

# Optimal Sampling Pattern for the Free Final Time Linear Quadratic Regulator

Pedro Balaguer and Carlos Ariño

**Abstract**—The optimal sampling problem is the selection of the optimal sampling instants together with the optimal control actions such that a given cost function is minimized. In this article we solve the optimal sampling problem for the free final time linear quadratic regulator. Each optimal sampling instant is computed as the minimization of a maximum eigenvalue problem that is formulated at each stage by previously applying dynamic programming. The solution provides the optimal sampling instants, control actions, and the optimal final time in a recursive and constructive way. Furthermore, the solution is optimal for any arbitrary number of control moves  $N \geq 1$ , as it is not based on asymptotic arguments. Two application examples show the feasibility of the approach.

**Index Terms**—Optimal sampling, linear quadratic regulator, control design, dynamic programming, optimal control, networked control, real-time systems.

## I. INTRODUCTION

THE technological developments of sensor networks (SN), the Internet of Things (IoT), and networked control systems (NCS) result in a multitude of nodes sending information for supervision [1], estimation [2], and control [3]. The reduction of the number of sampling instants  $N$  saves energy consumption, computing power, and communication bandwidth, which may be scarce resources.

Digital control systems were initially developed on the basis of periodic sampling [4]. However, for a given number of sampling instants  $N$ , non-periodic sampling may lead to a better performance than a periodic one. The optimal sampling problem, as defined in [5], is the selection of the  $N$  interarrival times  $\tau_k := t_{k+1} - t_k$  for  $k \in \{0, 1, \dots, N-1\}$  and control inputs  $u_k$  for  $k \in \{0, 1, \dots, N-1\}$ , such that a given cost function  $J$  is minimized.

The analytical solution of the sampling problem by deriving necessary conditions for the optimum is not feasible. Instead, numerical gradient based optimization algorithms are derived from necessary conditions, which are of high complexity. In [5] a numerical optimization algorithm is proposed that, for systems of order  $n$  and  $N$  sampling instants, only the computation of the gradient has complexity of  $O(N^2 n^3)$ .

The optimal sampling problem has similarities with the control of switched systems [6], [7]. The optimal control of switched systems is the selection of optimal continuous inputs and optimal switching sequences among a family of dynamical systems so as to minimize a given cost function. The proposed solutions are also based on numerical optimization methods.

Expressions for the gradient of the cost function with respect to switching times are derived, and gradient descent optimization algorithms are employed for numerical minimization. The derived numerical algorithms are computationally costly, not scalable with the number of sampling instants  $N$ , and, being a nonconvex problem, they lack any guarantee to find the global minimum [6], [7].

To overcome these problems, a new approach to solve the optimal sampling problem is proposed in [5], called the quantization-based sampling. The basic idea of quantization-based sampling is to approximate the optimal continuous-time control input  $u(t)^*$  by a piecewise constant function that provides the optimal interarrival times. Once the optimal interarrival times are calculated the optimal control inputs are computed by standard linear quadratic regulator (LQR). The approach is computationally tractable because for LQR the optimal control action  $u(t)^*$  is readily calculated. Furthermore, it is shown in [5] that quantization based sampling is optimal for first-order systems for a large number of samples  $N$ .

In this article we solve the optimal sampling problem for general order multiple-input multiple-output (MIMO) systems with quadratic cost function and free final time  $T_N$ . This is in contrast with the problem proposed in [5] where the final time  $T$  is fixed beforehand. As a result, we not only provide the optimal interarrival times  $\tau_k^*$  and optimal control inputs  $u_k^*$  but also the optimal final time  $T_N^*$ . It is the free final time condition  $T_N$  that allows the scalability and constructibility properties of the optimization algorithm derived in this article.

The solution of the optimal sampling problem is a complex problem. In a first step we keep the complexity of the problem bounded by applying dynamic programming, which is well suited for optimal control with quadratic cost function [8]. The application of dynamic programming does not solve the optimal sampling pattern per se. Instead, dynamic programming yields a family similar interrelated optimization problems for which the optimal interarrival time can be computed as the minimization of a maximum eigenvalue problem that must be performed at each stage. The resulting algorithm is

- optimal for arbitrary  $N \geq 1$  because it is not based on asymptotic arguments.
- scalable because the optimal interarrival times are obtained by solving  $N$  one-dimensional optimization problems.
- recursive because the optimal interarrival times are computed backwards from the last stage in a sequential manner.
- constructive because given the  $N$  optimal interarrival times, the solution for the new problem with  $N + 1$

interarrival times just requires to solve one more one-dimensional optimization problem to calculate the new optimal interarrival time.

## II. PROBLEM STATEMENT

Consider the free final time  $T_N$  linear quadratic regulator problem

$$P_{T_N} : \quad \min_{u \in \mathbb{U}_N} \int_0^{T_N} (x^T Q x + u^T R u) dt + x(T_N)^T S x(T_N) \quad (1)$$

s.t.

$$\dot{x} = Ax + Bu$$

$$x(0) = x_0$$

where  $x(t) \in \mathbb{R}^n$  is the state,  $u(t) \in \mathbb{R}^m$  the input signal,  $A \in \mathbb{R}^{n \times n}$ , and  $B \in \mathbb{R}^{n \times m}$  are the dynamical system matrices,  $Q \in \mathbb{R}^{n \times n}$ ,  $R \in \mathbb{R}^{m \times m}$ , and  $S \in \mathbb{R}^{n \times n}$  are the cost function weights, with  $Q$  and  $S$  positive semi-definite, and  $R$  positive definite. The control input signal  $u(t)$  is constrained to be in the set  $\mathbb{U}_N$  of  $N$ -complexity piecewise constant signals, where by piecewise constant  $N$ -complexity we mean that  $u(t)$  is a linear combination of  $N$  indicator functions of intervals

$$u(t) = u_k \quad \forall t \in [t_k, t_{k+1}), \quad k = 0, 1, \dots, N-1$$

with  $0 = t_0 < t_1 < \dots < t_N = T_N$ , where  $u_k$  is given as an state feedback  $u_k := F_k x_k$  without loss of generality. The sequence  $\{t_0, t_1, \dots, t_{N-1}, t_N\}$  is the sampling pattern  $\mathbb{T}$ , while  $t_k$  for  $k \in \{0, 1, \dots, N\}$  are called sampling instants. For  $t_0 = 0$ , the sampling pattern information is also given by the sequence  $\{\tau_0, \tau_1, \dots, \tau_{N-1}\}$  of interarrivals  $\tau_k := t_{k+1} - t_k$  for  $k \in \{0, 1, \dots, N-1\}$ , defined as the time between two consecutive sampling instants. Consequently, in what follows the sampling pattern  $\mathbb{T}$  is defined by the interarrival sequence.

The previous problem is solved by providing the optimal sampling pattern  $\mathbb{T}^* := \{\tau_0^*, \tau_1^*, \dots, \tau_{N-1}^*\}$  and the optimal control action gain sequence  $\mathbb{F}^* := \{F_0^*, F_1^*, \dots, F_{N-1}^*\}$ .

## III. PROBLEM DISCRETIZATION

For an arbitrary sampling pattern  $\mathbb{T} = \{\tau_0, \tau_1, \dots, \tau_{N-1}\}$ , with  $\sum_{k=0}^{N-1} \tau_k = T_N$ , the exact discretization of the LQR problem  $P_{T_N}$  is

$$\min_{u_k, \tau_k} \sum_{k=0}^{N-1} (x_k^T Q_k x_k + u_k^T R_k u_k + 2x_k^T P_k u_k) + x_N^T S x_N \quad (2)$$

s.t.

$$x_{k+1} = A_k x_k + B_k u_k \quad (3)$$

$$x_{k=0} = x_0 \quad (4)$$

with

$$A_k := e^{A\tau} \quad (5)$$

$$B_k := \int_0^\tau e^{A(\tau-\mu)} d\mu B \quad (6)$$

$$Q_k := \int_0^\tau A_k^T Q A_k d\mu \quad (7)$$

$$R_k := \int_0^\tau B_k^T Q B_k d\mu + R\tau \quad (8)$$

$$P_k := \int_0^\tau A_k^T Q B_k d\mu \quad (9)$$

This result follows by direct integration of the cost function and exact discretization of the dynamical system with the arbitrary sampling pattern  $\mathbb{T} = \{\tau_0, \tau_1, \dots, \tau_{N-1}\}$ . The details can be found in the Appendix A.

*Remark 1:* Note that cost function parameters  $Q_k$ ,  $P_k$ , and  $R_k$ , except  $S$ , become a function of the interarrivals  $\tau_k$  of the sampling pattern  $\mathbb{T}$ . The same happens to the discrete-time system parameters  $A_k$  and  $B_k$ . On the contrary  $x_k$  and  $u_k$  are the state and control action values at sampling instant  $t_k$ , that is  $x_k := x(t_k)$  and  $u_k := u(t_k)$ .

*Remark 2:* The cost function of the discretized problem depends on the product  $x_k u_k$  by means of cost weight  $P_k$  despite the continuous cost function lacks it. However, this is not a problem for optimization by dynamic programming.

If the continuous time dynamic matrix  $A$  is invertible (i.e.  $A$  is either stable or unstable but has no poles at origin), and there exists a matrix  $M$  that solves the following equation

$$A^T M + M A = Q \quad (10)$$

equations (5)-(9) can be analytically computed yielding the following expressions for a given  $\tau_k$ :

$$A_k = e^{A\tau_k} \quad (11)$$

$$B_k = (e^{A\tau_k} - I)A^{-1}B \quad (12)$$

$$Q_k = A_k^T M A_k - M \quad (13)$$

$$R_k = B^T A^{-T} (\tau_k Q + (I - A_k^T) A^{-T} Q + Q A^{-1} (I - A_k)) A^{-1} B + R\tau_k \quad (14)$$

$$P_k = (A_k^T M A_k - M + A^{-1} (I - A_k^T) Q) A^{-1} B \quad (15)$$

Problem  $P_{T_N}$  and its discrete-time counterpart are finite time problems. An infinite time cost is more desirable in order to take into account the future behaviour of the cost function once the  $N$  control actions have been applied. This can be accomplished by proper selection of the final stage cost  $S$ . For stable systems the cost of the free response (i.e.  $u(t) = 0$ ) is equal to  $-M$ , with  $M$  given by equation (10). In fact, in the free response case  $R_k$  and  $P_k$  are equal to zero, and  $Q_k$ , as given by equation (13), tends to  $-M$  as  $\tau_k \rightarrow \infty$ , that is  $\lim_{\tau_k \rightarrow \infty} Q_k = -M$  because  $\lim_{\tau_k \rightarrow \infty} A_k = 0_{n \times n}$  due to stability of  $A$ . As a result setting the final cost  $S = -M$  we are considering an infinite time optimization problem.

For unstable systems an appropriate feedback control action is required for stability. In this case we can consider a discrete-time periodic LQR with sampling time  $\tau_\infty$  with  $\tau_\infty \geq \tau_{N-1}$

and cost  $M_d$ . Setting the final cost  $S = M_d$  we are also considering an infinite time cost.

#### IV. DYNAMIC PROGRAMMING

The objective is to find the optimal sampling pattern  $\mathbb{T}^* := \{\tau_0^*, \tau_1^*, \dots, \tau_{N-1}^*\}$  and the optimal control gain sequence  $\mathbb{F}^* := \{F_0^*, F_1^*, \dots, F_{N-1}^*\}$ , with  $u_k = F_k^* x_k$ , that solve the problem given by equations (2)-(4). In this section we make use of dynamic programming to divide the original optimal sampling problem into  $N$  nested subproblems from which we can compute the optimal interarrival time. As shown in Fig. 1, at each stage we have to decide not only the optimal control gain  $F_k^*$  but also the optimal interarrival time  $\tau_k^*$ . The use of dynamic programming is an intermediate step to keep the complexity of the optimization problem bounded but it cannot solve the optimal sampling problem by itself. We begin by considering an arbitrary sampling pattern  $\mathbb{T} = \{\tau_0, \tau_1, \dots, \tau_{N-1}\}$ , with  $\sum_{k=0}^{N-1} \tau_k = T_N$ .



Fig. 1. Dynamic programming problem. At each stage  $k \in \{0, 1, \dots, N-1\}$  a control action  $u_k = F_k x_k$  is applied during the interarrival time  $\tau_k$ .

For an arbitrary intermediate stage  $k$ , the cost-to-go is given by

$$J_k = x_k^T Q_k x_k + u_k^T R_k u_k + 2x_k^T P_k u_k + J_{k+1}^*(x_{k+1}) \quad (16)$$

with  $J_{k+1}^*(x_{k+1})$  the optimal cost-to-go from stage  $k+1$ , calculated in the previous iteration of the dynamic program, and given by

$$J_{k+1}^* = x_{k+1}^T K_{k+1}^* x_{k+1} \quad (17)$$

Substituting the discrete dynamic system  $x_{k+1} = A_k x_k + B_k u_k$  in equations (16)-(17) we obtain

$$J_k = x_k^T (Q_k + A_k^T K_{k+1}^* A_k) x_k + u_k^T (R_k + B_k^T K_{k+1}^* B_k) u_k + 2u_k^T (P_k + B_k^T K_{k+1}^* A_k) x_k$$

The optimal control action is obtained by minimization of  $J_k$  with respect to  $u_k$  resulting in

$$u_k^\circ = \underbrace{-(R_k + B_k^T K_{k+1}^* B_k)^{-1}}_{F_k} (P_k + B_k^T K_{k+1}^* A_k) x_k \quad (18)$$

Note that we have written the optimized control action as  $u_k^\circ$  instead of  $u_k^*$ . This is to stress the fact that  $u_k^\circ$  depends on the interarrival time  $\tau_k$  through  $A_k$ ,  $B_k$ ,  $R_k$ , and  $P_k$ . We reserve the optimal control action  $u_k^*$  as the control action  $u_k^\circ$  for the optimal sampling interarrival  $\tau_k^*$ , that is  $u_k^* := u_k^\circ(\tau_k^*)$ .

Coming back to  $u_k^\circ$  in equation (18), for an arbitrary interarrival  $\tau_k$  it follows that the control action is proportional to the current state through  $F_k$  like in the classical LQR. Substitution of the optimal control action  $u_k^\circ$  on the cost-to-go function yields the following cost-to-go at stage  $k$

$$J_k = x_k^T K_k x_k \quad (19)$$

with

$$K_k := (Q_k + A_k^T K_{k+1}^* A_k) - (P_k + B_k^T K_{k+1}^* A_k)^T (R_k + B_k^T K_{k+1}^* B_k)^{-1} (P_k + B_k^T K_{k+1}^* A_k) \quad (20)$$

The cost-to-go function is proportional to the squared state through  $K_k$ , again like in the classical LQR. However, in contrast to periodic LQR [8], the proportional factors  $F_k$  and  $K_k$  are not constant but dependent on the interarrival  $\tau_k$ .

The optimal interarrival time  $\tau_k$  can be obtained by minimizing the cost function (20) with respect to  $\tau_k$ . In this article we consider the minimization of equation (20) for any direction  $x_k$  leading to a guaranteed cost minimization, as discussed in Section V.

Assume we have computed the optimal interarrival  $\tau_k^*$  by any means, thus the optimal control action  $u_k^*$  and optimal cost-to-go function  $J_k^*$  are given, for  $k = \{N-1, N-2, \dots, 1, 0\}$ , by

$$u_k^* := \underbrace{F_k(\tau_k^*)}_{F_k^*} x_k \quad (21)$$

$$J_k^* := x_k^T \underbrace{K_k(\tau_k^*)}_{K_k^*} x_k \quad (22)$$

Note that for  $k = N-1$  we have that  $K_N^* = S$ , so the cost-to-go function from stage  $N-1$  is

$$J_{N-1} = x_{N-1}^T Q_{N-1} x_{N-1} + u_{N-1}^T R_{N-1} u_{N-1} + 2x_{N-1}^T P_{N-1} u_{N-1} + x_{N-1}^T S x_{N-1} \quad (23)$$

##### A. Dynamic Programming Approach

Given the optimal cost-to-go gain at stage  $k+1$ , that is  $K_{k+1}^*$ , it is possible to derive the cost-to-go function gain at stage  $k$ ,  $K_k$ , as shown by equations (16)-(17). By computing the LQR optimal control action  $u_k^\circ$  given by equation (18) we are able to obtain the optimal cost-to-go  $K_k$  as a function of the interarrival time  $\tau_k$ . Note that dynamic programming itself does not solve the optimal sampling pattern problem. Instead dynamic programming defines, for each cost-to-go function  $k$ , an optimization problem which solution provides the optimal sampling pattern  $\tau_k^*$ . Furthermore, the optimization of interarrival times can be embedded into a family of similar problems such that each member of the family is easily related with the solution of the previous problem.

By the minimization of cost-to-go  $k$  defined by equations (19)-(20) we obtain at stage  $k$  the optimal interarrival  $\tau_k^*$  and the optimal cost-to-go gain  $K_k^*$ . In this way,

starting from the last stage  $N - 1$ , with cost-to-go gain defined by equation (23), we are able to compute backwards the optimal sampling pattern  $\{\tau_{N-1}^*, \tau_{N-2}^*, \dots, \tau_0^*\}$  by solving  $N$  one-dimensional optimization problems. Finally, we compute the optimal control action gain sequence  $\mathbb{F}^* := \{F_0^*, F_1^*, \dots, F_{N-1}^*\}$  as defined by equation (18).

## V. SAMPLING PATTERN COST FUNCTION

The cost-to-go function is a quadratic form of the gain  $K_k$ , that is

$$J_k := x_k^T \left( (Q_k + A_k^T K_{k+1}^* A_k) - (P_k + B_k^T K_{k+1}^* A_k)^T (R_k + B_k^T K_{k+1}^* B_k)^{-T} (P_k + B_k^T K_{k+1}^* A_k) \right) x_k \quad (24)$$

with  $Q_k$ ,  $P_k$ ,  $R_k$ ,  $A_k$ , and  $B_k$  functions of  $\tau_k$  and given by equations (11)-(15). In this article we consider the minimization of the interarrival time  $\tau_k$  for any state  $x_k$ , leading to a guaranteed cost minimization. This problem is equivalent to the minimization of the maximum eigenvalue of matrix  $K_k$  with respect to the interarrival  $\tau_k$ . Hence the optimal interarrival time  $\tau_k$  at each stage is obtained by solving the following optimization problem

$$\min_{\tau_k \in (0, \infty)} \lambda_{max}(K_k(\tau_k)) \quad (25)$$

In the next section we show the existence of solution of this optimization problem.

### A. Well-Posedness Optimization Problem

A necessary condition for the maximum singular value of problem (25) to be depend on  $\tau_k$  is that the rank of  $A^{-1}B$  is equal to the rank of  $Q$ , that is, the input is able to change the cost induced by the state  $x(t)$ . Otherwise the maximum eigenvalue might be insensitive to the control  $u(t)$  and hence to the value of  $\tau_k$ . In this case, one possible solution is to minimize the trace instead of the maximum eigenvalue, that is

$$\min_{\tau_k \in (0, \infty)} \text{trace}(K_k(\tau_k)) \quad (26)$$

because now the effects of the control action are captured in the cost function value. A similar discussion can be performed with the cost function dependent on the dynamical system output

$$\min_{u \in \mathbb{U}_N} \int_0^{T_N} (y^T Q y + u^T R u) dt + y(T_N)^T S y(T_N) \quad (27)$$

s.t.

$$y = Cx$$

$$\dot{x} = Ax + Bu$$

$$x(0) = x_0$$

where  $y(t) \in \mathbb{R}^p$  is the output of the dynamical system with  $Q \in \mathbb{R}^{p \times p}$ ,  $S \in \mathbb{R}^{p \times p}$ , and  $C \in \mathbb{R}^{p \times n}$ , in such a way that  $CA^{-1}B$  has rank  $p$ . In this case both optimization problems (25) and (26) are well-posed.

### B. Boundedness of $\tau_k$

At each stage we have to compute the optimal interarrival time  $\tau_k$ . In this section we show that the range of  $\tau_k$  is a bounded interval and that there exists a value  $\tau_k^*$  inside this interval that provides a minimum.

Consider the standard discrete LQR with constant sampling time. The cost-to-go gain  $K_k$  is obtained in this case as an iteration of the Riccati Difference Equation (RDE) taken  $\tau_k = \tau_{k+1}^*$  and, as initial matrix,  $K_{k+1}^*$ . By the convergence and monotonicity properties of the RDE [9], the cost-to-go gain  $K_k$  is monotonically decreasing, then we have that  $K_k(\tau_{k+1}^*) < K_{k+1}(\tau_{k+1}^*)$ . In case that  $\tau_k = 0$  there is no cost reduction<sup>1</sup>, that is  $K_{k+1} = K_k$ . As a result  $\tau_k = 0$  is not optimal, thus it must be the case that  $\tau_k^* > 0$ .

Now we show that the value  $\tau_k$  is upper bounded. For stable systems if  $\tau_k = \infty$  the cost-to-go gain is equal to the free response  $K_k = -M$ , hence the optimum  $\tau_k$  must be finite for effective cost reduction. For unstable systems  $\tau_k$  must be finite otherwise  $K_k = \infty$ . As a result  $\tau_k < \infty$ . For practical purposes  $\tau_k$  may be bounded by the system settling time.

## VI. PROPOSED OPTIMIZATION ALGORITHM

In this section we design the optimization algorithm to solve the optimal sampling control problem. The algorithm computes the optimal sampling pattern  $\mathbb{T}^* := \{\tau_0^*, \tau_1^*, \dots, \tau_{N-1}^*\}$  and the optimal control action gain sequence  $\mathbb{F}^* := \{F_0^*, F_1^*, \dots, F_{N-1}^*\}$  in a reverse order, as can be seen in the pseudocode of Fig. 2. The amplitude of the optimal control action  $u_k$  is computed in closed-loop as  $u_k = F_k^* x_k$  for  $k \in 0, 1, \dots, N - 1$ .

The optimal sampling pattern at stage  $k$  is computed by minimizing the maximum eigenvalue of  $K_k$ , as show in the minimization problem (25). For the optimization algorithm it is important to stress that  $K_k$  is a function of the decision variable  $\tau_k$  and the optimal cost-to-go of stage  $k + 1$ , that is  $K_k(\tau_k, K_{k+1}^*)$ . Note that the initialization is to set  $K_N^* = S$ . In this way the optimal sampling pattern is computed backward from stage  $N - 1$  to stage 0.

Being the optimization problem univariate and with decision variable domain convex and bounded (i.e.  $0 < \tau_k^* < \tau_{max}$ ), we propose its minimization by a gradient-descent algorithm. Recall that  $\tau_{max}$  is the settling time of the dynamical system free response.

The proposed optimization approach is able to provide a certificate of global optimality by solving the optimization problem twice at each stage. One optimization with starting point at  $\tau_k = 0$ , and the other optimization with starting point at  $\tau_k = \tau_{max}$ . If the optimal decision variables of both problems are equal, then the optimum is global.

At the same time that the optimal sampling pattern is computed, the optimal control action gain  $F_k(\tau_k^*)$ , as given by equation (18), is also computed. Finally, we obtain the optimal control action amplitude in closed loop as  $u_k = F_k^* x_k$ .

<sup>1</sup>We assume that the LQR formulation provides a solution that is different to the free response, otherwise  $u_k = 0$  is optimal.

Fig. 2. Optimal Sampling Pattern algorithm

```

1: procedure OPTIMALSAMPLING( $N, Q, R, S, A, B$ )
    $K_N^* \leftarrow S$ 
2:   for  $k \leftarrow (N - 1), 0$  do           ▷ Compute  $\mathbb{T}^*$  and  $\mathbb{F}^*$ 
3:      $\tau_k^* \leftarrow \min_{\tau_k} K_k(\tau_k, K_{k+1}^*)$  ▷ Defined by eq. (25)
4:      $K_k^* \leftarrow K_k(\tau_k^*, K_{k+1}^*)$      ▷ Defined by eq. (20)
5:      $F_k^* \leftarrow F_k(\tau_k^*)$            ▷ Defined by eq. (18)
6:   end for
7: end procedure

```

## VII. APPLICATION EXAMPLES

### A. Example 1: Second Order SISO System

Let us define a SISO continuous system as:

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx \end{aligned} \quad (28)$$

$$A = \begin{pmatrix} -10 & -5 \\ 5 & -10 \end{pmatrix}, \quad B = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad C = \begin{pmatrix} 1 & 0 \end{pmatrix}$$

Where the cost function weights are:

$$Q = 100 \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, \quad R = 1$$

The optimal interarrival times and control gains are obtained by the algorithm in Fig. 2 with  $N = 4$ . Figure 3 shows the maximal eigenvalue of the quadratic  $k$ -step cost  $K_k$  as a function of  $\tau_k$ . As can be seen, all functions fit to the same curve as  $\tau$  is increased. This curve corresponds to the standard discrete LQR optimal cost for constant sample time  $\tau$ . The optimal sampling pattern is  $\mathbb{T}^* := \{\tau_0^* = 0.0242, \tau_1^* = 0.0317, \tau_2^* = 0.0464, \tau_3^* = 0.0876\}$ . Note that all interarrival times are decreasing. The figure also shows that all the cost functions are quasiconvex and therefore there is an unique minimum value. The computation time for the complete algorithm was  $62ms$  in an Intel i7-8565U 1.8GHz.

Figure 4 compares the continuous time LQR optimal control action  $u^*(t)$  with the optimal sampling pattern control action  $u_k^*$ . The piecewise control action  $u_k^*$  approximates the continuous time control action  $u^*(t)$  for the first time instants, however beyond 0.2 sec there is no similarity between both control actions.

Finally, Figure 5 compares the optimal continuous time LQR output  $y^*(t)$  with the output  $y_k^*$  obtained with optimal state feedback controller with optimal sampling pattern previously computed in Fig 3. The numerical difference between the cost of the continuous time LQR cost and the cost achieved with the optimal sampling pattern with  $N = 4$  is only of 0.32%.

In order to compare our proposed algorithm with the one proposed in [5], the quantization-based sampling has been implemented for this example. In the cited work, the final time has to be set before the optimization. A first final time guess could be the optimal continuous state feedback controller settling time of  $T_f = 0.5s$ . With  $N = 4$ , the quantization-based sampling cost is 2.31% greater than the optimal continuous controller. If the algorithm is set with the

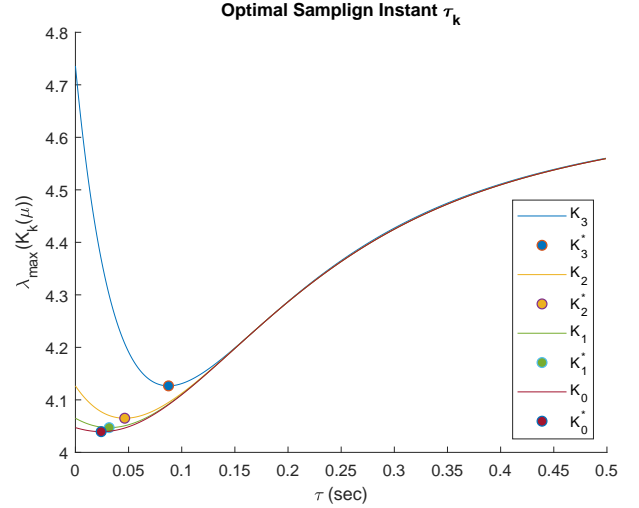


Fig. 3. Evolution of the maximum eigenvalue of the cost-to-go gain  $K_k(\tau)$  as a function of the interarrival time  $\tau$ . The filled dots mark the optimal interarrival time  $\tau_k^*$  for  $k = 0, 1, 2, 3$  that minimizes the maximum eigenvalue of  $K_k(\tau_k)$ .

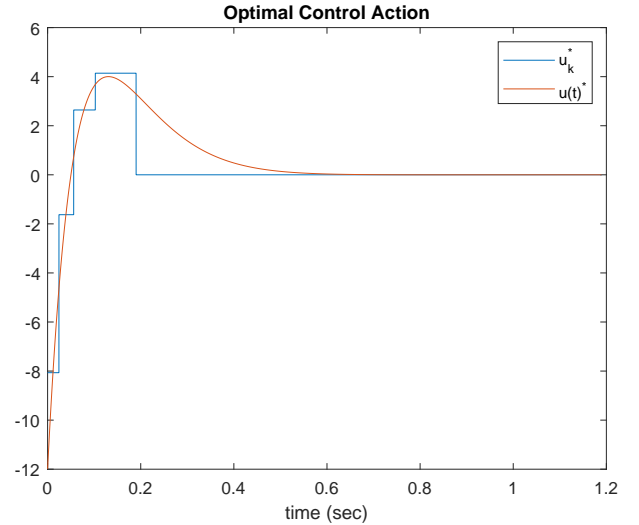


Fig. 4. Optimal control action  $u_k^*$  (blue) for the optimal sampling pattern with  $N = 4$  compared to the continuous-time optimal LQR control action  $u^*(t)$  (red).

optimal final time  $T_f^* = 0.2s$  the cost is only 0.82% greater than the continuous one and closer to the optimal of 0.32% obtained with our algorithm. Therefore, it can be stated that the knowledge of optimal final time is a decisive parameter for [5].

It has been also tested that both algorithms tend to the continuous cost as  $N$  is increased. For our presented algorithm, the error is  $8.6 \cdot 10^{-4}\%$  with  $N = 100$  and the computation time is  $0.54s$ , thus showing convergence.

### B. Example 2: IoT Heating System

In this example we consider the control of a home heating system. The house is divided into 5 rooms as presented in Fig. 6. Each room is equipped with a water radiator. Each radiator in rooms 1, 2, and 3 are equipped with a wireless

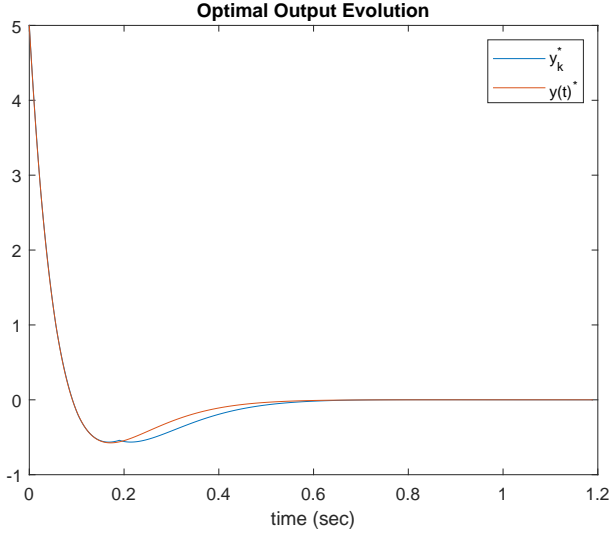


Fig. 5. Output of the system  $y_k^*$  (blue) obtained with the optimal sampling pattern with  $N = 4$  compared to the continuous-time optimal LQR output  $y^*(t)$  (red)

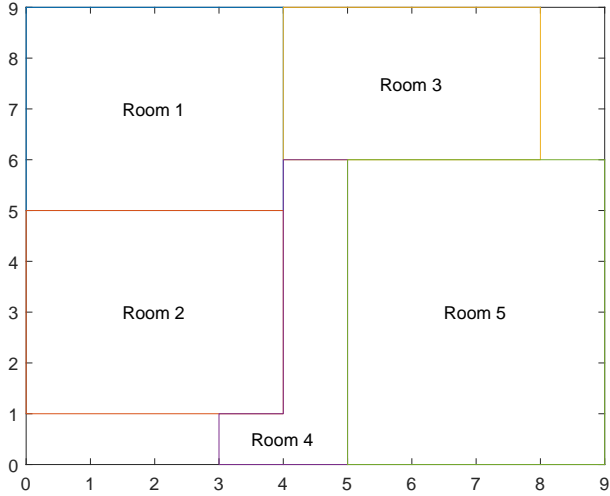


Fig. 6. House rooms distribution for the home heating system.

battery powered solenoid valve actuators. Radiators in rooms 4 and 5 share a unique wireless battery powered solenoid valve actuator. In this case limiting the movements in the solenoid valve is decisive in order to preserve the battery life.

The system has been modelled by considering that the temperatures of the five rooms are known together with the outside temperature. The control actions are measured in Watts. The thermal conductivity of the interior walls is  $K_i = 0.887 \frac{W}{m^{20}K}$ , and that of the exterior walls is  $K_e = 0.298 \frac{W}{m^{20}K}$ . We have also considered the conductivity of the doors  $K_d = 2.32 \frac{W}{m^{20}K}$ . The ceiling height is  $2.5m$ , the specific heat capacity is  $C = 1005.3 \frac{J}{KgK}$  and its density  $\rho = 1.22 \frac{Kg}{m^3}$ . Taking into account this parameters, the next model is obtained:

$$\dot{x} = Ax + Bu + Ev \quad (29)$$

where  $x = [T_1, T_2, T_3, T_4, T_5]^T$  are the room temperatures in Celsius,  $u = [Q_1, Q_2, Q_3, Q_{45}]^T$  are the water radiators power

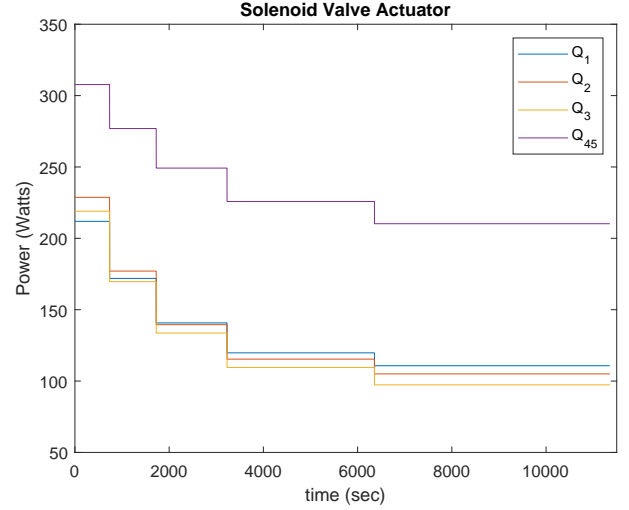


Fig. 7. Optimal sampling pattern control actions for each solenoid valve.

in Watts and  $v$  is the outside temperature. The dynamical system matrices are:

$$A = 10^{-4} \begin{pmatrix} -5.17 & 1.79 & 1.34 & 0.81 & 0 \\ 1.79 & -5.46 & 0 & 2.61 & 0 \\ 1.43 & 0 & -5.35 & 0.86 & 1.91 \\ 1.85 & 5.97 & 1.85 & -19.23 & 6.99 \\ 0 & 0 & 1.19 & 2.04 & -4.24 \end{pmatrix}$$

$$B = 10^{-5} \begin{pmatrix} 2.02 & 0 & 0 & 0 \\ 0 & 2.02 & 0 & 0 \\ 0 & 0 & 2.16 & 0 \\ 0 & 0 & 0 & 1.39 \\ 0 & 0 & 0 & 0.94 \end{pmatrix}$$

$$E = 10^{-4} (1.21 \quad 1.06 \quad 1.13 \quad 2.55 \quad 1.01)^T$$

The optimization problem allows only 4 solenoid valve movements, that is  $N = 4$ , and the cost function weights are:

$$Q = 100 \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \quad R = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

The optimal sampling pattern control is solved with a computing time is of  $0.313s$ . The optimal control actions for the actuators are shown in Fig. 7. As in the previous example the interarrival time of the optimal sampling pattern increases with time.

The controlled system response is presented in Figure 8. It can be seen that temperature of room 4 is not controlled as it has weight 0 in the cost function.

The final cost of the trajectory for the optimal sampling strategy is  $J = 172.517$ . The optimal cost of the continuous LQR controller is  $J_c = 170.572$ . The relative error is  $\frac{J - J_c}{J_c} = 1.1\%$ . As a result with the optimal sampling pattern the cost with respect to the optimal continuous solution is increased

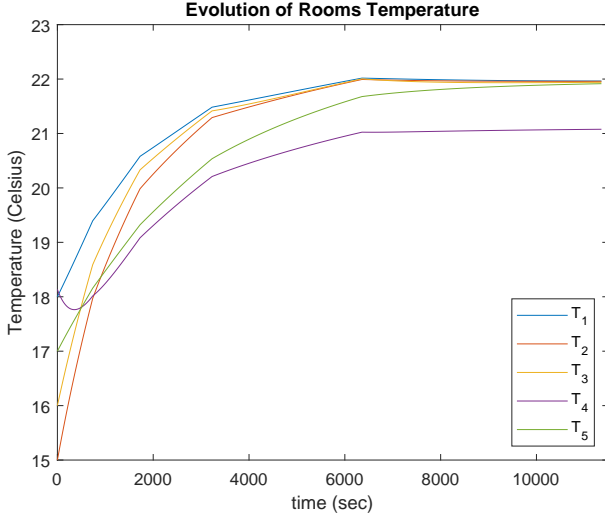


Fig. 8. Evolution of temperatures in each room. Note that temperature of room 4 is not taken into account in the cost function.

only in 1.1%. If the number of movements are increase to  $N = 9$  the relative error is decreased to 0.25%.

### VIII. CONCLUSION AND FUTURE WORK

In this article we have solved the optimal sampling problem for free final time LQR of arbitrary order and with multiple inputs and outputs. We have shown how the optimal sampling pattern problem, that is to decide the optimal sampling instants, can be embedded into a family of similar problems yielding an algorithm that is optimal for any value of  $N \geq 1$ , scalable with respect to  $N$ , recursive, and constructive. The proposed strategy is natural for many potential real-world applications of chemical, biological or medical processes, where the treatment application times is also a variable of interest. A current area of research is the extension of the presented results to include robustness properties on the sampling pattern and its extension to problems that are non-linear.

#### APPENDIX A

##### EXACT PROBLEM DISCRETIZATION

Consider the linear quadratic regulator problem

$$P_{T_N} : \quad \min_{u \in \mathbb{U}_N} \int_0^{T_N} (x^T Q x + u^T R u) dt + x(T_N)^T S x(T_N) \quad (30)$$

s.t.

$$\dot{x} = Ax + Bu$$

$$x(0) = x_0$$

The exact discretization of the dynamic equation with  $t > t_k$ , is

$$x(t) = \underbrace{e^{A(t-t_k)}}_{A_k} x_k + \underbrace{(e^{A(t-t_k)} - I)A^{-1}B}_{B_k} u_k \quad (31)$$

Given a sampling pattern  $\mathbb{T} = \{\tau_0, \tau_1, \dots, \tau_{N-1}\}$ , cost function (30) can be rewritten as the sum of  $N$  integrals

$$\int_0^{T_N} (x^T Q x + u^T R u) dt + x(T_N)^T S x(T_N) = \sum_{k=0}^{N-1} \int_{t_k}^{t_{k+1}} (x^T Q x + u^T R u) dt + x(T_N)^T S x(T_N) \quad (32)$$

For each integral term in (32) let us substitute the state  $x(t)$  by (31)

$$\begin{aligned} & \int_{t_k}^{t_{k+1}} (x^T Q x + u^T R u) dt = \\ & \int_{t_k}^{t_{k+1}} (A_k x_k + B_k u_k)^T Q (A_k x_k + B_k u_k) + u_k^T R u_k dt \\ & = x_k^T \underbrace{\int_{t_k}^{t_{k+1}} A_k^T Q A_k dt}_{Q_k} x_k + \\ & \quad + 2x_k^T \underbrace{\int_{t_k}^{t_{k+1}} A_k^T Q B_k dt}_{P_k} u_k + \\ & \quad + u_k \underbrace{\int_{t_k}^{t_{k+1}} (B_k^T Q B_k + R) dt}_{R_k} u_k \end{aligned}$$

Computing the integrals  $Q_k$ ,  $P_k$ , and  $R_k$  results in

$$\begin{aligned} Q_k &= A_k^T M A_k - M \\ R_k &= B^T A^{-T} (\tau_k Q + (I - A_k^T) A^{-T} Q + \\ & \quad + Q A^{-1} (I - A_k)) A^{-1} B + R \tau_k \\ P_k &= (A_k^T M A_k - M + A^{-1} (I - A_k^T) Q) A^{-1} B \end{aligned}$$

#### REFERENCES

- [1] J. Zheng and A. Jamalipour, *Wireless sensor networks : a networking perspective*. Wiley-IEEE Press, 2009.
- [2] X. Ge, F. Yang, and Q.-L. Han, "Distributed networked control systems: A brief overview," *Information Sciences*, vol. 380, pp. 117–131, feb 2017.
- [3] T. Tran and Q. Ha, "Dependable control systems with Internet of Things," *ISA Transactions*, vol. 59, pp. 303–313, nov 2015.
- [4] K. J. K. J. Astrom and B. Wittenmark, *Computer-controlled systems : theory and design*. Prentice Hall, 1997.
- [5] E. Bini and G. M. Buttazzo, "The optimal sampling pattern for linear control systems," *IEEE Transactions on Automatic Control*, vol. 59, no. 1, pp. 78–90, 2014.
- [6] X. Xu and P. Antsaklis, "Optimal Control of Switched Systems Based on Parameterization of the Switching Instants," *IEEE Transactions on Automatic Control*, vol. 49, no. 1, pp. 2–16, jan 2004. [Online]. Available: <http://ieeexplore.ieee.org/document/1259451/>
- [7] M. Egerstedt, Y. Wardi, and H. Axelsson, "Transition-Time Optimization for Switched-Mode Dynamical Systems," *IEEE Transactions on Automatic Control*, vol. 51, no. 1, pp. 110–115, jan 2006. [Online]. Available: <http://ieeexplore.ieee.org/document/1576861/>
- [8] B. Anderson and J. Moore, *Optimal Filtering*. Englewood Cliffs, NJ: Prentice-Hall, 1979.
- [9] R. R. Bitmead and M. Gevers, "Riccati Difference and Differential Equations: Convergence, Monotonicity and Stability," in *The Riccati Equation*. Berlin, Heidelberg: Springer Berlin Heidelberg, 1991, pp. 263–291. [Online]. Available: [http://link.springer.com/10.1007/978-3-642-58223-3\\_10](http://link.springer.com/10.1007/978-3-642-58223-3_10)