Stochastic Model Predictive Control for Networked Systems with Random Delays and Packet Losses in All Channels

Marijan Palmisano[1], Martin Steinberger[1] and Martin Horn[1]

***Abstract*— A stochastic Model Predictive Control strategy for control systems with communication networks between the sensor node and the controller and between the controller and the actuator node is proposed. Data packets are subject to random delays and packet loss is possible; acknowledgments for received packets are not provided.**

The expected value of a quadratic cost is minimized subject to linear constraints; the set of all initial states for which the resulting optimization problem is guaranteed to be feasible is provided. The state vector of the controlled plant is shown to converge to zero with probability one.

I. Introduction

A typical setup considered in Model Predictive Controller (MPC) design consists of a discrete-time plant and a discrete-time controller where the output of the controller at time step k is the actuating variable of the plant u_k and the input of the controller at time step k is the state vector of the plant x_k. The controller output is determined by minimizing a cost function which depends on current and predicted values of state and actuating variable. For details, see, e.g., [1].

In a networked control system (NCS), controller and plant are connected via communication networks. Data packets are subject to random delays and packet losses which are two of the major challenges in networked control; see [2] or [3].

While MPCs are also developed for NCS where no random delays are assumed to occur between controller and actuator node like in [4], [5], [6] or [7], such delays appear to be particularly challenging. There are numerous works on designing MPCs for NCS with such delays. For example, delayed packets are discarded in [8] which turns delays into packet losses.

In [9], buffers are used to compensate for the randomness of the delays by ensuring that the evolution of the actuating variable applied to the plant is equivalent to the evolution of the actuating variable used in the prediction. Effectively, this replaces the random delay with the maximal delay.

Another possible approach, e.g. presented in [10], [11] and [12], involves predicting future values of state and actuating variable using the nominal plant model, i.e. without including the random effects. This introduces a mismatch between the actual setup and the prediction model which can be expected to deteriorate the performance of the closed control loop.

An approach which avoids such a mismatch is to include the random effects in the prediction model and to minimize the expected value of the cost. It is typically assumed that information about data received by the plant is instantly available to the controller; e.g. (a) via 'error free receipt acknowledgments' in [13], (b) via 'TCP-like protocols' in [14] or (c) implicitly by using such information in the optimization problem solved by the MPC in [15] and [16].

While minimizing the expected value of the cost function results in the best expected performance, it is not always possible to provide 'error free receipt acknowledgments'. In [17], an MPC is presented which minimizes the expected value of the cost function while taking into account that the acknowledgments might be subject to random delays and packet losses. However, this MPC is designed for plants without constraints on the states and inputs; this also applies to the MPCs presented in [18]. A different strategy is presented in [19] where the MPC provides 'correction terms' to a local controller which is connected directly to the plant.

Another approach which avoids a mismatch between the actual setup and the prediction model is minimizing the worst-case cost like in [20] where a robust MPC design method is extended to NCS.

Main Contribution: To our knowledge, no MPC (i) minimizing the expected value of a cost function has yet been developed (ii) for NCS with random delays and packet losses between controller and actuator node where (iii) acknowledgments are also subject to random delays and packet losses and (iv) constraints on the states and inputs have to be satisfied. In this work, one such control strategy is presented for NCS with linear time-invariant (LTI) plants for a quadratic cost function and linear inequality constraints.

Please note that a preprint of this work is available at [22].

Consider the common control problem of designing an MPC for an LTI plant which is connected directly to the controller in order to minimize a quadratic cost while satisfying linear equality constraints. The only difference between this control problem and the control problem considered in this work is that plant and controller are separated, i.e. they can only communicate via communication networks.

The assumptions on the properties of these networks are quite common as well: (i) there are no instantly available acknowledgments about received packets, i.e. 'UDP-like protocols' are used, (ii) packets are timestamped and (iii) random delays and/or packet loss can occur where (iv) delays and number of consecutive packet losses are bounded. The considered random network effects are handled by minimizing the expected value of the cost in order to achieve the best possible expected performance.

Notation: $\succeq, \succ$ denote (semi-) definiteness; $\geq, >$ denote element-wise inequalities. $\otimes$ denotes the Kronecker product.

[1]Marijan Palmisano, Martin Steinberger and Martin Horn are with the Institute of Automation and Control, Graz University of Technology, Inffeldgasse 21/B, 8010 Graz, Austria martin.steinberger@tugraz.at

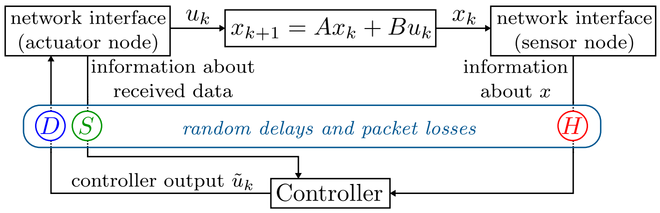

Fig. 1. Considered class of networked control systems; random network effects are represented by the stochastic processes $\mathbf{D}$, $\mathbf{H}$ and $\mathbf{S}$.

I_n is the $n \times n$ identity matrix, $1_{n \times m}$ is a $n \times m$ matrix where all elements are 1, $0_{n \times m}$ is a $n \times m$ matrix where all elements are 0, $1_n = 1_{n \times 1}$ and $0_n = 0_{n \times 1}$. $(a_k)_{i \le k \le j}$ with $i, j \in \mathbb{Z}$ denotes the sequence $(a_i, a_{i+1}, ..., a_j)$. $\mathbb{P}[A]$ denotes the probability of event A and $\mathbb{P}[A|B]$ denotes the conditional probability of event A given event B.

II. Considered Feedback Loop

The considered networked control systems consist of one plant and one controller; data packets transmitted from plant to controller and vice versa are subject to random delays and packet loss is possible. Delays and number of consecutive packet losses are bounded and packets are timestamped. Information about packets received at the actuator node is transmitted to the controller using a network connection and is subject to random delays and packet loss is possible.

This setup is depicted in Fig. 1, where the plant with the state vector $x_k \in \mathbb{R}^n$, the actuating variable $u_k \in \mathbb{R}^m$ and the initial state x_0 is a stabilizable discrete-time LTI system

$$x_{k+1} = Ax_k + Bu_k, \quad k \in \mathbb{N}_0. \tag{1}$$

State and actuating variable have to satisfy linear constraints

$$M_x x_k \le n_x \quad \forall k \in \mathbb{N}_0, \quad M_x \in \mathbb{R}^{a_x \times n}, \ n_x \in \mathbb{R}^{a_x} \tag{2a}$$
$$M_u u_k \le n_u \quad \forall k \in \mathbb{N}_0, \quad M_u \in \mathbb{R}^{a_u \times m}, \ n_u \in \mathbb{R}^{a_u} \tag{2b}$$

where $\{x_k \in \mathbb{R}^n | M_x x_k \le n_x\}$ and $\{u_k \in \mathbb{R}^m | M_u u_k \le n_u\}$ are closed sets containing the origin as an inner point.

A. Network Model

The network used to transmit data from the controller to the actuator node is represented by a discrete-time stochastic process $\mathbf{D} = (D_k)_{k \in \mathbb{N}_0}$ where D_k is the age of the most recently transmitted available packet like in [21] for $D_k \le k$. The first packet is transmitted at time step $k = 0$ and $D_k > k$ means that no packet has been received until time step k. As discussed in detail in [21], the random variable D_k can only increase by 1 each time step and is bounded by $\underline{d} \le D_k \le \overline{d}$ with $\underline{d}, \overline{d} \in \mathbb{N}_0$; for $\delta \in \mathbb{N}$, this can be written as

$$\mathbb{P}\left[D_{k+1} = \delta | D_k = d_k\right] > 0 \ \Rightarrow \ \underline{d} \le \delta \le \min\left(\overline{d}, d_k + 1\right). \tag{3}$$

The other networks are modeled analogously: the network used to transmit data from the sensor node to the controller is represented by $\mathbf{H} = (H_k)_{k \in \mathbb{N}_0}$ with $\underline{h} \le H_k \le \overline{h}$; the network used to transmit data from the actuator node to the controller is represented by $\mathbf{S} = (S_k)_{k \in \mathbb{N}_0}$ with $\underline{s} \le S_k \le \overline{s}$.

While there are no further assumptions on $\mathbf{H}$ and $\mathbf{S}$, $\mathbf{D}$ is assumed to be a homogeneous Markov Process, i.e.

$$\begin{aligned} \mathbb{P}\left[D_{k+1} = d_{k+1} \middle| (D_{\tilde{k}})_{0 \le \tilde{k} \le k} = (d_{\tilde{k}})_{0 \le \tilde{k} \le k}\right] \\ = \mathbb{P}\left[D_{k+1} = d_{k+1} | D_k = d_k\right] \ \forall k \in \mathbb{N}_0, \\ \mathbb{P}\left[D_{k+1} = \beta | D_k = \alpha\right] = \mathbb{P}\left[D_{\tilde{k}+1} = \beta | D_{\tilde{k}} = \alpha\right] \ \forall k, \tilde{k} \in \mathbb{N}_0. \end{aligned} \tag{4}$$

This Markov Process is described by the known initial distribution μ and the known transition probabilities Φ:

$$\mu(d_0) = \mathbb{P}\left[D_0 = d_0\right] > 0 \quad \forall d_0 \in \left[\underline{d}, \overline{d}\right] \tag{5a}$$
$$\Phi(d_k, d_{k+1}) = \mathbb{P}\left[D_{k+1} = d_{k+1} | D_k = d_k\right], \ k \in \mathbb{N}_0; \tag{5b}$$

Φ_n denotes the resulting n-step transition probabilities:

$$\Phi_n(d_k, d_{k+n}) = \mathbb{P}\left[D_{k+n} = d_{k+n} | D_k = d_k\right], \ k, n \in \mathbb{N}_0. \tag{6}$$

In order to avoid case distinctions, $\overline{d} > \underline{d}$ is assumed, i.e. controller outputs are actually subject to random effects.

B. Transmitted Data Packets

1) Sensor Node: Let K_h be the first time step at which a packet from the sensor node is received at the controller. Once $k \ge K_h$, the controller has access to K_h and $(H_{\tilde{k}})_{K_h \le \tilde{k} \le k}$. By transmitting $(x_{\tilde{k}})_{\min(0, k - \overline{h} + \underline{h}) \le \tilde{k} \le k}$ at each time step $k \ge 0$, it is ensured that the controller has access to the entire sequence $(x_{\tilde{k}})_{0 \le \tilde{k} \le k - H_k}$ at each time step $k \ge K_h$.

2) Controller: The controller transmits the prediction

$$\tilde{u}_k = \begin{bmatrix} \tilde{u}_k^{(\overline{d})^T} & \tilde{u}_k^{(\overline{d}-1)^T} & \dots & \tilde{u}_k^{(\underline{d})^T} \end{bmatrix}^T \in \mathbb{R}^{\tilde{m}} \tag{7}$$

where $\tilde{m} = (\overline{d} - \underline{d} + 1)m$ to the actuator node at all time steps $k \ge 0$. Let K_d be the first time step at which a data packet is received at the actuator node. Once $k \ge K_d$, u_k is determined from the most recent available packet via

$$u_k = \tilde{u}_{k - D_k}^{(D_k)}. \tag{8}$$

While $k < K_d$, $u_k = 0$ is applied which can also be written as (8) by defining $\tilde{u}_k = 0 \ \forall \, k < 0$.

Due to (8), $\tilde{u}_k^{(d)}$ in (7) is the value applied as u_{k+d} if $\tilde{u}_k$ is the most recent available controller output at time step $k+d$.

3) Actuator Node: Once $k \ge K_d$, the actuator node transmits $\left(D_{\tilde{k}}\right)_{\max(K_d, k - \overline{s} + \underline{s}) \le \tilde{k} \le k}$; empty packets are transmitted while $k < K_d$. Let K_s be the first time step at which a packet from the actuator node is received at the controller and let $K_{sd} \ge K_s$ be the first time step at which a non-empty data packet is received. Once $k \ge K_s$, the controller has access to K_s and $(S_{\tilde{k}})_{K_s \le \tilde{k} \le k}$. Once $k \ge K_{sd}$, the controller has access to K_{sd}, K_d and the entire sequence $\left(D_{\tilde{k}}\right)_{K_d \le \tilde{k} \le k - S_k}$. While $K_s \le k < K_{sd}$, the controller has the information that $K_d > k - S_k$. Note that this implies that $\left(u_{\tilde{k}}\right)_{0 \le \tilde{k} \le k - S_k}$ can be determined by the controller for all $k \ge K_s$.

4) Information Set: All data available to the controller at time step k assuming that all data is maintained is referred to as information set $\mathcal{I}_k$. This implies $\mathcal{I}_k \subseteq \mathcal{I}_{k+i} \, \forall i \ge 0$. The amount of storage required for maintaining this set is not bounded so it is not actually maintained by the controller. While the controller is designed assuming that $\mathcal{I}_k$ is available at time step k, maintaining a finite subset of $\mathcal{I}_k$ is sufficient for implementing the resulting control law.

III. Model Predictive Controller

Consider an MPC where an optimization problem

$$\min_{(\hat{u}_k^{(i)})_{\underline{d}\le i\le N-1}} \mathbb{E}\left\{\sum_{i=0}^{\infty}\left(x_{k+i}^T Q x_{k+i} + u_{k+i}^T R u_{k+i}\right)\middle| \mathcal{I}_k\right\} \quad (9a)$$

$$s.t.\ u_{k+i} = \begin{cases}\tilde{u}_{k+i-D_{k+i}}^{(D_{k+i})} & i < D_{k+i}\\ \hat{u}_k^{(i)} & D_{k+i}\le i\le N-1\\ \kappa_k^{(i)} & i\ge N\end{cases} \quad (9b)$$

$$M_u u_{k+i} \le n_u,\ M_x x_{k+i}\le n_x \quad \forall i\in[0,N-1] \quad (9c)$$

$$x_{k+N}\in\mathcal{X}_N(\mathcal{I}_k) \quad (9d)$$

with the finite optimization horizon $N\in\mathbb{N}$ is solved at each time step $k\ge K_h$ where $R=R^T\succ 0$ and $Q=C^TC\succeq 0$ with some C such that (A,C) is detectable.

Once $k\ge K_h$, the controller output (7) is chosen via $\tilde{u}_k^{(d)}=\hat{u}_k^{(d)}$ which requires that $N\ge\overline{d}+1$. While $k<K_h$, i.e. while the controller has not received any information about the state vector, the controller output is set to zero:

$$\tilde{u}_k = \begin{cases}\begin{bmatrix}\hat{u}_k^{(\overline{d})^T} & \hat{u}_k^{(\overline{d}-1)^T} & \dots & \hat{u}_k^{(\underline{d})^T}\end{bmatrix} & k\ge K_h\\ 0 & k<K_h.\end{cases} \quad (10)$$

In (9), the conditional expected value of a cost given all information available at time step k is minimized in order to achieve the best possible expected performance. The constraint (9b) takes into account that (a) u_{k+i} is chosen from previous controller outputs while no packet transmitted at time step k or later is received at the actuator node, (b) the optimization variables are the actuating variables while $D_{k+i}\le i\le N-1$ and (c) a control law $u_{k+i}=\kappa_k^{(i)}$ has to be chosen for $i\ge N$ in order to evaluate the cost function.

Theorem 1: Consider the feedback loop presented in Section II. For any $N\ge\overline{d}+1$, a stabilizing control law $u_{k+i}=\kappa_k^{(i)}\ \forall i\ge N$, a terminal constraint set $\mathcal{X}_N(\mathcal{I}_k)$ and a non-empty set $\mathcal{X}_0=\left\{x_0\in\mathbb{R}^n\middle|M_0x_0\le n_0\right\}$ exist such that choosing the controller output via (10) for all $k\ge 0$ where $(\hat{u}_k^{(i)})_{\underline{d}\le i\le\overline{d}}$ is determined by solving (9) for $k\ge K_h$ yields the following properties if the initial state satisfies $x_0\in\mathcal{X}_0$:

a) (9) is feasible at all time steps $k\ge K_h$.
b) The constraints (2) are guaranteed to be satisfied.
c) The state vector converges to zero with probability one, i.e. $\mathbb{P}\big[\lim_{k\to\infty}x_k=0\big]=1$.
d) There exists an optimization problem that (i) is equivalent to (9) in the sense that the resulting optimal value for $\hat{u}_k=\begin{bmatrix}\hat{u}_k^{(N-1)^T} & \dots & \hat{u}_k^{(\underline{d})^T}\end{bmatrix}^T$ is the same which (ii) can be written in the form

$$\min_{\hat{u}_k}\left(\hat{u}_k^TV_k\hat{u}_k+\hat{u}_k^Tv_k\right)\quad s.t.\quad W_k\hat{u}_k\le w_k \quad (11)$$

where V_k, v_k, W_k and w_k can be computed from (a) values that can be computed offline in advance and (b) probabilities and an extended state vector.

IV. Proof for Theorem 1

Remark: Due to the page limit, the lemmas used in the proof for Theorem 1 are presented without proof in this work; these proofs can be found in [22].

A. Stabilizing Control Law and Terminal Constraint Set

1) Stabilizing Control Law: Consider a controller with the input x_k and the output u_k and consider the cost function $J=\sum_{k=0}^{\infty}\left(x_k^TQx_k+u_k^TRu_k\right)$. Minimizing J with respect to $(u_k)_{k\ge 0}$ in the unconstrained case yields the well-known Linear Quadratic Regulator (LQR), i.e. $u_k=-Lx_k$ with $L=(R+B^TPB)^{-1}B^TPA$ resulting in $J=x_0^TPx_0$ where $P\succ 0$ is the solution of the Discrete-Time Algebraic Riccati Equation $P=Q+A^TPA-A^TPB\left(R+B^TPB\right)^{-1}B^TPA$.

The stabilizing control law $\kappa_k^{(i)}$ should be chosen such that $u_{k+i}=\kappa_k^{(i)}\ \forall i\ge N$ can actually be achieved by choosing the controller outputs $(\tilde{u}_{k+i})_{i\ge 0}$ accordingly. Otherwise, a mismatch between the actual setup and the prediction with (9b) is introduced. This issue occurs for $\kappa_k^{(i)}=-Lx_{k+i}$.

When choosing $\kappa_k^{(i)}=-L\mathbb{E}\left\{x_{k+i}|\mathcal{I}_k\right\}$ instead, one sequence $(u_{k+i})_{i\ge N}$ has to stabilize the plant for all possible x_{k+N} resulting in unnecessarily conservative constraints.

Definition 1 (Proposed Stabilizing Control Law): To avoid both of the above issues, the stabilizing control law

$$\kappa_k^{(i)} = -Lx_{k+i}+L\sum_{j=-\hat{H}_k^{(i)}}^{\overline{d}-1}A^{i-1-j}B\Big(u_{k+j}-\mathbb{E}\left\{u_{k+j}|\mathcal{I}_k\right\}\Big)$$

$$\tilde{H}_k = \begin{cases}H_k & k<K_s \quad (12a)\\ \min(H_k,S_k-1) & k\ge K_s \quad (12b)\end{cases}$$

$$\hat{H}_k^{(i)} = \min(\overline{h}+N-i-1,\overline{s}+N-i-2,\tilde{H}_k) \quad (12c)$$

is proposed. As shown in [22], $(\tilde{u}_{k+i})_{i\ge 1}$ can be chosen, i.e. $\tilde{u}_{k+i}$ can be computed from $\mathcal{I}_{k+i}$ for all $i\ge 1$, such that (9b) is actually applied to the plant for (12a). Since

$$\kappa_k^{(i)}=-Lx_{k+i}\ \ \forall\ i\ge\hat{N}=\overline{d}+N-1+\min(\overline{h},\overline{s}-1) \quad (13)$$

since $-\hat{H}_k^{(i)}\ge\overline{d}$ for all $i\ge\hat{N}$ due to (12c), $\lim_{k\to\infty}x_k=0$ certainly holds for any x_{k+N} if $u_{k+i}=\kappa_k^{(i)}$ for all $i\ge N$.

2) Terminal Constraint Set: If $u_{k+i}=-Lx_{k+i}\ \forall i\ge 0$, $M_xx_{k+i}\le n_x$ and $M_uu_{k+i}\le n_u$ hold for all $i\ge 0$ if

$$\begin{bmatrix}M_x^T & -(M_uL)^T\end{bmatrix}^T(A-BL)^ix_k\le\begin{bmatrix}n_x^T & n_u^T\end{bmatrix}^T. \quad (14)$$

Using Algorithm 3.2 in [23], the set of all x_k for which (14) holds can be written as $M_Nx_k\le n_N$.

The value of u_{k+i} for $D_{k+i}=d_{k+i}$ for $i\le N-1$ used for prediction at time step k according to (9b) is denoted by

$$\bar{u}_k^{(i)}(d_{k+i}) = \begin{cases}\tilde{u}_{k+i-d_{k+i}}^{(d_{k+i})} & i<d_{k+i}\\ \hat{u}_k^{(i)} & d_{k+i}\le i\le N-1\end{cases} \quad (15)$$

in order to distinguish it from u_{k+i} used for prediction at time step $k+1$ which is denoted by $\bar{u}_{k+1}^{(i-1)}(d_{k+i})$.

The sequence $(D_{k+i})_{-\tilde{H}_k\le i\le\overline{d}-1}$ is denoted by $\vec{D}_k$; the set of all possible sequences $\vec{D}_k$ given $\mathcal{I}_k$ is denoted as $\mathcal{D}_k$, i.e.

$$\vec{D}_k=(D_{k+i})_{-\tilde{H}_k\le i\le\overline{d}-1} \quad (16a)$$

$$\vec{d}_k\in\mathcal{D}_k\ \Leftrightarrow\ \mathbb{P}[\vec{D}_k=\vec{d}_k|\mathcal{I}_k]>0. \quad (16b)$$

Definition 2 (Proposed Terminal Constraint Set): $\mathcal{X}_N(\mathcal{I}_k)$ is chosen as the set of all x_{k+N} for which

$$M_u u_{k+i} \leq n_u,\ M_x x_{k+i} \leq n_x \quad \forall i \in [N, \hat{N}-1] \tag{17a}$$

$$M_N x_{k+\hat{N}} \leq n_N \tag{17b}$$

hold if u_{k+i} can be written in the form

$$u_{k+i} = \sum\nolimits_{\vec{d}_k \in \mathcal{D}_k} p_{k,i}(\vec{d}_k) \bar{\kappa}_k^{(i)}(\vec{d}_k), \tag{18a}$$

$$\bar{\kappa}_k^{(i)}(\vec{d}_k) = -L x_{k+i} + L \sum\nolimits_{j=-\hat{H}_k^{(i)}}^{\overline{d}-1} A^{i-1-j} B \left(\bar{u}_k^{(j)}(D_{k+j}) - \bar{u}_k^{(j)}(d_{k+j})\right) \tag{18b}$$

for all $i \in [N, \hat{N}-1]$ where the scalar factors $p_{k,i}(\vec{d}_k)$ satisfy

$$\sum\nolimits_{\vec{d}_k \in \mathcal{D}_k} p_{k,i}(\vec{d}_k) = 1, \quad p_{k,i}(\vec{d}_k) \geq 0 \ \ \forall \vec{d}_k \in \mathcal{D}_k. \tag{18c}$$

Due to (12a), $u_{k+i} = \kappa_k^{(i)}$ can be written in the form (18) with $p_{k,i}(\vec{d}_k) = \mathbb{P}\big[\vec{D}_k = \vec{d}_k \big| \mathcal{I}_k\big]$. Therefore, (9d) implies that (17) holds for (9b). Furthermore, (17) and (13) imply $M_u u_{k+i} \leq n_u$ and $M_x x_{k+i} \leq n_x$ for all $i \geq N$ for (9b).

In other words, $\mathcal{X}_N(\mathcal{I}_k)$ is chosen such that (9d) and (9c) imply that $M_u u_{k+i} \leq n_u$ and $M_x x_{k+i} \leq n_x$ are guaranteed to hold for all $i \geq 0$ if (9b) holds.

B. Feasibility and Admissible Initial States

Lemma 2 (Recursive Feasibility): If (a) $\kappa_k^{(i)}$ and $\mathcal{X}_N(\mathcal{I}_k)$ are chosen according to Definition 1 and 2 and (b) (9) is feasible at some time step $k \geq K_h$ then (9) is also feasible at all time steps $\tilde{k} \geq k$.

Definition 3 (Admissible Initial State): An initial state x_0 is considered admissible if $M_x A^k x_0 \leq n_x$ is guaranteed for all $k \in [0, K_h - 1]$ and that (9) is feasible at time step K_h.

Lemma 3 (Satisfied Constraints): If (a) $\kappa_k^{(i)}$ and $\mathcal{X}_N(\mathcal{I}_k)$ are chosen according to Definition 1 and 2 and (b) x_0 is admissible then the constraints (2) are satisfied.

Lemma 4: Assume that $\kappa_k^{(i)}$ and $\mathcal{X}_N(\mathcal{I}_k)$ are chosen according to Definition 1 and 2 and let $\tilde{\mathcal{X}}_0(K_h)$ be the set of all $x_0 \in \mathbb{R}^n$ for which $(u_{K_h+i})_{\overline{d} \leq i \leq N-1}$ exists such that

$$M_u u_k \leq n_u \ \forall k \in [K_h + \overline{d}, K_h + N - 1], \tag{19}$$

$$M_x x_k \leq n_x \ \forall k \in [0, K_h + N - 1], \quad M_N x_{K_h+N} \leq n_N$$

if $u_k = 0$ for all $k < K_h + \overline{d}$. Then (a) x_0 is admissible if and only if $x_0 \in \tilde{\mathcal{X}}_0(K_h)$ with (b) $\tilde{\mathcal{X}}_0(K_h)$ of the form

$$\tilde{\mathcal{X}}_0(K_h) = \left\{x_0 \in \mathbb{R}^n \middle| M_{0,K_h} x_0 \leq n_{0,K_h}\right\}. \tag{20}$$

Since K_h is not known in advance, x_0 is admissible if and only if $x_0 \in \tilde{\mathcal{X}}_0(K_h)$ for all $K_h \in [\underline{h}, \overline{h}]$.

Definition 4 (Set of Admissible Initial States): The set of admissible initial states $\mathcal{X}_0$ is chosen as

$$\mathcal{X}_0 = \left\{x_0 \in \mathbb{R}^n \middle| M_0 x_0 \leq n_0\right\} \tag{21}$$

with $M_0^T = \begin{bmatrix} M_{0,\underline{h}}^T & \cdots & M_{0,\overline{h}}^T \end{bmatrix}$ and $n_0^T = \begin{bmatrix} n_{0,\underline{h}}^T & \cdots & n_{0,\overline{h}}^T \end{bmatrix}$ from (20) which is the set of all admissible initial states.

Since $\mathcal{X}_0$ is of the form (21) and since x_0 is admissible if $x_0 \in \mathcal{X}_0$, property $a)$ in Theorem 1 holds due to Definition 3 and Lemma 2; property $b)$ holds due to Lemma 3.

C. Convergence

Lemma 5 (Convergence with Probability One): If $\kappa_k^{(i)}$ and $\mathcal{X}_N(\mathcal{I}_k)$ are chosen according to Definition 1 and 2 then $\mathbb{P}\big[\lim_{k\to\infty} x_k = 0\big] = 1$ if x_0 is admissible.

Due to Lemma 5, property $c)$ in Theorem 1 also holds.

D. Rewritten Control Law

In order to complete the proof for Theorem 1, it remains to be shown that property (d) holds for the optimization problem (9) with $\kappa_k^{(i)}$ and $\mathcal{X}_N(\mathcal{I}_k)$ from Definition 1 and 2. This optimization problem is rewritten by (i) inserting the equality constraint (9b) in the cost function and in the remaining constraints, by (ii) rewriting the terminal constraint (9d) as inequality constraints and by (iii) evaluating the expected value in (9a) which involves computing the probability distribution of $\vec{D}_k$ from (16a).

1) Rewritten Probabilities:

Lemma 6 (Rewritten Probability Distribution of $\vec{D}_k$): The probability distribution of $\vec{D}_k$ from (16a) given $\mathcal{I}_k$ can be written as $\mathbb{P}\big[\vec{D}_k = \vec{d}_k \big| \mathcal{I}_k\big] = P_k(d_{k-\tilde{H}_k}) \vec{P}(\tilde{H}_k, \vec{d}_k)$ where

$$P_k(d_{k-\tilde{H}_k}) = \mathbb{P}\left[D_{k-\tilde{H}_k} = d_{k-\tilde{H}_k} \middle| \mathcal{I}_k\right] \tag{22a}$$

$$\vec{P}(\tilde{H}_k, \vec{d}_k) = \prod\nolimits_{i=-\tilde{H}_k}^{\overline{d}-2} \Phi(d_{k+i}, d_{k+i+1}). \tag{22b}$$

Lemma 7 (Possible Sequences): Let $\tilde{\mathcal{D}}_k \subseteq [\underline{d}, \overline{d}]$ denote the set of all possible values of $D_{k-\tilde{H}_k}$ given $\mathcal{I}_k$, i.e. $\tilde{\mathcal{D}}_k = \left\{d \in \mathbb{N}_0 \middle| P_k(d) > 0\right\}$. Then, the set $\mathcal{D}_k$ of possible sequences from (16b) is the set of all $(d_{k+i})_{-\tilde{H}_k \leq i \leq \overline{d}-1}$ with $d_{k-\tilde{H}_k} \in \tilde{\mathcal{D}}_k$ for which (5b) yields a non-zero probability.

Lemma 8 (Expected Values): Let $f(\vec{D}_k)$ be some function of $\vec{D}_k$ and let $\hat{\mathcal{D}}_k(\tilde{\mathcal{D}}_k, \delta) \subseteq \mathcal{D}_k(\tilde{\mathcal{D}}_k)$ be the set of all $\vec{d}_k \in \mathcal{D}_k(\tilde{\mathcal{D}}_k)$ with $d_{k-\tilde{H}_k} = \delta$. Then,

$$\mathbb{E}\big\{f(\vec{D}_k) \big| \mathcal{I}_k\big\} = \sum_{\delta \in \tilde{\mathcal{D}}_k} P_k(\delta) \sum_{\vec{d}_k \in \hat{\mathcal{D}}_k(\tilde{\mathcal{D}}_k, \delta)} \vec{P}(\tilde{h}_k, \vec{d}_k) f(\vec{d}_k). \tag{23}$$

Lemma 9 (Relevant Information): Consider the finite set

$$\tilde{\mathcal{I}}_k = \begin{cases} \big\{S_k = s_k, \tilde{H}_k = \tilde{h}_k,\ D_{k-s_k} = d_{k-s_k}, & \\ D_{k+i} \in \Theta_{k,i} \,\forall\, i \in [-s_k + 1, -\tilde{h}_k - 1]\big\} & k \geq K_{sd} \\ \big\{\tilde{H}_k = \tilde{h}_k,\ D_{k+i} \in \Theta_{k,i} & \\ \qquad \forall\, i \in [\underline{d} - k, -\tilde{h}_k - 1]\big\} & k < K_s \\ \big\{S_k = s_k, \tilde{H}_k = \tilde{h}_k,\ D_{k+i} \in \hat{\Theta}_{k,i} & \\ \qquad \forall\, i \in [\underline{d} - k, -\tilde{h}_k - 1]\big\} & \text{otherwise} \end{cases}$$

where s_k and $\tilde{h}_k$ are known realizations of S_k and $\tilde{H}_k$ and

$$\Theta_{k,i} = \left\{d_{k+i} \in [\underline{d}, \overline{d}] \,\middle|\, x_{k+i+1} = A x_{k+i} + B \tilde{u}_{k+i-d_{k+i}}^{(d_{k+i})}\right\}$$

$$\hat{\Theta}_{k,i} = \begin{cases} \Theta_{k,i} & -S_k < i \leq -H_k - 1 \\ \left\{d \in [\underline{d}, \overline{d}] \,\middle|\, d \geq k + i\right\} & -H_k - 1 < i \leq -S_k \\ \left\{d \in \Theta_{k,i} \,\middle|\, d \geq k + i\right\} & \text{otherwise.} \end{cases}$$

This set is contained in $\mathcal{I}_k$ and contains all information that is relevant for evaluating (22a) in the sense that

$$P_k(\delta) = \mathbb{P}\left[D_{k-\tilde{H}_k} = \delta \middle| \tilde{\mathcal{I}}_k\right] \quad \forall\, k \geq K_h. \tag{24}$$

While computing $P_k(\delta)$ from $\tilde{\mathcal{I}}_k$ is not computationally expensive, the resulting expressions are quite lengthy and are therefore omitted in this work; details about computing $P_k(\delta)$ from $\tilde{\mathcal{I}}_k$ can be found in [22].

2) Extended State Vector: For $k \geq K_h$, x_{k-H_k} is known and $(u_{k+i})_{-H_k \leq i \leq -S_k}$ can be determined via (8). Therefore, $x_{k-\tilde{H}_k}$ can be computed from $\mathcal{I}_k$ so the extended state vector $\hat{x}_k = \begin{bmatrix} x_{k-\tilde{H}_k}^T & \tilde{u}_{k-1}^T & \dots & \tilde{u}_{k-\overline{d}-\overline{h}}^T \end{bmatrix} \in \mathbb{R}^{\hat{n}}$ with $\hat{n} = n + \tilde{m}(\overline{d} + \overline{h})$ is known at time step $k \geq K_h$.

As shown in detail in [22], the extended state vector $\hat{x}_k$ and the vector of optimization variables $\hat{u}_k$ can be used to write x_{k+i} with $-\tilde{H}_k \leq i \leq N$ in the form

$$x_{k+i} = \hat{A}(i, \tilde{H}_k, \vec{D}_k)\hat{x}_k + \hat{B}(i, \tilde{H}_k, \vec{D}_k)\hat{u}_k, \tag{25}$$

to write u_{k+i} with $-\tilde{H}_k \leq i \leq N-1$ in the form

$$u_{k+i} = \hat{A}_u(i, \tilde{H}_k, \vec{D}_k)\hat{x}_k + \hat{B}_u(i, \tilde{H}_k, \vec{D}_k)\hat{u}_k \tag{26}$$

and to write the stabilizing control law in the form

$$\begin{aligned} \bar{\kappa}_k^{(i)}(\vec{d}_k) &= -Lx_{k+i} \\ &+ \hat{L}_x(i, \tilde{H}_k, \vec{D}_k, \vec{d}_k)\hat{x}_k + \hat{L}_u(i, \tilde{H}_k, \vec{D}_k, \vec{d}_k)\hat{u}_k. \end{aligned} \tag{27}$$

These results are then used to write all constraints in (9) that depend on the optimization variables $\hat{u}_k$ in the form

$$\vec{M}_x(\tilde{H}_k, \tilde{\mathcal{D}}_k)\hat{x}_k + \vec{M}_u(\tilde{H}_k, \tilde{\mathcal{D}}_k)\hat{u}_k \leq \vec{n}(\tilde{\mathcal{D}}_k) \tag{28}$$

and to write the cost function in (9) in the form

$$\begin{aligned} &\sum\nolimits_{\delta_1 \in \tilde{\mathcal{D}}_k} \sum\nolimits_{\delta_2 \in \tilde{\mathcal{D}}_k} \sum\nolimits_{\delta_3 \in \tilde{\mathcal{D}}_k} P_k(\delta_1)P_k(\delta_2)P_k(\delta_3) \cdot \\ &\left(\hat{u}_k^T \hat{R}(\delta_1, \delta_2, \delta_3, \tilde{H}_k, \tilde{\mathcal{D}}_k)\hat{u}_k + 2\hat{u}_k^T \hat{H}(\delta_1, \delta_2, \delta_3, \tilde{H}_k, \tilde{\mathcal{D}}_k)\hat{x}_k\right) \\ &+ \text{terms constant w.r.t. } \hat{u}_k \end{aligned} \tag{29}$$

when choosing $\kappa_k^{(i)}$ and $\mathcal{X}_N(\mathcal{I}_k)$ from Definition 1 and 2.

3) Optimization Problem: Since the cost function in (9) can be written as (29) and since all constraints in (9) that depend on the optimization variables can be written as (28),

$$\begin{aligned} &\min_{\hat{u}_k} \left(\hat{u}_k^T \vec{R}_k \hat{u}_k + 2\hat{u}_k^T \vec{H}_k \hat{x}_k\right) \\ &s.t. \;\; \vec{M}_x(\tilde{H}_k, \tilde{\mathcal{D}}_k)\hat{x}_k + \vec{M}_u(\tilde{H}_k, \tilde{\mathcal{D}}_k)\hat{u}_k \leq \vec{n}(\tilde{\mathcal{D}}_k) \end{aligned} \tag{30}$$

with $\vec{R}_k = \sum\limits_{\delta_1 \in \mathcal{D}_k(\tilde{\mathcal{D}}_k)} \sum\limits_{\delta_2 \in \mathcal{D}_k(\tilde{\mathcal{D}}_k)} \sum\limits_{\delta_3 \in \mathcal{D}_k(\tilde{\mathcal{D}}_k)} P_k(\delta_1)P_k(\delta_2)P_k(\delta_3)$ $\hat{R}(\delta_1, \delta_2, \delta_3, \tilde{H}_k, \tilde{\mathcal{D}}_k)$ and $\vec{H}_k = \sum\limits_{\delta_1 \in \mathcal{D}_k(\tilde{\mathcal{D}}_k)} \sum\limits_{\delta_2 \in \mathcal{D}_k(\tilde{\mathcal{D}}_k)}$ $\sum\limits_{\delta_3 \in \mathcal{D}_k(\tilde{\mathcal{D}}_k)} P_k(\delta_1)P_k(\delta_2)P_k(\delta_3)\hat{H}(\delta_1, \delta_2, \delta_3, \tilde{H}_k, \tilde{\mathcal{D}}_k)$ is equal to (9) with $\kappa_k^{(i)}$ and $\mathcal{X}_N(\mathcal{I}_k)$ from Definition 1 and 2.

The matrices $\vec{M}_x(\tilde{H}_k, \tilde{\mathcal{D}}_k)$, $\vec{M}_u(\tilde{H}_k, \tilde{\mathcal{D}}_k)$, $\vec{n}(\tilde{\mathcal{D}}_k)$, $\hat{R}(\delta_1, \delta_2, \delta_3, \tilde{H}_k, \tilde{\mathcal{D}}_k)$ and $\hat{H}(\delta_1, \delta_2, \delta_3, \tilde{H}_k, \tilde{\mathcal{D}}_k)$ can be computed offline in advance where $\tilde{\mathcal{D}}_k \subseteq [\underline{d}, \overline{d}]$, $\delta_1, \delta_2, \delta_3 \in [\underline{d}, \overline{d}]$ and $\tilde{H}_k \in [\min(\underline{h}, \underline{s}-1), \overline{h}]$.

The optimization problem (30) can also be written in the form (11) which completes the proof for Theorem 1.

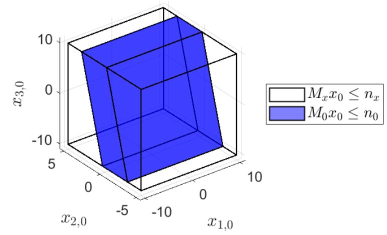

Fig. 2. Set of admissible initial states: x_0 satisfying $M_0 x_0 \leq n_0$.

V. Simulation Example

The plant is the controllable third order LTI system

$$x_{k+1} = \begin{bmatrix} 0.8 & 0.5 & 0 \\ 0 & -1.2 & 0.2 \\ 0 & 0 & 0.2 \end{bmatrix} x_k + \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} u_k.$$

The bounds for the random variables D_k, H_k and S_k are $\underline{d} = 0$, $\overline{d} = 2$, $\underline{h} = 0$, $\overline{h} = 1$, $\underline{s} = 1$, $\overline{s} = 3$ and the parameters of the homogeneous Markov Process $\mathbf{D}$ are given by $\begin{bmatrix} \mu(0) & \mu(1) & \mu(2) \end{bmatrix} = \begin{bmatrix} 0.2 & 0.4 & 0.4 \end{bmatrix}$ and

$$\begin{bmatrix} \Phi(0,0) & \Phi(0,1) & \Phi(0,2) \\ \Phi(1,0) & \Phi(1,1) & \Phi(1,2) \\ \Phi(2,0) & \Phi(2,1) & \Phi(2,2) \end{bmatrix} = \begin{bmatrix} 0.4 & 0.6 & 0 \\ 0.4 & 0.4 & 0.2 \\ 0.2 & 0.4 & 0.4 \end{bmatrix}.$$

The constraints on state and actuating variables are

$$\begin{bmatrix} -10 \\ -5 \\ -10 \end{bmatrix} \leq \begin{bmatrix} x_{1,k} \\ x_{2,k} \\ x_{3,k} \end{bmatrix} \leq \begin{bmatrix} 10 \\ 5 \\ 10 \end{bmatrix}, \quad \begin{bmatrix} -2 \\ -5 \end{bmatrix} \leq \begin{bmatrix} u_{1,k} \\ u_{2,k} \end{bmatrix} \leq \begin{bmatrix} 2 \\ 5 \end{bmatrix}$$

for all $k \geq 0$ which can be written in the form (2). The controller parameters are given by $N = 10$, $Q = \text{diag}(10, 100, 1)$ and $R = \text{diag}(1, 1)$. The resulting set of admissible x_0 is depicted in Fig. 2. All simulations are executed with $x_0 = \begin{bmatrix} -4.5 & -2.6 & -7 \end{bmatrix}^T$ which is admissible.

The proposed control strategy is applied in a simulation with randomly generated realizations of $\mathbf{D}$, $\mathbf{H}$ and $\mathbf{S}$. These realizations and the resulting evolution of x_k and u_k are depicted in Fig. 3 which illustrates that applying the proposed control strategy satisfies all constraints on state an actuating variables while x_k converges to zero with probability one.

The performance of the proposed stochastic MPC is compared to a deterministic MPC which is obtained by (a) using a buffer to ensure that $D_k = \overline{d}$ for all $k \geq 0$ and (b) solving (9) with $\kappa_k^{(i)} = -Lx_{k+i}$ and the terminal constraint $M_N x_{k+N} \leq n_N$. These control strategies are compared by comparing the evolution of $\tilde{J}_k = \sum_{i=0}^{k} \left(x_i^T Q x_i + u_i^T R u_i\right)$.

Three simulations are executed with $H_k = \overline{h} \; \forall k$: (1) the stochastic MPC with $D_k = \overline{d} \; \forall k$ and $S_k = \overline{s} \; \forall k$; (2) the stochastic MPC with $D_k = \underline{d} \; \forall k$ and $S_k = \underline{s} \; \forall k$; (3) the deterministic MPC (which yields the same results for all possible realizations of $\mathbf{D}$ and $\mathbf{S}$). The resulting evolution of $\tilde{J}_k$ is depicted in Fig. 4 which illustrates the potential increase in performance achieved by minimizing the expected value of the cost function: (i) the deterministic MPC is designed by minimizing $\tilde{J}_\infty$ for $D_k = \overline{d} \; \forall k$. Since this is not the only possible case, applying the stochastic

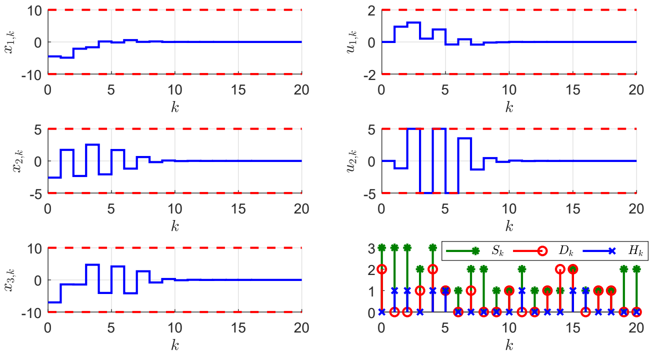

Fig. 3. Simulation with randomly generated realizations of $\mathbf{D}$, $\mathbf{H}$ and $\mathbf{S}$.

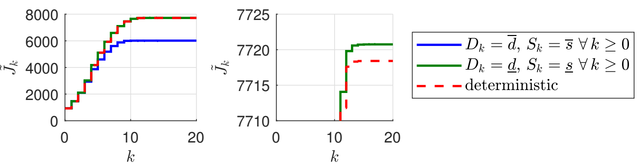

Fig. 4. Comparing stochastic and deterministic MPC for $H_k = \overline{h}\ \forall\, k$.

MPC yields a larger cost if $D_k = \overline{d}\ \forall\, k$. However, this difference is quite small. (ii) if $D_k = \underline{d}\ \forall\, k$ and $S_k = \underline{s}\ \forall\, k$, the proposed stochastic MPC yields a significantly lower value of the cost function than the deterministic MPC.

The simulation with maximal delays is repeated with (a) an LQR controller such that $u_k = 0\ \forall\, k < \overline{h}$, $u_k = -Lx_k\ \forall\, k \geq \overline{h}$ and (b) the stochastic MPC without the constraints which both results in constraint violations.

VI. Conclusion and Further Work

Compared to the MPC minimizing the value of the cost function for $D_k = \overline{d}\ \forall\, k$, the proposed stochastic MPC (i) potentially increases the performance significantly (ii) without deteriorating the performance for $D_k = \overline{d}\ \forall\, k$ significantly. Additionally, it is guaranteed that all constraints on state and actuating are always satisfied if x_0 is admissible.

While computing the constraints in the optimization problem (30) requires computationally expensive computations, these computations can be done offline in advance. Solving (30) is not considered to be computationally expensive in the context of model predictive control since it is an optimization problem with a quadratic cost and linear constraints.

However, implementing the proposed MPC requires that $x_{k+1} = Ax_k + Bu_k$ holds exactly which is typically not the case when considering real world applications. While an adaptation that is capable of handling this issue is discussed in [22], including a suitable model of disturbances in the plant model is considered to be the most important next step.

References

[1] D. Mayne, J. Rawlings, C. Rao, and P. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, 2000.

[2] D. Zhang, P. Shi, Q. Wang, and L. Yu, "Analysis and synthesis of networked control systems: A survey of recent advances and challenges," *ISA Transactions*, vol. 66, pp. 376–392, 2017.

[3] Y. Xia, Y. Gao, L. Yan, and M. Fu, "Recent progress in networked control systems — a survey," *International Journal of Automation and Computing*, vol. 12, pp. 343–367, 2015.

[4] M. Bahraini, M. Zanon, A. Colombo, and P. Falcone, "Optimal control design for perturbed constrained networked control systems," *IEEE Control Systems Letters*, vol. 5, no. 2, pp. 553–558, 2021.

[5] D. M. de la Pena and P. Christofides, "Lyapunov-based model predictive control of nonlinear systems subject to data losses," *IEEE Transactions on Automatic Control*, vol. 53, no. 9, pp. 2076–2089, 2008.

[6] P. Trodden, P. Loma-Marconi, and I. Esnaola, "Stability and the separation principle in output-feedback stochastic mpc with random packet losses," vol. 56, no. 2, 2023, pp. 3818–3823, 22nd IFAC World Congress.

[7] P. Loma-Marconi, P. Trodden, and I. Esnaola, "Stability of nonlinear model predictive control under consecutive packet losses," in *2024 UKACC 14th International Conference on Control*, 2024, pp. 295–300.

[8] D. Quevedo and D. Nešić, "Input-to-state stability of packetized predictive control over unreliable networks affected by packet-dropouts," *IEEE Transactions on Automatic Control*, vol. 56, no. 2, pp. 370–375, 2011.

[9] P. Varutti, B. Kern, T. Faulwasser, and R. Findeisen, "Event-based model predictive control for networked control systems," in *48h IEEE Conference on Decision and Control held jointly with 2009 28th Chinese Control Conference*, 2009, pp. 567–572.

[10] M. Reble, D. Quevedo, and F. Allgöwer, "Control over erasure channels: stochastic stability and performance of packetized unconstrained model predictive control," *International Journal of Robust and Nonlinear Control*, vol. 23, no. 10, pp. 1151–1167, 2013.

[11] D. Quevedo and D. Nešić, "On stochastic stability of packetized predictive control of non-linear systems over erasure channels," *IFAC Proceedings Volumes*, vol. 43, no. 14, pp. 557–562, 2010, 8th IFAC Symposium on Nonlinear Control Systems.

[12] D. Quevedo, E. Silva, and G. Goodwin, "Packetized predictive control over erasure channels," in *2007 American Control Conference*, 2007, pp. 1003–1008.

[13] P. Mishra, D. Chatterjee, and D. Quevedo, "Stochastic predictive control under intermittent observations and unreliable actions," *Automatica*, vol. 118, p. 109012, 2020.

[14] P. Li, Y. Kang, Y. Zhao, and Z. Yuan, "Packet-based model predictive control for networked control systems with random packet losses," in *2018 IEEE Conference on Decision and Control*, 2018, pp. 3457–3462.

[15] J. Wu, L. Zhang, and T. Chen, "Model predictive control for networked control systems," *International Journal of Robust and Nonlinear Control*, vol. 19, no. 9, pp. 1016–1035, 2009.

[16] A. Liu, L. Yu, W. Zhang, and D. He, "Model predictive control for networked control systems with random delay and packet disordering," in *2011 8th Asian Control Conference*, 2011, pp. 653–658.

[17] F. Rosenthal, M. Dolgov, and U. Hanebeck, "Sequence-based receding horizon control over networks with delays and data losses," in *2019 American Control Conference*, 2019, pp. 1450–1457.

[18] Y. Zou, T. Chen, and S. Li, "Network-based predictive control of multirate systems," *IET Control Theory & Applications*, vol. 4, pp. 1145–1156(11), 2010.

[19] Y. Mao, S. Liu, and J. Liu, "Robust economic model predictive control of nonlinear networked control systems with communication delays," *International Journal of Adaptive Control and Signal Processing*, vol. 34, no. 5, pp. 614–637, 2020.

[20] H. Pan, Y. Zhang, and J. Wang, "A synthesis approach for networked mpc of linear system with double-sided arbitrary bounded delays and packet losses," *International Journal of Control*, vol. 96, no. 11, pp. 2762–2772, 2023.

[21] M. Palmisano, M. Steinberger, and M. Horn, "Optimal finite-horizon control for networked control systems in the presence of random delays and packet losses," *IEEE Control Systems Letters*, vol. 5, no. 1, pp. 271–276, 2021.

[22] ——, "Stochastic model predictive control for networked systems with random delays and packet losses in all channels," 2025. [Online]. Available: https://arxiv.org/abs/2309.01679

[23] E. Gilbert and K. Tan, "Linear systems with state and control constraints: the theory and application of maximal output admissible sets," *IEEE Transactions on Automatic Control*, vol. 36, no. 9, pp. 1008–1020, 1991.