The authors remark that if it is possible to show that the delay varies "sufficiently slowly", the existence of these overlapping subspaces should allow stability to be inferred for the larger union of these regions. Although explicit criteria for maximum delay variation rate are not given, it is discussed as a topic for future work. Additionally, it is not completely clear from the presentation whether the same gain K was used in each subspace or if different gains were being chosen to affect the subspace size. If the gains were fixed for all the different subspaces, gain variation could provide a possible way to extend the tolerable delay even further.

Finally, experimental results are presented for a GPRS network control system tracking a square wave with a period of ~ 12 min. From the responses, the system exhibited a stable underdamped response under conditions where the observed delay averaged 18 s and became as large as 35 s. It is interesting to note that nearly all of the simulation and experimental results treat case 6 (Figure 3) of the possible client–server interactions, where the total delay exceeds the sample period.

3. Conclusions

This paper begins to address the key requirements for developing network control via GPRS. The primary contribution is the development of the discrete dynamics variable delay problem as a switched system whose stability is analyzed using a LMI-based switched Lyapunov function search. The results demonstrate that bands of delay can be identified where overall system stability can be guaranteed. Experimental results indicate that a controller evaluated using this technique is suitable for closed-loop control via a real mobile wireless network using GPRS.

Two natural questions arise from reading this paper that were not directly addressed by the authors. The first question is, "What types of systems would be able to best utilize this technology?" Given the large delay levels and relatively slow sampling rates of 0.1–1 Hz observed in the experimental GPRS moNCS, the control environment seems suited for distributed systems with very slow and stable dynamics. In addition, the capability to use existing phone networks would allow large spatial areas to be monitored and controlled at relatively low cost given the ubiquity of cellular hardware. Some potential applications could include control of slow thermal systems, or remote monitoring and actuation of civil structures, solar power plants or windfarms. The second question is motivated by the distributed nature of the above applications and GPRS capable networks. This preliminary examination focused on a single control loop with one server and one client. Taking full advantage of the technology likely would require enabling network connectivity between multiple nodes. This expanded interaction could include a single client interacting with multiple servers for distributed sensing and actuation, or coordination between multiple loops [1]. The feasibility of the above schemes within the client-centric framework and the extension of the presented tools for those applications would be another possible avenue of future study.

References

1. Ploplys N, Kawka P, Alleyne A. Closed loop control over wireless networks. IEEE Control Syst Mag 2004; 24(3): 58–71
2. Ji K, Ambike A, Kim W-J. Control strategies for distributed real-time control with time delays and packet losses. In: Proceedings of 2004 ASME international mechanical engineering congress and exposition, Paper No. 61733. November, 2004
3. Ray A, Liou LW. A stochastic approach to delay compensation in network-based distributed intelligent control systems. *Proc IEEE Conf Decision Control* 1991; 2: 1891–1892
4. Daafouz J, Riedinger P, Iung C. Stability analysis and control synthesis for switched systems: a switched Lyapunov function approach. IEEE Trans Autom Control 2002; 47(11): 1883–1887
5. Mahoud M. Robust Control and Filtering for Time Delay System. Marcel Dekker Inc., NY, 2000, pp 32–34