Proceedings of the
44th IEEE Conference on Decision and Control, and
the European Control Conference 2005
Seville, Spain, December 12-15, 2005

A Real-time Framework for Model Predictive Control of Continuous-Time Nonlinear Systems

Darryl DeHaan and Martin Guay

***Abstract*— A new formulation of continuous-time nonlinear model predictive control (NMPC) is developed which accounts for dynamics associated with minimization of the optimal control problem. In doing so, it is shown that the stability of NMPC can be maintained for fast processes in which the computation time is significant with respect to the process dynamics. Our framework generalizes recent results for piecewise constant NMPC of continuous-time processes.**

***Index Terms*— nonlinear model predictive control, real-time optimization, optimal control, piecewise constant control**

I. INTRODUCTION

Model predictive control (MPC) is a control approach in which the current control move is obtained by repeatedly solving online a finite horizon, open-loop optimal control problem. The unprecedented industrial success of MPC approaches based on linear, discrete-time process models motivates the development of predictive control approaches for systems which exhibit significant nonlinearities (i.e. NMPC).

Typical approaches to NMPC of continuous-time plants fall into two categories. Discrete-time approaches treat both the control decisions and plant dynamics as evolving in discrete time. While computationally appealing, this assumes knowledge of an exact plant difference equation, which is typically unavailable. Furthermore, important inter-sampling behaviour such as constraint violation can be missed. Continuous-time approaches, in which both the control decisions and plant dynamics evolve continuously, are of theoretical interest only as their optimization is over arbitrary input trajectories $u(t)$, which is computationally intractable.

More recently, the control of continuous-time plants by sampled-data controllers has been studied. Fontes [1] used ideas from [2] to describe the closed-loop dynamics when the optimal continuous-time input trajectory is implemented in piecewise-constant (PWC), sampled-data manner. Similarly, [3] proposes implementing the continuous-time input trajectory in a sampled-data manner, without the zero-order hold. Unlike these works, the approach of [4] performs the optimization itself over the class of PWC control moves, resulting in a more tractable search.

An issue which has received little, if any, attention in the literature is the impact on closed-loop stability of input delay associated with computation time for the optimal control problem. Traditional MPC literature assumes the calculated input is implemented instantaneously following state/output sampling. For fast or high-dimensional processes, this computational delay may become significant with respect to the process dynamics. Recent work involving suboptimal solutions ([5], [6]) attempt to minimize this delay by early termination of the minimization calculation, at the expense of failing to achieve the true minimum.

In contrast to the above, our approach allows for the minimization calculation to proceed throughout the entire sampling interval, rather than restricting it to terminate within a small subinterval of the controller period. We choose to view the optimal PWC control trajectory as a set of unknown parameters which can be identified in real time, using continuous-time adaptive control techniques. The input is continuously updated as these parameter estimates evolve, and thus the resulting closed-loop control action is piecewise continuous, but not necessarily piecewise constant. While existing approaches such as [4] are encompassed as a special case of our framework, the main contribution of this work is to show that stability can be preserved even when the minimization dynamics evolve in a comparable timescale to the process.

This paper is organized as follows. In Section II we describe the problem framework and relevant assumptions, while Section III establishes preliminary results on constraint handling and piecewise constant control. Section IV contains the main results for the general MPC framework we propose, while Section V outlines similarities and differences to standard MPC approaches. Section VI contains a brief simulation example, and proofs are contained in the appendix.

II. PROBLEM STATEMENT AND ASSUMPTIONS

In this paper, we will use the notation $\mathring{\mathbb{S}}$ to denote the open interior of a closed set $\mathbb{S}$, and $\partial\mathbb{S}$ for the boundary $\mathbb{S} \setminus \mathring{\mathbb{S}}$. A function $f(\cdot)$ will be described as being C^{m+} if it is C^m, with all derivatives of order m yielding locally Lipschitz functions. A continuous function $\gamma : [0, \infty) \to \mathbb{R}_{\geq 0}$ will be considered class $\mathcal{K}$ if it is strictly increasing from $\gamma(0) = 0$, and class $\mathcal{K}_\infty$ if it is furthermore radially unbounded. Finally, the notation $\|\cdot\|_F$ denotes a Frobenius matrix norm.

We consider the continuous time optimal control problem

$$\min_{u(\cdot)} J_\infty = \int_{t_0}^{\infty} L(x, u)\, d\tau \tag{1}$$

$$s.t. \qquad \dot{x} = f(x, u) \tag{2}$$

$$(x, u) \in \mathbb{X} \times \mathbb{U},\ \forall t \geq t_0 \tag{3}$$

The mapping $L : \mathbb{X} \times \mathbb{U} \to \mathbb{R}_{\geq 0}$ is assumed to be at least C^{1+}, and to satisfy $\gamma_L(\|x, u\|) \leq L(x, u) \leq \gamma_U(\|x, u\|)$, with $\gamma_L, \gamma_H \in \mathcal{K}_\infty$. The dynamics $f : \mathbb{X} \times \mathbb{U} \to \mathbb{R}^n$ are also assumed C^{1+}, and satisfy $f(0, 0) = 0$. The assumed

The authors are with the Department of Chemical Engineering, Queen's University, Kingston, Canada, K7L 3N6 (email: guaym@chee.queensu.ca)

differentiability is primarily for convenience; with additional computational expense, results could be reposed using generalized gradients with L and f being C^{0+}. The sets $\mathbb{X} \subseteq \mathbb{R}^n$ and $\mathbb{U} \subseteq \mathbb{R}^m$ are closed, convex, and satisfy $\mathring{\mathbb{X}} \neq \emptyset$.

For online calculation, (1) is approximated at time t by

$$J_{rh}(x(\cdot), u(\cdot)) = \int_t^{t+T} L(x, u)\, d\tau + W(x(t+T)) \quad (4)$$

$$x(t+T) \in \mathcal{X}_f \quad (5)$$

where the horizon length T is potentially variable. The penalty $W : \mathcal{X}_f \to \mathbb{R}_{\geq 0}$ is C^{1+} on the convex terminal set $\mathcal{X}_f \subseteq \mathbb{X}$, and satisfies $\gamma_{WL}(\|x\|) \leq W(x) \leq \gamma_{WU}(\|x\|)$ for $\gamma_{WL}, \gamma_{WU} \in \mathcal{K}$. The interior $\mathring{\mathcal{X}}_f$ is assumed nonempty.

We begin by defining $\kappa(x, T_\kappa)$ to be a known family (parameterized by T_κ) of PWC, locally Lipschitz feedbacks $\kappa : \mathcal{X}_f \times \mathbb{R}_{\geq 0} \to \mathbb{R}^m$. The sampling period T_κ is specified by a known, locally Lipschitz function $\delta : \mathcal{X}_f \to \mathbb{R}_{\geq 0}$ which is lower bounded by some class $\mathcal{K}$ function $\gamma_\delta(\|x\|)$, and uniformly upper bounded by some constant $M_\delta > 0$ for all $x \in \mathcal{X}_f$. Starting from $(t_i, x_i) \in \mathbb{R} \times \mathcal{X}_f$, we denote by $x^{\kappa_\delta}(t)$ the closed-loop solution to $\dot{x} = f(x, \kappa_\delta(x_i))$ on the interval $t \in [t_i, t_i + \delta(x_i)]$, where $\kappa_\delta(x_i) \triangleq k(x_i, \delta(x_i))$.

We will focus on stabilization of x to a target set Σ, which we assume to be of the form $\Sigma = \{x : W(x) \leq c_\Sigma\}$ with $c_\Sigma \geq 0$, with inner approximations given by

$$\Sigma^\varepsilon = \{x : W(x) \leq (c_\Sigma - \varepsilon)\} \subset \Sigma$$

for any $\varepsilon > 0$. If $\varepsilon > c_\Sigma$, then $\Sigma^\varepsilon \equiv \emptyset$. The following assumption is thus adapted from [7].

Assumption 1: Let $\Sigma \subset \mathcal{X}_f$ denote the target set. The functions $\kappa(\cdot,\cdot)$, $\delta(\cdot)$, $W(\cdot)$ and the set $\mathcal{X}_f$ satisfy

A1.1) $\mathcal{X}_f \subset \mathring{\mathbb{X}}$, $\mathcal{X}_f$ closed, $0 \in \Sigma \subset \mathring{\mathcal{X}}_f$.

A1.2) Σ compact, $\Sigma = \{x : W(x) \leq c_\Sigma\}$, $c_\Sigma \geq 0$.

A1.3) $\kappa_\delta(x) \in \mathbb{U}$ for all $x \in \mathcal{X}_f$.

A1.4) $\mathring{\mathcal{X}}_f$ is positive invariant under (2) with $u = \kappa_\delta(x)$

A1.5) For $\varepsilon > 0$ sufficiently small, $\exists\, \gamma(\cdot) \in \mathcal{K}$ such that for all $x \in \mathcal{X}_f \setminus \mathring{\Sigma}^\varepsilon$,

$$W(x') - W(x) + \int_t^{t'} L(x^{\kappa_\delta}, \kappa_\delta(x^{\kappa_\delta}))\, d\tau \leq -\gamma(\|x\|) \quad (6)$$

where $t' \triangleq t + \delta(x)$, and $x' \triangleq x^{\kappa_\delta}(t')$.

Remark 1: Using set stabilization provides greater flexibility in designing $\kappa_\delta(x)$ by allowing for practically-stabilizing methods (see [2], [8] and references therein). The (potentially) variable period $T_\kappa = \delta(x)$ is motivated by the observation in [2] that faster switching near the origin allows closer convergence using a practical-stabilizer.

III. Preliminary Results

A. Incorporation of State Constraints

Due to the continuous-time nature of the closed-loop state trajectories, the constraint $x \in \mathbb{X}$ effectively represent an infinite collection of (pointwise in time) constraints which must be imposed on the minimization in (1). Since the combinatorial complexity of active-set approaches tend to scale poorly, interior-point methods provide a much more straightforward approach to constraint enforcement. To this end, we assume that appropriately selected barrier functions B_x and B_f are defined on the sets $\mathbb{X}$ and $\mathcal{X}_f$, respectively. For our purposes, a barrier function $B : \mathbb{S} \to \mathbb{R}_{\geq 0} \cup \{\infty\}$ on a convex set $\mathbb{S}$ is deemed "appropriate" if it is sufficiently differentiable on the open, convex set $\mathring{\mathbb{S}}$, and satisfies $\lim_{s \to \partial\mathbb{S}} B(s) = \infty$. We refer the reader to [9] for more rigorous insight into the selection of suitable barriers.

Following [10], we employ a method of *gradient-recentering* to center the barriers about the origin. For any general barrier function $B(s)$ we define

$$B^o(s) = B(s) - B(0) - \nabla B(0)^T s. \quad (7)$$

The state constraints are incorporated into the design by augmenting the functions $L(\cdot, u)$ and $W(\cdot)$ as

$$L^a(x, u) = L(x, u) + \mu B_x^o(x) \quad (8)$$

$$W^a(x_f) = W(x_f) + \mu_f B_f^o(x_f) \quad (9)$$

with weighting constants μ, $\mu_f > 0$.

Assumption 2: The barriers $B_x(\cdot)$, $B_f(\cdot)$, and weightings μ, μ_f are chosen to satisfy

$$\mu_f \left[B_f^o(x') - B_f^o(x)\right] + \mu \int_t^{t'} B_x^o(x^{\kappa_\delta}(\tau))\, d\tau \leq \gamma(\|x\|) \quad (10)$$

or equivalently,

$$W^a(x') - W^a(x) + \int_t^{t'} L^a(x^{\kappa_\delta}, \kappa_\delta(x^{\kappa_\delta}))\, d\tau \leq 0 \quad (11)$$

$\forall\, x \in \mathring{\mathcal{X}}_f \setminus \mathring{\Sigma}^\varepsilon$, where x', t' and ε are from Assumption 1.

In general, selection of B_x, B_f such that (10) can be satisfied is not obvious. However, we show in the full version of this paper that there exists $\mu^* > 0$ such that (10) is satisfied for all $\mu, \mu_f \in (0, \mu^*]$ if i) $\mathring{\Sigma} \neq \emptyset$, ii) $\mathcal{X}_f$ is of the form $\{x : W(x) \leq b\}$, and iii) B_f is chosen as $B_f \triangleq \tilde{B}_f \circ W$, with $\tilde{B}_f$ a barrier for the scalar interval $W \in (-\infty, b]$.

B. Piecewise Constant Control Parameterizations

The primary motivation for using piecewise constant control trajectories in NMPC is the reduction of the optimal control problem to a finite-dimensional NLP. The discretization of the input trajectory is therefore motivated by *computational*, rather than physical, considerations. In most applications, sensor and actuator data are updated at rates significantly faster than the process or controller dynamics, and can be reasonably treated as continuous-time signals.

Let N denote the (constant) number of PWC control moves (per scalar input) by which the input trajectory will be parameterized. The PWC control parameters are contained in the matrix $\Theta \in \mathbb{R}^{m \times N}$, which has an associated vector of "switching times" $t^\theta \triangleq \{t_i^\theta : i = 1, \ldots N\} \in \mathbb{R}^N$ satisfying $0 \leq t_{i+1} - t_i \leq M_\delta$, $\forall i = \{1, \ldots N-1\}$. The pair (Θ, t^θ) specify the input trajectory $u = v(t, t^\theta, \Theta)$ as

$$v_i(t, \Theta, t^\theta) = \begin{cases} \Theta_{i1} & t \leq t_1^\theta \\ \Theta_{ij} & t \in (t_{j-1}^\theta, t_j^\theta],\ j = 2 \ldots N \\ 0 & \text{otherwise} \end{cases} \quad (12)$$

for $i = \{1, \dots m\}$. For the remainder of this work, we will denote by θ a vector containing the mN elements of Θ. With minor abuse of notation, we will refer to (12) by $v(t, \theta, t^\theta)$.

Definition 1: A *control parameterization* refers to any pair of the form $(\theta,\, t^\theta) \in \mathbb{R}^{(mN)} \times \mathbb{R}^N$.

Let $(t_0,\, x_0) \in \mathbb{R} \times \mathring{\mathbb{X}}$ be an arbitrary initial condition for (2), and let $(\theta,\, t^\theta)$ be an arbitrary, constant control parameterization satisfying $t_0 \leq t_1^\theta$. We denote the resulting solution (in the classical sense) to (2) and (12) by $x^p(t, t_0, x_0, \theta, t^\theta)$ and $u^p(t, t_0, x_0, \theta, t^\theta)$, defined over subintervals of $[t_0,\, t_N^\theta]$. The superscript stands for "prediction", since these solutions will be applicable only for the NMPC prediction model. At times we will condense this notation to $x^p(t)$, $u^p(t)$.

Definition 2: A control parameterization $(\theta,\, t^\theta)$ of length N is *feasible* if, $\forall t \in [t_0,\, t_N^\theta]$, the solution $x^p(t, t_0, x_0, \theta, t^\theta)$, $u^p(t, t_0, x_0, \theta, t^\theta)$ exists and satisfies $x^p(t) \in \mathring{\mathbb{X}}$, $u^p(t) \in \mathbb{U}$, $x^p(t_N^\theta) \in \mathring{\mathcal{X}}_f$. Let $\Phi^N(t_0, x_0) \subseteq \mathbb{R}^{(mN)} \times \mathbb{R}^N$ denote the set of all such feasible control parameterizations.

Throughout this paper, we interpret $\theta \in \mathbb{U}^N \subseteq \mathbb{R}^{mN}$ to mean that $\forall i \in \{1, \dots, N\}$, the control vectors given by $\Theta\, e_i$ (with $e_i \in \mathbb{R}^N$ elementary basis vectors) satisfy $(\Theta\, e_i) \in \mathbb{U}$.

Lemma 1: Let $\mathcal{X}^0 \subseteq \mathring{\mathbb{X}}$ denote the set of initial states x_0 for which there exists piecewise-continuous open-loop trajectories $x(t)$, $u(t)$ solving (2), defined on some interval $t \in [t_0,\, t_f]$, and satisfying $x(t) \in \mathring{\mathbb{X}}$, $u(t) \in \mathbb{U}$ and $x(t_f) \in \mathring{\mathcal{X}}_f$. Then, for every $(t_0,\, x_0) \in \mathbb{R} \times \mathcal{X}^0$, there exists $N^*(x_0)$ such that $\Phi^N(t_0, x_0)$ has positive Lebesgue measure in $\mathbb{U}^N \times \mathbb{R}^N$ for all $N \geq N^*(x_0)$.

IV. Generalized Continuous-time MPC Design

A. Description of Algorithm

Let $(t_0,\, x_0) \in \mathbb{R} \times \mathcal{X}^0$ denote an arbitrary initial condition for (2). Below we outline the steps involved in calculating our MPC controller. It will be useful to define $z \triangleq [x^T,\, \theta^T,\, {t^\theta}^T]^T$, the vector of closed-loop states.

For ε chosen according to A1.5, we define the following smoothed indicator-type function for Σ:

$$\rho(x) \triangleq \begin{cases} 1 & x \notin \mathring{\Sigma} \\ \rho_0(W(x)) & x \in \Sigma \setminus \Sigma^\varepsilon \\ 0 & x \in \Sigma^\varepsilon \end{cases} \tag{13}$$

where $\rho_0 : (c_\Sigma - \varepsilon,\, c_\Sigma) \to (0, 1)$ is a C^1 monotonic function satisfying $\lim_{W \to c_\Sigma^-} \left\{\rho_0, \frac{d\rho_0}{dW}\right\} = \{1, 0\}$ and $\lim_{W \to (c_\Sigma - \varepsilon)^+} \left\{\rho_0, \frac{d\rho_0}{dW}\right\} = \{0, 0\}$. We then use the following modified version of (4) as our cost function:

$$\begin{aligned} J(t, z) = \int_t^{t_N^\theta} & L_\rho^a(x^p(\tau, t, z), u^p(\tau, t, z))\, d\tau \\ & + W_\rho^a(x^p(t_N^\theta, t, z)) \end{aligned} \tag{14}$$

$$L_\rho^a(x, u) = \rho(x) L^a(x, u)$$
$$W_\rho^a(x) = \rho(x) W^a(x)$$

where $J(t, z) \equiv J(t, x, \theta, t^\theta)$. Since W is not necessarily a Lyapunov function inside Σ^ϵ, this will prevent the minimization of J from compromising the forward invariance of Σ^ε achieved under $\kappa_\delta(x)$.

Step 1: Parameterization Initialization

As is a common starting point for many numerical NMPC approaches, the first step in our procedure requires initialization of the control parameterization to any known value $(\theta,\, t^\theta)_0$ in the feasible set $\Phi^N(t_0, x_0)$, which has positive Lebesgue measure (for sufficiently large N) by Lemma 1. If the PWC stabilizer $\kappa_\delta(x)$ is globally stabilizing, then $(\theta,\, t^\theta)_0$ can be determined by solving (2) under $u = \kappa_\delta(x)$. Otherwise, a dual programming program could be solved to identify a feasible initial control parameterization.

Step 2: Continuous Flow under Dynamic Feedback

At any instant $t \in [t_0,\, t_1^\theta]$, we assume that we can "instantaneously" compute the model prediction $x^p(\tau, t, z(t))$, $u^p(\tau, t, z(t))$ solving (2), (12) over the interval $\tau \in [t,\, t_N^\theta]$. Since the parameterization $(\theta,\, t^\theta)$ is constant with respect to τ, the predicted $u^p(\tau, t, z(t))$ is PWC. Using this prediction we calculate the receding-horizon cost (14).

Since this step only deals with the interval $t \in [t_0,\, t_1^\theta]$, we will (with admitted abuse of notation) rewrite (12) as simply $u_i = v_i(\theta) \triangleq \Theta_{i1}$, $i \in \{1, \dots m\}$. The closed-loop dynamics (w.r.t *ordinary* time t) are therefore written as

$$\dot{z} = \begin{bmatrix} \dot{x} \\ \dot{\theta} \\ \dot{t}^\theta \end{bmatrix} = \begin{bmatrix} f(x,\, v(\theta)) \\ \Psi(t, z) \\ 0_{N \times 1} \end{bmatrix} \tag{15}$$

in which θ evolves as a dynamic controller state.

Criterion 1: The update law $\Psi(t, z)$ must be chosen to ensure the following:

C1.1) $\langle\, \nabla_\theta J,\, \Psi(t, z)\, \rangle \leq 0$

C1.2) $(\theta(t),\, t^\theta(t)) \in \Phi^N(t, x(t))$, $\forall t \in [t_0,\, t_1^\theta]$, where $\theta(t)$ and $x(t)$ denote solutions to (15).

C1.3) $\Psi(t, z)$ is continuous w.r.t t, and locally Lipschitz w.r.t z, for all $(\theta,\, t^\theta) \in \Phi^N(t, x)$.

The term $\nabla_\theta J$ is the gradient of (14) with respect to θ, the calculation of which is discussed in the full version of this paper. Examples of update laws satisfying Criterion 1 are discussed in section V.

Step 3: Discrete Transitions at Switching Times

At the completion of step 2, when $t = t_1^\theta$, the controller states $(\theta,\, t^\theta)$ are updated by the following jump mapping

$$\begin{bmatrix} x^+ \\ \theta^+ \\ t^{\theta+} \end{bmatrix} = \begin{bmatrix} x \\ \Upsilon(t, z) \\ \begin{bmatrix} 0_{N-1} & I_{N-1} \\ 0 & 0_{N-1}^T \end{bmatrix} t^\theta + \begin{bmatrix} 0_{N-1} \\ t_N^\theta + \delta(x^p(t_N^\theta, \theta, t^\theta)) \end{bmatrix} \end{bmatrix} \tag{16}$$

where the notation z^+ denotes the post-jump values, still at the same value of time t (while the state t_1^θ has meanwhile been redefined by (16)). The function $\delta(\cdot)$ is the same as that in Assumption 1.

Criterion 2: The jump mapping $\Upsilon(t, z)$ is chosen to ensure the following:

C2.1) $J(t, z^+) - J(t, z) \leq 0$

C2.2) $(\theta^+,\, t^{\theta+}) \in \Phi^N(t, x)$

Step 4: Iteration of Steps 2 and 3

From criterion C2.2, the post-jump values $(\theta^+, t^{\theta+})$ constitute a feasible control parameterization for the current state (t, x), and thus satisfy the conditions of the initialization in step 1. The procedure thus repeats back to step 2, with appropriate redefinition of (t_0, x_0).

B. A Meaningful Notion of Closed-Loop Solutions

The feedback control resulting from the above algorithm is a dynamic, time-varying control law which is is set-valued at the switching times. The notion of a "solution" to (2) is unclear, since neither classical nor "sample-and-hold" [2] solutions apply, while Filippov solutions are too general to be of use. Instead, since the z dynamics exhibit both continuous and discrete transitions, we will adopt our notion of solution from the hybrid system literature.

If we augment z to include time as an additional state (i.e. $z_a \triangleq [z^T, \pi]^T$, $\pi_0 = t_0$, $\dot{\pi} = 1$, $\pi^+ = \pi$), then (15) has the form $\dot{z}_a = F(z_a)$ on the *flow domain*

$$S_F \triangleq \left\{ z_a : \pi \le t_1^\theta \text{ and } (\theta, t^\theta) \in \Phi^N(\pi, x) \right\} \subset \mathbb{R}^{n+mN+N+1}$$

Likewise, (16) has the form $z_a^+ = H(z_a)$ on the *jump domain*

$$S_H \triangleq \left\{ z_a : \pi \ge t_1^\theta \text{ and } (\theta, t^\theta) \in \Phi^N(\pi, x) \right\} \subset \mathbb{R}^{n+mN+N+1}$$

The hybrid dynamics of z_a are therefore of the form discussed in [11], [12]. We follow the notation of [11] when we explicitly denote a solution $z_a(t, k)$ as evolving over both ordinary time t and event time k, jointly referred to as *hybrid time*. Since from (16) we have that $x(t, k+1) = x(t, k)$, we will write $x(t) \equiv x(t, k)$ with the understanding that $x(t)$ is still a component of the hybrid-time trajectory $z_a(t, k)$. Existence and uniqueness of the hybrid time solution $z_a(t, k)$ follows from [12, Theorem III.1] and [12, Lemma III.2] respectively.

C. Main Result

We are now ready to present the main result of this paper. While the theorem itself may appear to be a straightforward consequence of Criteria 1 and 2, the usefulness of its generality will become apparent in section V when we provide examples of $\Psi(t, z)$ and $\Upsilon(t, z)$ meeting these criteria.

Theorem 1: Let $\kappa_\delta(\cdot)$, $W(\cdot)$, $\mathcal{X}_f$ be chosen to satisfy Assumption 1 for given Σ, and let B_x, B_f, μ, μ_f satisfy Assumption 2. For any initial conditions $(t_0, x_0) \in \mathbb{R} \times \mathcal{X}^0$ of (2), and any initial feasible parameterization $(\theta, t^\theta)_0 \in \Phi^N(t_0, x_0)$, the set $x \in \Sigma$ is asymptotically stabilized by the algorithm of Section IV-A. Furthermore, the resulting trajectories satisfy all input, state, and terminal constraints.

V. Flow and Jump Mappings Ψ and Υ

As shown in Theorem 1, asymptotic convergence to $x \in \Sigma$ is guaranteed as long as $\Psi(t, z)$ and $\Upsilon(t, z)$ do not cause increases in $J(t, z)$. Of course, since the initial control parameterization was assumed to be feasible, this is no real surprise. Our interest is in the ability of both Ψ and Υ to improve upon the initial control parameterization. To this end, we will look at each mapping individually.

A. Decrease by Υ: The Standard MPC Approach

The primary difference between our approach and the "standard" NMPC literature is the time-varying nature of our control parameterization vector θ. However, if we make the following choices for Ψ and Υ

$$\Psi(t, z) \equiv 0 \tag{17}$$

$$\Upsilon(t, z) = \arg\min_{\theta^+ \in \mathbb{U}^N} J(t, x, \theta^+, t^{\theta+}) \tag{18}$$

then the resulting trajectories will be identical to those generated by standard approaches to MPC using PWC controls.

Criterion 1 is trivial, while Criterion C2.1 follows from

$$\begin{aligned} J(t, z^+) - J(t, z) &= \int_t^{t_N^{\theta+}} L_\rho^a(x^p(\tau, t, z^+), u^p(\tau, t, z^+))\, d\tau \\ &\quad - \int_t^{t_N^\theta} L_\rho^a(x^p(\tau, t, z), u^p(\tau, t, z))\, d\tau \\ &\quad + W_\rho^a(x^p(t_N^{\theta+}, t, z^+)) - W_\rho^a(x^p(t_N^\theta, t, z)) \\ &\le \int_{t_N^\theta}^{t_N^{\theta+}} L_\rho^a(x^p(\tau, t_N^\theta, \bar{z}), u^p(\tau, t_N^\theta, \bar{z}))\, d\tau \\ &\quad + W_\rho^a(x^p(t_N^{\theta+}, t_N^\theta, \bar{z})) - W_\rho^a(x_f^p) \\ &\le \rho(x_f^p) \left[\int_{t_N^\theta}^{t_N^{\theta+}} L^a(x^p(\tau, t_N^\theta, \bar{z}), u^p(\tau, t_N^\theta, \bar{z}))\, d\tau \right. \\ &\quad \left. + W^a(x^p(t_N^{\theta+}, t_N^\theta, \bar{z})) - W^a(x_f^p) \right] \\ &\le 0 \end{aligned} \tag{19}$$

where $x_f^p \triangleq x^p(t_N^\theta, t, z)$, and $\bar{z} \triangleq [x_f^p, \theta^+, t^{\theta+}]$. Criterion C2.2 follows from C2.1 and the fact that $J(t, z^+) \to \infty$ continuously as $x^p(t_f) \to \partial\mathcal{X}_f$ or any point $x^p(\tau)$ approaches $\partial\mathbb{X}$. The input constraint is enforced by the indicated minimization.

While the theoretical merit of this approach is well established in the literature, it requires the assumption that the mapping $\Upsilon(t, z)$ is instantaneous. This mapping involves a potentially difficult minimization in which *every iteration* of the nonlinear program involves solving prediction trajectories $x^p(\tau)$ and $u^p(\tau)$, evaluating $J(t, x, \theta^+, t^{\theta+})$, and solving all sensitivity equations necessary to calculate $\nabla_{\theta^+} J$ (used in descent-based NLP's).

B. Decrease by Ψ: A New Approach to MPC

Our main goal in this work is to demonstrate that minimization of $J(t, z)$ can be performed in a manner which more realistically accounts for the dynamics associated with asymptotic convergence of an NLP. In this approach, the burden of minimization is carried by Ψ, which allows Υ to be simplified to an explicit, optimization-free expression. The simplest choice for Υ is

$$\Upsilon(t, z) := \Theta_{i,j}^+ = \begin{cases} \Theta_{i,j+1} & j \in \{1, \dots N-1\} \\ \left(\kappa_\delta \left(x^p(t_N^\theta, t, z) \right) \right)_i & j = N \end{cases} \qquad \forall i \in \{1, \dots m\} \tag{20}$$

Satisfaction of criterion C2 follows from (19) and the associated arguments of the previous section.

A general descent-type continuous time update law $\Psi(t, z)$ can be described as

$$\Psi(t, z) = \text{Proj}\left\{\vartheta(t, z),\, \Gamma(t, z),\, \theta,\, \mathbb{U}^N\right\} \qquad (21)$$
$$\vartheta(t, z) = -k_\theta \Gamma(t, z)\nabla_\theta J$$

where $\Gamma : \mathbb{R} \times \mathbb{X} \times \Phi^N(t, x) \rightarrow M^{mN}_{>0}$ is a locally Lipschitz function, $M^{mN}_{>0}$ is the space of positive definite square matrices of size mN, and $k_\theta > 0$ is a constant. The gain k_θ controls the rate of descent, while $\Gamma(t, z)$ is the variable metric used to select the descent direction. Possible examples include $\Gamma(t, z) = I$ (steepest descent), $\Gamma(t, z) = \left[\nabla^2_{\theta\theta} J + \left(\left\|\nabla^2_{\theta\theta} J\right\|_F + \epsilon_J\right) I\right]^{-1}$ (Newton's Method with trust region), or any number of other choices based on descent direction-based NLP methods.

The Proj$\{\cdot, \cdot, \cdot, \cdot\}$ operator is a locally Lipschitz parameter projection, discussed in [13], [14] in the context of nonlinear adaptive control, which ensures that each of the N columns of Θ satisfy $\Theta_i \in \mathbb{U}$. For constraint sets $\mathbb{U}$ with smooth boundary $\partial\mathbb{U}$ and $\mathring{\mathbb{U}} \neq \emptyset$, one standard definition is:

$$\text{Proj}\left\{\vartheta, \Gamma, s, \mathbb{S}\right\} \triangleq \begin{cases} \vartheta & s \in \mathring{\mathbb{S}}_r \\ & \text{or } v_\perp^T \vartheta \leq 0 \\ \left(I - c(s)\Gamma \frac{v_\perp v_\perp^T}{v_\perp^T \Gamma v_\perp}\right)\vartheta & s \in \mathbb{S} \setminus \mathring{\mathbb{S}}_r \\ & \text{and } v_\perp^T \vartheta > 0 \end{cases}$$
$$c(s) = \min\left\{1,\, \frac{r - \epsilon}{r}\right\}, \qquad r > 0 \qquad (22)$$

where $\mathbb{S}_\epsilon$, $\epsilon \in [0, r]$, denotes a family of closed inner approximations to $\mathbb{S}$ strictly satisfying $\mathbb{S}_\epsilon \subset \mathbb{S}_{\epsilon'}$ for $\epsilon > \epsilon'$, and where $\partial\mathbb{S}_\epsilon$ continuously approaches $\partial\mathbb{S}$ as $\epsilon \rightarrow 0^+$. The vector $v_\perp$ is outward normal to $\mathbb{S}_\epsilon$, with $\epsilon \equiv \epsilon(s)$ the level curve satisfying $s \in \partial\mathbb{S}_\epsilon$. As an obvious extension of [14, Lemma E.1], we assert (without proof) the following:

Assertion 1: i) Proj$\{\vartheta, \Gamma, s, \mathbb{S}\}$ is locally Lipschitz in all of its arguments. ii) The solution to $\dot{s} = \text{Proj}\{\vartheta, \Gamma, s, \mathbb{S}\}$ from $s_0 \in \mathbb{S}$ satisfies $s(t) \in \mathbb{S}$, for all $t \geq 0$. iii) $\vartheta^T \Gamma^{-1} \text{Proj}\{\vartheta, \Gamma, s, \mathbb{S}\} \leq \vartheta^T \Gamma^{-1} \vartheta$.

Assertion 1.iii and the positive-definiteness of $\Gamma(t, z)$ imply that (21) satisfies criterion C1.1. By Assertion 1.ii, the resulting control $v(t, \Theta, t^\theta)$ from (12) satisfies the input constraints. Condition C1.2 follows from the fact that C1.1 ensures $\dot{J} < 0$ (see proof of Theorem 1), and that $J(t, z) \rightarrow \infty$ continuously as $x^p(t_f) \rightarrow \partial\mathcal{X}_f$ or any point $x^p(\tau) \rightarrow \partial\mathbb{X}$.

VI. Simulation Example

To illustrate the concept of real-time optimization proposed in this work, we consider a simple nonlinear example from [15],

$$\dot{x}_1 = x_2 + (0.5 + 0.5x_1)u$$
$$\dot{x}_2 = x_1 + (0.5 - 2x_2)u$$

with definitions $\mathbb{U} = [-2, 2]$ and $L(x, u) = 0.5\left\|x\right\|^2 + u^2$.

The PWC local stabilizer was designed using the exact discretization of the linearized process for a constant switching interval of $\delta = 0.5$, yielding the feedback $\kappa(x) = [0.1402,\ 0.1402]x$. The terminal penalty

$$W(x) = x^T \begin{bmatrix} 3.6988 & 2.8287 \\ 2.8287 & 3.6988 \end{bmatrix} x$$

was obtained from a Lyapunov equation, and the corresponding terminal region $\mathcal{X}_f = \{x : W(x) \leq 0.141\}$ was enforced using a logarithmic barrier.

The system was simulated from $x_0 = [-0.683,\ -0.864]$, using several different controllers. The four "real-time" (RT-) controllers are based on section V-A, using a simple steepest descent choice for Γ. For comparison, the "standard" (S) controller is based on section V-B. Controller parameters and their accumulated costs $\int_0^\infty L(x, u)d\tau$ are given in Table I.

TABLE I
CONTROLLER PARAMETERS AND PERFORMANCE

name	Υ	Ψ	Γ	δ	N	k_θ	cost
RT-1	(20)	(21)	-	0.5	3	0	6.329
RT-2	(20)	(21)	I	0.5	3	1	5.134
RT-3	(20)	(21)	I	0.5	3	10	4.822
RT-4	(20)	(21)	I	0.5	3	100	4.800
S	(18)	(17)	-	0.5	3	-	4.807

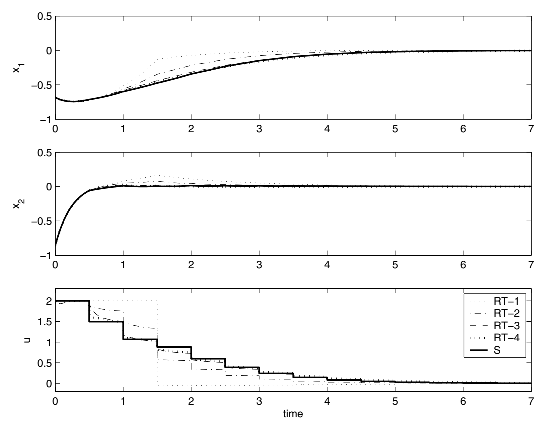

Fig. 1. Closed-loop response of different controllers

It can be seen from the results in Figure 1 that all five controllers stabilize the origin, and that as k_θ is increased the trajectories of the RT controllers approach very close to that attained by the standard MPC controller. In fact, controller RT-4 marginally outperforms controller S, due to the fact that the input is recomputed throughout the switching interval.

VII. Conclusions

In this work, we have proposed a framework for continuous-time MPC in which the dynamics associated with the NLP incorporated into the controller design, and allowed to evolve in the same timescale as the process dynamics without compromising closed-loop stability. By allowing for stabilization to a set, a broad range of design methods for the piecewise constant local stabilizer is permitted. While our result only guarantees local improvement of an initial feasible input trajectory, this limitation is shared by most other implementable approaches such as those using SQP's.

Appendix

A. Proof of Lemma 1

Let $x^o(t)$, $u^o(t)$ be a specific open-loop trajectory satisfying the stated conditions, for a particular $x_0 \in \mathcal{X}^0$. Let $\mathbb{S}$ be any compact set satisfying $x^o(t) \in \mathring{\mathbb{S}}, \forall t \in [t_0, t_f]$ and $\mathbb{S} \subset \mathring{\mathbb{X}}$, and define $d_{\mathbb{S}} = \min_{(s,\, t) \in \partial\mathbb{S} \times [t_0, t_f]} \|s - x^o(t)\|$. Likewise, let $x^o(t_f) \in \mathbb{S}_f \subset \mathring{\mathcal{X}}_f$ and define $d_{\mathbb{S}_f} = \min_{s \in \partial \mathbb{S}_f} \|s - x^o(t_f)\|$. Finally, define $d \triangleq \min\{d_{\mathbb{S}},\, d_{\mathbb{S}_f}\}$.

Let $w(t)$ denote the solution to the perturbed system

$$\dot{w} = f(w, u^o(t)) + g(t, w, \theta, t^\theta), \qquad w(t_0) = x_0$$

where $g(t, w, \theta, t^\theta) \triangleq f(w, v(t,\theta), t^\theta) - f(w, u^o(t))$ for an (as-yet unspecified) parameterization (θ, t^θ). Define $M_g \equiv M_g(\theta, t^\theta) \triangleq \int_{t_0}^{t_f} \|g(\tau, w(\tau))\| \, d\tau$, and let K_x be the Lipschitz constant of $f(x,u)$ w.r.t. $x \in \mathbb{S}$, uniformly for $u \in \mathbb{U}^o$, a compact subset of $\mathbb{U}$ such that $u^o(t) \in \mathbb{U}^o, \forall t \in [t_0,\, t_f]$. Using the Gronwall-Bellman inequality [14, Lemma B11]

$$\begin{aligned}
\|x^o(t) - w(t)\| &\leq \int_{t_0}^{t_f} \|f(x^o, u^o) - f(w, u^o)\| \, d\tau + M_g \\
&\leq \int_{t_0}^{t_f} K_x \, \|x^o(\tau) - w(\tau)\| \, d\tau + M_g \\
&\leq M_g \exp[K_x(t_f - t_0)]
\end{aligned}$$

from which the result follows if it can be shown that the set of parameterizations satisfying $M_g \leq d \exp[K_x(t_0 - t_f)] \triangleq M_d$ has positive measure.

Let N_d denote the number of discontinuities in $u^o(t)$ on $t \in [t_0,\, t_f]$, and assume $N \gg N_d$. Let (θ, t^θ) be a corresponding parameterization such that θ is defined by $\Theta_{i,j} = \lim_{t \to (t_j^\theta)^+} u_i^o(t)$, and $t_N^\theta = t_f$. Define π_d to be the (disjoint) set of intervals of the form $(t_{i-1}^\theta,\, t_i^\theta]$ containing the discontinuities. Likewise, define $\pi \triangleq [t_0,\, t_f] \setminus \pi_d$, and let $d_t \triangleq \max_i \left| t_i^\theta - t_{i-1}^\theta \right|$. Then

$$\begin{aligned}
M_g \leq& \int_\pi \left\| g(\tau, w, \theta, t^\theta) \right\| d\tau + \int_{\pi_d} \left\| g(\tau, w, \theta, t^\theta) \right\| d\tau \\
\leq& (t_f - t_0) \min\left\{ M_f,\, \left(K_u \sup_{\tau \in \pi} \left\| u^o(\tau) - v(\tau, \theta, t^\theta) \right\| \right) \right\} \\
&+ N_d d_t M_f
\end{aligned} \tag{23}$$

where $M_f \triangleq \max_{(x, u_1, u_2) \in \mathbb{S} \times \mathbb{U}^o \times \mathbb{U}^o} \|f(x, u_1) - f(x, u_2)\|$, and K_u is the Lipschitz constant of $f(x,u)$ w.r.t u, on the set $(x, u) \in \mathbb{S} \times \mathbb{U}^o$. By the continuity of $u^o(t)$ on $t \in \pi$, the supremum in (23) approaches zero continuously as $d_t \to 0$, which implies that $M_g \leq M_d$ follows for sufficiently small d_t. If $M_g < M_d$, then the conclusion holds for small perturbations in θ or t^θ in directions feasible w.r.t. $\mathbb{U}^N$. ∎

B. Proof of Theorem 1

Using the cost (14), we will use a hybrid systems Invariance Principle [12, Theorem IV.1] requiring nonincrease of $J_k(t) \triangleq J(t, z(t,k))$ under both flow and jump dynamics.

Ordinary-time Evolution

By standard arguments, C1.3 guarantees $z(t,k)$ exists and is continuous on some nonzero subinterval of $[t_i^\theta,\, t_{i+1}^\theta]$, with constant k. Using $\dot{J}_k$ to denote $\frac{d}{dt} J_k(t)$, from (14)

$$\dot{J}_k = \nabla_t J + \langle \nabla_x J,\, f(x(t,k), u(t,k)) \rangle + \langle \nabla_\theta J, \dot{\theta} \rangle$$

where $u(t,k) \triangleq v(t, \theta(t,k), t^\theta(k))$. From (14) it follows $\nabla_t J = \langle \nabla_x J,\, f(x(t,k), u(t,k)) \rangle - L_\rho^a(x(t,k), u(t,k))$, so

$$\begin{aligned}
\dot{J}_k =& -L_\rho^a(x(t,k), u(t,k)) + \langle \nabla_\theta J, \dot{\theta} \rangle \\
\leq& -L(x(t,k), u(t,k)) + \langle \nabla_\theta J, \dot{\theta} \rangle, \quad \forall x \in \mathbb{X} \setminus \mathring{\Sigma} \\
\leq& -\gamma_L(\|x(t,k), u(t,k)\|), \qquad \forall x \in \mathbb{X} \setminus \mathring{\Sigma}
\end{aligned} \tag{24}$$

where the second inequality follows from C1.1. From (13), the first line of (24) implies that $\dot{J}_k \leq 0$ when $x \in \mathring{\Sigma}$.

Event-time Evolution

Defining $S \triangleq \left\{ z_a : (\theta, t^\theta) \in \Phi^N(\pi, x) \right\}$ as the set of feasible states, we have $S_H = S \setminus \mathring{S}_F$, guaranteeing that $H(z_a)$ is defined whenever continuous flows are not. Criterion C2.1 then directly gives that $J_{k+1}(t) \leq J_k(t)$ under mapping H.

From [12, Theorem IV.1], the above implies that z_a converges asymptotically to the invariant set $M = \{z_a : J_{k+1} - J_k = 0 \text{ under } H(\cdot)\} \cup \{z_a : \dot{J}_k = 0 \text{ under } F(\cdot)\}$. With $dom\{H\}$ and $rng\{H\}$ disjoint, (24) gives $z_a \in M \implies x \in \Sigma$. Feasibility follows directly from C1.1 and C2.2. ∎

References

[1] F. Fontes, "A general framework to design stabilizing nonlinear model predictive controllers," *Systems and Control Letters*, vol. 42, no. 2, pp. 127–143, 2001.

[2] F. Clarke, Y. Ledyaev, E. Sontag, and A. Subbotin, "Asymptotic controllability implies feedback stabilization," *IEEE Trans. Automat. Contr.*, vol. 42, no. 10, pp. 1394–1407, 1997.

[3] R. Findeisen and F. Allgöwer, "Stabilization using sampled-data open-loop feedback – a nonlinear model predictive control perspective," in *Proc. IFAC Symposium on Nonlinear Control Systems*, 2004, pp. 735–740.

[4] L. Magni and R. Scattolini, "Model predictive control of continuous-time nonlinear systems with piecewise constant control," *IEEE Trans. Automat. Contr.*, vol. 49, no. 6, pp. 900–906, 2004.

[5] D. Henriksson, A. Cervin, J. Åkesson, and K. E. Årzén, "On dynamic real-time scheduling of model predictive controllers," in *Proc. IEEE Conf. on Decision and Control*, 2002, pp. 1325–1330.

[6] P. O. M. Scokaert, D. Q. Mayne, and J. B. Rawlings, "Suboptimal model predictive control (feasibility implies stability)," *IEEE Trans. Automat. Contr.*, vol. 44, no. 3, pp. 648–654, 1999.

[7] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, pp. 789–814, 2000.

[8] D. Nešić and A. Teel, "A framework for stabilization of nonlinear sampled-data systems based on their approximate discrete-time models," *IEEE Trans. Automat. Contr.*, vol. 49, pp. 1103–1122, 2004.

[9] Y. Nesterov and A. Nemirovskii, *Interior-Point Polynomial Algorithms in Convex Programming*. Philadelphia: SIAM, 1994.

[10] A. Wills and W. Heath, "Barrier function based model predictive control," *Automatica*, vol. 40, pp. 1415–1422, 2004.

[11] R. Goebel, J. Hespanha, A. Teel, C. Cia, and R. Sanfelice, "Hybrid systems: generalized solutions and robust stability," in *Proc. IFAC Symposium on Nonlinear Control Systems*, 2004, pp. 1–12.

[12] J. Lygeros, K. Johansson, S. Simić, J. Zhang, and S. Sastry, "Dynamical properties of hybrid automata," *IEEE Trans. Automat. Contr.*, vol. 48, no. 1, pp. 2–17, 2003.

[13] J. B. Pomet and L. Praly, "Adaptive nonlinear regulation: Estimation from the lyapunov equation," *IEEE Trans. Automat. Contr.*, vol. 37, no. 6, pp. 729–740, 1992.

[14] M. Krstic, I. Kanellakopoulos, and P. Kokotovic, *Nonlinear and Adaptive Control Design*. New York: Wiley and Sons, 1995.

[15] H. Chen and F. Allgöwer, "A quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability," *Automatica*, vol. 34, no. 10, pp. 1205–1217, 1998.