Proceedings of the 42nd IEEE
Conference on Decision and Control
Maui, Hawaii USA, December 2003

TuP06-5

Minimizing Down-Link Traffic in Networked Control Systems via Optimal Control Techniques

Daniel E. Quevedo* dquevedo@ieee.org

Graham C. Goodwin* eegcg@ee.newcastle.edu.au

James S. Welsh* jwelsh@ee.newcastle.edu.au

Abstract—**This paper presents a control strategy for multivariable plants where controller and actuators are connected via a digital data-rate limited channel. In order to minimize bandwidth utilization, a communication constraint is imposed, which restricts all data transmitted to belong to a finite set and only permits one plant input to be addressed at a time. We develop a new scheme, which aims at optimizing quadratic performance under the above communication constraint. A key aspect of this contribution is the implementation of the control scheme to a real laboratory-scale system.**

I. INTRODUCTION

Networked Control Systems (NCS) are control systems, in which controller and plant are connected via a serial communication channel. They have attracted much recent interest, see e.g. [1], [2]. Practical applications abound. They have been made possible by technological developments, including the development of MEMS arrays, and may deploy wireless links (e.g. Bluetooth or IEEE 802.11), Ethernet (e.g. IEEE 802.3) or specialized protocols such as CAN.

While the usage of digital communication channels enables novel teleoperating applications, also new and interesting challenges arise. The network itself is a dynamical system that exhibits characteristics which traditionally have not been taken into account in control system design. These special characteristics include quantization and time-delays and are a consequence of the fact that practical channels have only a limited bandwidth. Thus, a networked controller should be designed to take into account the communication channel.

The specific problem addressed here is the design of a methodology for minimizing network traffic between a centralized controller and several actuators, without compromising the complexity at the actuator side. In particular, we restrict the controller so that only one actuator can be addressed at any one time and, moreover, only one of a finite number of levels can be transmitted. The design of the resulting system is aimed at optimizing performance subject to these constraints. In this context we choose to send increments in the control signal, rather than their actual values. In-between updates, all inputs are kept at their previous values.

Our design problem includes the issue of allocating communication time and, thus has connections to *limited communication control* as treated in [3], [4], [5], see also [6], [7]. Allocation of different forms of network resources has also been explored by [8], [9], [10], [11], [12]. Our strategy is characterized by the fact that we do not fix the control law a-priori, but instead determine the input via an on-line optimization, which takes account, *inter-alia*, of disturbances, speed of response, open loop unstable poles, etc. A key, and distinguishing, feature is that we use a finite-set constraint for all control moves, thus limiting the number of bits needed to be transmitted at any one time.

Our scheme is particularly suited to protocols such as *Modbus*, *Profibus* and *Control Net*, where the message size can be adjusted. However, this does not mean that bandwidth reduction via our proposal is only a consequence of minimizing message length. Bandwidth is also conserved due to the dynamic optimization of the system with respect to supplying control increments only when they are required.

An overview of the remainder of this paper is as follows: In Section II we provide additional background to our contribution. In Section III, we formulate the networked control system design problem under consideration. Section IV describes our proposed strategy which we call *Receding Horizon Networked Controller*. It relies on solving a finite set constrained optimization problem. By utilizing our previous results contained in [13], in Section V we state its closed form solution. This allows us to implement the state estimate controller by means of linear filters and a standard vector quantizer. Section VI documents a laboratory-scale experiment which illustrates the main characteristics of our controller. Section VII draws some conclusions.

II. BACKGROUND TO NETWORKED CONTROL SYSTEMS

As already stated in the Introduction, in NCS a digital communication medium is utilized to transmit data between controller-, sensor- and actuator-nodes. Due to the digital nature of the communication channel, every signal transmitted is expressed as a finite number of bits, hence it needs to be quantized to a finite set. As is well known in the nonlinear dynamical systems community, see e.g. [14], the introduction of quantizers has a strong impact on closed loop dynamics, see also [15]. Moreover, since a practical channel has only a limited data rate, the sampling rate and the size of the quantized sets are limited as well. This relation has been explored by several authors, such as [2], [16], [17], [18], [19] aiming at different notions of stabilizability, often at the expense of deploying very sophisticated decoders at the actuator side.

*School of Electrical Engineering & Computer Science, The University of Newcastle, Callaghan, NSW 2308, Australia

Another consequence of utilizing channels with limited bandwidth is the introduction of time-delays. Since the medium is data-rate limited, signals may have to *queue-up* before being transmitted [20], leading to delays in the *up-link* (between sensors and controller) and in the *down-link* (between controller and actuator). Without taking into account quantization effects, several problems have been studied in the literature, see e.g. [21], [22] which address control system design with random time delays and [23] which addresses the effect of time delays on stability for a fixed control law.

In many cases it is possible to *time-stamp* the up-link data [24], so that the delays are known at the receiving end. Thus, without taking into account quantization issues, plant state estimates can be obtained at the controller side by simply solving a standard Kalman Filtering Problem [25], [26], see also [27], [28], [29] for alternative approaches. On the other hand, time-stamping is not usually helpful in order to compensate for delays in the down-link, since they occur after the control calculations. Thus, deterministic down-link traffic is critical for performance.

In order to obtain fixed down-link delays, up-link data should be given lower priority and overall traffic should be kept at a minimum. This can be achieved by deploying event-based (nonuniform) sampling of the plant outputs, as described in [30]. Sensor data should be sent only when needed, see also [28]. At the down-link, traffic can be reduced by sending control increments, instead of control values and by sending data only to one actuator at a time. This is the set-up considered in the present work.

III. Problem Statement

As foreshadowed in the preceding sections, we consider the following problem: Suppose we want the outputs of a linear time-invariant MIMO plant with m inputs and s outputs to follow prescribed reference trajectories. The plant is connected to a central controller via a digital channel of limited bandwidth, which constitutes a significant bottleneck in the design. Following the reasoning of Sec. II, we assume that the link between the controller and actuators is characterized by a known and fixed time-delay and that data is sent at a bounded rate. This is achieved by imposing the following two communication constraints on the design:

Restriction 1. *The data sent from the controller to each actuator is restricted to belong to a (small and fixed) finite set of scalars,* $\mathbb{U}$.

Restriction 2. *Only data corresponding to one input of the plant can be transmitted at a time. In-between updates (which may be separated by several sampling periods) all plant inputs are held at their previous values.*

The delay between controller and plant can be incorporated into the discrete time plant model:

$$x(t+1) = Ax(t) + Bu(t), \quad y(t) = Cx(t) \qquad (1)$$

where $x(t) \in \mathbb{R}^{n\times 1}$, $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{s\times n}$ and

$$u(t) = \begin{bmatrix} u_1(t) & u_2(t) & \dots & u_m(t) \end{bmatrix}^T. \qquad (2)$$

The design problem can be stated as that of developing a control strategy, which makes the model (1) track a given reference $r(t)$, while not violating Restrictions 1 and 2. Thus, the control strategy for the networked system is characterized by choosing, at each time step, *which* of the m inputs in (2) to access and *what* to send. The controller needs to divide its actions between the plant inputs.

Remark 1 (Resetting the control signals). *It should be emphasized here that, although keeping the inputs to the plant at their previous values between updates as prescribed in Restriction 2 appears a natural choice, some authors utilize alternative strategies in which inputs are reset to zero after each sampling period, see e.g. [6], [7] and compare also to [5].*

Remark 2 (Control of multiple plants). *The above formulation encompasses the problem of controlling a collection of geographically separate plants. Simply note that a set of p plants, each described by:*

$$\begin{aligned} x_i(t+1) &= A_i x_i(t) + B_i u_i(t), \\ y_i(t) &= C_i x_i(t), \quad i = 1, \dots, p \end{aligned} \qquad (3)$$

can be put into the form (1), by defining x as the overall state:

$$x(t) \triangleq \begin{bmatrix} (x_1(t))^T & \dots & (x_p(t))^T \end{bmatrix}^T. \qquad (4)$$

With this, the matrices in the realization (1) are given as: $A = \operatorname{diag}(A_1, \dots, A_p)$, $B = \operatorname{diag}(B_1, \dots, B_p)$ *and* $C = \operatorname{diag}(C_1, \dots, C_p)$.

IV. The Receding Horizon Networked Controller

It is useful to decouple the overall design procedure into state estimation and control law design based on state estimates. Whilst in this paper we concentrate upon the control law, we refer the interested reader to our parallel work on estimation with quantized measurements [31].

Our strategy is characterized by sending finite set constrained control signal increments. These are calculated by a networked controller, which operates in a receding horizon fashion. In our scheme, computations are mainly done on the controller side.

A. *Specification of the Down-link*

Rather than sending the control signals directly, we propose to send their increments:

$$\Delta u_i(t) \triangleq u_i(t) - u_i(t-1) \qquad (5)$$

when nonzero. This choice, which is inspired by principles of *Delta-Modulation*, see e.g. [32, Chap.3], generally requires less bits to specify the control signal. The pair $(\Delta u_i(t), i)$ is received at the actuator node specified by the index i. The actual signal $u_i(t)$ is readily reconstructed by discrete time integration as shown in Fig. 1.[1]

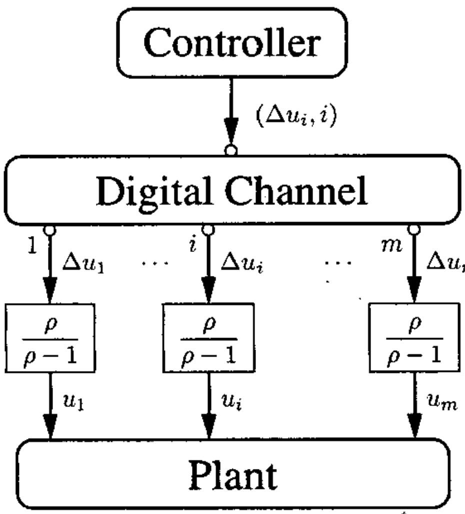

Fig. 1. Down-Link Design

The Restrictions 1 and 2 can be summarized by means of a simple finite-set constraint on the increments (5). More precisely, at every time instant t, the vector

$$\Delta u(t) \triangleq u(t) - u(t-1) \in \mathbb{R}^m \tag{6}$$

is restricted to belong to the set $\mathbb{V}$, which is defined as:

$$\mathbb{V} \triangleq \Big\{ V \in \mathbb{R}^m \text{ such that } \exists \mu \in \mathbb{U}: \\ V = \begin{bmatrix} 0 & \dots & 0 & \nu & 0 & \dots & 0 \end{bmatrix}^T \Big\}. \tag{7}$$

As depicted in Fig. 2; this set contains all vectors formed by one element of $\mathbb{U}$, whilst all its other components are zero.

B. Performance measure

Incorporating *preview* for the reference $r(t)$, the tracking performance of the model (1) over a finite horizon N starting at time $t = k$ can be quantified by means of the cost:[2]

$$V_N \triangleq \sum_{t=k+1}^{k+N} \|\widetilde{y}(t) - r(t)\|^2 + \sum_{t=k}^{k+N-1} \|\widetilde{\Delta u}(t)\|_R^2, \tag{8}$$

with $R > 0$. In this cost, $\widetilde{y}(t)$ and $\widetilde{\Delta u}(t)$ are predicted quantities that follow the plant dynamics (1), i.e.:

$$\begin{aligned} \widetilde{x}(t+1) &= A\widetilde{x}(t) + B\widetilde{u}(t), \quad \widetilde{y}(t) = C\widetilde{x}(t) \\ \widetilde{\Delta u}(t) &= \widetilde{u}(t) - \widetilde{u}(t-1), \quad t = k, \dots, k+N-1 \end{aligned} \tag{9}$$

[1] ρ denotes the forward shift operator, $\rho v(k) = v(k+1)$.

[2] $\|u\|_R^2$ denotes the quadratic form $u^T R u$.

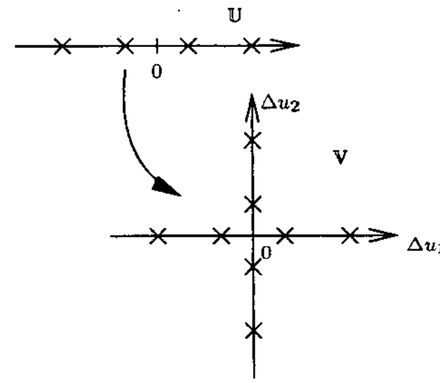

Fig. 2. Construction of the set $\mathbb{V}$

with initial conditions $\widetilde{x}(k) = x(k)$, the current state, which is assumed to be known and $\widetilde{u}(k-1) = u(k-1)$, the previous control value.

The decision variables of V_N can be grouped into:

$$\Delta\vec{u}(k) \triangleq \left[(\widetilde{\Delta u}(k))^T \quad \dots \quad (\widetilde{\Delta u}(k+N-1))^T \right] \in \mathbb{R}^{Nm}. \tag{10}$$

Thus, we write $V_N(\Delta\vec{u}(k))$.

C. Receding Horizon Formulation

Based upon the cost (8), we propose to utilize a receding horizon scheme as deployed in Model Predictive Control, see e.g. [33]. Therefore, at each time step, we solve the finite-set constrained quadratic programme:

$$\Delta\vec{u}^\star(k) \triangleq \arg \min_{\Delta\vec{u}(k) \in \mathbb{V}^N} V_N(\Delta\vec{u}(k)), \tag{11}$$

with $\mathbb{V}^N \triangleq \mathbb{V} \times \cdots \times \mathbb{V}$.

Instead of implementing the entire control sequence contained in $\Delta\vec{u}^\star(k)$, we only choose its first m components:

$$\Delta u^\star(k) = L^T \Delta\vec{u}^\star(k), \quad \text{where:}^3 \tag{12}$$

$$L \triangleq \begin{bmatrix} I_m & 0_m & \dots & 0_m \end{bmatrix}^T \in \mathbb{R}^{Nm \times m}. \tag{13}$$

This vector contains the data corresponding to all m inputs. Since $\Delta u^\star(k) \in \mathbb{V}$, not more than one of its components is nonzero. Only this value is sent to the plant input determined by its index, see Fig. 1. The other $m - 1$ inputs are left unchanged as prescribed in Restriction 2. As can be seen, the strategy consists of only sending the most relevant control increment, as quantified by the cost (8).

The networked controller specified by (11) and (12) constitutes the principal contribution of the present work. We will call it the *Receding Horizon Networked Controller* (RHNC). It makes decisions based upon overall future performance of the plant which respect the communication constraints contained in Restrictions 1 and 2.

[3] I_m denotes the identity matrix in $\mathbb{R}^{m \times m}$ and $0_m \triangleq 0 \cdot I_m$.

Note that, not only the control increment to be applied at each time is provided, but also the question of which input to access is addressed. Thus, the RHNC yields the optimal *communication sequence* (borrowing terminology from [3]).

V. Implementation of the RHNC

In this section we develop a closed form solution to the optimization problem (11). As will be apparent, the RHNC presented in the previous section can be implemented by utilizing linear filters and a vector quantizer. For that purpose, it is useful to define the vectors

$$\vec{u}(k) \triangleq \begin{bmatrix} \tilde{u}(k) \\ \tilde{u}(k+1) \\ \vdots \\ \tilde{u}(k+N-1) \end{bmatrix}, \quad \vec{y}(k) \triangleq \begin{bmatrix} \tilde{y}(k+1) \\ \tilde{y}(k+2) \\ \vdots \\ \tilde{y}(k+N) \end{bmatrix} \tag{14}$$

and to note that (8) can be re-written in vector form as:

$$V_N(\Delta\vec{u}(k)) = \|\vec{y}(k) - \vec{r}(k)\|^2 + \|\Delta\vec{u}(k)\|^2_{\overline{R}}, \tag{15}$$

with

$$\overline{R} \triangleq \operatorname{diag}(R, \dots, R).$$

Furthermore, iteration of (1) yields that

$$\vec{y}(k) = \Phi\vec{u}(k) + \Lambda x(k), \tag{16}$$

where:

$$\Phi \triangleq \begin{bmatrix} CB & 0 & \dots & 0 \\ CAB & CB & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA^{N-1}B & CA^{N-2}B & \dots & CB \end{bmatrix}, \Lambda \triangleq \begin{bmatrix} CA \\ CA^2 \\ \vdots \\ CA^N \end{bmatrix}. \tag{17}$$

In order to include the predicted increments (10) in these expressions, it can easily be verified that:

$$\Delta\vec{u}(k) = K\vec{u}(k) - Lu(k-1), \tag{18}$$

where L is defined in (13) and:

$$K \triangleq I_{Nm} - M, \; M \triangleq \begin{bmatrix} 0_m & \cdots & & \cdots & 0_m \\ I_m & 0_m & & & \vdots \\ 0_m & \ddots & \ddots & & \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0_m & \cdots & 0_m & I_m & 0_m \end{bmatrix}. \tag{19}$$

Given these relations, the controller which yields the receding horizon law (12) can be characterized in a simple fashion as described in Theorem 1 below. It makes use of a *nearest neighbour vector quantizer*, defined as follows:

Definition 1 (Nearest Neighbour Vector Quantizer). *Given a countable set of non-equal vectors $\mathcal{B} = \{b_1, b_2, \dots\} \subset \mathbb{R}^{n_\mathcal{B}}$, the nearest neighbour quantizer is defined as a mapping $q_\mathcal{B}\colon \mathbb{R}^{n_\mathcal{B}} \to \mathcal{B}$ which assigns to each vector $c \in \mathbb{R}^{n_\mathcal{B}}$ the closest element of $\mathcal{B}$, i.e., $q_\mathcal{B}(c) = b \in \mathcal{B}$ if and only if c satisfies: $\|c - b\| \leq \|c - b_i\|, \quad \forall b_i \in \mathcal{B}$.*

(A thorough treatment of quantizers can be found in [34].)

Theorem 1 (Closed Form Solution). *The optimizer $\Delta\vec{u}^\star(k)$ in (11) satisfies:*

$$\Delta\vec{u}^\star(k) = W^{-1/2} q_{\tilde{\mathbb{V}}^N}(-W^{-T/2}F), \quad \text{where:} \tag{20}$$

$$\begin{aligned} W &\triangleq K^{-T}\Phi^T\Phi K^{-1} + \overline{R} \\ F &\triangleq K^{-T}\Phi^T(\Phi K^{-1}Lu(k-1) + \Lambda x(k) - \vec{r}(k)) \end{aligned} \tag{21}$$

and $W^{1/2}$ is square and defined via $W^{T/2}W^{1/2} = W$.

The nonlinearity $q_{\tilde{\mathbb{V}}^N}(\cdot)$ is the nearest neighbour quantizer described in Definition 1. Its image is the set:

$$\tilde{\mathbb{V}}^N \triangleq \left\{\tilde{V} \in \mathbb{R}^{Nm} \colon \exists V \in \mathbb{V}^N \colon \tilde{V} = W^{1/2}V\right\}. \tag{22}$$

Proof: The proof follows closely that of our result in [13]. Given (15) to (18) it follows that

$$\begin{aligned} V_N(\Delta\vec{u}(k)) = \|\Delta\vec{u}(k)\|^2_W + 2(\Delta\vec{u}(k))^T F \\ + \overline{V}_N(x(k), \vec{r}(k), u(k-1)), \end{aligned} \tag{23}$$

where $\overline{V}_N(x(k), \vec{r}(k), u(k-1))$ does not depend upon $\Delta\vec{u}(k)$, and F and W are defined in (21).

We introduce the change of variables, $\vec{\mu}(k) = W^{1/2}\Delta\vec{u}(k)$. This transforms $\mathbb{V}^N$ into $\tilde{\mathbb{V}}^N$ defined in (22). Eq. (23) then allows one to rewrite the optimizer (11) as:

$$\Delta\vec{u}^\star(k) = W^{-1/2} \arg \min_{\vec{\mu}(k) \in \tilde{\mathbb{V}}^N} J_N(\vec{\mu}(k)),$$

$$J_N(\vec{\mu}(k)) \triangleq (\vec{\mu}(k))^T\vec{\mu}(k) + 2(\vec{\mu}(k))^T W^{-T/2}F.$$

The level sets of J_N are spheres in $\mathbb{R}^{Nm}$, centred at $-W^{-T/2}F$. Hence, the constrained optimizer is given by:

$$\arg \min_{\vec{\mu}(k) \in \tilde{\mathbb{V}}^N} J_N(\vec{\mu}(k)) = q_{\tilde{\mathbb{V}}^N}(-W^{-T/2}F),$$

which establishes the result given in (20). □

As a consequence, the RHNC can be characterized as in Fig. 3.

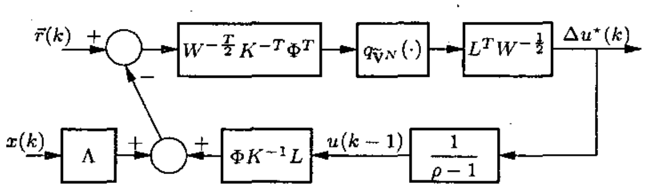

Fig. 3. Implementation of the RHNC

Remark 3. *The term F in the solution (20) contains the previous control value $u(k-1)$. Although we propose to calculate it directly in the controller by integrating all previous increments as illustrated in Fig. 1, this signal could alternatively be fed back from the plant, at the expense of additional traffic in the up-link.*

VI. Experimental Results

In order to illustrate the main characteristics of the Receding Horizon Networked Controller, we consider a practical example undertaken in our control laboratory where a multi-plant system consisting of level control for several identical tanks was configured. We focus on the attention given by the controller to the plants over a limited bandwidth channel.

A. System Configuration

Communications between the controller and plant actuators was over the IEEE 802.3 standard Ethernet, 10Base-T configuration utilizing the TCP/IP protocol. The communications interface employed was that of a client / server style architecture. The controller is the client and the plants are defined as servers. Thus, our NCS consisted of the controller running on a desktop PC with each plant also having a PC to act as a communication gateway for the plant actuators.

In each of the five tanks we seek to maintain a constant level by controlling the inflow via a pump. Outflow is through a constant sized orifice. Control is required to reject disturbances from unmeasured inflows and increased outflows. In our set-up, a single tank having as output, the fluid level, and input, the pump voltage, is described by the discrete-time model (3) with a sampling period of 1 second and matrices:

$$A_i = 0.9716, \ B_i = 0.125, \ C_i = 0.1979.$$

Here static scheduling would divide the attention of the controller evenly between all the tanks. In the case of the RHNC, the scheduling of the control is determined dynamically, such that the plant or plants in most need of attention receive the control necessary to maintain the required levels. In order to emphasize the quantized nature of the problem we choose the set $\mathbb{U} = \{-1, 0, 1\}$. Now essentially, only one bit is needed to transmit the control action. Indeed, under these conditions it is only necessary, in principle, to transmit 4 bits per second (3 for the address and 1 for the level change up or down).

An issue, not specifically addressed by our network set-up, is that of reduced bandwidth due to the size of the data required to be sent to each actuator. The reason for this is due to the Ethernet packet transmitted being of a fixed minimum length which is much larger than we require. Channel bandwidth on the down-link was artificially limited by allowing the controller to write only to the network once per second, i.e. only one plant would receive a control signal each second. Up-link traffic did not have this restriction.

B. Results

In this section we present the experimental results. We control the five tanks by means of the RHNC with horizon $N = 2$ following Remark 2.

Once the tanks have reached the desired level (0V) after the initial start-up, an unmeasured inflow disturbance was introduced into Tank 2 at 35 seconds and an extra outflow valve was opened on Tank 3 at 1090 seconds. Figs. 4–6, show the control increments sent by the controller to each plant. The control signal applied to the plant by the actuator is also shown together with the level measurement from the tank. It can be seen from these figures that, when the disturbance occurs in Tank 2, most down-link bandwidth is dedicated to the control of level in this tank. Also, it is easily observed that the controller pays attention to the other tanks when the measured level in tank 2 approaches the desired level at around 210 seconds into the experiment. The same effect can be observed when tank 3 is disturbed.

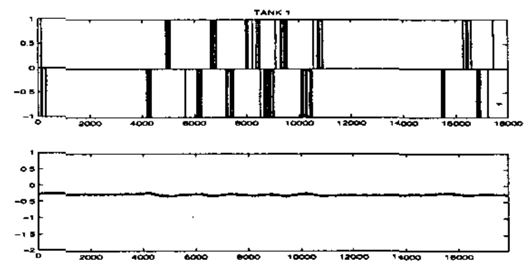

Fig. 4. Top: Control action sent by Controller, Bottom: Level Measurement

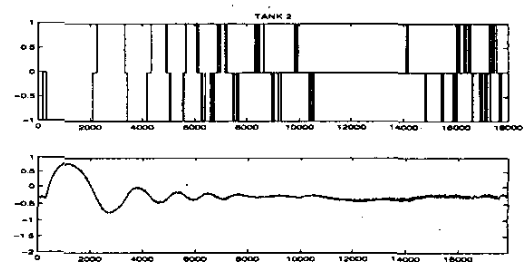

Fig. 5. Top: Control action sent by Controller, Bottom: Level Measurement

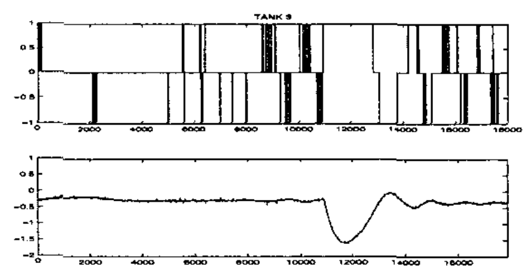

Fig. 6. Top: Control action sent by Controller, Bottom: Level Measurement

This aspect of sharing attention is also apparent in the histograms contained in Fig. 7. Most control attention ($\Delta u_i = \pm 1$) was applied to those tanks, for the given time period, to which the disturbances occurred.

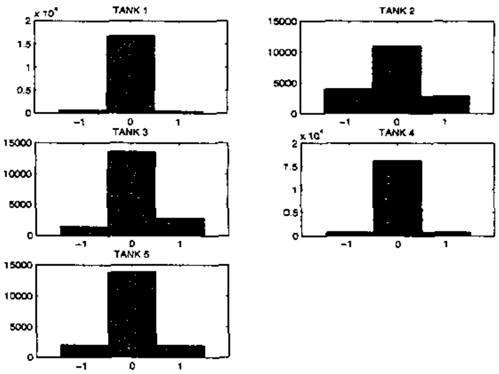

Fig. 7. Histogram of Control Increments

VII. CONCLUSIONS

We have presented a novel scheme for minimizing downlink traffic in Networked Control Systems. We have termed the method the Receding Horizon Networked Controller and illustrated its main features by means of a laboratory-scale experiment. It provides a direct solution to a design problem for Networked Control Systems under communication constraints which respect the finite-set nature of digital data.

VIII. REFERENCES

[1] L. Bushnell, "Special section on networks and control," *IEEE Control Systems Magazine*, vol. 21, pp. 57–99, Feb. 2001.

[2] H. Ishii and B. A. Francis, *Limited Data Rate in Control Systems with Networks.* Springer, 2002.

[3] R. W. Brockett, "Stabilization of motor networks," in *Proc. of the 34th IEEE CDC*, pp. 1484–1488, 1995.

[4] D. Hristu and K. Morgansen, "Limited communication control," *Sys. & Contr. Letters*, vol. 37, no. 4, pp. 193–205, 1999.

[5] D. Hristu, "Feedback control systems as users of a shared network: Communication sequences that guarantee stability," in *Proc. of the 40th IEEE CDC*, pp. 3631–3636, 2001.

[6] B. Lincoln and B. Bernhardsson, "LQR optimization of linear system switching," *IEEE Trans. Automat. Contr.*, vol. 47, no. 10, pp. 1701–1705, 2002.

[7] L. Palopoli, A. Bicchi, and A. Sangiovanni-Vincentelli, "Numerically efficient control of systems with communication constraints," in *Proc. of the 41st IEEE CDC*, 2002.

[8] L. Xiao, M. Johansson, H. Hindi, S. Boyd, and A. Goldsmith, "Joint optimization of communication rates and linear systems," *IEEE Trans. Automat. Contr.*, vol. 48, no. 1, pp. 148–153, 2003.

[9] G. C. Walsh, H. Ye, and L. G. Bushnell, "Stability analysis of networked control systems," *IEEE Trans. Automat. Contr.*, vol. 10, no. 3, pp. 438–446, 2002.

[10] G. C. Walsh, O. Beldiman, and L. G. Bushnell, "Error encoding algorithms for networked control systems," *Automatica*, vol. 38, pp. 261–267, 2002.

[11] M. S. Branicky, S. M. Phillips, and W. Zhang, "Scheduling and feedback co-design for networked control systems," in *Proc. of the 41st IEEE CDC*, pp. 1211–1217, 2002.

[12] H. S. Park, Y. H. Kim, D.-S. Kim, and W. H. Kwon, "A scheduling method for network-based control systems," *IEEE Trans. Automat. Contr.*, vol. 10, no. 3, pp. 318–330, 2002.

[13] D. E. Quevedo, G. C. Goodwin, and J. A. De Doná, "Finite constraint set receding horizon control," *International Journal of Robust and Nonlinear Control*, vol. 13, 2003, to appear.

[14] O. Feely, "A tutorial introduction to non-linear dynamics and chaos and their application to Sigma-Delta modulators," *International Journal of Circuit Theory and Applications*, vol. 25, pp. 347–367, 1997.

[15] J. Baillieul, "Feedback coding for information-based control: Operating near the data-rate limit," in *Proc. of the 41st IEEE CDC*, 2002.

[16] S. Tatikonda, A. Sahai, and S. Mitter, "Control of LQG systems under communication constraints," in *Proceedings of the 37th IEEE CDC*, pp. 1165–1170, 1998.

[17] W. S. Wong and R. W. Brocket, "Systems with finite communication bandwidth constraints–Part II: Stabilization with limited information feedback," *IEEE Trans. Automat. Contr.*, vol. 44, no. 5, pp. 1049–1053, 1999.

[18] G. N. Nair and R. J. Evans, "Exponential stabilisability of finite-dimensional linear systems with limited data rates," *Automatica*, vol. 39, pp. 585–593, 2003.

[19] N. Elia and S. K. Mitter, "Stabilization of linear systems with limited information," *IEEE Trans. Automat. Contr.*, vol. 46, no. 9, pp. 1384–1400, 2001.

[20] F. L. Lian, J. R. Moyne, and D. M. Tilbury, "Performance evaluation of control networks," *IEEE Control Syst. Mag.*, vol. 21, pp. 66–83, Feb. 2001.

[21] L. W. Liou and A. Ray, "A stochastic regulator for integrated communication and control systems," *Transactions of the ASME*, vol. 113, pp. 604–619, 1991.

[22] J. Nilsson, B. Bernhardsson, and B. Wittenmark, "Stochastic analysis and control of real-time systems with random time delays," *Automatica*, vol. 34, no. 1, pp. 57–64, 1998.

[23] 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, Chicago, Illinois*, pp. 2352–2357, 2000.

[24] H. Chan and Ü. Özgüner, "Closed-loop control of systems over a communications network with queues," *International Journal of Control*, vol. 62, no. 3, pp. 493–510, 1995.

[25] G. C. Goodwin and K. S. Sin, *Adaptive Filtering Prediction and Control.* Englewood Cliffs, NJ: Prentice-Hall, 1984.

[26] B. D. O. Anderson and J. Moore, *Optimal filtering.* Englewood Cliffs, N.J.: Prentice Hall, 1979.

[27] O. Beldiman and G. Walsh, "Predictors for networked control systems," in *Proceedings of the American Control Conference, Chicago, Illinois*, pp. 2347–2351, 2000.

[28] P. H. Bauer, M. Sichitiu, C. Lorand, and K. Premaratne, "Total delay compensation in LAN control systems and implications for scheduling," in *Proceedings of the American Control Conference, Arlington, VA*, pp. 4300–4305, 2001.

[29] P. V. Zhivoglyadov and R. H. Middleton, "Networked control design for linear systems," *Automatica*, vol. 39, pp. 743–750, 2003.

[30] K. J. Åström, "Riemann and Lebesgue sampling," in *Advances in System Theory: A Symposium in Honor of Sanjoy K. Mitter, October 15–16, Cambridge, MA*, 1999.

[31] H. Haimovich, G. C. Goodwin, and D. E. Quevedo, "Rolling horizon Monte Carlo state estimation for linear systems with output quantization," in *Proc. of the 42nd IEEE CDC*, 2003.

[32] J. G. Proakis, *Digital Communications.* McGraw-Hill, 1995.

[33] J. M. Maciejowski, *Predictive Control with Constraints.* Prentice-Hall, 2002.

[34] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression.* Kluwer Academic, 1992.