On the Control of Lossy Networks with Hold Strategy

Milton Armando Cunguara, Tomás António Mendes Oliveira e Silva
DETI/IEETA/University of Aveiro
Aveiro, Portugal
{milton.cunguara,tos}@ua.pt

Paulo Bacelar Reis Pedreiras
DETI/IT/University of Aveiro
Aveiro, Portugal
pbrp@ua.pt

***Abstract*—The discipline of linear control in centralized systems has been extensively studied and optimal results can already be found in the literature. However, in recent years, new variants of control systems have appeared, among them control communication networks with various degrees of reliability. Two approaches have been defined to cope with missing control values in such non-ideal networks: output zero and hold. The optimal control law over output zero lossy networks has already been presented in the literature. In this paper, we present the optimal control law of generalized output schemes. The control law derived from such generalized output scheme has more internal structure, being equal to optimal controller of each one of its sub-cases, including the zero and the hold approaches. Furthermore, it is presented the optimal output strategy to which the optimal control under it produces the smallest cost. Two proofs are provided for the proposed control law, one based on the collection/gathering terms and another based on a recursive differential expansion. The novel control law is tested via simulation and the obtained results are in perfect agreement with the presented theory.**

I. Introduction

Control theory is nowadays a well developed theory, with mathematically sound and experimentally proven foundations. Optimal control of linear centralized systems, i.e., systems where all of its components are located in the same physical device, was naturally the first to be discovered.

Improvements in networking technologies and computational platforms made Networked Control Systems (NCS) a reality. Despite bringing important benefits (deployment cost, deployment time, maintenance cost, to name just a few) the adoption of NCS also introduced several non-ideal aspects that were not taken into account in the initial formulation of optimal control. Such non-ideal aspects include jitter (variance in the control message arrival time), latency (average delay in the communication) and packet losses.

These effects have been subject of intense research and appropriate solutions for some of them are nowadays reported in the literature. For example, latency can be dealt with using predictive controllers that choose the best inputs based on the moment in which the control value will actually be outputted. For the case of lossy networks, there are two possible output strategies: zero and hold. In the zero approach, whenever a control value is not delivered, it is outputted a zero. In an equivalent scenario, the hold approach outputs the value that was outputted in the previous control cycle.

The optimal control law in output zero networks is now a standard part of control theory. The actual result maintains the overall structure of the optimal controller in centralized systems. However, the hold strategy leads to an optimal controller that has a structure that is significantly different from the optimal control law in centralized networks, though in the limit of no network error it converges to the optimal centralized control. It is the opinion of the authors that this fact influenced the timing of the discoveries.

In this paper it is presented a novel control law that is optimal in lossy control networks, in which the actuator applies a generalized linear function of the last outputted value whenever there is a control message drop, as well as it is presented the optimal matrix of such linear function.

The remaining of this article is organized as follows: in Section II it is made a review of the contributions in the literature that are relevant to this work. In Section III it is presented the notation used throughout the paper. In Section IV it is presented a solution for a similar problem which in turn will be compared to the solution presented in this paper and later on is used for a comparison of the structure of the respective problems. Section V is the main section of this paper presenting two different, though equivalent, methods to solve the identified problem. The first one is an extension of classical approaches which keeps track of all the matrices that emerge due to the hold strategy. The second one uses the *Hessian* matrix to reduce the quadratic cost function into a more tractable form. Section VI discusses aspects related to presented solutions as well as of the other types of control. In Section VIII it is made an evaluation, via simulation, of the control laws devised in this paper. Finally, Section IX summarizes the paper and presents some concluding remarks.

II. Related Work

Optimal control is a relatively well-established discipline, arguably around 90 years old, which started with the work of Pontryagin and Richard Bellman, with Bellman's contribution being

$$V_k(x) = x_k' Q x_k + \min_{u_k} \left(u_k' R u_k + x_{k+1}' P_{k+1} x_{k+1} \right) \quad (1)$$

This work was partially supported by the Portuguese Government through FCT - Fundacao para a Ciencia e a Tecnologia in the scope of project Serv-CPS -PTDC/EEA-AUT/122362/2010 and Ph.D. grant - SFRH/BD/62178/2009/J53651387T4E

where $V_k = x'_k P_k x_k$ is the value function to be minimized, x is the state, u is the control signal, and Q and R are the weight/cost matrices. This equation assumes full state knowledge, i.e. all state are measurable with no sensor noise, and no state perturbations whatsoever. Hence, from a control practice standpoint, its usefulness is limited. This changed with the advent of Kalman filters, leading to the much heralded *Linear Quadratic Gaussian Regulators*. Estimation and control decoupling was explained by the application of the certainty equivalence principle, which states that if $u_k = -L_k x_k$, i.e. L_k is the optimal controller gain if the state is perfectly known, then if only a state estimation ($\hat{x}_{k|k}$) is available the optimal control value is $u_k = -L_k \hat{x}_{k|k}$ [1]. In [2] it was shown that the certainty equivalence principle holds if and only if the estimation error covariance matrix is not a function of the state i.e.

$$\frac{\partial E\left[\left(\hat{x}_{k|k} - x_k\right)'\left(\hat{x}_{k|k} - x_k\right)\right]}{\partial x_k} = \mathbf{0}_{n\times n\times n} \qquad (2)$$

Recent contributions in this line of work show that the certainty equivalence principle is still widely accepted (e.g. [3]–[11]), though some works put its applicability in question for certain classes of systems. E.g. [12] and [13] show that in systems in which the parameters are unknown or because the system is non-linear, the use of the certainty equivalence leads to a suboptimal control, hence it is proposed an *heuristic* called *partial* certainty equivalence. Similar remarks were made in [14] regarding self-tuned controllers.

Obviously, the subject of estimation and control over lossy networks is not as old as the subject of optimal control. The first appearances of optimal control in lossy networks in the literature tried to find a correspondence between the Shannon information theory [15] and control theory. More specifically, tried to answer the age old question: *what is smallest bit-rate necessary to stabilize a given system* or, equivalently, what is the maximum transmission error rate that a network can deliver without compromising the stability of its systems? However, *Which techniques achieve this threshold?* is a matter that, to the best of the authors knowledge, is still unexplored. A number of early attempts to answer the first question are presented in [16] [17].

In [18] the authors attempt to extend the classical theory of optimal control [19] [20]. However, they conclude that the separation principle — which allows a simplification of the results — does not hold in the general case. However, other simpler results are shown, such as, that a system is stable if the independent and identically distributed (i.i.d.) variables that describe the error rate (α) verify $\alpha = \max \lambda(\mathrm{A}^{-2})$, assuming that the system is unstable, where $\lambda(\mathrm{A})$ is the set of all eigenvalues of matrix A and A is the state-transition matrix. Similar claims are made in [21].

[22] makes a more in depth study of such issues, presenting formally the UDP and TCP-like protocol cases. The TCP-like protocol is more amenable to the existing optimal control theory, whereas UDP-like protocol, under the assumptions made therein — $\mathrm{u_k} = 0$, if $\delta_\mathrm{k} = 0$[2]— leads to a non simple optimization problem, since the estimation error covariance matrix at any given time step will depend on previous control values. It is shown in the same reference that, under their assumptions, TCP-like protocols have a error-rate similar to the one found in the previous papers, but UDP-like protocols have lower error-rate bounds that guarantee stability.

Another somewhat related development is given in [23], in which state information is sent in more that one packet. This fragmentation is done primarily in an attempt to reduce the impact of packet losses, since loosing a packet with a significant large number of state variables would be worst than loosing one packet with only one variable.

[24] takes a radically different approach. It uses passive networks as a mean to provide jitter immunity to the controlled system. However, no reasons to believe that passive networks are more immune to jitter is presented, and no mechanism to design passive controllers to achieve certain metrics, e.g. state-tracking or input tracking under noisy conditions, are provided.

In [25] it is presented an analysis similar to the analysis presented in this paper, in which an optimal controller was introduced. However, the authors placed a strong emphasis on the maximum drop rate that allowed for the stabilization of the system. The actual controller for the hold case was not fully presented, rather it was presented a reference to another article that provided the main intuition for the correctness of the proposed controller. Note, however, that the results presented herein do not contradict the results in [25]. In fact, the results presented herein extend them.

In [26] the output schemes are extended from the zero versus hold spectrum by allowing the value that is held at the actuator to decay with time, that is,

$$x_{k+1} = Ax_k + Bs_k + w_k \qquad (3a)$$

$$s_k = L_k r_k \qquad (3b)$$

$$r_k = \theta_k u_k + (1-\theta_k) M_k r_{k-1} \qquad (3c)$$

in which s_k is the value that is actually outputted, r_k is the actuator internal state representation which is allowed to be a matrix, u_k is the control value computed at the controller using information from the sensors and it is transmitted in a bulk message, M_k is a state transition for the actuator state and L_k is the respective controller gain (in fact, the authors of [26] assume that s_k is equal to the first column of r_k, hence, constraining M_k). The authors of the paper in question proceeded by proposing a suboptimal controller given their control architecture. Nevertheless, no motivation for this particular control architecture was presented. In a fact, a case can be made that their approach both sends a large number of messages and has a complex actuator. Second, the actuator internal state is matrix, i.e. rank 2 tensor, whereas it is well known from realization theory that the total number of entries of such state can be chosen such that it is not larger that the size of the state *per se*.

[2] $\delta_\mathrm{k} = 1$ if the message is delivered and $\delta_\mathrm{k} = 0$ otherwise

III. Problem Definition and Notation

In this paper there are considered discrete-time Linear Time-Invariant (LTI) systems, which will be described using a state space representation, i.e.

$$x_{k+1} = Ax_k + Bz_k \tag{4a}$$
$$y_k = Cx_k \tag{4b}$$

with x_k, z_k and y_k representing the system state, input (as outputted by the actuator) and output, as sensed by the sensors, vectors respectively, and A, B and C the state transition, the input and the output matrices respectively. It will also be used a variable u_k to denote the control value computed by the controller. It is assumed that the variables x_k, z_k, u_k and y_k are column vectors of sizes $(n\times1), (r\times1), (r\times1)$ and $(p\times1)$ respectively. A, B, C are matrices of appropriate dimensions. Two standard matrices are used frequently, namely the $\mathbf{0}_{n\times m}$ which is a matrix of zeros with n lines and m columns, and the matrix I_n (identity matrix) which is a square matrix composed of 1 in the diagonal entries and 0 in all other entries.

The goal of the controller is to minimize, without loss of generality[3], a quadratic cost function, namely

$$J = \sum_{k=0}^{N} x_k' Q_k x_k + z_k' R_k z_k \tag{5}$$

with Q_k and R_k semi-positive definite matrices.

This cost function is computed in a step by step manner — cost-to-go — which is essentially the cost of the current step plus the (minimum) cost-to-go of all future steps, i.e. $V_k(x_k) = x_k' Q_k x_k + z_k' R_k z_k + V_{k+1}(x_{k+1})$.

Furthermore, it is assumed that the applied control value (z_k) depends on the delivery of a certain control massage as

$$z_k = \theta_k u_k + (1-\theta_k) M z_{k-1} \tag{6}$$

in which θ_k is a boolean variable that is '1' if the message is delivered and '0' otherwise and u_k is the control vector as produced by the controller. M_k is a square matrix, of suitable size. Whenever the actuator receives a new control value (u_k), i.e. θ_k ='1', it outputs the received value, otherwise, i.e. θ_k ='0', it outputs a vector equal to the matrix M_k times the previous outputted value. Note that if $M_k = 0$ then output strategy will be equal to the output zero strategy and if $M_k = I_r$, then it will be equal to the hold strategy.

The main reason why the system is solved in discrete-time is that as the sampling time goes to zero, the control value become a *Lebesgue function* composed by a series of discontinuities between two given points. This problem does not appear in physical systems because as the sampling time goes to zero the respective θ_k become correlated. However, in this article it is assumed that θ_k are independent and identically distributed (i.i.d.) variables. The issue of correlated θ_k will be addressed in future communications.

[3]it could have been minimized the more complete quadratic cost function

$$J = \sum_{k=0}^{N} \begin{bmatrix} x_k \\ z_k \end{bmatrix}' \begin{bmatrix} Q_k & N_k \\ N_k' & R_k \end{bmatrix} \begin{bmatrix} x_k \\ z_k \end{bmatrix}$$

but as will be shown, the result of both minimizations have essentially the same structure.

On a related note, though from a control perspective, network protocols can be organized in two groups, namely, TCP-like and UDP-like protocols, as discussed in previous sections. The main results of this paper only consider TCP-like protocols. This choice was imposed by the fact that UDP-like protocols do not respect the separation principle, see [2].

IV. Optimal Control over Output Zero Lossy Networks

Even though the solution of this particular problem can be found in the literature (see [22]), it will be presented a different derivation here to allow for a comparison with the generalized optimal controller, presented in this paper.

Theorem 1. *The value of the cost-to-go in a output zero lossy network with no estimation (and input) error is*

$$V_k(x_k) = x_k' P_k x_k \tag{7a}$$
$$P_k = Q_k + (A - \bar{\theta} B L_k)' P_{k+1} A \tag{7b}$$
$$L_k^* = (B' P_{k+1} B + R_k)^{-1} B' P_{k+1} A \tag{7c}$$

and $P_N = Q_N$.

Proof: This theorem is proven by induction. Obviously it is valid for $k = N$, in which case the definition of the cost function and of V_k coincide. Considering previous steps leads to

$$\begin{aligned} V_k(x_k) = x_k' Q_k x_k + \theta_k u_k' R u_k + \\ (Ax_k + \theta_k B u_k)' P_{k+1} (Ax_k + \theta_k B u_k) \end{aligned} \tag{8}$$

calculating the derivative of $V_k(x_k)$ in order to u_k and equating to zero yields

$$\frac{\partial V_k(x_k)}{2\partial u_k} = \theta_k u_k' R + \theta_k (Ax_k + \theta_k B u_k)' P_{k+1} B = \mathbf{0}_{1\times r} \tag{9}$$

which implies that

$$\begin{aligned} V_k(x_k) - \frac{\partial V_k(x_k)}{2\partial u_k}\Big|_{u_k=u_k^*} u_k^* = x_k' Q_k x_k \\ + (Ax_k + \theta_k B u_k^*)' P_{k+1} A x_k. \end{aligned} \tag{10}$$

The statement regarding the value of L_k^* (Equation (7c)) is proven by solving Equation (9) in order to u_k following by a comparison with the definition of L_k^*. However, it should be noticed that the value of u_k is relevant only if $\theta_k = 1$, since otherwise it will not be applied. The statement regarding P_k (Equation (7b)) is proven by comparing Equation (10) (after taking expectation over θ_k) with the definition of $V_k(x_k)$. ■

V. Optimal Control over Hold Lossy Networks

In this section there will be presented two different proofs of the theorem regarding optimal control over output hold lossy networks. They both produce the same optimal control law.

A. Recursive Matrix Derivation

Theorem 2. *The cost-to-go function of control in a lossy network is given by* $V_k(x_k, z_{k-1}) = \theta_k x_k' P_k x_k + (1 - \theta)(z_{k-1}' R_k z_{k-1} + V_{k+1}(x_{k+1}, z_{k-1}))$ *with*

$$P_k = Q_k + \left(A' I_{k+1} - (BL_k^*)' H_{k+1}\right) A \quad (11a)$$

$$I_k = \theta_k P_k + (1-\theta_k)\left(Q_k + A' I_{k+1} A\right) \quad (11b)$$

$$H_k = (1-\theta_k) M' \left(B' I_{k+1} A + H_{k+1} A\right) \quad (11c)$$

$$F_k = (1-\theta_k) M' \left(B' I_{k+1} B + H_{k+1} B + B' H_{k+1} + F_{k+1}\right) M \quad (11d)$$

$$\hat{R}_k = R_k + (1-\theta_k) M' \hat{R}_{k+1} M \quad (11e)$$

$$L_k^* = \left(B' F_{k+1} B + \hat{R}_k\right)^{-1} B' H_{k+1} A \quad (11f)$$

and $P_N = Q_N, I_N = Q_N, H_N = \mathbf{0}_{r\times n}, F_N = \hat{R}_N = \mathbf{0}_{r\times r}$.

Proof: For this output strategy it must be taken into account the fact that the cost-to-go function includes a term that does not depend on u_k, which is applied whenever $\theta_k = 0$. Hence, the cost-to-go function can be written as

$$\begin{aligned} V_k(x_k;\theta_k, z_{k-1}) &= x_k' Q_k x_k + \\ &\theta_k\Big(u_k' R_k u_k + \theta_{k+1} x_{k+1}' P_{k+1} x_{k+1} + \\ &(1-\theta_{k+1}) V_{k+1}(x_{k+1};\theta_{k+1}, u_k)\Big) + \\ &(1-\theta_k)\Big(z_k' R_k z_k + V_{k+1}(x_{k+1};\theta_{k+1}, z_k)\Big). \end{aligned} \quad (12)$$

Obviously, the term with $1-\theta_k$ cannot be minimized by u_k. However, Equation (16) applies to $V_i, k < i \le N$ and in that case it is useful to consider cases that were not minimized upon the computation of the respective u_i, i.e. expansions of the term $(1-\theta_k)(z_k' R_k z_k + V_{k+1}(x_{k+1};\theta_{k+1}, z_k))$. Recall also, that these equations are solved backwards in time, starting at $k = N$ and finishing at $k = 0$.

The proof *per se*, is done by induction: according to the theorem $V_N(x_N) = x_N' Q_N x_N$ which agrees with the cost-to-go in the last step. Given that the previous, which in this case (due to it being backwards in time) is actually the next, step held and under the condition of the theorem plus the structure of V_k as shown in Equation (12), then

$$\begin{aligned} V_k(x_k;\theta_k, z_{k-1}) &= x_k' Q_k x_k + \\ &\theta_k\Big(u_k' R_k u_k + \theta_{k+1}(Ax_k + Bu_k)' P_{k+1}(Ax_k + Bu_k) + \\ &(1-\theta_{k+1})(V_{k+1}(Ax_k + Bu_k;\theta_{k+1}, u_k))\Big) + \\ &(1-\theta_k)\Big(z_k' R_k z_k + V_{k+1}(Ax_k + Bz_k;\theta_{k+1}, z_k)\Big) \end{aligned} \quad (13)$$

which with the recursive expansion of all the terms enclosed within $V_i, \forall_i k < i \le N$, yields

$$\begin{aligned} V_k(x_k;\theta_k, z_{k-1}) &= x_k' Q_k x_k + \\ &\theta_k \sum_{i=k}^{N-1}\left[\prod_{j=k+1}^{i}(1-\theta_j)\right]\Big(u_k'\left(M^{i-k}\right)' R_i M^{i-k} u_k + \\ &\theta_{i+1} x_{i+1}' P_{i+1} x_{i+1} + (1-\theta_{i+1}) x_{i+1}' Q_{i+1} x_{i+1}\Big) + \\ &(1-\theta_k)\Big(z_{k-1}' M' R_k M z_{k-1} + V_{k+1}(x_{k+1};\theta_{k+1}, M z_{k-1})\Big) \end{aligned} \quad (14)$$

At this point we would like to stress, once more, that the terms in θ_j (as opposed to terms in $1-\theta_j$) were not expanded because they are encapsulated into P_{i+1}. By further expanding x_{i+1} (for the case in which $\theta_k = 1$ and $\theta_{[k+1,i]} = 0$) into

$$x_{i+1} = A^{i-k} x_k + \sum_{j=k}^{i} A^{i-j} B M^{j-k} u_k \quad (15)$$

leads equation (14) into (16) (shown at the bottom). Equation (16) itself can be rewritten into

$$\begin{aligned} V_k(x_k;\theta_k, z_{k-1}) &= x_k' Q_k x_k + \\ &\theta_k\Big(u_k' \hat{R}_k u_k + (Ax_k)' I_{k+1} A x_k + (Bu_k)' H_{k+1} A x_k + \\ &(Ax_k)' H_{k+1}' B u_k + (Bu_k)' F_{k+1} B u_k\Big) \\ &+ (1-\theta_k)\Big(z_k' R_k z_k + V_{k+1}(x_{k+1};\theta_{k+1}, z_{k-1})\Big) \end{aligned} \quad (17)$$

with

$$\hat{R}_k = \sum_{i=k}^{N-1}\left[\prod_{j=k+1}^{i}(1-\theta_j)\right](M^{i-k})' R_i M^{i-k} \quad (18a)$$

$$I_k = \sum_{i=k}^{N-1}\left[\prod_{j=k+1}^{i}(1-\theta_j)\right](A^{i-k})' \Big(\theta_{i+1} P_{i+1} + (1-\theta_{i+1}) Q_{i+1}\Big) A^{i-k} \quad (18b)$$

$$H_k = \sum_{i=k}^{N}\left[\prod_{j=k+1}^{i}(1-\theta_j)\right]\left(\sum_{j=k}^{i} A^{i-j} B M^{j-k}\right)' \Big(\theta_{i+1} P_{i+1} + (1-\theta_{i+1}) Q_{i+1}\Big) A^{i-k} \quad (18c)$$

$$F_k = \sum_{i=k}^{N-1}\left[\prod_{j=k+1}^{i}(1-\theta_j)\right]\left(\sum_{j=k}^{i} A^{i-j} B M^{j-k}\right)' \Big(\theta_{i+1} P_{i+1} + (1-\theta_{i+1}) Q_{i+1}\Big)\left(\sum_{j=k}^{i} A^{i-j} B M^{j-k}\right). \quad (18d)$$

$$\begin{aligned} V_k(x_k;\theta_k, z_{k-1}) &= x_k' Q_k x_k + (1-\theta_k)(z_{k-1}' M' R_k M z_{k-1} + V_{k+1}(x_{k+1};\theta_{k+1}, M z_{k-1})) + \\ \theta_k \sum_{i=k}^{N-1}&\left[\prod_{j=k+1}^{i}(1-\theta_j)\right]\left(u_k'(M^{i-k})' R_i M^{i-k} u_k + \left(A^{i-k} x_k + \sum_{j=k}^{i} A^{i-j} B M^{j-k} u_k\right)' (\theta_{i+1} P_{i+1} + (1-\theta_{i+1}) Q_{i+1})\left(A^{i-k} x_k + \sum_{j=k}^{i} A^{i-j} B M^{j-k} u_k\right)\right) \end{aligned} \quad (16)$$

All these variables have been defined in a more economic manner beforehand. Taking the derivative of V_k (equation (17)) in order to u_k and equation to zero yields

$$\begin{aligned}\frac{\partial J_k(x_k;\theta_k,z_{k-1})}{2\partial u_k} &= \\ \theta_k\Big(u_k'\hat{R}_k + (Ax_k)'H_k'B + (Bu_k)'F_kB\Big) &= \mathbf{0}_{1\times r}.\end{aligned} \tag{19}$$

Subtracting an appropriate form of equation (19) from (17) yields

$$\begin{aligned}V_k(x_k;\theta_k,z_{k-1}) - \frac{\partial J_k(x_k;\theta_k,z_{k-1})}{2\partial u_k}u_k &= \\ x_k'Q_kx_k + \theta_k\Big((Ax_k)'I_{k+1} + (Bu_k)'H_{k+1}\Big)Ax_k \\ +(1-\theta_k)\Big(z_k'M'R_kMz_k + V_{k+1}(x_{k+1};\theta_{k+1},Mz_{k-1})\Big)\end{aligned} \tag{20}$$

from which, substituting $u_k^* = -L_k^*x_k$, taking expectations over θ_k makes the term of V_k that appears multiplied by θ_k independent of u_k. Furthermore, such term is multiplied by x_k and x_k' and it coincides with the definition of P_k. By expanding the terms multiplied by $1-\theta_k$ and remembering that, under such circumstances $z_k = Mz_{k-1}$, concludes the proof with respect to V_k.

Regarding the optimal controller gain, starting from equation (19), transposing and grouping terms with u_k and x_k, yields

$$\theta_k\left(B'F_kB + \hat{R}_k\right)u_k^* + \theta_kB'H_kAx_k = \mathbf{0}_{n\times 1} \tag{21}$$

in which it is assumed that $\theta_k = 1$, since otherwise the cost function does not depend on u_k. Sending the second term to the right-side and considering the definition of L_k^* concludes the proof. ■

B. Differential Matrix Derivation

Theorem 3. *The cost-to-go function of control in a lossy network is given by*

$$V_k(x_k) = \frac{1}{2}\begin{bmatrix}x_k\\ z_{k-1}\end{bmatrix}'\begin{bmatrix}P^-_{xx,k} & P^-_{xu,k}\\ P^-_{ux,k} & P^-_{uu,k}\end{bmatrix}\begin{bmatrix}x_k\\ z_{k-1}\end{bmatrix} \tag{22a}$$

$$\begin{bmatrix}P^+_{xx,k} & P^+_{xu,k}\\ P^+_{ux,k} & P^+_{uu,k}\end{bmatrix} = \hat{A}'\begin{bmatrix}P^-_{xx,k+1} & P^-_{xu,k+1}\\ P^-_{ux,k+1} & P^-_{uu,k+1}\end{bmatrix}\hat{A} + \begin{bmatrix}Q_k & \mathbf{0}_{n\times r}\\ \mathbf{0}_{r\times n} & R_k\end{bmatrix} \tag{22b}$$

$$\hat{A} = \begin{bmatrix}A & B\\ \mathbf{0}_{r\times n} & I_n\end{bmatrix} \tag{22c}$$

$$\begin{bmatrix}P^-_{xx,k} & P^-_{xu,k}\\ P^-_{ux,k} & P^-_{uu,k}\end{bmatrix} = \begin{bmatrix}P^+_{xx,k} - \theta_kP^+_{xu,k}L_k^* & (1-\theta_k)P^+_{xu,k}M\\ (1-\theta_k)M'P^+_{ux,k} & (1-\theta_k)M'P^+_{uu,k}M\end{bmatrix} \tag{22d}$$

$$L_k^* = \left(P^+_{uu,k}\right)^{-1}P^+_{ux,k} \tag{22e}$$

and

$$\begin{bmatrix}P^-_{xx,N} & P^-_{xu,N}\\ P^-_{ux,N} & P^-_{uu,N}\end{bmatrix} = \begin{bmatrix}Q_N & \mathbf{0}_{n\times r}\\ \mathbf{0}_{r\times n} & \mathbf{0}_{r\times r}\end{bmatrix}. \tag{23}$$

Proof: Since the cost function is quadratic it is possible to rewrite it in the form

$$V_k(x_k, z_{k-1}) = \frac{1}{2}\begin{bmatrix}x_k\\ z_{k-1}\end{bmatrix}'\frac{\mathrm{d}^2J_k}{\mathrm{d}\begin{bmatrix}x_k\\ z_{k-1}\end{bmatrix}\mathrm{d}\begin{bmatrix}x_k\\ z_{k-1}\end{bmatrix}'}\begin{bmatrix}x_k\\ z_{k-1}\end{bmatrix}. \tag{24}$$

The derivation is carried out in order to J_k instead of $V_k(x_k)$ because

$$V_k(x_k, z_{k-1}) \overset{def}{=} \min_{u_k} J_k(x_k, u_k, z_{k-1}).$$

Hence, V_k is not a function of u_k, being assumed that u_k is equal to a given value u_k^*. Continuing, deriving Equation (12) (twice) in order to $[x_k' \; u_k']'$ using the chain rule leads to

$$\frac{\partial J_k(x_k)}{\partial x_k} = 2x_k'Q_k + \frac{\partial J_{k+1}(x_{k+1})}{\partial x_{k+1}}\frac{\partial x_{k+1}}{\partial x_k} \tag{25a}$$

$$\begin{aligned}\frac{\partial J_k(x_k)}{\partial z_k} =& 2\theta_kz_k'R_k + \frac{\partial J_{k+1}(x_{k+1})}{\partial x_{k+1}}\frac{\partial x_{k+1}}{\partial z_k} + \\ &\frac{\partial J_{k+1}(x_{k+1})}{\partial z_k}\frac{\partial z_k}{\partial z_k}\end{aligned} \tag{25b}$$

taking the derivative of these two equations in order to x_k and z_k lead to (the derivative in order to $\partial x_k'\partial z_k$ was omitted due to the symmetry of the equations)

$$\frac{\partial^2 J_k(x_k)}{\partial x_k'\partial x_k} = 2Q_k + \left(\frac{\partial x_{k+1}}{\partial x_k}\right)'\frac{\partial^2 J_{k+1}(x_{k+1})}{\partial x_{k+1}'\partial x_{k+1}}\frac{\partial x_{k+1}}{\partial x_k} \tag{26a}$$

$$\begin{aligned}\frac{\partial^2 J_k(x_k)}{\partial z_k'\partial x_k} =& \left(\frac{\partial x_{k+1}}{\partial z_k}\right)'\frac{\partial^2 J_{k+1}(x_{k+1})}{\partial x_{k+1}'\partial x_{k+1}}\frac{\partial x_{k+1}}{\partial x_k} + \\ &\left(\frac{\partial z_k}{\partial z_k}\right)'\frac{\partial J_{k+1}(x_{k+1})}{\partial z_k'\partial x_{k+1}}\frac{\partial x_{k+1}}{\partial x_k}\end{aligned} \tag{26b}$$

$$\begin{aligned}\frac{\partial^2 J_k(x_k)}{\partial z_k'\partial z_k} =& 2R_k + \\ &\left(\frac{\partial x_{k+1}}{\partial z_k}\right)'\frac{\partial^2 J_{k+1}(x_{k+1})}{\partial x_{k+1}'\partial x_{k+1}}\frac{\partial x_{k+1}}{\partial z_k} + \\ &\left(\frac{\partial z_k}{\partial z_k}\right)'\frac{\partial^2 J_{k+1}(x_{k+1})}{\partial z_k'\partial x_{k+1}}\frac{\partial x_{k+1}}{\partial z_k} + \\ &\left(\frac{\partial x_{k+1}}{\partial z_k}\right)'\frac{\partial^2 J_{k+1}(x_{k+1})}{\partial x_{k+1}'\partial z_k}\frac{\partial z_k}{\partial z_k} + \\ &\left(\frac{\partial z_k}{\partial z_k}\right)'\frac{\partial^2 J_{k+1}(x_{k+1})}{\partial z_k'\partial z_k}\frac{\partial z_k}{\partial z_k}\end{aligned} \tag{26c}$$

The last set of equations is recursive which can be made more evidently if the substitutions

$$\hat{A} \equiv \frac{\partial\begin{bmatrix}x_{k+1}\\ z_k\end{bmatrix}}{\partial\begin{bmatrix}x_k\\ z_k\end{bmatrix}} = \begin{bmatrix}A & B\\ \mathbf{0}_{r\times n} & I_r\end{bmatrix} \tag{27}$$

the vector being derived (in the numerator) as a term in z_k as opposed to z_{k+1} because (as it will be discussed below) the act of producing a control value transforms the expression in

z_{k+1} into an expression in z_k. Define also

$$\hat{P}_k^- \stackrel{def}{=} \begin{bmatrix} P_{xx,k}^- & P_{xu,k}^- \\ P_{ux,k}^- & P_{uu,k}^- \end{bmatrix} = \frac{1}{2}\begin{bmatrix} \frac{\partial^2 J_k}{\partial x_k' \partial x_k} & \frac{\partial^2 J_k}{\partial x_k' \partial z_k} \\ \frac{\partial^2 J_{k+1}}{\partial z_k' \partial x_k} & \frac{\partial^2 J_k}{\partial z_k' \partial z_k} \end{bmatrix} \quad (28)$$

(with $P_{xu,k}^{\pm} = (P_{ux,k}^{\pm})'$). Putting it all together

$$\hat{P}_k^+ = \begin{bmatrix} Q_k & \mathbf{0}_{n\times r} \\ \mathbf{0}_{r\times n} & R_k \end{bmatrix} + \hat{A}'\hat{P}_{k+1}^-\hat{A}. \quad (29)$$

Due to the quadratic nature of the cost function (before minimization $J_k(x_k, z_k)$ is a function of $\hat{P}_k^+$)

$$J_k(x_k) = \frac{1}{2}\begin{bmatrix} x_k \\ z_k \end{bmatrix}' \hat{P}_k^+ \begin{bmatrix} x_k \\ z_k \end{bmatrix} \quad (30)$$

which taking the derivative in order to u_k leads to

$$\frac{\partial J_k(x_k)}{\partial u_k} = \begin{bmatrix} x_k \\ z_k \end{bmatrix}' \hat{P}_k^+ \begin{bmatrix} \mathbf{0}_{n\times r} \\ \theta_k I_r \end{bmatrix} = \mathbf{0}_{1\times r} \quad (31)$$

which is solved into (i.e. its minimum)

$$u_k = -\left(P_{uu,k}^+\right)^{-1} P_{ux,k}^+ x_k \quad (32)$$

which defines the controller gain L_k. Remembering that $z_k = \theta_k u_k + (1-\theta_k)Mz_{k-1}$ and as just shown $u_k^* = L_k x_k$, this can be substituted into (30) from which performing the matrix multiplication taking into account that $\theta_k(1-\theta_k) = 0$ and Equation (32) and grouping terms in x_k yields

$$V_k(x_k) = \frac{1}{2}\begin{bmatrix} x_k \\ z_{k-1} \end{bmatrix}' \hat{P}_k^- \begin{bmatrix} x_k \\ z_{k-1} \end{bmatrix} \quad (33)$$

with (due to the idempotency of the product in Boolean algebra $(1-\theta_k)(1-\theta_k) = (1-\theta_k)$)

$$\hat{P}_k^- = \begin{bmatrix} P_{xx,k}^- - \theta_k P_{xu,k}^-\left(P_{uu,k}^-\right)^{-1} P_{ux,k}^- & (1-\theta_k)P_{xu,k}^- M \\ (1-\theta_k)M'P_{ux,k}^- & (1-\theta_k)M'P_{uu,k}^- M \end{bmatrix} \quad (34)$$

■

As referred above, the computation of u_k^* due to the fact that it is a function of x_k transformed Equation (30) with terms in z_k into Equation (33) with terms in z_{k-1}.

VI. Discussion

Two important notes are in order: first, previous sections presented two proofs of the optimal controller over hold lossy networks. The control laws that came out of these two sections appear to be different. However, the various sub-matrices of the *Hessian* matrix that appear in the second proof are related according to

$$\hat{P}_k^- = 2\begin{bmatrix} I_k & H_k \\ H_k' & F_k + \hat{R}_k \end{bmatrix}. \quad (35)$$

The factor of two in the last equation stems from the slightly different cost functions used in each proof. The last equality can be established by simply comparing the values of the respective variables.

Second, in general this control law tracks a matrix of size $(n+r)\times(n+r)$, which is larger than the P_k matrix that is tracked in classical optimal control and in control over lossy networks with the output zero strategy. Once more, it seems to be a discrepancy. This stems from the (borrowing from the estimation lexicon) correction part, i.e. the mechanism that updates $\hat{P}_k$ once the value of u_k is known. Since, in the hold case, after the correction what remains in z_k is $(1-\theta_k)z_{k-1}$ (as pointed out above, the term in u_k is joined with the term in x_k). Whereas in centralized control (as well as in the hold strategy), once u_k is found and joined to the terms in x_k, z_k becomes equal to zero. Hence, the three sub-matrices associated with it become unnecessary, since the product will be equal to zero.

This fact implies that only $P_{xx,k}$ needs to be tracked. This has profound implications and is what lowers the effective rank of the other approaches.

It was pointed out in Section III that the introduction of cost terms relating to the correlation between the input and the output would not change the structure of the optimal controller. That statement can now be substantiated by noting that by repeating the proofs presented herein under such circumstances Equation (29) becomes

$$\hat{P}_k^+ = \begin{bmatrix} Q_k & N_k \\ N_k' & R_k \end{bmatrix} + \hat{A}'\hat{P}_{k+1}^-\hat{A} \quad (36)$$

whereas Equation (32) and Equation (34) remain untransformed.

As pointed out before, this control strategy generalizes the types of actuator outputs. Hence, it is no surprise that for $M = 0$ it is equal to the controller presented in [22] and reproduced in Section IV.

A. Domain of Application

Even though the novel optimal controller was derived under a number of strict conditions, i.e. no noise/error of any kind, the optimal control law has a wider domain of application. That is, substituting this controller on certain systems that do not comply with the error constraints still yield the optimal control law.

This control law can be trivially extended into the case of a (perfectly, e.g. full state measurement) known state but with state disturbances, Equation (37), since the effects of w_k cannot be minimized by u_k.

$$x_{k+1} = Ax_k + Bu_k + w_k \quad (37)$$

More generally, if the separation principle holds, then by definition the presented controller will be optimal. This fact can be used to extend this controller into the TCP-like network protocols, i.e. network protocols in which the delivery status is always known by the sender (note that this does not imply retransmissions).

VII. Optimal Actuator State Transition Matrix

In past sections, it was derived the optimal control over a lossy network, in which whenever there was no new control message, the actuator applied a linear function of the last applied value. Such linear function was multiplication by a constant matrix M. However, the optimality was with respect only to u_k, i.e. on the controller side.

This section deals with the optimal value of the matrix M and, of course, its effects on the optimal value that the actuator should apply. In the previous section it was shown that z_k, hence both $M_k z_{k-1}$ and u_k, only has a bearing on the cost function starting from Equations (29)-(30). Finding an optimal M_k implies the minimization of such equations in order to both u_k and M_k, however, the minimum over u_k is already known from past sections.

Using the standard method for determining extremum points, a derivative of Equation (29) is taken in order to M_k and equated to zero. It is known that such extremum is a minimum because Equation (29) is convex in z_k, with a semi-positive definite hessian. Continuing,

$$\frac{\partial J_k\left(x_k, z_{k-1}\right)}{2\partial M_k} = z_{k-1}\begin{bmatrix} x_k \\ z_k \end{bmatrix}' \hat{P}_k \begin{bmatrix} \mathbf{0}_{n\times r} \\ (1-\theta_k)I_r \end{bmatrix} = \mathbf{0}_{r\times r} \quad (38)$$

which can be extended into

$$(1-\theta_k)z_{k-1}\left(x_k' P_{xu,k} + z_k' P_{uu,k}\right) = \mathbf{0}_{r\times r} \quad (39)$$

or

$$z_{k-1}\left(x_k' P_{xu,k} + \left(M_k^* z_{k-1}\right)' P_{uu,k}\right) = \mathbf{0}_{r\times r} \quad (40)$$

hence

$$\left(P_{uu,k}^{-1} P_{ux,k} x_k + M_k^* z_{k-1}\right) z_{k-1}' = \mathbf{0}_{r\times r} \quad (41)$$

which implies that

$$\left(M_k^* z_{k-1} - u_k^*\right) z_{k-1}' = \mathbf{0}_{r\times r}. \quad (42)$$

Equation (42) shows that M^* is a projection matrix that provides the optimal estimate of u_k^* given z_{k-1}, which is an unsurprising result. M_k^* can be readily computed from the equations that describe the evolution of the system. Just as L_k in the controller, M_k can be preprogrammed into the actuator in order to reduce the online computational load. Furthermore, the equation that describe both u_k^* and M_k^* converge to constant values.

Regarding $V_k\left(x_k, z_{k-1}\right)$ given M_k^*, Equation (34) which defines the hessian of $V_k\left(x_k, z_{k-1}\right)$ still holds, however, two of the terms of Equation (34) with M_k cancel each other, more concretely,

$$\begin{aligned} &\left(M_k z_{k-1}\right)' P_{ux,k} x_k + x_k' P_{xu,k} M_k z_{k-1} + \\ &\left(M_k z_{k-1}\right)' P_{xu,k} M_k z_{k-1} = \left(M_k z_{k-1}\right)' P_{ux,k} x_k + \\ &x_k' P_{xu,k} M_k z_{k-1} + \left(M_k z_{k-1}\right)' P_{xu,k} M_k z_{k-1} + \\ &\left(M_k z_{k-1}\right)' P_{xu,k} M_k z_{k-1} - \left(M_k z_{k-1}\right)' P_{xu,k} M_k z_{k-1} \end{aligned} \quad (43)$$

the last equation has the trace of Equation (41) (and its transpose) times M_k^*, which is zero by definition of M_k^*. This implies that the first four terms of the previous equation are equal to $\mathbf{0}_{r\times r}$. Hence, the hessian of the $V_k(x_k, z_{k-1})$ given that $M_k = M_k^*$ is

$$\hat{P}_k = \begin{bmatrix} P_{xx,k} - \theta_k P_{xu,k} P_{uu,k}^{-1} P_{ux,k} & \mathbf{0}_{n\times r} \\ \mathbf{0}_{r\times n} & -(1-\theta_k) M_k^{*\prime} P_{uu,k} M_k^* \end{bmatrix}. \quad (44)$$

VIII. Evaluation

This section evaluates the optimality of the controller proposed in this paper. To this end, an unstable system was chosen at random. The cost function (according to cost matrices that are defined below) are evaluated for a set of (linear) control laws during a finite amount of time. The controller gain matrix that minimizes the cost function is compared to the theoretically derived controller gain matrix. The agreement between these values informs on the optimality of the proposed control law.

Note, however, that it was only tested for $M_k = \mathbf{0}_{r\times r}$. This choice was made because testing the optimality of the choice of M_k would require performing many tests similar to the one presented in this section. However, due to timing constraints it will not be possible to include such results on this paper.

A. A First Order System

The system had a state-space representation

$$x_{k+1} = [1.5]\, x_k + [1]\, u_k \quad (45)$$

$$y_k = [1]\, x_k \quad (46)$$

which means that it is unstable with a discrete pole at z = 1.5. The cost function was

$$J_k = x_{300}^2 + \sum_{k=0}^{299}\left(x_k^2 + 0.7u_k^2\right). \quad (47)$$

The controller, as in $u_k = -L_k x_k$, was chosen from 300 linearly spaced points in the interval $[0.5\ 1.4]$. To smooth out the inherent variance of θ, $5\cdot 10^6$ simulations were made for each value of L_k. Each simulation last for 300 steps. All simulations were initialized with $x_0 = 1$. The simulations were made with $\bar{\theta} = \frac{3}{4}$. All simulations were performed without any type of noise of perturbation. Figure 1 depicts the results of such simulations. A logarithmic scale was used in the y-axis to allow a proper visualization of the many orders of magnitude that are spawned by the graph.

There is an asymmetry between the error behavior on the left side, i.e. close to $L_{ss} = .5$, and the behavior on the right side, i.e. close to $L_{ss} = 1.4$, regarding the variance of the curve, since as can be seen, in the former case, the curve is rather well behaved, whereas in the later case, the curve is rather jagged, having many peaks and valleys. This behavior stems from the fact that when $L_{ss} = 0.5$ the closed-loop system has a pole close to $z = 1$, which in turn implies that the control value changes slowly, therefore, the previous value is not much different than the current control value. Hence, the various simulations that differ only by the message error sequence have similar cost, contributing to the observed low variance. Such effect is not present on the right end of the graph

As can be seen, there is a perfect agreement between theory (i.e. L = 0.78) and practice (the simulation results).

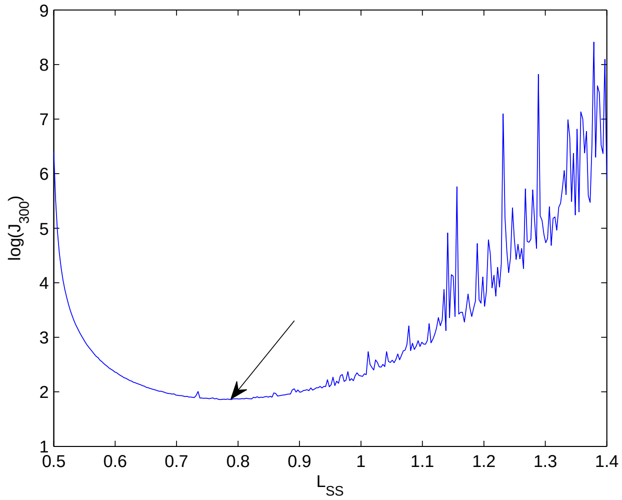

Fig. 1: Controller Gain versus Cost.

IX. Conclusion and Future Work

The problem of optimal linear control has already been solved in the literature, as well as the problem of control in networks that output zero in the absence of a new control value. The present article solved the problem of (linear) control in systems in which the past output value is kept in case the present control value is unknown.

The problem solved in this article is mathematical richer than the output zero case. In fact, previous sections discussed the conditions in which this case degenerates into the output zero case.

Simulation results are in perfect agreement with the optimal control gain computed theoretically.

Nevertheless, from an engineering perspective, now that the optimal controller of both cases is known, it is paramount to know, before deployment, which output strategy produces a lower cost. This question will be addressed in future work. Another aspect that will be addressed in future work regards the independence of θ_k for both output strategies.

References

[1] H. Witsenhausen, "Separation of estimation and control for discrete time systems," *Proceedings of the IEEE*, vol. 59, no. 11, pp. 1557 –1566, nov 1971.

[2] E. Tse and Y. Bar-Shalom, "Generalized certainty equivalence and dual effect in stochastic control," *Automatic Control, IEEE Transactions on*, vol. 20, no. 6, pp. 817 – 819, Dec 1975.

[3] L. Bao, M. Skoglund, and K. Johansson, "A scheme for joint quantization, error protection and feedback control over noisy channels," in *American Control Conference, 2007. ACC '07*, july 2007, pp. 4905 – 4910.

[4] L. P. Hansen and T. J. Sargent, "Certainty equivalence and model uncertainty," *Proceedings, Board of Governors of the Federal Reserve System (U.S.)*, pp. 17–38, 2005.

[5] C. Ebenbauer, T. Raff, and F. Allgower, "Certainty-equivalence feedback design with polynomial-type feedbacks which guarantee iss," *Automatic Control, IEEE Transactions on*, vol. 52, no. 4, pp. 716 –720, april 2007.

[6] T. Li and J.-F. Zhang, "Decentralized adaptive games for large population coupled arx systems with unknown coupling strength," in *Decision and Control (CDC), 2010 49th IEEE Conference on*, dec 2010, pp. 3110 –3115.

[7] Z. Ma, D. Callaway, and I. Hiskens, "Decentralized charging control for large populations of plug-in electric vehicles: Application of the nash certainty equivalence principle," in *Control Applications (CCA), 2010 IEEE International Conference on*, sept. 2010, pp. 191 –195.

[8] G. Nair, F. Fagnani, S. Zampieri, and R. Evans, "Feedback control under data rate constraints: An overview," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 108 –137, Jan 2007.

[9] A. Molin and S. Hirche, "On lqg joint optimal scheduling and control under communication constraints," in *Decision and Control, 2009. CDC/CCC 2009. Proceedings of the 48th IEEE Conference on*, dec. 2009, pp. 5832 –5838.

[10] C. Ramesh, H. Sandberg, L. Bao, and K. Johansson, "On the dual effect in state-based scheduling of networked control systems," in *American Control Conference (ACC), 2011*, vol. 29, no. 1, july 2011, pp. 2216 –2221.

[11] S. Battilotti, "Robust detectability from the measurements plus state feedback stabilization imply semiglobal stabilization from the measurements," *Automatic Control, IEEE Transactions on*, vol. 51, no. 9, pp. 1542 –1547, sept. 2006.

[12] N. Filatov and H. Unbehauen, "Design of improved adaptive controllers using partial certainty equivalence principle," in *American Control Conference. Proceedings of the 1998*, vol. 2, jun 1998, pp. 1068 –1072.

[13] N. Filatov and H. Unbehausen, "Adaptive predictive control policy for nonlinear stochastic systems," *Automatic Control, IEEE Transactions on*, vol. 40, no. 11, pp. 1943 –1949, nov 1995.

[14] C. Charalambous, "Stochastic nonlinear minimax dynamic games with noisy measurements," *Automatic Control, IEEE Transactions on*, vol. 48, no. 2, pp. 261 – 266, feb. 2003.

[15] C. E. Shannon, "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, p. 379423 & 623656, July 1948.

[16] J. Hespanha, A. Ortega, and L. Vasudevan, "Towards the control of linear systems with minimum bit-rate," in *In Proc. of the Int. Symp. on the Mathematical Theory of Networks and Syst*, 2002.

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

[18] B. Azimi-sadjadi, "Stability of networked control systems in the presence of packet losses," in *in Proceedings of 2003 IEEE Conference on Decision and Control*, 2003, pp. 676–681.

[19] D. P. Bertsekas, *Dynamic Programming and Optimal Control, Vol. II*, 3rd ed. Athena Scientific, 2007.

[20] B. W. Karl J. Astrom, *Computer Controlled Systems: Theory and Design*, 3rd ed. Prentice Hall, 1997.

[21] S. Hu and W.-Y. Yan, "Stability robustness of networked control systems with respect to packet loss," *Automatica*, vol. 43, no. 7, pp. 1243 – 1248, 2007.

[22] L. Schenato, B. Sinopoli, M. Franceschetti, K. Poolla, and S. Sastry, "Foundations of control and estimation over lossy networks," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 163 –187, jan. 2007.

[23] S. Hu and W.-Y. Yan, "Stability of networked control systems under a multiple-packet transmission policy," *Automatic Control, IEEE Transactions on*, vol. 53, no. 7, pp. 1706 –1711, aug. 2008.

[24] X. Koutsoukos, N. Kottenstette, J. Hall, P. Antsaklis, and J. Sztipanovits, "Passivity-based control design of cyberphysical systems," in *In International Workshop on CyberPhysical Systems - Challenges and Applications (CPSCA08*, 2008.

[25] L. Schenato, "To zero or to hold control inputs with lossy links?" *Automatic Control, IEEE Transactions on*, vol. 54, no. 5, pp. 1093–1099, 2009.

[26] M. Kogel and R. Findeisen, "Robust suboptimal control over lossy networks using extended input schemes," in *Computer-Aided Control System Design (CACSD), 2011 IEEE International Symposium on*, 2011, pp. 210–215.