Whither Discrete Time Model Predictive Control?

G. Pannocchia, J.B. Rawlings, D.Q. Mayne, G.M. Mancuso

Published in *IEEE Transactions on automatic control*, 2015, v. 60 (1), pp. 246-252

Whither discrete time model predictive control?

Gabriele Pannocchia and James B. Rawlings and
David Q. Mayne and Giulio M. Mancuso

Abstract—**This note proposes an efficient computational procedure for the continuous time, input constrained, infinite horizon, linear quadratic regulator problem (CLQR). To ensure satisfaction of the constraints, the input is approximated as a piecewise linear function on a finite time discretization. The solution of this approximate problem is a standard quadratic program. A novel lower bound on the infinite dimensional CLQR problem is developed, and the discretization is adaptively refined until a user supplied error tolerance on the CLQR cost is achieved. The offline storage of the required quadrature matrices at several levels of discretization tailors the method for online use as required in model predictive control (MPC). The performance of the proposed algorithm is then compared with the standard *discrete time* MPC algorithms. The proposed method is shown to be significantly more efficient than standard discrete time MPC that uses a sample time short enough to generate a cost close to the CLQR solution.**

I. INTRODUCTION

In this paper we are concerned with the infinite horizon, *continuous time* optimal control problem for a linear system subject to input bounds. For brevity, we refer to this as the CLQR (constrained linear quadratic regulator) problem. It is perhaps the *simplest* optimal control problem of significant interest after the classical, unconstrained LQR. One of the compelling features of both LQR and CLQR is the guarantee of nominal, closed-loop stability that they provide for unconstrained and input constrained linear systems, respectively. Model predictive control, which is based on implementing solutions to optimal control problems as state measurements (or state estimates) become available, is arguably the most important advanced industrial control design method in use today. Besides linearity, however, another feature of almost all industrial MPC methods is the use of *discrete time* models. It is this use of discrete time that we would like to examine in this paper. Is it necessary? Is it convenient? Is it as good as, or even better than, continuous time? To provide a sound basis for comparison, we first develop and present a new algorithm for solving the CLQR problem. The problem is *doubly* infinite dimensional, first because the input is a continuous time *function*, and second because the cost is defined on an *infinite* horizon. We show that neither feature causes insurmountable computational difficulties, and we can solve this problem reasonably efficiently with a guarantee of proximity to optimality.

The paper is organized as follows. In Section II, we develop the basic numerical discretization of the continuous time problem using a piecewise linear input parameterization. In Section III, we present quadrature formulas, based on matrix exponentiation, that can be computed and stored offline for fast, repetitive online calculation, as is required in MPC. Because the original CLQR problem is (strictly) convex, we are able to develop a novel lower bound on the optimal cost; this is presented in Section IV. This lower bound enables a stopping criterion that meets a user specified proximity to optimality. In Section V, we propose an algorithm for refining the discretization to solve the CLQR problem and discuss stopping criteria. Next in Section VI, we show that this algorithm converges. In Section VII, we provide numerical examples that show the algorithm can be solved quickly. Finally in Section VIII we draw conclusions of the study.

Notation: $\mathbb{R}$ and $\mathbb{Z}$ denote the fields of reals and integers, respectively. Given $a, b \in \mathbb{Z}$ $\mathbb{Z}_{a:b}$ denotes $\{x \in \mathbb{Z} \mid a \leq x \leq b\}$. The symbol $'$ is the transpose operator. Inequalities are meant: component wise for vectors, in positive (semi)definite sense for symmetric matrices. Given $x, y \in \mathbb{R}^n$, $(x, y) \triangleq \left[\begin{smallmatrix} x \\ y \end{smallmatrix}\right]$, $\langle x, y\rangle$ denotes the inner product, $|x| \triangleq \sqrt{\langle x, x\rangle}$, $|x|_Q^2 \triangleq \langle x, Qx\rangle$. For $A \in \mathbb{R}^{n\times m}$ and $a, b, c, d \in \mathbb{Z}_{>0}$, $A_{a:b,c:d}$ denotes the submatrix with rows a to b and columns c to d, and $A_{a:b,:}$ denotes the submatrix of rows a to b and all columns, $\mathbf{I}_m$ is the identity matrix (dimension $m \times m$, possibly omitted), $\lambda_{\min}(A)$ is the smallest eigenvalue of a symmetric matrix A, and $\mathrm{int}(S)$ is the interior of S.

II. PRELIMINARIES

A. Continuous time optimal control problem

In this paper we address the computation of the optimal solution to the continuous time, infinite horizon, input constrained, linear quadratic regulation problem:

$$\mathbb{P}_\infty(x):\ \inf_{u(\cdot)} \left\{ V_\infty(x, u(\cdot)) \triangleq \int_0^\infty \ell(x(t), u(t))dt \right\}, \tag{1a}$$

s.t. $x(0) = x$,

$$\dot{x} = f(x, u) \triangleq Ax + Bu, \quad \text{for all } t \in [0, \infty), \text{ and} \tag{1b}$$

$$u(\cdot) \in \mathcal{U}_\infty, \tag{1c}$$

in which $\mathcal{U}_\infty$ is the class of measurable controls defined on $[0, \infty)$ and taking values in $\mathbb{U} = \prod_{i=1}^m \mathbb{U}_i$, where $\mathbb{U}_i \triangleq [u_i^{\min}, u_i^{\max}]$, $0 \in \mathrm{int}(\mathbb{U})$, which is a compact, convex subset of $\mathbb{R}^m$. The state $x \in \mathbb{R}^n$, and the function $\ell(\cdot)$ is quadratic: $\ell(x, u) \triangleq \frac{1}{2}(x'Qx + u'Ru)$.

Assumption 1: The pair (A, B) is stabilizable and $(Q^{1/2}, A)$ is observable. $Q \geq 0$ and $R > 0$.

As discussed in [1], Assumption 1 can be relaxed to $(Q^{1/2}, A)$ detectable. Let $\mathbb{X}_\infty$ be the set of initial states x for which there exists $u(\cdot) \in \mathcal{U}_\infty$ such that $V_\infty(x, u(\cdot))$ is finite. Thus $V_\infty : \mathbb{X}_\infty \times \mathcal{U}_\infty \to \mathbb{R}_{\geq 0}$. Existence and uniqueness of a solution to $\mathbb{P}_\infty(x)$ for each $x \in \mathbb{X}_\infty$ is established after (4).

In order to rewrite the *infinite horizon* problem $\mathbb{P}_\infty(x)$ as an equivalent *finite horizon* problem, we define a suitable ellipsoid invariant set. Let P be the unique symmetric positive definite solution to the (continuous time) algebraic Riccati equation:

$$0 = Q + A'P + PA - PBR^{-1}B'P. \tag{2}$$

Given a positive scalar α, we consider the compact set: $\mathbb{X}_f \triangleq \{x \in \mathbb{R}^n \mid x'Px \leq \alpha\}$. Clearly, $\mathbb{X}_f$ is an invariant (ellipsoidal) set for the *unconstrained* closed-loop system: $\dot{x} = Ax + Bu$, in which $u = Kx$, with $K = -R^{-1}B'P$. Because $\mathbb{U}$ contains the origin in its interior, if α is chosen sufficiently small, $x \in \mathbb{X}_f$ implies $Kx \in \mathbb{U}$. Hence, $u(t) = Kx(t)$ remains feasible at all times with respect to the constraint (1c) once $x(t)$ has entered $\mathbb{X}_f$. Let $\mathbb{U}$ be rewritten as $\{u \in \mathbb{R}^m \mid Du \leq d\}$. Given the eigenvalue decomposition $P = S\Lambda S'$, it is easy to show that the largest admissible value of α is given by: $\alpha = \min_i \left\{ \frac{d_i}{|M_{i,:}|^2} \right\}$ with $M = DKS\Lambda^{-1/2}$.

Given $T \in \mathbb{R}_{>0}$, we replace $\mathbb{P}_\infty(x)$ by the following finite horizon optimal control problem:

$$\mathbb{P}_T(x):\ \min_{u(\cdot)} \left\{ V_T(x, u(\cdot)) \triangleq \int_0^T \ell(x(t), u(t))dt + V_f(x(T)) \right\},$$

$$\text{s.t. } x(0) = x, \tag{3a}$$

$$\text{model (1b) for all } t \in [0, T], \text{ and } u(\cdot) \in \mathcal{U}_T, \tag{3b}$$

G. Pannocchia is with Dept. of Civil and Industrial Eng., Univ. of Pisa, Italy (g.pannocchia@diccism.unipi.it). J.B. Rawlings is with Dept. of Chemical and Biological Eng., Univ. of Wisconsin, Madison, USA (rawlings@engr.wisc.edu). D.Q. Mayne is with Dept. of Electrical & Electronic Eng., Imperial College London, UK (d.mayne@imperial.ac.uk). G.M. Mancuso is with United Technologies, Rome, Italy (giuliomancuso@gmail.com).

in which: $V_f(x) \triangleq \frac{1}{2}x'Px$ with $P > 0$ computed from (2), $\mathcal{U}_T$ is the class of measurable controls defined on $[0,T]$ and taking values in $\mathbb{U}$. Thus, $V_T : \mathbb{R}^n \times \mathcal{U}_T \to \mathbb{R}_{\geq 0}$. $\mathbb{P}_T(x)$ has a unique solution for any $x \in \mathbb{R}^n$ [2, Thm. 14, Chapter 3]. Let $u_T^0(\cdot)$ be the (finite time) input trajectory solution to $\mathbb{P}_T(x)$ and $x_T^0(\cdot)$ the associated (finite time) state trajectory.

Proposition 2 (See [1]): For each $x \in \mathbb{X}_\infty$, there exists $\bar{T} \in \mathbb{R}_{>0}$ such that $x_T^0(T) \in \mathbb{X}_f$ $\forall T \geq \bar{T}$, and $\lim_{T\to\infty} x_T^0(T) = 0$.

For $x \in \mathbb{X}_\infty$, if $T \in \mathbb{R}_{\geq 0}$ is large enough that $x_T^0(T) \in \mathbb{X}_f$, since $V_f(x)$ is the optimal infinite horizon cost for any $x \in \mathbb{X}_f$, by the principle of optimality it follows that the (infinite time) input and state trajectories defined as:

$$\left(u_\infty^0(\cdot), x_\infty^0(\cdot)\right) \triangleq \begin{cases} \left(u_T^0(t), x_T^0(t)\right) & \text{if } t \in [0,T], \\ (Ke^{(A+BK)(t-T)}x_T^0(T), e^{(A+BK)(t-T)}x_T^0(T)) & \text{if } t > T, \end{cases} \tag{4}$$

are, respectively, the minimizer of $\mathbb{P}_\infty(x)$ and its associated state trajectory. Thus, $\mathbb{P}_\infty(x)$ and $\mathbb{P}_T(x)$ yield the same solution, i.e. $V_T^0(x) \triangleq V_T(x, u_T^0(\cdot)) = V_\infty^0(x) \triangleq V_\infty(x, u_\infty^0)$. From the above discussion it follows that $\mathbb{P}_\infty(x)$ has a unique solution for all $x \in \mathbb{X}_\infty$. In discrete time, earlier but less general results on existence and uniqueness of solutions to $\mathbb{P}_\infty$ can be found in [3], [4].

There is a rich literature on solution methods for finite horizon nonlinear constrained optimal control. In most approaches a piecewise constant input parameterization is considered and numerical discretization is deployed to derive and solve (approximate) optimality conditions (see e.g. [5]–[7] and references therein). On the other hand, methods specifically tailored to constrained linear systems are less common, but some interesting results and methods can be found in [8]–[14]. We remark that one distinguishing feature of our method is that it computes a solution to $\mathbb{P}_\infty(x)$ that is accurate to a user defined tolerance. Furthermore, our method is based on the solution of strictly convex quadratic programming problems, for which reliable algorithms exist, and has no specific restriction on the system (state and input) dimensions.

Remark 3: We are assuming that the actuator hardware, if digital, is able to implement the continuous time input solution to $\mathbb{P}_\infty(x)$ without introducing noticeable discretization effects.

B. Input parameterizations and discretized optimal control problem

For all $T \in \mathbb{R}_{>0}$, let γ be a *partition* of the interval $[0,T]$, defined as a sequence of $J_\gamma \in \mathbb{Z}_{>0}$ intervals $\{I_j \triangleq [t_j, t_{j+1}] \mid j \in \mathbb{Z}_{0:J_\gamma-1}\}$ such that $0 = t_0 < t_1 < \cdots < t_{J_\gamma} = T$. Let $\Delta_j \triangleq t_{j+1} - t_j$ denote the length of I_j; we assume that each Δ_j satisfies $\Delta_j = 2^{q_j}\delta$ with $q_j \in \mathbb{Z}_{\geq 0}$ and $\delta > 0$, in which case we say that $\gamma \in \Gamma_\delta^T$. In order to consider a finite parameterization of the function $u : [0,T] \to \mathbb{R}^m$, it is customary in sampled data control of continuous time systems (see, e.g. [15]) to assume that the input is constant in each interval I_j, i.e.,

$$u(t) = u_j \qquad \text{for all } t \in I_j. \tag{5}$$

Formally, given a partition γ of $[0,T]$, we define $\mathcal{U}_T^{\gamma,\mathrm{ZOH}}$ as the set of all functions $u(\cdot) \in \mathcal{U}_T$ satisfying the zero-order hold (ZOH) parameterization (5) in which $u_j \in \mathbb{U}$ for all $j \in \mathbb{Z}_{0:J_\gamma-1}$. Since $u(\cdot)$ is piecewise constant it is measurable. Besides the fact that restricting $u(\cdot)$ to the set $\mathcal{U}_T^{\gamma,\mathrm{ZOH}}$ makes $\mathbb{P}_T(x)$ finite dimensional, it also ensures that $u(t) \in \mathbb{U}$ for all $t \in [0,T]$. In [16] we argued that a *better* choice is to assume the input piecewise *linear* in each interval:

$$u(t) = (1-\eta_j(t))u_j + \eta_j(t)v_j, \text{ for all } t \in I_j, \text{ with } \eta_j(t) \triangleq \frac{t-t_j}{\Delta_j}. \tag{6}$$

Formally, given a partition γ of $[0,T]$, we define $\mathcal{U}_T^{\gamma,\mathrm{PWLH}}$ as the set of all functions $u(\cdot) \in \mathcal{U}_T$ satisfying the piecewise linear hold (PWLH) parameterization (6) in which $(u_j, v_j) \in \mathbb{U}^2$ for all $j \in \mathbb{Z}_{0:J_\gamma-1}$. Notice that for all $j \in \mathbb{Z}_{0:J_\gamma-1}$, we have that $\eta_j(t_j) = 0$ and $\eta_j(t_{j+1}) = 1$. Thus, if $(u_j, v_j) \in \mathbb{U}^2$, then $u(t) \in \mathbb{U}$ for all $t \in I_j$, all $j \in \mathbb{Z}_{0:J_\gamma-1}$.

Given a partition γ and choosing either ZOH or PWLH, i.e., $\mathcal{U}_T^\gamma \triangleq \mathcal{U}_T^{\gamma,\mathrm{ZOH}}$ or $\mathcal{U}_T^\gamma \triangleq \mathcal{U}_T^{\gamma,\mathrm{PWLH}}$, we can obtain a suboptimal solution to $\mathbb{P}_T(x)$ by solving the discretized optimal control problem:

$$\mathbb{P}_T^\gamma(x): \quad \min_{u(\cdot)\in\mathcal{U}_T^\gamma} V_T(x, u(\cdot)) \qquad \text{s.t. } x(0) = x \text{ and model (1b)}. \tag{7}$$

In most existing approaches to solve CLQR (and general nonlinear optimal control) problems, the time partition is uniform, but a number of methods exist which take advantage of (offline predetermined) non-uniform partition schemes [12], [16], [17].

III. ODE SOLVER FREE DISCRETIZATION

A. LQR discretization for ZOH via matrix exponential

Given an interval I_j, assuming to use the ZOH parameterization (5), it is well known that we can compute an equivalent discrete time system evolution as: $x_{j+1} = A_j x_j + B_j u_j$, in which $x_j \triangleq x(t_j)$ and $A_j = e^{A\Delta_j}$, $B_j = \int_0^{\Delta_j} e^{As}B ds$. Moreover, we have that: $V_T(x, u(\cdot)) = \sum_{j=0}^{J_\gamma-1} \ell_j(x_j, u_j) + V_f(x(T))$, with $\ell_j(x_j, u_j) \triangleq \int_{t_j}^{t_{j+1}} \ell(x,u)dt = \frac{1}{2}(x_j'Q_jx_j + u_j'R_ju_j + 2x_j'M_ju_j)$, in which $\left[\begin{smallmatrix} Q_j & M_j \\ M_j' & R_j \end{smallmatrix}\right] = \int_0^{\Delta_j} e^{\left[\begin{smallmatrix} A & B \\ 0 & 0 \end{smallmatrix}\right]'s} \left[\begin{smallmatrix} Q & 0 \\ 0 & R \end{smallmatrix}\right] e^{\left[\begin{smallmatrix} A & B \\ 0 & 0 \end{smallmatrix}\right]s} ds$. The matrices $(A_j, B_j, Q_j, R_j, M_j)$ can be obtained by solving a system of ordinary differential equations (ODE). However, Van Loan [18] showed that all of these matrices can be found by means of a single matrix exponentiation

$$C \triangleq \begin{bmatrix} -A' & I & 0 & 0 \\ & -A' & Q & 0 \\ & & A & B \\ & & & 0 \end{bmatrix}, \quad e^{C\tau} \triangleq \begin{bmatrix} F_1(\tau) & G_1(\tau) & H_1(\tau) & K_1(\tau) \\ & F_2(\tau) & G_2(\tau) & H_2(\tau) \\ & & F_3(\tau) & G_3(\tau) \\ & & & F_4(t) \end{bmatrix},$$
$$A_j = F_3(\Delta_j), \quad B_j = G_3(\Delta_j),$$
$$Q_j = F_3'(\Delta_j)G_2(\Delta_j), \quad M_j = F_3'(\Delta_j)H_2(\Delta_j),$$
$$R_j = R\Delta_j + \left[B'F_3'(\Delta_j)K_1(\Delta_j)\right] + \left[B'F_3'(\Delta_j)K_1(\Delta_j)\right]'. \tag{8}$$

B. LQR discretization for PWLH via matrix exponential

Numerical experience shows that computation of $(A_j, B_j, Q_j, R_j, M_j)$ for ZOH via matrix exponential formulas (8) is faster and (typically) more accurate than via an ODE solver. A similar procedure can be implemented for PWLH. To this aim, in each interval I_j, we can define an augmented state $z \triangleq (z^{(1)}, z^{(2)}) \in \mathbb{R}^{2n}$, with $z^{(1)}(t) \triangleq x(t)$ and $z^{(2)}(t) \triangleq u(t) - u_j = \eta_j(t)(v_j - u_j)$, and a constant input $w_j = (u_j, v_j) \in \mathbb{R}^{2m}$. The augmented state $z(t)$ evolves as:

$$\dot{z} = \left[\begin{smallmatrix} A & B \\ 0 & 0 \end{smallmatrix}\right] z + \begin{bmatrix} B & 0 \\ -\frac{\mathbf{I}_m}{\Delta_j} & \frac{\mathbf{I}_m}{\Delta_j} \end{bmatrix} w_j, \qquad t \in I_j.$$

If we set $A^* \triangleq \left[\begin{smallmatrix} A & B \\ 0 & 0 \end{smallmatrix}\right]$, $B^* \triangleq \begin{bmatrix} B & 0 \\ -\frac{\mathbf{I}_m}{\Delta_j} & \frac{\mathbf{I}_m}{\Delta_j} \end{bmatrix}$, $Q^* \triangleq \left[\begin{smallmatrix} Q & 0 \\ 0 & 0 \end{smallmatrix}\right]$ and we define C and its partitioned exponential as in (8) with (A, B, Q) replaced by (A^*, B^*, Q^*), under PWLH (6) we obtain: $z_{j+1} = A_j^* z_j + B_j^* w_j$ and $\ell_j^*(z_j, w_j) \triangleq \int_{t_j}^{t_{j+1}} \ell(x,u)dt = \frac{1}{2}(z_j'Q_j^*z_j + w_j'R_jw_j + 2z_j'M_j^*w_j)$, where

$$A_j^* = F_3(\Delta_j), \quad B_j^* = G_3(\Delta_j), \quad Q_j^* = F_3'(\Delta_j)G_2(\Delta_j),$$
$$M_j^* = F_3'(\Delta_j)H_2(\Delta_j), \quad R_j = (1/6)\left[\begin{smallmatrix} 2R & R \\ R & 2R \end{smallmatrix}\right]\Delta_j +$$
$$\left[B'F_3'(\Delta_j)K_1(\Delta_j)\right] + \left[B'F_3'(\Delta_j)K_1(\Delta_j)\right]'. \tag{9}$$

Finally, by noticing that $z^{(2)}(t_j) = 0$, we can remove the extra component $z^{(2)}$ to obtain:

$$x_{j+1} = A_j x_j + B_j w_j, \tag{10}$$
$$V_T(x, u(\cdot)) = \sum_{j=0}^{J_\gamma - 1} \ell_j(x_j, w_j) + V_f(x(T)),$$
$$\ell_j(x_j, w_j) = \tfrac{1}{2}(x_j' Q_j x_j + w_j' R_j w_j + 2x_j' M_j w_j),$$

where $A_j = A^*_{j\,1:n,1:n}$, $B_j = B^*_{j\,1:n,:}$, $Q_j = Q^*_{j\,1:n,1:n}$, $M_j = M^*_{j\,1:n,:}$, and R_j is defined in (9). We observe that in (10) the discrete time evolution of the system under PWLH is still described by a linear system with the original state x_j and an augmented input $w_j = (u_j, v_j)$.

For the sake of brevity, from now on we focus solely on PWLH, but all derivations and results will apply directly to ZOH, which can be seen as a *particular* PWLH in which $w_j = (u_j, u_j)$.

Given the above premises, let $\mathbf{u} \triangleq (w_0, w_1, \ldots, w_{J_\gamma - 1})$ be an augmented input sequence of length J_γ. Thus, problem $\mathbb{P}_T^\gamma(x)$ can be rewritten as a conventional discrete time CLQR problem:

$$\mathbb{P}_T^\gamma(x): \min_{\mathbf{u} \in \mathbb{U}^{2J_\gamma}} \left\{ V_T^\gamma(x, \mathbf{u}) \triangleq \sum_{j=0}^{J_\gamma - 1} \ell_j(x_j, w_j) + V_f(x_{J_\gamma}) \right\},$$
$$\text{s.t. } x_0 = x \text{ and model (10).} \tag{11}$$

Clearly, if $u(\cdot) \in \mathcal{U}_T^\gamma$ and $\mathbf{u} \in \mathbb{U}^{2J_\gamma}$ is its parameterization vector, then $V_T(x, u(\cdot)) = V_T^\gamma(x, \mathbf{u})$. We observe that $V_T^\gamma : \mathbb{R}^n \times \mathbb{U}^{2J_\gamma} \to \mathbb{R}_{\geq 0}$, and we notice that for each γ and each x, the map $\mathbf{u} \mapsto V_T^\gamma(x, \mathbf{u})$ is continuous, differentiable and convex.

IV. Lower Bounds on Optimal Cost of $\mathbb{P}_T(x)$ and $\mathbb{P}_T^\gamma(x)$

Exploiting the convexity of both $\mathbb{P}_T(x)$ and $\mathbb{P}_T^\gamma(x)$, we can obtain a lower bound of the optimal cost of each problem, given any feasible input $u(\cdot)$.

A. Lower bound of the continuous time optimal cost

$V_T : \mathbb{R}^n \times \mathcal{U}_T \to \mathbb{R}_{\geq 0}$ is defined by: $V_T(x, u(\cdot)) \triangleq \int_0^T \ell(x^u(t;x)), u(t) dt + V_f(x^u(T;x))$, in which $x^u(t;x$ is the solution of (1b) at time t given that the initial state is x at time 0 and the control is $u(\cdot) \in \mathcal{U}_T$. Similarly, the cost due to another control $\nu(\cdot) \in \mathcal{U}_T$ is $V_T(x, \nu(\cdot)) \triangleq \int_0^T \ell(x^\nu(t;x)), \nu(t) dt + V_f(x^\nu(T;x))$. Let $\Delta u(\cdot) \triangleq \nu(\cdot) - u(\cdot)$ and let $\Delta x(\cdot) \triangleq x^\nu(\,\cdot\,;x) - x^u(\,\cdot\,;x)$ for all $t \in [0,T]$. Because $\ell(x,u) = \frac{1}{2}(x'Qx + u'Ru)$ and $V_f(x) = \frac{1}{2}x'Px$, we can write:

$$V_T(x, \nu(\cdot)) - V_T(x, u(\cdot)) = \int_0^T (\langle \Delta x(t), Qx^u(t;x) \rangle +$$
$$\langle \Delta u(t), Ru(t) \rangle) dt + \langle \Delta x(T), Px^u(T) \rangle +$$
$$\tfrac{1}{2} \int_0^T (\langle \Delta x(t), Q\Delta x(t) \rangle + \langle \Delta u(t), R\Delta u(t) \rangle) dt +$$
$$\tfrac{1}{2} \langle \Delta x(T), P\Delta x(T) \rangle. \tag{12}$$

The first order terms may be computed in the usual way [19, pp. 148-149]:

$$\int_0^T \langle \Delta x(t), Qx^u(t;x) \rangle + \langle \Delta u(t), Ru(t) \rangle dt +$$
$$\langle \Delta x(T), Px^u(T;x) \rangle =$$
$$\int_0^T \langle \nabla_u \mathcal{H}(x^u(t;x), u(t), \lambda^u(t;x)), \Delta u(t) \rangle dt,$$

in which the Hamiltonian $\mathcal{H} : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^n \to \mathbb{R}$ is defined by $\mathcal{H}(x, u, \lambda) \triangleq \ell(x, u) + \lambda'(Ax + Bu)$, and $\lambda^u(t;x)$ is the solution at time t of the adjoint system:

$$-\dot{\lambda}(t) = A'\lambda(t) + Qx^u(t;x), \quad \lambda(T) = Px^u(T;x).$$

Since $Q \geq 0$ and $P > 0$, for any $R^* \in \mathcal{R} \triangleq \{S \mid 0 \leq R^* \leq R\}$, we have:

$$V_T(x, \nu(\cdot)) - V_T(x, u(\cdot)) \geq$$
$$\int_0^T \langle g(x, u(\cdot))(t), (\nu(t) - u(t)) \rangle + \tfrac{1}{2} |\nu(t) - u(t)|^2_{R^*} dt,$$

for all $\nu(\cdot) \in \mathcal{U}_T$, in which $g(\cdot)$ is defined by: $g(x, u(\cdot))(t) \triangleq \nabla_u \mathcal{H}(x^u(t;x), u(t), \lambda^u(t;x))$.

We now define the optimality function [20] $\theta : \mathbb{R}^n \times \mathcal{U}_T \to \mathbb{R}_{\leq 0}$ for problem $\mathbb{P}_T(x)$ as:

$$\theta(x, u(\cdot)) \triangleq \inf_{\nu(\cdot) \in \mathcal{U}_T} \int_0^T \langle g(x, u(\cdot))(t), \nu(t) - u(t) \rangle +$$
$$\tfrac{1}{2} |\nu(t) - u(t)|^2_{R^*} dt$$
$$= \int_0^T \min_{v \in \mathbb{U}} \{ \langle \nabla_u \mathcal{H}(x^u(t;x), u(t), \lambda^u(t;x)), v - u(t) \rangle +$$
$$\tfrac{1}{2} |v - u(t)|^2_{R^*} \} dt, \tag{13}$$

where the last equality holds as shown in [1] using results in [21, p.107-108]. Thus, for any $\nu(\cdot) \in \mathcal{U}_T$, we have: $V_T(x, \nu(\cdot)) - V_T(x, u(\cdot)) \geq \theta(x, u(\cdot))$. Hence we have proved:

Proposition 4: For any $(x, u(\cdot), T) \in \mathbb{R}^n \times \mathcal{U}_T \times \mathbb{R}_{>0}$, the following inequality holds:

$$V_T^0(x) \geq V_T(x, u(\cdot)) + \theta(x, u(\cdot)).$$

B. Lower bound of the optimal cost for a given partition

In various stages of the algorithm described in § V, it is useful to have a lower bound on the optimal cost of the discretized problem $\mathbb{P}_T^\gamma(x)$. Given an input $u(\cdot) \in \mathcal{U}_T^\gamma$ defined on a partition γ and its associated parameterization vector $\mathbf{u} = (w_0, w_1, \ldots, w_{J_\gamma - 1}) \in \mathbb{U}^{2J_\gamma}$, then:

$$V_T(x, u(\cdot)) = V_T^\gamma(x, \mathbf{u}) \triangleq \sum_{j=0}^{J_\gamma - 1} (\tfrac{1}{2}\langle x_j, Q_j x_j \rangle + \tfrac{1}{2}\langle w_j, R_j w_j \rangle +$$
$$\langle x_j, M_j w_j \rangle) + \tfrac{1}{2}\langle x_{J_\gamma}, P x_{J_\gamma} \rangle.$$

For another $\nu(\cdot) \in \mathcal{U}_T^\gamma$, with associated parameterization vector $\boldsymbol{\nu} = (\nu_0, \nu_1, \ldots, \nu_{J_\gamma - 1})$ then:

$$V_T^\gamma(x, \boldsymbol{\nu}) - V_T^\gamma(x, \mathbf{u}) = \sum_{j=0}^{J_\gamma - 1} (\langle \Delta x_j, Q_j x_j \rangle + \langle \Delta w_j, R_j w_j \rangle +$$
$$\langle \Delta x_j, M_j w_j \rangle) + \langle \Delta x_{J_\gamma}, P x_{J_\gamma} \rangle + \tfrac{1}{2} \sum_{j=0}^{J_\gamma - 1} (\langle \Delta x_j, Q_j \Delta x_j \rangle +$$
$$\langle \Delta w_j, R_j \Delta w_j \rangle + 2\langle \Delta x_j, M_j \Delta w_j \rangle) + \tfrac{1}{2}\langle \Delta x_{J_\gamma}, P\Delta x_{J_\gamma} \rangle. \tag{14}$$

The first order terms may be computed in the usual way [19, pp. 43-47]:

$$\sum_{j=0}^{J_\gamma - 1} (\langle \Delta x_j, Q_j x_j \rangle + \langle \Delta w_j, R_j w_j \rangle + \langle \Delta x_j, M_j w_j \rangle) +$$
$$\langle \Delta x_{J_\gamma}, P x_{J_\gamma} \rangle = \langle g^\gamma(x, \mathbf{u}), \boldsymbol{\nu} - \mathbf{u} \rangle,$$

in which $g^\gamma(x,\mathbf{u}) \triangleq (g_0^\gamma(x,\mathbf{u}), g_1^\gamma(x,\mathbf{u}), \ldots, g_{J_\gamma-1}^\gamma(x,\mathbf{u}))$ with:

$$g_j^\gamma(x,\mathbf{u}) \triangleq \nabla_{w_j}\mathcal{H}_j(x_j,w_j,\lambda_{j+1}) = M_j'x_j + R_jw_j + B_j'\lambda_{j+1}, \quad j\in\mathbb{Z}_{0:J_\gamma-1}, \quad (15)$$

where $\mathcal{H}_j:\mathbb{R}^n\times\mathbb{R}^{2m}\times\mathbb{R}^n\to\mathbb{R}$ is the Hamiltonian defined by $\mathcal{H}_j(x,w,\lambda)\triangleq \ell_j(x,w)+\lambda'(A_jx+B_jw)$ and $\{\lambda_0,\lambda_1,\ldots,\lambda_{J_\gamma}\}$ is the solution of the discrete time adjoint system:

$$\lambda_j = A_j'\lambda_{j+1} + M_j'w_j + Q_jx_j, \qquad \lambda_{J_\gamma} = Px_{J_\gamma}.$$

The second row of (14) consists of the second order terms. Forming the Schur complement, we note that: $(\langle\Delta x_j, Q_j\Delta x_j\rangle + \langle\Delta w_j, R_j\Delta w_j\rangle + 2\langle\Delta x_j, M_j\Delta w_j\rangle) \geq \langle\Delta w_j,(R_j - M_j'Q_j^{-1}M_j)\Delta w_j\rangle$. Since $P>0$, it follows from (14) and (15) that, for all $\nu(\cdot)$ and $u(\cdot)$ in $\mathcal{U}_T^\gamma$:

$$V_T^\gamma(x,\boldsymbol{\nu}) - V_T^\gamma(x,\mathbf{u}) \geq \langle g^\gamma(x,\mathbf{u}), \boldsymbol{\nu}-\mathbf{u}\rangle + \tfrac{1}{2}\langle\boldsymbol{\nu}-\mathbf{u}, \mathbf{R}^*(\boldsymbol{\nu}-\mathbf{u})\rangle,$$

with $\mathbf{R}^*$ a block diagonal matrix formed by matrices $R_j^*\in\mathcal{R}_j\triangleq\{S \mid 0\leq S\leq R_j - M_j'Q_j^{-1}M_j\}$, $j\in\mathbb{Z}_{0:J_\gamma-1}$. We now define the optimality function $\theta^\gamma:\mathbb{R}^n\times\mathcal{U}_T^\gamma\to\mathbb{R}_{\leq 0}$ for $\mathbb{P}_T^\gamma(x)$ as:

$$\theta^\gamma(x,u(\cdot)) \triangleq \min_{\boldsymbol{\nu}\in\mathbb{U}^{2J_\gamma}} \langle g^\gamma(x,\mathbf{u}),\boldsymbol{\nu}-\mathbf{u}\rangle + \tfrac{1}{2}\langle\boldsymbol{\nu}-\mathbf{u},\mathbf{R}^*(\boldsymbol{\nu}-\mathbf{u})\rangle = \sum_{j=0}^{J_\gamma-1}\theta_j^\gamma(x,u(\cdot)), \quad (16)$$

$$\theta_j^\gamma(x,u(\cdot)) \triangleq \min_{w\in\mathbb{U}^2}\langle g_j^\gamma(x,\mathbf{u}), w-w_j\rangle + \tfrac{1}{2}\langle w-w_j, R_j^*(w-w_j)\rangle. \quad (17)$$

Thus, for any $\boldsymbol{\nu}\in\mathbb{U}^{2J_\gamma}$, we have: $V_T^\gamma(x,\boldsymbol{\nu}) - V_T^\gamma(x,\mathbf{u}) \geq \theta^\gamma(x,u(\cdot))$. Hence we have proved:

Proposition 5: For any $(x,u(\cdot),T)\in\mathbb{R}^n\times\mathcal{U}_T^\gamma\times\mathbb{R}_{>0}$, the following inequality holds:

$$V_T^{\gamma,0}(x) \geq V_T(x,u(\cdot)) + \theta^\gamma(x,u(\cdot)). \quad (18)$$

Finally, let $\theta^\delta(x,u(\cdot))$ denote $\theta^\gamma(x,u(\cdot))$ for the *uniform* partition $\gamma^\delta\in\Gamma_\delta^T$ in which each constituent interval has length δ. Recalling that for any partition in Γ_δ^T, all intervals I_j have a length that is a multiple of δ (or at most equal to δ), we will refer to γ^δ as the *finest* partition.

V. ALGORITHM: CONCEPTUAL DESIGN AND PRACTICAL IMPLEMENTATION

A. Conceptual algorithm

We solve $\mathbb{P}_T^\gamma(x)$ repeatedly, refining γ at each iteration, until we obtain a satisfactory solution of $\mathbb{P}_\infty(x)$. We refer to $\tilde\gamma\in\Gamma_\delta^T$ as a *refinement* of $\gamma\in\Gamma_\delta^T$ if some of the intervals $\{\tilde I_j\}$ defining $\tilde\gamma$ are obtained by bisecting one or more intervals in the set $\{I_j\}$ that defines γ and if the remaining intervals in $\tilde\gamma$ are the same as the corresponding ones in γ. If $V_T^0(x)$ and $V_T^{\gamma,0}(x)$ are, respectively, the optimal value functions of $\mathbb{P}_T(x)$ and $\mathbb{P}_T^\gamma(x)$ then, clearly $V_T^{\gamma,0}(x)\geq V_T^0(x)$, for all $x\in\mathbb{R}^n$, all $\gamma\in\Gamma_\delta^T$, all permissible $\delta\in(0,T)$. Moreover, if $\tilde\gamma$ is a refinement of γ, it follows that $V_T^{\gamma,0}(x)\geq V_T^{\tilde\gamma,0}(x)$. We now state the (conceptual) optimization algorithm to solve $\mathbb{P}_\infty(x)$.

Algorithm 6 (Conceptual Alg.):

Initialize: $\delta,\epsilon>0$, $\gamma\in\Gamma_\delta^T$, $c\in(0,1)$, $T,\Delta T>0$.

1: Solve $\mathbb{P}_T^\gamma(x)$ yielding control $u(\cdot)\in\mathcal{U}_T^\gamma$ and state trajectory $x(\cdot)$. Compute $\theta^\delta(x,u(\cdot))$.

2: Refine γ (repeatedly if necessary) until $\theta^\gamma(x,u(\cdot))\leq c\theta^\delta(x,u(\cdot))$.

3: If $\theta^\delta(x,u(\cdot))\leq-\epsilon$, go to Step 1. Else, go to Step 4.

4: If $x(T)\notin\mathbb{X}_f$, define $I_{J_\gamma}=[T,T+\Delta T]$, and $\gamma\leftarrow\{\gamma,I_{J_\gamma}\}$, $T\leftarrow T+\Delta T$, $J_\gamma\leftarrow J_\gamma+1$.

5: Replace $\epsilon\leftarrow\epsilon/2$, $\delta\leftarrow\delta/2$. Go to Step 1.

A procedure for refining γ (repeatedly if necessary) is given in Section V-B. In Step 5, $\epsilon\leftarrow\epsilon/2$ and $\delta\leftarrow\delta/2$ may be replaced, respectively, by $\epsilon\leftarrow c_1\epsilon$ and $\delta\leftarrow c_2\delta$, with $c_1\in(0,1)$ and $c_2=(1/2)^q$ (with $q\in\mathbb{Z}_{>0}$).

Remark 7: The control $u(\cdot)\in\mathcal{U}_T^\gamma$ obtained in Step 1 satisfies $\theta^\gamma(x,u(\cdot))=0$; if $\tilde\gamma$ is the refined partition obtained in Step 2, and $u(\cdot)$ is not optimal for $\mathbb{P}_T^{\tilde\gamma}(x)$, then $\theta^{\tilde\gamma}(x,u(\cdot))<0$.

Remark 8: T is increased in Step 4 if the (implicit) terminal constraint $x(T)\in\mathbb{X}_f$ is not satisfied. As shown later by Theorem 10, this step occurs only a finite number of iterations.

B. Refinement strategy

Since the length of each interval in the current partition γ is an even multiple of the current δ and since the length of all intervals in the refined partition should also be a multiple of δ, the refinement strategy of Step 2 consists of bisecting each interval with length greater than or equal to 2δ and selecting a subset whose bisection satisfies the condition in Step 2.

Suppose the current partition γ consists of the intervals $\{I_0,I_1,...,I_{J_\gamma-1}\}$. Because the current $u(\cdot)$ is optimal for $\mathbb{P}_T^\gamma(x)$, then $\theta_j^\gamma(x,u(\cdot))=0$ for all $j\in\mathcal{J}_\gamma\triangleq\mathbb{Z}_{0:J_\gamma-1}$. If I_j is bisected, yielding $I_{j1}=[t_j,t_{j1}]$ and $I_{j2}=[t_{j1},t_{j+1}]$ with $t_{j1}=\frac{t_j+t_{j+1}}{2}$, let $w_j\triangleq(u_j,v_j)$ be replaced by $w_{j1}=(u_j,\frac{u_j+v_j}{2})$ in I_{j1} and $w_{j2}=(\frac{u_j+v_j}{2},v_j)$ in I_{j2}, and let x_{j1} and λ_{j1} denote the value of $x(\cdot)$ (the current state trajectory) and $\lambda(\cdot)$ at time t_{j1}. Then the gradients $g_{j1}^\gamma(x,u(\cdot))$ and $g_{j2}^\gamma(x,u(\cdot))$ of the cost with respect to w_{j1} and w_{j2} may be computed from (15) yielding:

$$\theta_j^{\tilde\gamma}(x,u(\cdot))\triangleq\theta_{j1}^{\tilde\gamma}(x,u(\cdot))+\theta_{j2}^{\tilde\gamma}(x,u(\cdot)), \quad (19)$$

where $\theta_{j1}^{\tilde\gamma}(x,u(\cdot))$ and $\theta_{j2}^{\tilde\gamma}(x,u(\cdot))$ are defined as in (17), respectively, for I_{j1} and I_{j2}. Notice that $\theta_j^{\tilde\gamma}(x,u(\cdot))\leq 0$ is a lower bound on the cost reduction obtainable by bisecting I_j. Given a candidate set of intervals to be bisected, $\mathcal{J}\subseteq\mathcal{J}_\gamma$, we obtain: $\theta^{\tilde\gamma}(x,u(\cdot))=\sum_{j\in\mathcal{J}}\theta_j^{\tilde\gamma}(x,u(\cdot))$. By ordering $\theta_j^{\tilde\gamma}(x,u(\cdot))$ in ascending manner, i.e., from the most negative to the least negative, $\mathcal{J}$ is chosen as the subset of $\mathcal{J}_\gamma$ with smallest cardinality such that the condition in Step 2 is satisfied by $\theta^{\tilde\gamma}(x,u(\cdot))$. If no such $\mathcal{J}$ can be found even if all intervals I_j are bisected, i.e., if $\mathcal{J}=\mathcal{J}_\gamma$, the procedure is repeated with γ replaced by the partition with *every* I_j bisected.

C. Practical considerations and algorithm with stopping conditions

The discrete time matrices appearing in the various steps of Algorithm 6 can be computed and stored offline for a (finite) number of possible interval sizes, in geometric sequence of ratio 2, using the formulas of § III. The minimization in (17) is analytic if R_j^* are chosen diagonal, due to the fact that $\mathbb{U}$ (and hence $\mathbb{U}^2$ also) is a box constraint set. The choice of R_j^* diagonal is always possible, e.g. $R_j^*\triangleq\lambda_{\min}(R_j-M_j'Q_j^{-1}M_j)\mathbf{I}_{2m}$ is a valid choice because $0<R_j^*\leq R_j-M_j'Q_j^{-1}M_j$. For a general polytopic set $\mathbb{U}$, the minimization in (17) is a small dimensional convex QP, namely in $2m$ decision variables.

For a given δ, the loop in Steps 1-3 is always exited in a finite number of iterations because, otherwise, the refinement of γ would reach γ^δ and then we would have $\theta^\gamma(x,u(\cdot))=\theta^\delta(x,u(\cdot))=0$, which makes the condition to proceed to Steps 4-5 true. However, as written, Algorithm 6 never terminates because it would keep entering Step 5, reducing δ (and ϵ) and then going to Step 1. A practical variant could terminate after Step 1 and return the computed solution $u(\cdot)$ when $\theta(x,u(\cdot))\geq-\rho$, for a given $\rho>0$. By doing so, there is a guarantee that the achieved cost $V_T(x,u(\cdot))$ satisfies: $V_T(x,u(\cdot))-V_T^0(x)\leq\rho$.

However, evaluation of $\theta(x,u(\cdot))$ from (13) requires computing a numerical integral of the scalar function $\psi : [0,T] \to \mathbb{R}_{\leq 0}$ given by: $\psi(t) \triangleq \min_{v\in\mathbb{U}}\{\langle \nabla_u \mathcal{H}(x^u(t;x),u(t),\lambda^u(t;x)), v-u(t)\rangle + \frac{1}{2}|v-u(t)|^2_{R^*}$, i.e., $\theta(x,u(\cdot)) \triangleq \int_0^T \psi(t)dt$. Notice that if $R^* \leq R$ is chosen as a diagonal matrix, the previous minimization can be performed analytically. Thus, evaluating $\theta(x,u(\cdot))$ and ensuring an exact bound on the termination error is achievable, but for fast closed-loop implementations a simpler alternative is to terminate after Step 1 when

$$\theta^\delta(x,u(\cdot)) \geq -\rho. \quad (20)$$

In this way the computed solution is guaranteed to satisfy $V_T(x,u(\cdot)) - V_T^{\delta,0}(x) \leq \rho$, i.e., the solution to $\mathbb{P}_T^\gamma(x)$ is a ρ-close approximation to the problem $\mathbb{P}_T^{\gamma^\delta}(x)$ at the *current* finest partition γ^δ. Notice that ρ should be chosen (significantly) smaller than the initial value of ϵ.

VI. Properties of the Algorithm

The next result follows from Theorem 3.1 in [22].

Theorem 9: For all $(x,T) \in \mathbb{R}^n \times \mathbb{R}_{>0}$, $u_T^0 : [0,T] \to \mathbb{U}$ is Lipschitz continuous.

Let $u_i(\cdot)$, $x_i(\cdot)$, ϵ_i, γ_i, δ_i and T_i denote, respectively, the values of $u(\cdot)$, $x(\cdot)$, ϵ, γ, δ and T at iteration i of Algorithm 6, where $i \in \mathbb{Z}_{\geq 1}$ increases each time Step 1 is executed.

We can now state the main result, which is proved in [1].

Theorem 10: For each $x \in \mathbb{X}_\infty$, there exists an $i^* \in \mathbb{Z}_{\geq 1}$ and a $T^* \in \mathbb{R}_{>0}$ such that $T_i = T^*$ and $x_i(T_i) \in \mathbb{X}_f$ for all $i \geq i^*$. Also $V_{T_i}(x,u_i(\cdot)) \to V^0_{T^*}(x) = V^0_\infty(x)$ and $u_i(\cdot) \to u^0_{T^*}(\cdot)$ in $L_p(T^*)$ for all $p \in \mathbb{Z}_{\geq 1}$ as $i \to \infty$ (with $u^0_{T^*}(\cdot) = u^0_\infty(\cdot)$ restricted to $[0,T^*]$).
Note that to prove this theorem [1], we do *not* have to assume that the largest interval in the partition goes to zero as $i \to \infty$.

VII. Application Example

Computations are performed in Matlab (R2012b) on a MacBook Air (1.8 GHz Intel Core i7, 4 GB of RAM). Problems $\mathbb{P}_T^\gamma(x)$ are solved using the function `quadprog.m` [1], in which both input and state sequences ($\{w_j\}$ and $\{x_j\}$) are the QP decision variables (see, e.g., [23]). Timing is measured with the functions `tic` and `toc`. The following performance indicators are considered during the iterations of Algorithm 6: (i) number of intervals, J_γ; (ii) continuous time cost error bound, $-\theta(x,u(\cdot))$; (iii) cumulative solution time (previous and current iteration).

We consider a stable three-state one-input system defined by the (continuous time) matrices: $A = \begin{bmatrix} -0.1 & 0 & 0 \\ 0 & -2.0 & -6.25 \\ 0 & 4.0 & 0 \end{bmatrix}$, $B = \begin{bmatrix} 0.25 \\ 2.0 \\ 0 \end{bmatrix}$, $Q = \mathbf{I}_3$, $R = 0.1$. Given the initial state $x(0) = \begin{bmatrix} 1.3440 & -4.5850 & 5.6470 \end{bmatrix}'$, we consider the first five iterations of Algorithm 6 in three different variants: (i) using PWLH and adaptive refinement as in § V-B; (ii) using PWLH and fixed refinement in which all intervals are bisected; (iii) using ZOH and fixed refinement in which all intervals are bisected. In all cases we use the following parameters $T = 10$, $c = 0.8$, and initial values of $\delta = 0.125$ and $\epsilon = 0.1$. The storage of the required discretized matrices, for 11 different interval sizes, takes 7 kB. Comparative results are depicted in Fig. 1 (left). From these results we observe that: adaptive PWLH is much more effective and efficient than ZOH as the same solution error is achieved with fewer intervals and much less computation time. In order to further examine the efficiency of the adaptive refinement procedure we depict in Fig. 1 (right) the input $u(\cdot)$ achieved during each iteration of Algorithm 6 using PWLH. In this picture the partition intervals at each iteration are also reported. It is clear that the devised adaptive procedure is able to detect which intervals require (possibly repeated) bisection. The partitions do not have an a-priori conceivable pattern, e.g. finer in the early part of $[0,T]$ and gradually more sparse as assumed in DT MPC implementations with move blocking. Hence, the algorithm is parsimonious in the use of intervals and hence of decision variables to solve $\mathbb{P}_T^\gamma(x)$.

[1] With options: `'interior-point-convex'` algorithm, 'function tolerance' of 10^{-12} and 'variable tolerance' of 10^{-10}.

TABLE I
Closed-loop performance comparison of continuous time CLQR and discrete time MPC. Results are averaged over 50 closed-loop simulations of 20 s, each starting from a different random initial state.

Controller	(T_C/T_s) Mean	(T_C/T_s) Max	$(V_{CL}-V^0_{CL})/V^0_{CL}$ Mean	$(V_{CL}-V^0_{CL})/V^0_{CL}$ Max
CTCLQR 1 ($\rho = 5\cdot 10^{-4}$, $T_s = 1$ s)	0.0229	0.1144	–	–
CTCLQR 2 ($\rho = 5\cdot 10^{-3}$, $T_s = 1$ s)	0.0178	0.0585	0.0018	0.0053
DTMPC 1 ($N = 10$, $T_s = 1$ s)	0.0096	0.0141	0.1223	0.2820
DTMPC 2 ($N = 200$, $T_s = 0.05$ s)	0.8141	1.3879	0.0002	0.0012

Next, we discuss the closed-loop performance of the proposed continuous time CLQR and compare it with that of *standard* discrete time MPC, formulated as in [23] and solved using the function `quadprog.m` with same options and tolerances used in solving problems $\mathbb{P}_T^\gamma(x)$. Every sampling time T_s, given the current state x, we solve $\mathbb{P}_T(x)$ and inject the first portion, $t \in [0,T_s)$, of the computed input $u(\cdot)$. Three controllers, which use a sampling time of $T_s = 1$ s, are compared: CTCLQR 1 and CTCLQR 2 use Algorithm 6 with PWLH and stopping condition (20) for $\rho = 5\cdot 10^{-4}$ and $\rho = 5\cdot 10^{-3}$, respectively; DTMPC 1 is a discrete time MPC with horizon $N = 10$. DTMPC 2 instead uses a horizon of $N = 200$ and hence a sampling time of $T_s = T/N = 0.05$ s. In Table I we report: the ratio between computation time T_C and sampling time T_s, the closed-loop suboptimality with respect to CTCLQR 1 defined as $(V_{CL}-V^0_{CL})/V^0_{CL}$ in which $V_{CL} = \int_0^{T_f} \ell(x,u)dt$, $T_f = 20$ s, is the closed-loop cost achieved with a given controller and V^0_{CL} is that achieved with CTCLQR 1. We can observe that CTCLQR 2 has a small relative suboptimality of (up to 0.53%), whereas DTMPC 1 has a suboptimality (up to 28.2%). By decreasing the sample time, DTMPC 2 achieves a small suboptimality (up to 0.12%). However, CTCLQR 1, CTCLQR 2 and DTMPC 1 have computation times significantly smaller than their sampling time. DTMPC 2, instead, have computation times similar and even larger than its sampling time. See [1] for other examples.

VIII. Conclusions

The method presented in this paper solves the input constrained, infinite horizon, continuous time linear quadratic regulator problem to a user specified accuracy. The algorithm is efficient due to the storage of matrix exponentials for exact solution of the model, cost, and gradients at several levels of discretization. The storage requirement for these matrices is minor.

The advantages of finite horizon *discrete time* linear MPC, and the reasons for its dominant position in industrial implementations, are mainly computational. All that is required for implementation is the solution of a strictly convex, finite dimensional quadratic program, and standard software exists for solving the QP to near machine precision in a finite number of iterations. If one wants a controller with a guarantee of even nominal recursive feasibility and closed-loop stability, however, more is required. Current theory requires a terminal penalty and an implicit or explicit terminal constraint. The terminal constraint restricts the set of feasible initial states that can be handled. One method to offset this reduction in the feasible set is to increase the horizon length. Since computational cost increases at least linearly

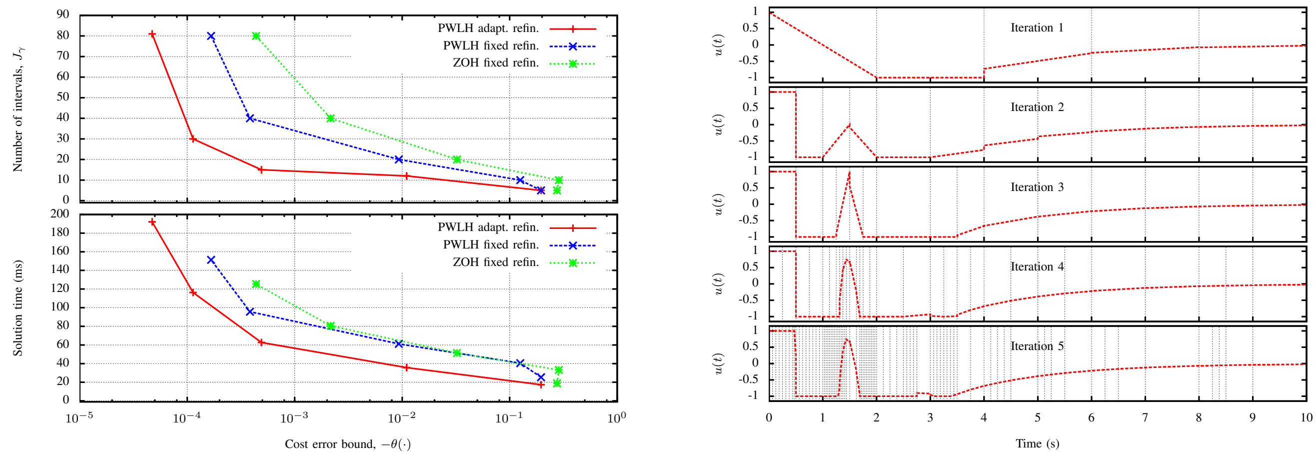

Fig. 1. Left: Performance indicators during the first five iterations of Algorithm 6 using PWLH with adaptive refinement (red), PWLH with fixed refinement (blue), ZOH with fixed refinement (green). Right: Input $u(\cdot)$ computed during the first five iterations of Algorithm 6 using PWLH, and associated (adaptive) intervals.

with horizon length, this approach creates an unpleasant tradeoff between the size of the feasible set and computational efficiency. The same difficult tradeoff applies to the choice of sample time. If chosen too large, closed-loop performance and robustness to disturbances degrade; if chosen too small, the horizon and computation time are excessively large or the feasible set of initial states is excessively small.

If we can instead solve efficiently the infinite horizon, continuous time CLQR, then nominal recursive feasibility and closed-loop stability follow directly, and the CLQR feasible set is the largest set possible. But can we solve online the infinite horizon problem? As shown in this paper, the answer is already yes for some systems. And when computing an infinite horizon solution for a challenging system, CT offers significant advantages over DT. If a system is highly constrained and requires a long time to enter the terminal set, the flexible CT partition requires far fewer decision variables than DT. When a system displays fast dynamics at constraint switching times, the CT algorithm automatically locates these times and places discretization points only in intervals where they are required, which is not possible with DT MPC.

It remains to be seen whether the CT approach can handle the largest industrial applications, which currently consist of hundreds or thousands of state variables. Because all notions of sample time are removed from the regulation problem, sample time can be chosen as a design parameter relevant to sensor hardware and robustness to disturbances, without consideration of the underlying regulation problem. This comment, of course, presumes that the regulation computation is several times faster than the desired sampling rate. Research directed at further improving the online efficiency of solving the CLQR is therefore always relevant.

References

[1] G. Pannocchia, J. B. Rawlings, D. Q. Mayne, and G. Mancuso, “Whither discrete time model predictive control?” Available at http://jbrwww.che.wisc.edu/pubs_techreports.html, TWCCC Technical Report, Tech. Rep. 2014–01, April 2014.

[2] E. B. Lee and L. Markus, *Foundations of Optimal Control Theory*. New York: John Wiley & Sons, 1967.

[3] D. Chmielewski and V. Manousiouthakis, “On constrained infinite-time linear quadratic optimal control,” *Sys. Cont. Let.*, vol. 29, pp. 121–129, 1996.

[4] P. O. M. Scokaert and J. B. Rawlings, “Constrained linear quadratic regulation,” *IEEE Trans. Auto. Contr.*, vol. 43, no. 8, pp. 1163–1169, 1998.

[5] W. W. Hager, “Runge-Kutta methods in optimal control and transformed adjoint system,” *Numer. Math*, vol. 87, pp. 247–282, 2000.

[6] C. Büskens and H. Maurer, “SQP-methods for solving optimal control problems with control and state constraints: Adjoint variables, sensitivity analysis and real-time control,” *J. Comput. Appl. Math.*, vol. 120, pp. 85–108, 2000.

[7] A. Hartwich, M. Schlegel, L. Würth, and W. Marquardt, “Adaptive control vector parameterization for nonlinear model-predictive control,” *Int. J. Rob. Nonlin. Contr.*, vol. 18, no. 8, pp. 845–861, 2008.

[8] M. Cannon and B. Kouvaritakis, “Infinite horizon predictive control of constrained linear systems,” *Automatica*, vol. 36, pp. 943–955, 2000.

[9] A. Kojima and M. Morari, “LQ control of constrained continuous-time systems,” *Automatica*, vol. 40, pp. 1143–1155, 2004.

[10] V. Sakizlis, J. D. Perkins, and E. N. Pistikopoulos, “Explicit solutions to optimal contol problems for contrained continuous-time linear systems,” *IEE Proc.-Control Theory Appl.*, vol. 152, no. 4, pp. 443–452, 2005.

[11] R. Goebel and M. Subbotin, “Continuous time linear quadratic regulator with control constraints via convex duality,” *IEEE Trans. Auto. Contr.*, vol. 52, no. 5, pp. 886–892, 2007.

[12] S. Longo, E. C. Kerrigan, K. V. Ling, and G. A. Constantinides, “A parallel formulation for predictive control with nonuniform hold constraints,” *Annual Reviews in Control*, vol. 35, no. 2, pp. 207–214, 2011.

[13] F. Bayat and T. A. Johansen, “Multi-resolution explicit model predictive control: Delta-model formulation and approximation,” *IEEE Trans. Auto. Contr.*, 2013, DOI: 10.1109/TAC.2013.2259982.

[14] V. Costanza, P. S. Rivadeneira, and A. H. González, “Minimizing control energy in a class of bounded-control linear-quadratic regulator problems,” *Opt. Contr. Appl. and Meth.*, 2013, DOI: 10.1002/oca.2072.

[15] J. Yuz, G. Goodwin, A. Feuer, and J. De Doná, “Control of constrained linear systems using fast sampling rates,” *Syst. Contr. Lett.*, vol. 54, pp. 981–990, 2005.

[16] G. Pannocchia, J. B. Rawlings, D. Q. Mayne, and W. Marquardt, “On computing solutions to the continuous time constrained linear quadratic regulator,” *IEEE Trans. Auto. Contr.*, vol. 55, no. 9, pp. 2192–2198, 2010.

[17] J. W. C. Robinson and P. Ögren, “On the use of gradual dense–sparse discretizations in receding horizon control,” *Opt. Contr. Appl. and Meth.*, 2013, DOI: 10.1002/oca.2065.

[18] C. F. Van Loan, “Computing integrals involving the matrix exponential,” *IEEE Trans. Auto. Contr.*, vol. 23, no. 3, pp. 395–404, 1978.

[19] A. E. Bryson and Y. Ho, *Applied Optimal Control*. New York: Hemisphere Publishing, 1975.

[20] E. Polak, *Optimization: Algorithms and Consistent Approximations*. Springer Verlag, 1997.

[21] R. B. Vinter, *Optimal Control*. Boston: Birkhaüser, 2000.

[22] G. N. Galbraith and R. B. Vinter, “Lipschitz continuity of optimal controls for state constrained problems,” *SIAM J. Control Optim.*, vol. 42, no. 5, pp. 1727–1744, 2003.

[23] C. V. Rao, S. J. Wright, and J. B. Rawlings, “Application of interior-point methods to model predictive control,” *J. Optim. Theory Appl.*, vol. 99, pp. 723–757, 1998.