RESEARCH

Open Access

CrossMark

Multiple descriptions for packetized predictive control

Jan Østergaard[1*] and Daniel Quevedo[2]

Abstract

In this paper, we propose to use multiple descriptions (MDs) to achieve a high degree of robustness towards random packet delays and erasures in networked control systems. In particular, we consider the scenario, where a data-rate limited channel is located between the controller and the plant input. This forward channel also introduces random delays and dropouts. The feedback channel from the plant output to the controller is assumed noiseless. We show how to design MDs for packetized predicted control (PPC) in order to enhance the robustness. In the proposed scheme, a quantized control vector with future tentative control signals is transmitted to the plant at each discrete time instant. This control vector is then transformed into M redundant descriptions (packets) such that when receiving any $1 \leq J \leq M$ packets, the current control signal as well as $J - 1$ future control signals can be reliably reconstructed at the plant side. For the particular case of LTI plant models and i.i.d. channels, we show that the overall system forms a Markov jump linear system. We provide conditions for mean square stability and derive upper bounds on the operational bit rate of the quantizer to guarantee a desired performance level. Simulations reveal that a significant gain over conventional PPC can be achieved when combining PPC with suitably designed MDs.

Keywords: Quantization, Networked control, Multiple descriptions

1 Introduction

In networked control systems (NCSs), the controller communicates with the plant via a general purpose communication network [1, 2]. When compared to using dedicated hardwired control networks, the use of general purpose and possibly wireless communication technology brings significant benefits in terms of efficiency, interoperability, deployment costs, etc. However, the use of practical communication technology also leads to new challenges, since the network needs to be taken into account in the overall design, see also [1–7].

In this paper, we will focus on the existence of a digital network between the controller and the plant input. This network contains either a single channel that introduces i.i.d. packet delays and erasures or multiple independent channels with i.i.d. packet delays and erasures. The channel between the plant output and the controller is considered ideal, i.e., noiseless and instantaneous. For example, this could be a situation where the controller

and plant communicates over wireless channels. The controller could be battery driven and therefore with limited transmission power. On the other hand, the plant might not have a limitation on the transmission power. In this case, the reverse channel from the plant to the controller has a significantly greater SNR than the forward channel between the controller and the plant. There are many other practical sitations with wireless controller-actuator links but direct sensor-controller connections, e.g., groups of agents/vehicles/robots/drones. Their positions/formation are sensed via a system comprising a camera and attached controller. Activation commands are then sent wirelessly to the agents.

The main contributions of this work is the theoretical analysis and practical design of the quantized control signals. In particular, we propose to combine a recent robust control strategy known as (quantized) packet predictive control (PPC) [8–11] with a joint source-channel coding strategy based on multiple description (MD) coding [12, 13]. We provide computable upper bounds on the operational bit rate required for coding the quantized control signals (descriptions) and provide a practical design based on our theoretical analysis. The simulation study

*Correspondence: jo@es.aau.dk
[1] Department of Electronic Systems, Aalborg University, Fredrik Bajers Vej 7b, Aalborg, Denmark
Full list of author information is available at the end of the article

shows that the combination of MDs and PPC provides a significant improvement over PPC in the case of large packet loss ratios.

In quantized PPC, a control vector with the current and $N-1$ future predicted plant inputs is constructed at the controller side to compensate for random delays and packet dropouts in the channel. Thus, in the case of packet erasures (and if not too many consecutive dropouts occur), the buffer will feed the plant with the appropriate future predicted control value [8]. The key principle of MDs is to encode a source signal into a number of descriptions (packets) that are transmitted over separate channels. Each description is able to approximate the source signal to within a prescribed quality. Moreover, if several descriptions are received, they can be combined to further improve the reconstruction quality. Thus, in the case of packet erasures, it is possible to achieve a graceful degradation of the reconstruction quality [13].

The design of optimal quantized control strategies subject to data rate limitations defines a complicated problem that lies in the intersection of signal processing and controls. In particular, if the quantizers are designed using conventional open-loop source-coding strategies, it cannot be guaranteed that the overall system will be stable, when used in closed-loop control. Indeed, the resulting data rate could exceed the bandwidth of the digital channel, the data rate could be too low to capture the plant uncertainty and thereby not guarantee stability, or the non-linear effects due to quantization could have a negative impact on the overall stability when fed back into the system [8, 9, 14, 15].

The combination of MDs and PPC has to the best of the authors' knowledge not been considered before (except in the conference contributions of the authors [16–18]). In [16], MDs were used for power control in wireless sensor networks. The quantizers were designed under high-resolution assumptions, and no stability assessment was provided. In [17, 18], the preliminary ideas for the current work (without analysis and proofs) were presented. MDs for state-estimation was considered in [19, 20] under high-resolution quantization assumptions. The design of lattice quantizers for PPC without MDs was treated in [16, 21] for the cases of entropy-constrained and resolution-constrained quantization, respectively.

In this work, we will focus on LTI plant models, which are (possibly) open-loop unstable. Thus, it is necessary to provide quantized control signals to the plant in a reliable way to guarantee stability in the presence of data rate limitations, random packet delays and erasures. Our key idea is to design and use MDs in a novel way that differs from how it is traditionally used. Traditionally, when the received descriptions are combined at the decoder, the approximation of a given source signal is improved. On the other hand, in the proposed work, when the received descriptions are combined at the decoder, then rather than improving existing control signals, new future controls signals are instead recovered.

There exists a vast amount on literature on MJLS with delays, cf.,[22–25]. In the present work, we show that the overall system with delays, erasures, quantization effects, and multiple descriptions, can be cast as a Markov jump linear system (MJLS), which makes it possible to use general stability results from the MJLS literature [26, 27].

The paper is organized as follows. Section 2 contains background information on quantized PPC. Section 3 contains the system analysis of a theoretical joint PPC and MD scheme. Section 4 presents the design of the combined practical PPC and MD scheme. Section 5 provides a simulation study of the proposed scheme. Section 6 contains the conclusions. Proofs of lemmas and theorems are deferred to the appendices.

1.1 Notation
Let $S^{\downarrow}$ be the down-shift-by-one matrix operator, which replaces the jth row of an $N \times M$ matrix by its $(j-1)$th row for $j = N, \ldots, 2$. Similarly, define $S^{\uparrow}$ as the up-shift-by-one matrix operator. Let e_i denote the unit-vector aligned with the ith axis of the Cartesian coordinate system, e.g., $e_2 = [0, 1, 0, \cdots, 0]^T$, where the dimension of e_i will be clear from the context. Let $\mathbf{1}_i \in \mathbb{R}^i$ be the all-ones vector of dimension i. Let γ_i be the matrix operator that takes the ith diagonal of an $N \times N$ matrix, where $i = 1$ is the main diagonal and $i > 1$ are diagonals above the main diagonal. Thus, $\gamma_i(A) \in \mathbb{R}^{N-i+1}$ if $A \in \mathbb{R}^{N \times N}$. We will use $\sigma_r(A)$ to denote the spectral radius of the matrix A, and $A \otimes B$ denotes the usual Kronecker product between the matrices A and B. The squared and weighted l_2-norm of a vector, say x, is written as $\|x\|_P^2 = x^T P x$, where $P \succeq 0$, i.e., P is a positive semidefinite matrix.

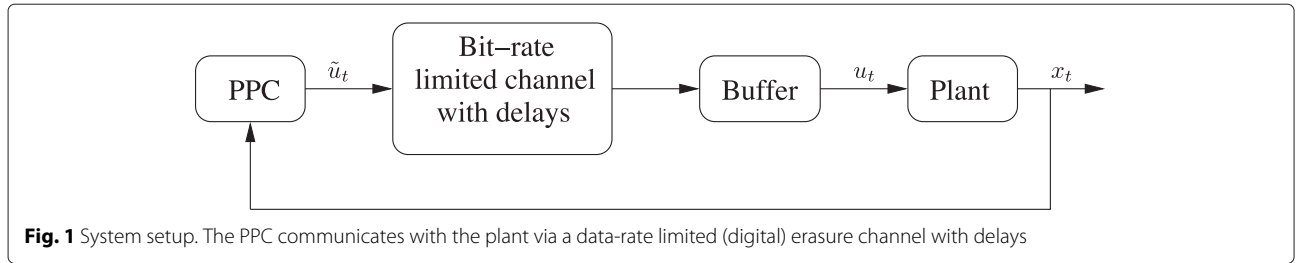## 2 Quantized packetized control over erasure channels
In this section, we provide a summary of existing results on quantized PPC and relate them to the present situation. The system considered is shown in Fig. 1. For a more detailed presentation of quantized PPC, see [11].

2.1 System model
We consider the following discrete-time stochastic linear time invariant (LTI) possibly unstable dynamical plant with state $x_t \in \mathbb{R}^z$, $z \geq 1$ and scalar input $u_t \in \mathbb{R}$:

$$x_{t+1} = Ax_t + B_1 u_t + B_2 w_t, \quad t \in \mathbb{N}. \tag{1}$$

In (1), $w_t \in \mathbb{R}^{z'}$, $z' \geq 1$, is an unmeasured disturbance, modeled as an arbitrarily distributed (and with possibly unbounded support) zero-mean stochastic process with bounded covariance matrix Σ_w, and $B_1 \in \mathbb{R}^z$ and $B_2 \in$

Fig. 1 System setup. The PPC communicates with the plant via a data-rate limited (digital) erasure channel with delays

$\mathbb{R}^{z \times z'}$. We do not assume that $A \in \mathbb{R}^{z \times z}$ is stable; however, we will assume that the pair (A, B_1) is stabilizable. The initial state x_0 is arbitrarily distributed with bounded variance.

2.2 Cost function

In MPC, at each time instant t and for a given plant state x_t, one often uses a linear quadratic cost function on the form [28]:

$$V(\bar{u}', x_t) \triangleq \|x'_N\|_P^2 + \sum_{\ell=0}^{N-1} \left(\|x'_\ell\|_Q^2 + \lambda (u'_\ell)^2 \right), \qquad (2)$$

where $N \geq 1$ is the horizon length, and the design variables $P \succeq 0$, $Q \succeq 0$ and $\lambda > 0$ allow one to trade-off control performance versus control effort. The variables x'_ℓ and $\bar{u}'_l$ denote tentative variables and are defined below. The final state weighting $\|x'_N\|_P^2$ in (2) aids in stabilizing the feedback loop by approximating the effect of the infinite-horizon behaviour [28]. For example, one may choose P as the unique positive semidefinite solution to the discrete algebraic Riccati equation:

$$P = A^T P A + Q - A^T P B_1 \left(\lambda + B_1^T P B_1 \right)^{-1} B_1^T P A, \quad (3)$$

which exists if the system (1) is stabilizable [28].

The cost function in (2) examines a prediction of the plant model over a finite horizon of length N. It is common to assume that the predicted state trajectories at time t are independent of the buffer contents at the decoder (i.e., they are independent of what has been received at the plant input side), network effects, and the external disturbances w_t, and are generated by

$$x'_{\ell+1} = A x'_\ell + B_1 u'_\ell, \qquad (4)$$

$x'_0 = x_t$, while the entries in $\bar{u}' = \left[u'_0, \ldots, u'_{N-1} \right]^T$ represent the associated predicted plant inputs. Thus, the current control vector

$$\bar{u}_t = \left[u_t(1), \ldots, u_t(N) \right]^T$$

contains the control signal $u_t(1)$ for the current time instant t as well as $N-1$ future predictive control signals for time up to $t + N - 1$.

One may include the effect of the channel delays in the cost function (2) by, for example, formulating the individual stage costs in terms of their expected stage costs, i.e., weighting by the probabilities of control signals being delayed:

$$\mathbb{E} \sum_{\ell=1}^{N-1} \left(\|x'_\ell\|_Q^2 + \lambda (u'_\ell)^2 \right) = \sum_{\ell=1}^{N-1} \left(\|x'_\ell\|_Q^2 + \lambda (u'_\ell)^2 \right) p_\ell, \quad (5)$$

where p_ℓ denotes the probability of using the control signal u'_ℓ. Moreover, in this work, we will also model the effect of the quantizer directly in the design of the control signal u'_ℓ, see Section 2.4 for details.

Following the ideas underlying PPCs, see, e.g., [29], at each time instant t, and for current state x_t, the controller sends the *entire* optimizing sequence, $\bar{u}_t$, to the actuator node. Depending upon future packet dropout scenarios, a subsequence of $\bar{u}_t$ will be applied at the plant input, or not. Following the receding horizon paradigm, at the next time instant, x_{t+1} is used to carry out another optimization, yielding $\bar{u}_{t+1}$, etc.

2.3 Network effects

As illustrated in Fig. 1, we shall assume that the backward channel of the network is noiseless and instantaneous, whereas the forward channel is a packet erasure channel, where packets can be delayed and also be received out-of-order. In fact, we allow the delay to be unbounded, which means that packets can be lost. In our setup, if a transmitted packet has not been received within N consecutive time slots, it is considered lost. In MD coding, it is common to assume the availability of either M separate and independent channels or a single (compound) channel where the M packets can be sent simultaneously and yet be subject to independent erasures and delays [13]. Formally, we define $\tau_t^i \in \mathbb{N}_0 \cup \infty$ to be the delay experienced by the ith packet that is constructed at time t. Thus, τ_t^i is a property of the ith channel. We will assume that the delays $\{\tau_t^i\}$ experienced by the different packets are independent and identically distributed (i.i.d.). With this notation, we model transmission effects via the discrete

processes $\left\{ d_{t,t'}^i \right\}_{t'=t}^{\infty}$, where $0 \le t \le t'$ and $i = 1, \ldots, M$, defined via:

$$d_{t,t'}^i \triangleq \begin{cases} 1, & \text{if } \tau_t^i \le t' - t, \\ 0, & \text{else,} \end{cases}$$

where $\tau_t^i \le t' - t$ implies that the ith packet constructed and transmitted at time t has experienced a delay no more than $t' - t$ time instances. We note that even though $\tau_t^i, \forall t$, are mutually independent, the processes $d_{t,t'}^i$ are generally not i.i.d., since if a packet constructed at time t experiences a delay of τ_t^i, then $d_{t,t'}^i = 1$ for all $t' \ge t + \tau_t^i$. However, for $t' = t$, the outcomes $d_{t,t}^i, i = 1, \ldots, M, t \ge 0$, are assumed mutually independent. We will also assume that the packet reception at time t is conditionally independent of the past packet receptions prior to time $t - N$, given the knowledge of the packet reception between time t and $t - N + 1$. Specifically, for $t' \ge t + N$,

$$\text{Prob}\left(d_{t,t'}^i = 1 | d_{t,t'-1}^i, d_{t,t'-2}^i, \ldots, d_{t,t}^i \right)$$
$$= \text{Prob}\left(d_{t,t'}^i = 1 | d_{t,t'-1}^i, d_{t,t'-2}^i, \ldots, d_{t,t'-N+1}^i \right).$$

Finally, we assume that the channel statistics are stationary so that $\text{Prob}\left(d_{t,t'}^i = 1 | d_{t,t'-1}^i, d_{t,t'-2}^i, \ldots, d_{t,t'-N+1}^i \right)$ does not depend upon t. We will make explicit use of the above stationarity and Markov assumptions in Lemma 3.2.

2.4 Quantization constraints

We consider a bit-rate limited digital network between controller output and plant input and all data to be transmitted needs therefore to be quantized. This introduces a quantization constraint into the problem of minimizing $V(\bar{u}', x_t)$.

Let $\overline{Q} \triangleq \text{diag}(Q, \ldots, Q, P) \in \mathbb{R}^{zN \times zN}$ and define:

$$\Phi \triangleq \begin{bmatrix} B_1 & 0 & \ldots & 0 \\ AB_1 & B_1 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A^{N-1}B_1 & A^{N-2}B_1 & \ldots & B_1 \end{bmatrix} \in \mathbb{R}^{zN \times N}, \quad (6)$$

$$\Upsilon \triangleq \begin{bmatrix} A \\ A^2 \\ \vdots \\ A^N \end{bmatrix} \in \mathbb{R}^{zN \times z}, \quad (7)$$

$$F \triangleq \Upsilon^T \overline{Q} \Phi \in \mathbb{R}^{z \times N}, \quad (8)$$

$$\Gamma \triangleq -\Psi^{-T} F^T \in \mathbb{R}^{N \times z}, \quad (9)$$

$$\Psi^T \Psi = \Phi^T \overline{Q} \Phi + \lambda I \in \mathbb{R}^{N \times N},$$

$$\xi_t \triangleq \Gamma x_t \in \mathbb{R}^N. \quad (10)$$

Then using the above and (4), the cost function (2) can be rewritten as

$$V(\bar{u}', x_t) = x_t^T \Upsilon^T \Upsilon x_t + \bar{u}' \Psi^T \Psi \bar{u}' + 2 x_t^T F \bar{u}', \quad (11)$$

which has the unique (unquantized) minimizer $\bar{u}^*$ given by

$$\bar{u}^* = (\Psi^T \Psi)^{-1} F^T x_t \quad (12)$$

$$= \Psi^{-1} \Gamma x_t = \Psi^{-1} \xi_t \in \mathbb{R}^N. \quad (13)$$

We note that Ψ is fixed and we may at this point either directly quantize $\bar{u}^*$ or instead quantize ξ_t and then apply the mapping Ψ^{-1} in order to obtain the quantized control vector.[1] Since Ψ is invertible, and we are transmitting the entire quantized control vector, the resulting coding rate is not affected by this operation [30].

When using entropy-constrained (subtractively) dithered (lattice) quantization (ECDQ), a dither vector ζ_t is added to the input prior to quantization and then subtracted again at the decoder to obtain the reconstruction [31].[2] Specifically, let $\mathcal{Q}_\Lambda$ denote an ECDQ with underlying lattice Λ. Then the discrete output ξ_t' of the ECDQ is given by $\xi_t' = \mathcal{Q}_\Lambda(\xi_t + \zeta_t)$. Furthermore, the reconstruction $\hat{\xi}_t$ at the decoder is then obtained by subtracting the dither, i.e., by forming $\hat{\xi}_t = \xi_t' - \zeta_t$. Interestingly, this quantization operation may be exactly modeled by an additive noise channel, i.e., we have $\hat{\xi}_t = \xi_t + n_t$, where the noise n_t is zero-mean with variance σ_n^2 and independent of ξ_t, see [31] for details. With this, the quantized (and reconstructed) control variable $\bar{u}_t$ can be written as

$$\bar{u}_t = \Psi^{-1}(n_t + \xi_t), \quad (14)$$

where n_t and ξ_t are mutually independent and $\xi_t = \Gamma x_t$. We note that $\bar{u}_t$ is the quantized (and reconstructed) control signal, which has been found by using an ECDQ on ξ_t. Thus, $\bar{u}_t$ is a continuous variable whereas $\tilde{u}_t = \Psi^{-1} \xi_t'$ is the corresponding discrete valued variable, which is entropy coded and thereby converted into a bit-stream (to be transmitted over the network), see Fig. 1. Throughout this work, we will use $u_t(i)$ to refer to the ith element of the vector $\bar{u}_t$.

2.5 MD coding for PPC

We design the MDs by explicitly exploiting the *layered* construction of the control signals. In particular, we first generate a quantized control vector based on the principles of PPC. This vector contains the current control signal and $N - 1$ future control signals. Then, we construct M descriptions based on this control vector. The descriptions are constructed so that the current control signal and $J - 1$ future control signals can be obtained by combining any subset of $J \in \{1, \ldots, M\}$ descriptions. Thus, the more packets that are received at the plant, the more future plant predictions become available. Note that on reception of at least one packet out of the M packets, the current quantized control signal can be completely recovered at the plant input side. When receiving

and combining more descriptions, the quality of this control signal is not improved. Instead new control signals become available. With this approach, we thus avoid the issue of having to guarantee stability subject to a probabilistic and time-varying accuracy of the control signals. Instead, we can use ideas from quantized PPC, when assessing the stability. A detailed design of the MDs is provided in Section 4.

3 Theoretical analysis of the PPC-MDC scheme

3.1 Markov jump linear system

Let $\bar{x}_t \in \mathbb{R}^{zN \times 1}$ be the $N-1$ past and the present system state vectors, i.e.,

$$\bar{x}_t \triangleq \left[x_t^T, \ldots, x_{t+1-N}^T \right]^T, \tag{15}$$

where x_t is given by (1), and let $\bar{n}_t$ be the $N-1$ past and the present quantization noise vectors, i.e.,

$$\bar{n}_t = \left[n_t^T, \ldots, n_{t+1-N}^T \right]^T \in \mathbb{R}^{N^2 \times 1}, \tag{16}$$

where n_t is introduced in (14). Moreover, let $\Xi_t \in \mathbb{R}^{N(z+1) \times 1}$ be the augmented state variable given by

$$\Xi_t \triangleq \begin{bmatrix} \bar{x}_t \\ \bar{f}_{t-1} \end{bmatrix}, \tag{17}$$

where $\bar{f}_t = [f_t(1), \ldots, f_t(N)]^T \in \mathbb{R}^{N \times 1}$ represents the buffer with the control signals to be applied by the actuator at the plant input side. This buffer holds the present and the $N-1$ tentative future control values. In particular, $f_t(1)$ is the control value to be applied at current time t, and $f_t(i)$ is to be applied at time $t+i-1$. In addition, there is also a buffer $\bar{f}'_t$ at the plant side, which holds all received packets that are no older than $t-N+1$ time instances.

Let $\Delta_t \in \mathbb{R}^{N \times N}$ be an indicator matrix with binary elements $\{0,1\}$ indicating the complete buffer contents of $\bar{f}'_t$ at time t. In particular, if Δ_t has a "1" at entry (i,j), it shows that at least j packets from time $t-i+1$ have been received and the buffer therefore contains at least $u_{t-i+1}(1), u_{t-i+1}(2), \ldots, u_{t-i+1}(j)$. If, in addition, entry $(i, j+1) = 1$, it further means that the buffer also contains $u_{t-i+1}(j+1)$. To better illustrate the relationship between Δ_t and the buffers $\bar{f}'_t$ and $\bar{f}_t$ consider the following example.

Example 3.1. *Let $N = 3$ and assume that $\bar{f}'_t$ is empty and that $\bar{f}_t$ is initialized to zero. Moreover, let the three packets constructed at time t be denoted by $s_t(i), i = 1, \ldots, 3$. Then at time t, assume that two packets, say $s_t(1)$ and $s_t(3)$, constructed at time t are received, which implies that $u_t(1)$ and $u_t(2)$ can be recovered. At time $t+1$, a single packet, say $s_{t+1}(1)$, from time $t+1$ is received. Finally, at time $t+2$, the third and remaining packet $s_t(2)$ from*

time t is received. This leads to the following sequence of variables:

$$\bar{f}'_t = \{s_t(1), s_t(3)\} \Rightarrow \Delta_t = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$\Rightarrow \bar{f}_t = \begin{bmatrix} u_t(1) \\ u_t(2) \\ 0 \end{bmatrix}$$

$$\bar{f}'_{t+1} = \{s_{t+1}(1), s_t(1), s_t(3)\} \Rightarrow \Delta_{t+1} = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$\Rightarrow \bar{f}_{t+1} = \begin{bmatrix} u_{t+1}(1) \\ 0 \\ 0 \end{bmatrix}$$

$$\bar{f}'_{t+2} = \{s_{t+1}(1), s_t(1), s_t(2), s_t(3)\}$$

$$\Rightarrow \Delta_{t+2} = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \Rightarrow \bar{f}_{t+2} = \begin{bmatrix} u_t(3) \\ 0 \\ 0 \end{bmatrix}.$$

□

In order to present a formal relationship between Δ_t and the buffer $\bar{f}_t$, we introduce U_t as the upper triangular matrix containing the relevant control signals, that is

$$U_t = \begin{bmatrix} u_t(1) & u_t(2) & u_t(3) & \cdots & u_t(N) \\ 0 & u_{t-1}(2) & u_{t-1}(3) & \cdots & u_{t-1}(N) \\ \vdots & 0 & & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & u_{t-N+1}(N) \end{bmatrix}. \tag{18}$$

The control signal to be applied at time t is given by one of the elements on the main diagonal of U_t, and the control signal to be applied at time $t + j$ is an element on the jth diagonal above the main diagonal (unless the buffer is changed in the mean time). Let

$$\gamma_i(U_t) = [u_t(i), u_{t-1}(i+1), \cdots, u_{t-N+i}(N)]^T$$

and let $\delta_i \triangleq \gamma_i(\Delta_t)$. Moreover, let

$$\tilde{\delta}_i(k) \triangleq \delta_i(k) \prod_{j=1}^{k-1} (1 - \delta_i(j)),$$

where $\delta_i(k)$ is the kth element of the vector δ_i. Thus, for a given i, at most one element of the vector $\tilde{\delta}_i = [\tilde{\delta}_i(1), \ldots, \tilde{\delta}_i(N-i+1)]^T$ is 1 and the others are 0. The control signal to be applied at time $t+i-1$ is, thus, given by $\tilde{\delta}_i^T \gamma_i(U_t)$, which could be zero if $\tilde{\delta}_i$ is the all zero vector. With this notation, it follows that

$$\bar{f}_t(i) = \tilde{\delta}_i^T \gamma_i(U_t), \quad i = 1, \ldots, N. \tag{19}$$

To avoid updating the buffer $\bar{f}_t$ with information from packets that were already received in previous time instances, it is useful to look only at the changes between Δ_t and Δ_{t-1}. Towards that end, let $\Delta'_t \in \{0,1\}^{N \times N}$ be the

difference indicator matrix that only indicates the packets that are received at current time t, i.e.,

$$\Delta'_t = \Delta_t - S^{\downarrow}\Delta_{t-1}. \tag{20}$$

In the following, we will show that the number of distinct difference indicator matrices is finite for bounded N, and that the sequence of difference indicator matrices $\{\Delta'_t\}$ is stationary Markov and ergodic. These properties will be helpful in the subsequent analysis.

Lemma 3.1. *The number L of distinct difference indicator matrices is upper bounded by:*

$$L \leq (N+1)\left(1 + \frac{1}{2}N(N+1)\right)^{N-1} \tag{21}$$

with equality if $N = M$, i.e., if the number of packets is equal to the horizon length. △

Proof. See Appendix 1. □

Lemma 3.2. *The sequence of difference indicator matrices $\{\Delta'_t\}$ is stationary Markov and ergodic.* △

Proof. See Appendix 2. □

Example 3.2. *Let us briefly consider the special case without delays, i.e., where we do not allow for late packet arrivals but simply discard late packets. Let us assume that $M = N$, i.e., the number of packets equals the horizon length. In this case, the difference indicator matrices Δ'_t take the form of the all-zero matrix except for the first row, which has J_t consecutive ones starting at the beginning of the row. Here J_t denotes the number of packets received at the current time (excluding any late packets). Thus, the number of distinct difference indicator matrices reduces to $L = M + 1$. Let J_{t-1} denote the number of packets received in the previous time slot. Then the transition probability $p_{J_t|J_{t-1}}$, i.e., the probability of receiving J_t packets conditioned upon receiving J_{t-1} packets in the previous time slot does not depend upon J_{t-1}. Indeed, in this particular case:*

$$p_{J_t|J_{t-1}} = \binom{N}{J_t}(1-p)^{J_t}p^{N-J_t}, \quad J_t = 0,\dots,N. \tag{22}$$

□

We are now in a position to introduce the main technical result of this section, which shows that the sequence of augmented state variables $\{\Xi_t\}$ in (17) and the sequence of difference indicator matrices $\{\Delta'_t\}$ in (20) are *jointly* Markovian and form a Markov jump linear system.

Theorem 3.1. *Let $v_t = \left[w_t^T, \bar{n}_t^T\right]^T$ be the vector containing the external disturbances and quantization noises. Moreover, let $\delta'_i \triangleq \gamma_i(\Delta'_t) \in \mathbb{R}^{N-i+1}$ and let*

$$\tilde{\delta}'_i(k) \triangleq \delta'_i(k)\prod_{j=1}^{k-1}(1 - \delta'_i(j)),$$

where $\delta'_i(k)$ is the kth element of the vector δ'_i. Then, $\{\Xi_t, \Delta'_t\}$ forms a Markov jump linear system with a state recursion that can be written in the following form:

$$\Xi_{t+1} = \mathcal{A}(\Delta'_t)\Xi_t + \mathcal{B}(\Delta'_t)v_t, \tag{23}$$

where the two switching matrices

$$\mathcal{A}(\Delta'_t) \triangleq \begin{bmatrix} \mathcal{A}_1(\Delta'_t) & \mathcal{A}_2(\Delta'_t) \\ \mathcal{A}_3(\Delta'_t) & \mathcal{A}_4(\Delta'_t) \end{bmatrix} \in \mathbb{R}^{(zN+N)\times(zN+N)}$$

and

$$\mathcal{B}(\Delta'_t) \triangleq \begin{bmatrix} \mathcal{B}_1(\Delta'_t) & \mathcal{B}_2(\Delta'_t) \\ \mathcal{B}_3(\Delta'_t) & \mathcal{B}_4(\Delta'_t) \end{bmatrix} \in \mathbb{R}^{(zN+N)\times(z'+N^2)}$$

are given by:

$$\mathcal{A}_1(\Delta'_t) = \begin{bmatrix} A & \mathbf{0}_{z\times z(N-1)} \\ \mathbf{0}_{z(N-1)\times z} & \mathbf{0}_{z(N-1)\times z(N-1)} \end{bmatrix}$$
$$+ \begin{bmatrix} B_1\tilde{\delta}_1'^T E_1 \\ \mathbf{0}_{z(N-1)\times zN} \end{bmatrix} \in \mathbb{R}^{zN\times zN} \tag{24}$$

$$\mathcal{A}_2(\Delta'_t) = \begin{bmatrix} B_1\left(1 - \mathbf{1}_N^T\tilde{\delta}'_1\right)e_1^T S^{\uparrow} \\ \mathbf{0}_{z(N-1)\times N} \end{bmatrix} \in \mathbb{R}^{zN\times N} \tag{25}$$

$$\mathcal{A}_3(\Delta'_t) = \begin{bmatrix} \tilde{\delta}_1'^T E_1 \\ \tilde{\delta}_2'^T E_2 & 0 \\ \vdots \\ \tilde{\delta}_N'^T E_N & 0 & \cdots & 0 \end{bmatrix} \in \mathbb{R}^{N\times zN} \tag{26}$$

$$\mathcal{A}_4(\Delta'_t) = \begin{bmatrix} \left(1 - \mathbf{1}_N^T\tilde{\delta}'_1\right)e_1^T \\ \vdots \\ \left(1 - \mathbf{1}_1^T\tilde{\delta}'_N\right)e_N^T \end{bmatrix} S^{\uparrow} \in \mathbb{R}^{N\times N} \tag{27}$$

and

$$\mathcal{B}_1(\Delta'_t) = \begin{bmatrix} B_2 \\ \mathbf{0}_{z(N-1)\times z'} \end{bmatrix} \in \mathbb{R}^{zN\times z'} \tag{28}$$

$$\mathcal{B}_2(\Delta'_t) = \begin{bmatrix} B_1\tilde{\delta}_1'^T E'_1 \\ \mathbf{0}_{z(N-1)\times N^2} \end{bmatrix} \in \mathbb{R}^{zN\times N^2} \tag{29}$$

$$\mathcal{B}_3(\Delta'_t) = \mathbf{0}_{N\times z'} \tag{30}$$

$$\mathcal{B}_4(\Delta'_t) = \begin{bmatrix} \tilde{\delta}_1'^T E'_1 \\ \tilde{\delta}_2'^T E'_2 & 0 \\ \vdots \\ \tilde{\delta}_N'^T E'_N & 0 & \cdots & 0 \end{bmatrix} \in \mathbb{R}^{N\times N^2}, \tag{31}$$

where $E_i \in \mathbb{R}^{(N-i+1)\times(N-i+1)z}$ and $E_i' \in \mathbb{R}^{(N-i+1)\times(N-i+1)N}$ are given by

$$
E_i = \begin{bmatrix} e_i^T \Psi^{-1}\Gamma & & & \\ & e_{i+1}^T \Psi^{-1}\Gamma & & \\ & & \ddots & \\ & & & e_N^T \Psi^{-1}\Gamma \end{bmatrix} \tag{32}
$$

$$
E_i' = \begin{bmatrix} e_i^T \Psi^{-1} & & & \\ & e_{i+1}^T \Psi^{-1} & & \\ & & \ddots & \\ & & & e_N^T \Psi^{-1} \end{bmatrix}. \tag{33}
$$

$\triangle$

Proof. See Appendix 3. $\qquad\square$

3.2 Stability and steady state system analysis

At time step $t+1$, the switching variable jumps from some particular state, say $\Delta_t' = \Delta$ to some state, say $\Delta_{t+1}' = \tilde{\Delta}$, where it is possible that $\Delta = \tilde{\Delta}$. Let the number of distinct states be L, see Lemma 3.1. Thus, without loss of generality, we can enumerate the L (not necessarily distinct) pairs of system matrices that are associated with the L states by $\{(\mathcal{A}(1),\mathcal{B}(1)),(\mathcal{A}(2),\mathcal{B}(2)),\cdots,(\mathcal{A}(L),\mathcal{B}(L))\}$. We note that even though some of the system matrices might be identical, there is a bijection between the state Δ and the index i of the pair of system matrices. Let $p_{i|j} = \text{Prob}(\Delta_t' = i|\Delta_{t-1}' = j)$, i.e., the transition probability due to jumping from state j to state i, where we note that $p_{i|j}$ is independent of t due to stationarity of the switching sequence, see Lemma 3.2.

In order to assess the stability of the MJLS in (23) and find its stationary first- and second-order moments, we will first introduce some new notation and then directly invoke Proposition 3.37 in [27], which we for completeness[3] include as Lemma 3.3 below.

Define $\mathfrak{A}$ and $\mathfrak{B}$ as in (34) and (35), respectively.

$$
\mathfrak{A} = \begin{bmatrix} p_{1|1}\mathcal{A}(1)\otimes\mathcal{A}(1) & p_{1|2}\mathcal{A}(2)\otimes\mathcal{A}(2) & \cdots & p_{1|L}\mathcal{A}(L)\otimes\mathcal{A}(L) \\ p_{2|1}\mathcal{A}(1)\otimes\mathcal{A}(1) & p_{2|2}\mathcal{A}(2)\otimes\mathcal{A}(2) & \cdots & p_{2|L}\mathcal{A}(L)\otimes\mathcal{A}(L) \\ \vdots & & \ddots & \vdots \\ p_{L|1}\mathcal{A}(1)\otimes\mathcal{A}(1) & p_{L|2}\mathcal{A}(2)\otimes\mathcal{A}(2) & \cdots & p_{L|L}\mathcal{A}(L)\otimes\mathcal{A}(L) \end{bmatrix}. \tag{34}
$$

$$
\mathfrak{B} = \begin{bmatrix} p_{1|1}\mathcal{A}(1) & p_{1|2}\mathcal{A}(2) & \cdots & p_{1|L}\mathcal{A}(L) \\ p_{2|1}\mathcal{A}(1) & p_{2|2}\mathcal{A}(2) & \cdots & p_{2|L}\mathcal{A}(L) \\ \vdots & & \ddots & \\ p_{L|1}\mathcal{A}(1) & p_{L|2}\mathcal{A}(2) & \cdots & p_{L|L}\mathcal{A}(L) \end{bmatrix}. \tag{35}
$$

Moreover, let $q = [q_1,\ldots,q_L] \triangleq (I - \mathfrak{B})^{-1}\psi$, $\psi \triangleq [\psi_1,\ldots,\psi_L]$, where

$$
\psi_j \triangleq \sum_{i=1}^{L} p_{j|i}\mathcal{B}(i)\gamma\pi_i, \tag{36}
$$

where $\gamma = \lim_{t\to\infty}\mathbb{E}[v_t]$ and π_i are the state priors. Define the operators ϕ and $\hat{\phi}$ as follows:

$$
\phi(V_i) \triangleq \begin{bmatrix} \bar{v}_{i,1} \\ \bar{v}_{i,2} \\ \vdots \\ \bar{v}_{i,L} \end{bmatrix}, \quad \hat{\phi}(V) \triangleq \begin{bmatrix} \phi(V_1) \\ \phi(V_2) \\ \vdots \\ \phi(V_L) \end{bmatrix}, \tag{37}
$$

where $\bar{v}_{i,j} \in \mathbb{R}^m$, $V_i = [\bar{v}_{i,1},\ldots,\bar{v}_{i,L}]$ and $V = [V_1,\ldots,V_L]$. Then define

$$
Q \triangleq \hat{\phi}^{-1}((I - \mathfrak{A})^{-1}\hat{\phi}(R(q))), \tag{38}
$$

where

$$
R(q) \triangleq [R_1(q),\ldots,R_L(q)], \tag{39}
$$

$$
\begin{aligned}
R_j(q) \triangleq &\sum_{i=1}^{L} p_{j|i}(\mathcal{B}(i)W\mathcal{B}(i)^*\pi_i \\
&+ \mathcal{A}(i)q_i\gamma^*\mathcal{B}(i)^* + \mathcal{B}(i)\gamma q_i^*\mathcal{A}(i)^*),
\end{aligned} \tag{40}
$$

where

$$
\begin{aligned}
W &= \lim_{t\to\infty}\mathbb{E}[v_t v_t^T] \\
&= \text{diag}\left(\Sigma_w, \sigma_n^2,\ldots,\sigma_n^2\right) \in \mathbb{R}^{(z'+N^2)\times(z'+N^2)}.
\end{aligned}
$$

Definition 3.1 (Definitions 3.8 and 3.32 in [27]). *The MJLS in* (23) *is mean square stable (MSS) if and only if for any initial condition* (Ξ_0, Δ_0') *and ergodic Markov jump sequence* $\{\Delta_t'\}$, *there exists* μ_Ξ *and* Σ_Ξ *such that*

$$
\|\mathbb{E}[\Xi_t] - \mu_\Xi\|_2 \to 0 \quad as \quad t\to\infty, \tag{41}
$$

$$
\|\mathbb{E}[\Xi_t \Xi_t^T] - \Sigma_\Xi\|_2 \to 0 \quad as \quad t\to\infty. \tag{42}
$$

Lemma 3.3 (Proposition 3.37 in [27]). *If* $\sigma_r(\mathfrak{A}) < 1$, *then the system in* (23) *is MSS.*

Remark 1. *Lemma* 3.3 *shows that there is an upper limit on the spectral radius of the matrix* $\mathfrak{A}$ *given by (34) above which the system cannot be stabilized. This matrix* $\mathfrak{A}$ *depends on the packet loss rates via* $p_{i|j}$ *and on the delays via the different switching matrices* $\mathcal{A}(i), i = 1,\ldots,L$.

The MJLS in (23) is in general not stationary. However, as can be observed from Definition 3.1, if the system is MSS then asymptotically as $t\to\infty$, its first- and second-order moments do not depend on t. This observation is formalized in ([27] Theorem 3.33), which we include in part below.

Theorem 3.2 ([27] Theorem 3.33). *If the MJLS in (23) is MSS, then it is also asymptotically wide sense stationary (AWSS) and vice versa.*

Lemma 3.4 (Proposition 3.37 in [27]). *If the MJLS is AWSS, then its first- and second-order asymptotically stationary (non-centralized) moments are given by:*

$$\mu_\Xi \triangleq \lim_{t \to \infty} \mathbb{E}[\, \Xi_t \,] = \sum_{i=1}^{L} q_i, \qquad (43)$$

$$\Sigma_\Xi \triangleq \lim_{t \to \infty} \mathbb{E}[\, \Xi_t \Xi_t^T \,] = \sum_{i=1}^{L} Q_i. \qquad (44)$$

△

In our case, we note that $\gamma = 0$ since the external disturbance w_t and the quantization noise n_t both are zero mean. This implies that $q_i = 0, \forall i$, in (43).

3.3 Assessing the coding rate of the quantizer

Recall from Section 2.4 that the quantized control vector $\tilde{u}_t$ is obtained by quantizing ξ_t to get the quantized vector ξ'_t and then using $\tilde{u}_t = \Psi^{-1} \xi'_t$. The following result establishes an upper bound on the bit rate required for transmitting ξ'_t.

Theorem 3.3. *Let the system (23) be AWSS. Then, for a given horizon length N, the total coding rate using M = N descriptions of the quantized control vector $\tilde{u}_t$, can be upper bounded by R_u:*

$$R_u \triangleq \frac{N}{2} \log_2 \left(\prod_{i=1}^{N} \left(1 + \frac{\sigma^2_{\bar{\xi}(i) | \bar{\xi}(1), \cdots, \bar{\xi}(i-1)}}{\sigma_n^2} \right)^{\frac{1}{i}} \right)$$
$$+ \frac{N}{2} \log_2 \left(\frac{\pi e}{6} \right) + 1, \qquad (45)$$

where $\sigma^2_{\bar{\xi}(i) | \bar{\xi}(1), \cdots, \bar{\xi}(i-1)}$ denotes the conditional variance of $\bar{\xi}(i)$ given $(\bar{\xi}(1), \cdots, \bar{\xi}(i-1))$, and where $\bar{\xi}$ denotes Gaussian random variables with the same first- and second-order moments as the asymptotically stationary moments of ξ_t.

Proof. See Appendix 4. □

Remark 2. *It is straight-forward to extend Theorem 3.3 to the case of $M \leq N$ descriptions by considering M (instead of N) subsets of the vector ξ_t. For example, if $N = 4$ and $M = 3$, one could make the split $\{\xi'_t(1), \xi'_t(2), (\xi'_t(3), \xi'_t(4))\}$, where upon receiving a single description only $\xi'_t(1)$ is recovered, receiving any two descriptions makes it possible to recover $\xi'_t(1)$ and $\xi'_t(2)$, and receiving all $M = 3$ descriptions, the entire vector $\xi'_t(1), \ldots, \xi'_t(4)$ is recovered.* □

Remark 3. *In (45), the conditional variances can easily be obtained using Schur's complement on the covariance matrix Σ_ξ of ξ, which is implicitly given via Σ_Ξ in (44) using (10), that is*
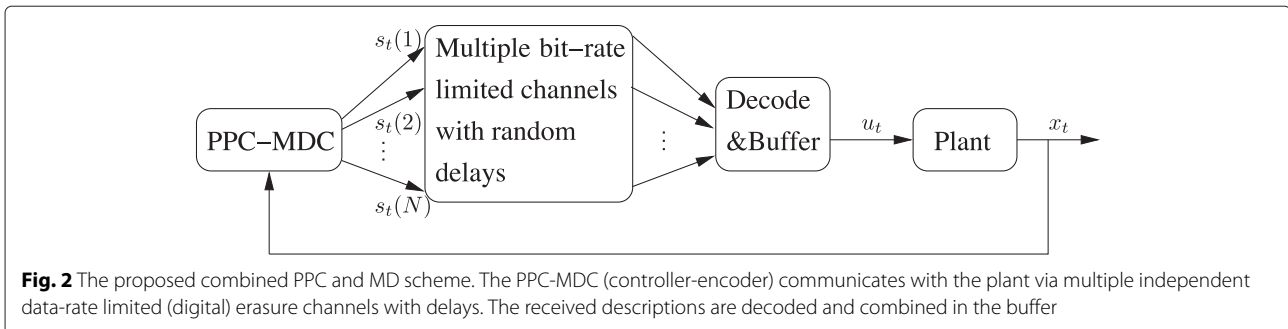
$$\Sigma_\xi = [\Gamma \quad 0] \, \Sigma_\Xi \, [\Gamma \quad 0]^T. \qquad (46)$$

This makes the upper bound on the bit rate in (45) computable and thereby relevant from a practical perspective. Indeed, we show in the simulation study in Section 5, that the bound in (45) is very close to (only 1 bit above) the resulting operational bit rate.[4] □

4 Practical design of the PPC-MDC scheme

In this section, we design a scheme that satisfies the theoretical analysis provided in the previous section. We first present the idea behind our design of MDs and then show the connection to PPC that was sketched in Section 2.5. The proposed scheme is illustrated in Fig. 2.

There are many ways to design MD coding schemes, for example, by use of lattice quantization and index assignment techniques [32, 33], frame expansions followed by quantization [34], oversampling and delta-sigma quantization [35], or layered source coding followed by unequal error protection [36, 37]. In this work, we will be using the latter technique, where the source is decomposed into a number of layers and encoded in such a way that upon reception of say k descriptions, all layers up till the kth layer are revealed [36]. In particular, we rely on a common practical implementation of this strategy, which is based on conventional forward error correction (FEC)

Fig. 2 The proposed combined PPC and MD scheme. The PPC-MDC (controller-encoder) communicates with the plant via multiple independent data-rate limited (digital) erasure channels with delays. The received descriptions are decoded and combined in the buffer

codes that are applied on the individual source layers [37]. It will be shown that there exists a natural connection between PPC and MD based on FEC codes, in the sense that a quantized control vector $\tilde{u}_t$ with $N - 1$ future predictions, can be split into $M \leq N$ "layers", where each layer contains at least one control value. Then, based on these M "layers", we construct M packets $s_t(i), i = 1, \ldots, M$, so that upon reception of any $k \leq M$ packets, the control signals $\tilde{u}_t(1), \ldots, \tilde{u}_t(k)$ can be exactly obtained at the decoder. Thus, as more packets are received, more information about future predicted control signals will become available at the plant input side.

4.1 Forward error correction codes

Consider an (n, k)-erasure code, which as input takes k symbols $y_t^k = (y_t(1), \ldots, y_t(k))$ and outputs n symbols $\tilde{y}_t^n = (\tilde{y}_t(1), \ldots, \tilde{y}_t(n))$, where $n \geq k$, and where $y_t, \tilde{y}_t$ belong to some (yet to be specified) discrete alphabets. With an (n, k)-erasure code, the original k input symbols can be completely recovered using any subset of at least k output symbols. For example, a $(3, 2)$-erasure code may be constructed by letting $\tilde{y}_t(1) = y_t(1), \tilde{y}_t(2) = y_t(2)$, and $\tilde{y}_t(3) = y_t(1)\text{XOR}y_t(2)$, where the XOR operation is performed on, e.g., the binary expansions of $y_t(1)$ and $y_t(2)$. Thus, using any two $\tilde{y}_t(i), \tilde{y}_t(j), i \neq j$ both $y_t(1)$ and $y_t(2)$ may be perfectly recovered. This principle extends to any $n > k$ by using, e.g., erasure codes that are maximum distance separable cf. [38].

4.2 Combining PPC- and FEC-based MDs

For the NCS studied, we apply a sequence of erasure codes on the quantized control vector $\tilde{u}_t = (\tilde{u}_t(1), \ldots, \tilde{u}_t(N))$ in order to obtain M packets. This process is illustrated in Fig. 3 and described in detail below. We first split $\tilde{u}_t$ into M subsets. For example, if $M = N$, the kth set consists of the kth control signal (i.e., $\tilde{u}_t(k)$). In general, we allow several control signals within the same set so that $M < N$. To simplify the exposition and without loss of generality, we will in the following assume that $M = N$. Due to quantization, each distinct $\tilde{u}_t(k)$ can be mapped (entropy coded) to a unique bit stream (codeword), say $b_t(k)$. The bitstream is then split into k non-overlapping sub-bitstreams $b_t^{(i)}(k), i = 1, \ldots, k$ of equal length.[5] These k bitstreams (whose union yields $b_t(k)$) are now considered as input to an (M, k)-erasure code, whose M outputs are denoted by $\phi_t^{(i)}(k), i = 1, \ldots, M$. To summarize, $\tilde{u}_t(1)$ is first mapped to bits $b_t(1)$ and then an $(M, 1)$-erasure code is applied,

which outputs M symbols $\phi_t^{(i)}(1), i = 1, \ldots, M$. Then, the second control signal $\tilde{u}_t(2)$ is mapped to $b_t(2)$. Hereafter, $b_t(2)$ is split into two bitstreams $b_t^{(1)}(2)$ and $b_t^{(2)}(1)$ and an $(M, 2)$-erasure code is applied, which outputs $\phi_t^{(i)}(2), i = 1, \ldots, M$. This process is repeated for all the M control signals.

The M packets $s_t(i), i = 1, \ldots, M$, to be sent over the network at time t are then finally constructed as:

$$s_t(i) = (\phi_t^{(i)}(1), \phi_t^{(i)}(2), \ldots, \phi_t^{(i)}(M)), i = 1, \ldots, M.$$

To further illustrate the usefulness of the above approach, consider the case where $M = 5$ and where the decoder receives three packets say $s_t(2), s_t(3)$, and $s_t(5)$. Then from say $s_t(2)$, we first recover $\phi_t^{(2)}(1)$, which is in fact identical to $\tilde{u}_t(1)$. Then, from say $s_t(2)$ and $s_t(3)$, we then recover $\phi_t^{(2)}(2)$ and $\phi_t^{(3)}(2)$ from which we can decode $\tilde{u}_t(2)$. Finally, using all three received packets, we recover $\phi_t^{(2)}(3), \phi_t^{(3)}(3)$, and $\phi_t^{(5)}(3)$, which can be uniquely decoded to obtain $\tilde{u}_t(3)$.

The foregoing discussion shows that the presence of packet dropouts together with the use of MDs makes the length of the received control packets stochastic and time-varying, while the prediction horizon N is fixed. This aspect makes the analysis of the resultant NCS significantly more involved than that of earlier PPC schemes, as presented in [11]. For example, the number of switching states L, as given by Lemma 3.1, grows exponentially in the horizon length N, whereas in [11] it was enough to consider only two states irrespective of the horizon length.

4.3 Buffering and reconstruction of control signals

At time t, the buffer at the plant input side contains all received packets, which are not older than $t - N + 1$. These will be used for obtaining the current control signal $\hat{u}_t$ giving preference to newer data. For example, assume the buffer is initially empty. Then, for the case of $M = N = 3$, if we at time t receive $s_t(2)$, then clearly we obtain $\hat{u}_t = u_t(1)$. If we then at time $t + 1$ receive $s_{t+1}(1)$ and the delayed packet $s_t(3)$ then we should form $\hat{u}_{t+1} = u_{t+1}(1)$ from $s_{t+1}(1)$ and, thus, simply ignore $s_t(3)$. However, if we now at time $t + 2$, only receive the very late $s_t(1)$, then we recover $\hat{u}_{t+2} = u_t(3)$. Thus, we use the older packets to obtain the control signal. This process is clarified in Table 1 for $M = N = 3$.

$$\tilde{u}_t(k) \longleftrightarrow b_t(k) \longleftrightarrow \underbrace{\left(b_t^{(1)}(k), b_t^{(2)}(k), \ldots, b_t^{(k)}(k) \right)}_{k \text{ input symbols}} \xleftrightarrow{\text{FEC}} \underbrace{\left(\phi_t^{(1)}(k), \phi_t^{(2)}(k), \ldots, \phi_t^{(M)}(k) \right)}_{M \text{ output symbols}}$$

Fig. 3 MPC control vector conversion into MDs. The kth control signal at time t is mapped into M output symbols

Table 1 Control value $\hat{u}_t$ at time t from available buffer contents

$\hat{u}_t$	$s_t(1)$	$s_t(2)$	$s_t(3)$	$s_{t-1}(1)$	$s_{t-1}(2)$	$s_{t-1}(3)$	$s_{t-2}(1)$	$s_{t-2}(2)$	$s_{t-2}(3)$
$u_t(1)$	1	x	x	x	x	x	x	x	x
$u_t(1)$	x	1	x	x	x	x	x	x	x
$u_t(1)$	x	x	1	x	x	x	x	x	x
$u_{t-1}(2)$	0	0	0	1	1	x	x	x	x
$u_{t-1}(2)$	0	0	0	1	x	1	x	x	x
$u_{t-1}(2)$	0	0	0	x	1	1	x	x	x
$u_{t-2}(3)$	0	0	0	x	0	0	1	1	1
$u_{t-2}(3)$	0	0	0	0	x	0	1	1	1
$u_{t-2}(3)$	0	0	0	0	0	x	1	1	1

"1" indicates that the packet is in the buffer and "0" indicates that it is not. "x" indicates that the control value does not depend on the given packet. In all other cases, we set $\hat{u}_t = 0$

4.4 Quantization and coding rates

In order to construct the MDs, we need to split the quantized control vector into individual components. It is therefore not possible to directly quantize the vector ξ_t by use of vector quantization as we have done in our previous work on NCS [11], which did not include the use of MDs. Instead, we will in this work use a scalar quantizer separately along each dimension of the vector ξ_t. Of course, a scalar quantizer is not as efficient as a vector quantizer, but the gap from optimality, which is given by $N/2 \log_2(\pi e/6)$, is included in the upper bound in (52). Interestingly enough, we can still do vector entropy coding by making use of conditional entropy coding. In particular, we first entropy code the first element of the quantized control vector, i.e., $\tilde{u}_t(1)$. This results in an average discrete entropy of $H(\tilde{u}_t(1)|\zeta_t)$. Next, we conditional entropy code the second element $\tilde{u}_t(2)$, which results in an average entropy of $H(\tilde{u}_t(2)|\tilde{u}_t(1), \zeta_t)$. This procedure is repeated for the entire vector $\tilde{u}_t$. The FEC code is now applied on outputs of the conditional entropy coders following the approach described in Section 4.2.

As pointed out in Section 2.4, we transmit the elements of $\tilde{u}_t$ and not those of ξ'_t. The reason for this is that if we receive $\xi'_t(1)$ for the case of $N > 1$, then we are actually not able to reconstruct $\tilde{u}_t(1)$, since $\tilde{u}_t = \Psi^{-1}\xi'_t$. Thus, $\tilde{u}_t(1)$ depends upon the whole vector ξ'_t and not just the first element. Since Ψ^{-1} is fixed and full rank, it simply maps elements one from discrete set into another discrete set. Thus, the coding rate is not affected by sending $\tilde{u}_t(i)$ instead of $\xi'_t(i)$.

The size R (in bits) of a single packet is then on average given by:

$$R = H(\tilde{u}_t(1)|\zeta_t) + \frac{1}{2}H(\tilde{u}_t(2)|\tilde{u}_t(1), \zeta_t) + \cdots$$
$$+ \frac{1}{M}H(\tilde{u}_t(N)|\tilde{u}_t(1), \ldots, \tilde{u}_t(N-1), \zeta_t). \quad (47)$$

Since we have M of these packets, i.e., we have M descriptions, the resulting coding rate is RM.

5 Simulation study

We will now use the analysis and design presented in Sections 3 and 4 in a simulation study in MATLAB.[6]

5.1 System setup

In the state recursion given in (1), we let $z = 5$ and randomly select the system matrix $A \in \mathbb{R}^{z \times z}$ to be

$$A = \begin{bmatrix} -0.1065 & -0.4330 & -0.0006 & -0.8232 & -0.9397 \\ -1.0164 & -1.0668 & -0.1995 & 0.1945 & -0.8169 \\ -1.3309 & 0.8582 & 0.3173 & -1.0053 & -0.3214 \\ -0.5629 & -0.5697 & -0.2112 & -0.2778 & 0.1390 \\ 0.2247 & -0.0090 & -1.3312 & -0.7531 & -0.0929 \end{bmatrix},$$

where the absolute values of the eigenvalues of A are $\{1.9829, 1.2265, 1.2265, 0.9455, 0.9455\}$. Thus, the system is open-loop unstable. We let the external disturbance $w_t \in \mathbb{R}^2$ in (1) be Gaussian distributed with zero mean and covariance matrix $\Sigma_w = I_2$, where I_2 denotes the 2×2 identity matrix. The remaining constants in (1) are set to $B_1 = \mathbf{1}_z$ and $B_2 = [B_1, B_1]$. In these simulations, we have used $T = 4 \times 10^6$ vectors each of dimension $z = 5$ in the sequence $\{x_t\}_{t=0}^{T}$ in (1). x_0 is initialized to the zero vector.

5.2 Cost function

For the cost function in (2), we let $Q = I_5, \lambda = 1/20$, and P is found by (3) and given by:

$$P =$$
$$\begin{bmatrix} 259.5872 & -100.8986 & -76.8526 & -63.0725 & -59.5344 \\ -100.8986 & 46.9687 & 40.4038 & 15.9182 & 15.0465 \\ -76.8526 & 40.4038 & 73.9883 & -10.3694 & -32.3071 \\ -63.0725 & 15.9182 & -10.3694 & 34.9787 & 42.6824 \\ -59.5344 & 15.0465 & -32.3071 & 42.6824 & 68.9741 \end{bmatrix}.$$

5.3 Horizon length and number of packets

We consider the cases where $N = 1, 2, 3$ and compare the proposed scheme that includes multiple descriptions, with the same scheme without multiple descriptions, i.e., that of our earlier work [11]. The two schemes are hereafter referred to as PPC-MDC and PPC, respectively. For the case of PPC-MDC, we let the number of packets M be equal to the horizon length N. For the case of PPC, the entire N-horizon vector is encoded into a single packet. For the case of $N = 1$, the two schemes are identical.

5.4 Network

To simplify the simulations and to be able to compare to existing works on PPC, we will not consider delayed or out-of-order packets. Specifically, if at time t, packet $s_{t-\ell}, \ell > 0$ is received, it is discarded. This means that for the case of $N = M = 3$, the number of jump states reduces to $L = 4$ instead of $L = 196$ as given by Lemma 3.1. Note that even though we do not consider late packet arrivals, control signals can still be applied out of order. To see this, assume that $M = N = 3$, and that all three packets $\{s_t(1), s_t(2), s_t(3)\}$ are received at time t. Then, at time $t+1$, a single packet is received, say $s_{t+1}(1)$, and at time $t + 2$ no packets are received. Then, the control signal $u_{t+1}(1)$ applied at time $t + 1$ is constructed later than the control signal $u_t(3)$ to be applied at time $t + 2$.
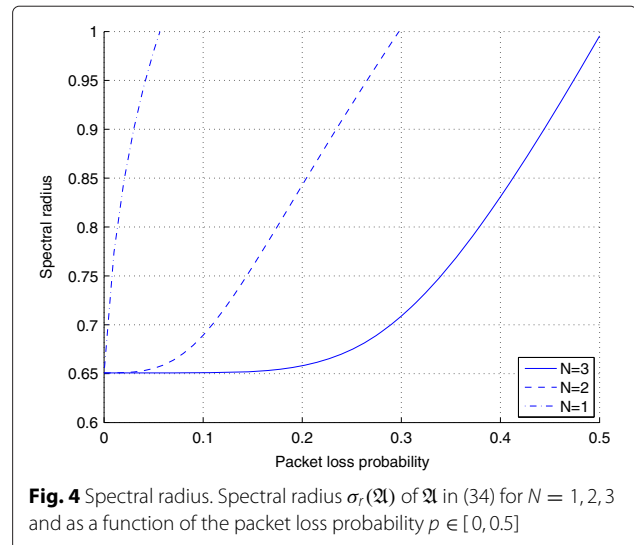
We let the packet losses be mutually independent and identically distributed with probability p that a packet is lost (erased). For this case, the state transition probabilities are given by (22).

5.5 Stability

To assess the stability of the system, we need to compute the spectral radius $\sigma_r(\mathfrak{A})$ of $\mathfrak{A}$ in (34). In order to compute $\mathfrak{A}$ we simply insert the above presented system and network parameters into (24) − (27) and (34). We then obtain the spectral radius by using MATLAB to find the eigenvalue of $\mathfrak{A}$ with the largest absolute value. For the case of $N = 1, 2, 3$, we have in Fig. 4 shown the spectral radius $\sigma_r(\mathfrak{A})$ as a function of the packet loss probability $p \in [0, 0.5]$. According to Lemma 3.3, the MJLS is MSS and AWSS if $\sigma_r(\mathfrak{A}) < 1$. As can be observed from Fig. 4, the MJLS is guaranteed to be MSS for $p < 0.06, p < 0.3$, and $p < 0.5$ for the cases of $N = 1, N = 2$, and $N = 3$, respectively. Thus, choosing a larger horizon brings stability benefits.

5.6 Quantization

Each scalar control value in the control vector $\bar{u}_t$ is quantized using a uniform scalar quantizer with some step size δ. Specifically, for the case of PPC, we simply keep the step size fixed at $\delta = 10$. On the other hand, for the case of PPC-MDC, we need to use a larger step than what is used for PPC, since PPC-MDC introduces redundancy across
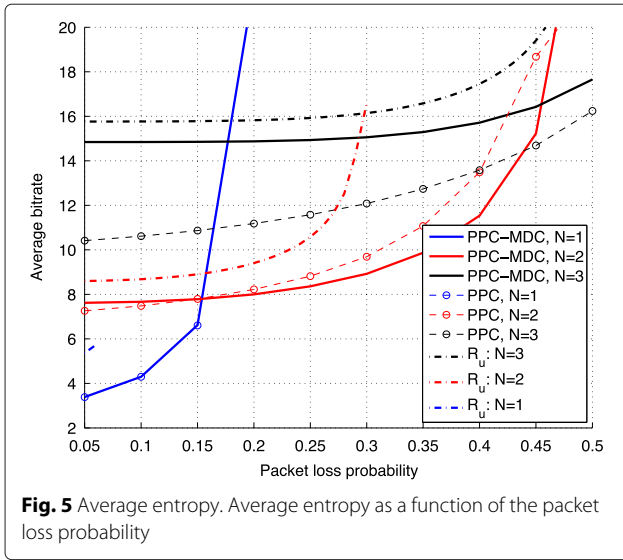

Fig. 4 Spectral radius. Spectral radius $\sigma_r(\mathfrak{A})$ of $\mathfrak{A}$ in (34) for $N = 1, 2, 3$ and as a function of the packet loss probability $p \in [0, 0.5]$

the $M = N$ descriptions. Thus, to keep the bit rate from growing too much as a function of N, we have experimentally found that $\delta = 25N^2$ to be a suitable choice, i.e., $\delta = 25, 100, 225$, for $N = 1, 2, 3$, respectively.

5.7 Bit-rates

In order to compute the upper bound (45) on the bit-rate, we need to estimate the conditional variances $\sigma^2_{\bar{\xi}(i)|\bar{\xi}(1),\cdots,\bar{\xi}(i-1)}$ for $i = 1, \ldots, N$, and the quantization noise variance σ_n^2. To find $\sigma^2_{\bar{\xi}(i)|\bar{\xi}(1),\cdots,\bar{\xi}(i-1)}$, we first find Σ_Ξ in (44) by use of (34) – (40). Then, we use (46) to obtain Σ_ξ from Σ_Ξ, where Σ_ξ is the steady state covariance matrix of ξ_t. Finally, we simply use the Schur complement [39] of Σ_ξ to obtain the desired conditional variances. To estimate the quantization noise variance σ_n^2, we use the relationship $\sigma_n^2 \approx \delta^2/12$, which is exact for a dithered uniform quantizer and a good approximation for a non-dithered scalar uniform quantizer. We have plotted the theoretical upper bound (45) in Fig. 5 as a function of the packet loss probability and for $N = 1, 2, 3$.

To estimate the bit-rate of the quantized control signals, we use (47), which require the computations of discrete conditional entropies. To estimate these conditional entropies, we use a histogram-based entropy estimation on the sequence of discrete (quantized) control signals $\{\tilde{u}_t\}_{t=0}^T$. Specifically, we first estimate $H(\tilde{u}_t(1))$ directly from $\{\tilde{u}_t(1)\}_{t=0}^T$. Then, we estimate $H(\tilde{u}_t(1), \tilde{u}_t(2))$ from $\{\tilde{u}_t(1), \tilde{u}_t(2)\}_{t=0}^T$ and use that $H(\tilde{u}_t(2)|\tilde{u}_t(1)) = H(\tilde{u}_t(1), \tilde{u}_t(2)) - H(\tilde{u}_t(1))$. We obtain $H(\tilde{u}_t(3)|\tilde{u}_t(1), \tilde{u}_t(2)) = H(\tilde{u}_t) - H(\tilde{u}_t(1), \tilde{u}_t(2))$ in a similar way. Finally, these estimates of the conditional entropies are inserted into (47) in order to approximate the resulting operational bit-rate R. The resulting total discrete entropy $R_T = RM$ obtained by adding the entropies

Fig. 5 Average entropy. Average entropy as a function of the packet loss probability

Fig. 6 Average state power. Average state power as a function of the packet loss probability

of the M descriptions is shown in Fig. 5 as a function of the packet loss rate and $M = N$.

It may be noticed in Fig. 5 that the upper bound (45) is approximately 1 bit above the estimate R_T of the operational bit-rates except in the region, where the packet loss rates approach and exceed the critical point, where the system becomes unstable. This excess 1 bit accounts for the theoretical loss of an entropy coder. While we have not applied actual entropy coding, it is well known that the loss of the entropy coder diminishes at moderate to large bit rates.

Note that the multiple descriptions of PPC-MDC have a certain amount of controlled redundancy, and one might therefore expect that the total coding rate for all $M = N$ descriptions would be much greater than what is used for the single description in PPC. However, due to being a closed-loop system, packet losses affect the variance of the input to the quantizer. Consequently, the resulting coding rate for PPC as well as for PPC-MDC also depend upon the packet loss rate.

5.8 Performance
We have measured the performance of the system in terms of the average state power $\frac{1}{T} \sum_{t=1}^{T} \|x_t\|_2^2$. This is shown in Fig. 6. For smaller packet loss rates, the performance of PPC is better than that of PPC-MDC for $N > 1$. This is because the negative impact on the performance due to quantization in PPC-MDC out-weights the impact due to using future predicted control values in PPC in case of packet losses. Recall that the quantizer in PPC-MDC is coarser than that used in PPC. When the packet loss rate is increased, PPC-MDC is often able to apply the most recent control value $\tilde{u}_t(1)$ due to the construction of the MDs. On the other hand, PPC will frequently be applying the future predicted control values $\tilde{u}_t(2)$ and $\tilde{u}_t(3)$ due the
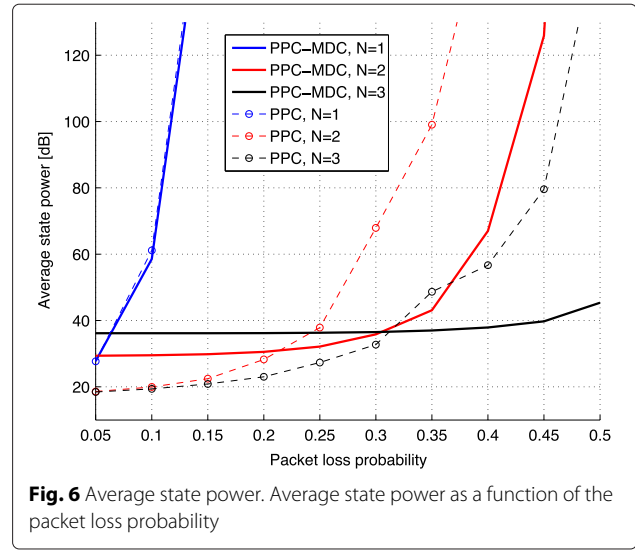
packet dropouts. This leads to a significant performance gain of PPC-MDC at higher packet loss rates.

5.9 Complexity
From the analysis of the MJLS in Section 3, it is not easy to assess the computational burden required, when using the proposed system in practice. In this section, we provide a brief overview of the complexity of the encoder and decoder. The encoder includes the controller, quantizer, entropy coder, and channel (FEC) coder. The decoder includes channel decoder, entropy decoder, buffering, and selection of the control values:

Encoder

1. At any given time, say t, the control vector $\bar{u}_t$ is constructed as in (10) and (13), which amount to a few matrix vector multiplications. The matrices in question are $\Gamma \in \mathbb{R}^{N \times z}$ and $\Psi \in \mathbb{R}^{N \times N}$, where N is the horizon length and z is the state dimension. For many applications, both the horizon length and the state dimension are moderately small.

2. Each scalar element in either the control vector $\bar{u}_t \in \mathbb{R}^N$ or in $\xi_t \in \mathbb{R}^N$ is quantized using a scalar quantizer as described in Section 5.6. This amounts to N simple rounding operations, which can be done efficiently in hardware.

3. The quantized elements are entropy encoded either independently, conditionally, or jointly. In either case, it is done in practice by look-up tables and is therefore of low complexity, i.e., $\mathcal{O}(N)$.

4. The resulting bitstream after entropy coding is converted into M packets by applying M FEC codes, which amounts to matrix-vector multiplications over finite fields [40]. If we use $M = N$ packets, and

thereby split the control vector into N "layers", then the ith layer uses $i \times N$ multiplications due to the (N, i) FEC code. Thus, the total number of multiplications is $N \times (1 + 2 + \cdots + N) = \mathcal{O}(N^3)$.

Decoder

1. At the decoder at time t, all received packets that are no older than time $t - N + 1$, are stored in a buffer. Moreover, all decoded control values that are no older than time $t - N + 1$ time delays are stored in another buffer. Thus, since there can be $M \leq N$ packets in each time slot, the storage complexity is $\mathcal{O}(MN)$.
2. Decoding of received packets involves decoding the FEC code and decoding the entropy code. Decoding the FEC code can be done by, e.g., Gaussian elimination, which has complexity $\mathcal{O}(N^3)$ per layer, and therefore at most $\mathcal{O}(N^4)$ for decoding the entire control vector. Decoding of the entropy code is done by a look-up table and has, thus, complexity $\mathcal{O}(N)$, since the control vector contains N elements.
3. If the decoded control signals are stored in U_t (18), then the selection of the control signal from the buffer can be done as suggested in (19). This includes construction of the vector $\tilde{\delta}_i$ in addition to forming the inner product of $\tilde{\delta}_i$ and the diagonal of U_t indexed by γ_i. The inner product has complexity $\mathcal{O}(N)$.

6 Conclusions

We have shown how to combine multiple description coding with quantized packetized predictive control, in order to get a high degree of robustness towards packet delays and erasures in network control systems. We focused on a digital network located between the controller and the plant input. In our scheme, when any single packet is received, the most recent control value becomes available at the plant input. Moreover, when any J out of M packets are received, the most recent control value and $J - 1$ future predicted control values become available at the plant input. These future-predicted control values can then be applied at time instances, where no packets are received. The key motivation for this design was twofold. From a practical point of view, it was shown that a significant gain over existing packetized predictive control was possible in the range of large packet loss rates. Moreover, from a theoretical point of view, computable guarantees for stability and upper bounds on the operational bit rate could be established. Indeed, a simulation study revealed that the upper bounds on the bit rate was a good indicator for the operational bit rate of the system in the range of packet loss probabilities that were not too close to the region of system instability.

Future works could include source coding in the feedback channel as well as the forward channel, which is a non-trivial extension. Indeed, the design and analysis of optimal joint controller, encoders, and decoders in both forward and backward channels is an open problem even in the absence of erasures and delays. The main difficulty is that the design of the source coder in the forward channel hinges heavily on the design of the source coder in the backward channel as well as on the controller. Another interesting open research direction is to establish lower bounds on the bit rates, which will then make it possible to assess the optimality of the overall system architecture from an information theoretic point of view.

Endnotes
[1]For the case of quantized MPC with fixed-rate quantization and without dithering, it was shown in [41], that the optimal quantized control vector is given by nearest neighbour quantization of ξ_t in (10).
[2]It follows that we require the dither sequence to be known both at the encoder and at the decoder.
[3]We will explicitly make use of (34) − (40) and Lemma 3.3, when assessing the stability of the system in the simulation study in Section 5.
[4]The excess 1 bit is due to the conservative estimate of the loss of the entropy coder, which is characterized by 1 bit.
[5]If they are not of equal length, it is always possible to augment one of the sub-bitstreams with a fixed (known) bit pattern to make them of equal length.
[6]Matlab code to reproduce all results (figures and tables) will be made available online on the authors webpage.
[7]Of course, information about what time instances the packets were received can be learned from past Δ's. However, we are not exploiting this knowledge here.

Appendix 1: Proof of Lemma 3.1
Let us first consider the case $M = N$. In this case, each row of Δ_t can take on $N + 1$ distinct patterns, i.e.,

$$[\overbrace{1 \cdots 1}^{m} \overbrace{0 \cdots 0}^{N-m}], \quad m = 0, \ldots, N,$$

where m describes the number of packets received for the time slot corresponding to that particular row. The first row of Δ_t' is equivalent to the first row of Δ_t. The remaining rows of Δ_t' can each either be the zero vector or any one of the following:

$$[\overbrace{0 \cdots 0}^{m-k} \overbrace{1 \cdots 1}^{k} \overbrace{0 \cdots 0}^{N-m}], m = 0, \ldots, N, k = 1, \ldots, m,$$

where k describes the number of packets received at time t and which contain control signals for that particular row

in the buffer. Thus, the number of distinct patterns for each of these rows are $1 + \sum_{m=1}^{N} m$. Since there is a total of $N-1$ of such rows, the total number of distinct difference matrices is

$$(N+1)\left(1+\sum_{m=1}^{N} m\right)^{N-1} = (N+1)\left(1+\frac{1}{2}N(N+1)\right)^{N-1}.$$

The case of $M < N$ follows easily from the above analysis. In this case, each row of Δ_t can only take on $M+1$ distinct patterns, i.e., the zero vector, or a vector containing the number of consecutive ones corresponding to the number of control values that are recovered, when receiving J out of the M packets, where $J = 1, \ldots, M$. It follows immediately that the number of possible difference indicator matrices is less for $M < N$ compared to $M = N$.

□

Appendix 2: Proof of Lemma 3.2

We first prove ergodicity. Clearly, from the all zero difference indicator matrix, it is possible to get to any other difference indicator matrix in a finite number of steps. Moreover, the probability of not receiving any packets in N consecutive time steps is positively bounded away from zero for any finite N. The all zero difference indicator matrix can therefore be reached in a finite number of steps (from any other difference indicator matrix). Thus, it is possible to jump between any two difference indicator matrices in a finite number of steps. We may therefore view the difference indicator matrices as being the different nodes in a fully connected graph. In this graph, any node can be reached at irregular times. Thus, the nodes are recurrent and aperiodic, which implies that they are are ergodic and the sequence $\{\Delta_t'\}$ of difference indicator matrices is therefore also ergodic.

We now prove the Markovian property. Observe that the matrices in the sequence $\{\Delta_t\}$ are not mutually independent. However, the sequence does satisfy a first-order Markov condition due to the Markov assumption on the data reception, see Section 2.3, i.e.,

$$\Delta_0^{t-1} \leftrightarrow \Delta_t \leftrightarrow \Delta_{t+1}, \quad \forall t, \tag{48}$$

which implies that knowledge of the buffer $\bar{f}_{t-1}'$ does not bring more *useful*[7] information about the buffer $\bar{f}_{t+1}'$ if the buffer $\bar{f}_t'$ is already known. Similarly, it is easy to see that the sequence $\{\Delta_t'\}$ of difference matrices form a Markov chain similar to (48), i.e.,

$$\{\Delta_i'\}_{i=0}^{t-1} \leftrightarrow \Delta_t' \leftrightarrow \Delta_{t+1}', \quad \forall t. \tag{49}$$

Finally, the stationarity of the channel, see Section 2.3, implies that the sequence of difference matrices $\{\Delta_t'\}$ is stationary. This proves the lemma. □

Appendix 3: Proof of Theorem 3.1

In Lemma 3.2, we have established ergodicity and Markov properties of the switching sequence $\{\Delta_t'\}$ as is required by Lemma 3.3. We then need to derive the recursive form for the system evolution, which guarantees that the combined system $\{\Xi_t, \Delta_t'\}$ will be Markovian.

Recall from (19) that $\bar{f}_t(i) = \tilde{\delta}_i^T \gamma_i(U_t)$ for $i = 1, \ldots, N$. However, to avoid updating the buffer with information about packets that was already received in previous time instances, we need to look only at the changes between Δ_t and Δ_{t-1}. Towards that end, let $\delta_i' \triangleq \gamma_i(\Delta_t - S^{\downarrow}\Delta_{t-1})$. Define $\tilde{\delta}_i'$ in a similar manner as $\tilde{\delta}_i$. If δ_i' is the all zero vector for some i, it means that no new control signals to be used at time $t + i - 1$ has been received yet. Thus, the ith element of the buffer is then simply obtained by using older control signals, i.e., $\bar{f}_t(i) = \bar{f}_{t-1}(i+1)$. With this, we obtain the following recursion:

$$\bar{f}_t(i) = \tilde{\delta}_i'^T \gamma_i(U_t) + \left(1 - \mathbf{1}_{N-i+1}^T \tilde{\delta}_i'\right)\bar{f}_{t-1}(i+1). \tag{50}$$

Using (10) and (14), it is possible to write $\gamma_i(U_t)$ as a function of $\bar{x}_t$ and $\bar{n}_t$, that is:

$$\gamma_i(U_t) = [u_t(i), u_{t-1}(i+1), \cdots, u_{t-N+i}(N)]^T$$

$$= \begin{bmatrix} e_i^T \Psi^{-1} \Gamma x_t \\ e_{i+1}^T \Psi^{-1} \Gamma x_{t-1} \\ \ddots \\ e_N^T \Psi^{-1} \Gamma x_{t-N+i} \end{bmatrix} + \begin{bmatrix} e_i^T \Psi^{-1} n_t \\ e_{i+1}^T \Psi^{-1} n_{t-1} \\ \vdots \\ e_N^T \Psi^{-1} n_{t-N+i} \end{bmatrix}.$$

With the above notation, the system state vector recursions can be written as:

$$x_{t+1} = A x_t + B_1(\tilde{\delta}_1'^T \gamma_1(U_t)$$
$$+ (1 - \mathbf{1}_N^T \tilde{\delta}_1')e_1^T S^{\uparrow} \bar{f}_{t-1}) + B_2 w_t. \tag{51}$$

Using that $\Xi_t = \begin{bmatrix} \bar{x}_t \\ \bar{f}_t \end{bmatrix}$ and combining (51) and (50) and using the matrix definitions in (24) – (33) yields (23). This proves the theorem. □

Appendix 4: Proof of Theorem 3.3

In order to provide an upper bound on the required bit rate for transmitting the quantized control vector ξ_t', we assume that the system is designed such that the loop is AWSS. For such a system, the bit rate R of the ECDQ is related to the discrete entropy $H(\xi_t'|\zeta_t)$ of the quantized signal ξ_t', conditioned upon the dither signal ζ_t [31]. That is,

$$H(\xi_t'|\zeta_t) \leq R \leq H(\xi_t'|\zeta_t) + 1/N, \tag{52}$$

where the term $1/N$ is the loss due to using entropy coding on finite dimensional vectors [42]. At this point, we could continue upper bounding $H(\xi_t'|\zeta_t)$, which would then provide an upper bound on the bit rate required for coding the entire control vector. However, recall that we need to

send ξ'_t using M descriptions such that upon receiving any $0 < J \le M$ descriptions, the J control signals $\tilde{u}_t(1), \ldots, \tilde{u}'_t$ can be reliably recovered. This is clearly not possible if ξ'_t is arbitrarily split into M sub-streams having a total bit rate of $H(\xi'_t|\zeta_t)$. In Section 4.2, we introduced a practical scheme for MDs based on forward error correction codes. With this scheme, a description is constructed by concatenating the entire bitstream used for representing the encoded version of $\tilde{u}_t(1)$ with half the bitstream used for $\tilde{u}_t(2)$, one third of the bits allocated for $\tilde{u}_t(3)$, and so on. With such a scheme in mind, we first invoke the chain rule of entropies, in order to expand $H(\xi'_t|\zeta_t)$ in (52) as:

$$H(\xi'_t|\zeta_t) = H(\xi'_t(1)|\zeta_t) + H(\xi'_t(2)|\xi'_t(1), \zeta_t) + \cdots$$
$$+ H(\xi'_t(N)|\xi'_t(1), \ldots, \xi'_t(N-1), \zeta_t). \tag{53}$$

The ith term on the r.h.s. of (53), describes the minimum bit rate required for conditionally encoding $\xi'_t(i)$. With this, and using the MD construction sketched above, the total rate R_T required for all $M = N$ descriptions is given by:

$$R_T \le M\Big(H(\xi'_t(1)|\zeta_t) + \frac{1}{2}H(\xi'_t(2)|\xi'_t(1), \zeta_t) + \cdots$$
$$+ \frac{1}{M}H(\xi'_t(N)|\xi'_t(1), \ldots, \xi'_t(N-1), \zeta_t)\Big) + 1. \tag{54}$$

Since we are using an ECDQ, the discrete entropy of the quantized variables satisfies:

$$H(\xi'_t|\zeta_t) = I(\xi_t; \hat{\xi}_t) \tag{55}$$
$$= I(\xi_t; \xi_t + n_t) \tag{56}$$
$$\le I(\bar{\xi}_t; \bar{\xi}_t + \bar{n}_t) + \mathcal{D}(n_t \| \bar{n}_t), \tag{57}$$

where equality in (55) follows from [31] and where $I(\xi_t; \hat{\xi}_t)$ denotes the mutual information [30] between the input ξ_t and the output $\hat{\xi}_t$ of the ECDQ [31]. In (56), the equality follows by replacing the quantization operation by its additive noise model, which is exact from a statistical point of view [31]. The upper bound in (57) follows from ([43] Lemma 2) by replacing the variables in play by their Gaussian counterparts, i.e., $\bar{\xi}_t$ and $\bar{n}_t$ are Gaussian distributed with the same first- and second moments as ξ_t and n_t, respectively. The Divergence operator $\mathcal{D}(n_t \| \bar{n}_t)$ describes the Kullback-Leibler distance (in bits) between the distribution of the quantization noise n_t to that of a Gaussian distribution [30] and is in our case upper bounded by $\mathcal{D}(n_t \| \bar{n}_t) \le N/2 \log_2(\pi e/6)$. This upper bound is achieved if n_t is uniformly distributed over an N-dimensional cube [44]. We may now proceed by expressing the mutual information in terms of differential entropies provided the latter exists [30]. Thus, we obtain that $I(\bar{\xi}_t; \bar{\xi}_t + \bar{n}_t) = h(\bar{\xi}_t + \bar{n}_t) - h(\bar{n}_t)$. Using the same idea on the conditional estimates $\xi_t(i)|\xi_t(1), \cdots, \xi_t(i-1)$ instead of the entire vector ξ_t leads to:

$$H(\xi'_t(i)|\xi'_t(1), \ldots, \xi'_t(i-1), \zeta_t)$$
$$\le h(\bar{\xi}_t(i)|\bar{\xi}_t(1), \cdots, \bar{\xi}_t(i-1) + \bar{n}_t(i)) - h(\bar{n}_t(i))$$
$$+ 1/2 \log_2(\pi e/6)$$
$$= \frac{1}{2} \log_2 \left(1 + \frac{\sigma^2_{\xi'(i)|\xi'(1), \cdots, \xi'(i-1)}}{\sigma^2_n} \right) + \frac{1}{2} \log_2 \left(\frac{\pi e}{6} \right), \tag{58}$$

where the last equality follows from the definition of differential entropy of a Gaussian variable [30]. Inserting (58) into (54) and (52) yields (45). This completes the proof. $\square$

Author details
[1]Department of Electronic Systems, Aalborg University, Fredrik Bajers Vej 7b, Aalborg, Denmark. [2]Department of Electrical Engineering (EIM-E), Paderborn University, Paderborn, Germany.

References
1. JP Hespanha, P Naghshtabrizi, Y Xu, A survey of recent results in networked control systems. Proc. IEEE. **1**(95), 138–162 (2007)
2. GN Nair, F Fagnani, S Zampieri, RJ Evans, Feedback control under data rate constraints: an overview. Proc. IEEE. **95**(1), 108–137 (2007)
3. S Wong, RW Brockett, Systems with finite communication band- width constraints ii: stabilization with limited information feedback. IEEE Trans. Autom. Control. **44**(5), 1049–1053 (1999)
4. M Trivellato, N Benvenuto, State control in networked control systems under packet drops and limited transmission bandwidth. IEEE Trans. Commun. **58**(2), 611–622 (2010)
5. DE Quevedo, A Ahlén, J Østergaard, Energy efficient state estimation with wireless sensors through the use of predictive power control and coding. IEEE Trans. Signal Process. **58**(9), 4811–4823 (2010)
6. T Wang, Y Zhang, J Qiu, H Gao, Adaptive fuzzy backstepping control for a class of nonlinear systems with sampled and delayed measurements. IEEE Trans. Fuzzy Syst. **23**(2), 302–312 (2015)
7. T Wang, H Gao, J Qiu, A combined adaptive neural network and nonlinear model predictive control for multirate networked industrial process control. IEEE Trans. Neural Netw. Learn. Syst. **27**(2), 416–425 (2016)
8. PL Tang, CW de Silva, Compensation for transmission delays in an Ethernet-based control network using variable-horizon predictive control. IEEE Trans. Contr. Syst. Technol. **14**(4), 707–718 (2006)
9. G Liu, Y Xia, J Chen, D Rees, W Hu, Networked predictive control of systems with random network delays in both forward and feedback channels. IEEE Trans. Ind. Electron. **54**(3), 1282–1297 (2007)
10. DE Quevedo, D Nešić, Input-to-state stability of packetized predictive control over unreliable networks affected by packet-dropouts. IEEE Trans. Autom. Control. **56**(2), 370–375 (2011)
11. DE Quevedo, J Østergaard, Nesić, Packetized predictive control of stochastic systems over digital channels with random packet loss. IEEE Trans. Autom. Control. **56**(12), 2854–2868 (2011)
12. AAE Gamal, TM Cover, Achievable rates for multiple descriptions. IEEE Trans. Inf. Theory. **IT-28**(6), 851–857 (1982)
13. VK Goyal, Multiple description coding: compression meets the network. IEEE Signal Proc. Mag. **18**(5), 74–93 (2001)
14. El Silva, MS Derpich, J Østergaard, An achievable data-rate region subject to a stationary performance constraint for LTI plants. IEEE Trans. Autom. Control. **56**, 1968–1973 (2011)

15. AS Leong, S Dey, GN Nair, Quantized filtering schemes for multi-sensor linear state estimation: stability and performance under high rate quantization. IEEE Trans. Signal Process. **61**(15), 3852–3865 (2013)

16. J Østergaard, DE Quevedo, A Ahlen, in *IEEE Int. Conference on Audio, Speech and Signal Processing*. Predictive power control and multiple-description coding for wireless sensor networks (IEEE, 2009), pp. 2785–2788

17. J Østergaard, DE Quevedo, in *9th IEEE International Conference on Control & Automation*. Multiple descriptions for packetized predictive control over erasure channel (IEEE, Santiago, Chile, 2011)

18. J Østergaard, DE Quevedo, in *IEEE Data Compression Conference*. Multiple description coding for closed loop systems over erasure channels (IEEE, 2013)

19. Z Jin, V Gupta, R Murray, State estimation over packet dropping networks using multiple description coding. IEEE Trans. Autom. Control. **42**(9), 1441–1452 (2006)

20. DE Quevedo, J Østergaard, A Ahlen, A power control and coding formulation for state estimation with wireless sensors. IEEE Trans. Control Syst. Technol. **22**(2), 413–427 (2014)

21. E Peters, DE Quevedo, J Østergaard, Shaped Gaussian dictionaries for quantized networked control systems with correlated dropouts. IEEE Trans. Signal Process. **64**(1), 203–213 (2016)

22. Y Ge, Q Chen, M Jiang, Y Huang, J. Control Sci. Eng. **2013** (2013)

23. Y Wei, J Qiu, H Karimi, M Wang, A new design of H_∞ filtering for continuous-time Markovian jump systems with time-varying delay and partially accessible mode information. Signal Process. **93**(9), 2392–2407 (2013)

24. Y Wei, J Qiu, H Karimi, M Wang, Filtering design for two-dimensional Markovian jump systems with state-delays and deficient mode information. Inf. Sci. **269**(10), 316–331 (2014)

25. J Qiu, Y Wei, H Karimi, New approach to delay-dependent H_∞ control for continuous-time Markovian jump systems with time-varying delay and deficient transition descriptions. J. Frankl. Inst. **352**, 189–215 (2015)

26. Y Fang, KA Loparo, Stochastic stability of jump linear systems. IEEE Trans. Autom. Control. **7**, 1204–1208 (2002)

27. OLV Costa, MD Fragoso, RP Marques, *Discrete-time Markov Jump Linear Systems*. (Springer, 2005)

28. JB Rawlings, DQ Mayne, *Model Predictive Control: Theory And Design*. (Nob Hill Publishing, 2009)

29. DE Quevedo, E Silva, GC Goodwin, Packetized predictive control over erasure channels. Proc. Amer. Contr. Conf, 1003–1008 (2007)

30. TM Cover, JA Thomas, *Elements of Information Theory*, 2nd edn. (Wiley-Interscience, 2006)

31. R Zamir, M Feder, On universal quantization by randomized uniform/lattice quantizers. IEEE Trans. Inform. Theory. **38**(2), 428–436 (1992)

32. VA Vaishampayan, NJA Sloane, SD Servetto, Multiple-description vector quantization with lattice codebooks: design and analysis. IEEE Trans. Inf. Theory. **47**(5), 1718–1734 (2001)

33. J Østergaard, J Jensen, R Heusdens, n-channel entropy-constrained multiple-description lattice vector quantization. IEEE Trans. Inf. Theory. **52**(5), 1956–1973 (2006)

34. J Kovačević, PL Dragotti, VK Goyal, Filter bank frame expansions with erasures. IEEE Trans. Inf. Theory. **48**(6), 1439–1450 (2002)

35. J Østergaard, R Zamir, Multiple description coding by dithered delta-sigma quantization. IEEE Trans. Inf. Theory. **55**(10), 4661–4675 (2009)

36. R Yeung, R Zamir, in *Proceedings of IEEE International Symposium on Information Theory*. Multilevel diversity coding via successive refinement (IEEE, Ulm, Germany, 1996)

37. R Puri, K Ramchandran, Conf. Record Thirty-Third Asilomar Conf Signals Syst. Comput. **1**, 342–346 (1999)

38. RC Singleton, Maximum distance q-nary codes. IEEE Trans. Inf. Theory. **10**(2), 116–118 (1964)

39. S Boyd, L Vandenberghe, *Convex Optimization*. (Cambridge University Press, 2004). Appendix A.5.5

40. J Sundararajan, D Shah, M Medard, S Jakubczak, M Mitzenmacher, J Barros, Network coding meets tcp: theory and implementation. Proc. IEEE. **99**(3), 490–512 (2011)

41. DE Quevedo, GC Goodwin, JA De Doná, Finite constraint set receeding horizon quadratic control. Int. J. Robust Nonlin. Contr. **14**(4), 355–377 (2004)

42. CE Shannon, A mathematical theory of communications. Bell Syst. Tech. J. **27** (1948)

43. MS Derpich, J Østergaard, DE Quevedo. Achieving the quadratic Gaussian rate-distortion function for source uncorrelated distortions, (2008). Electronically available on arXiv.org: http://arxiv.org/pdf/0801.1718.pdf. Accessed Oct 2015

44. R Zamir, M Feder, On lattice quantization noise. IEEE Trans. Inf. Theory. **42**(4), 1152–1159 (1996)