Dual-Sampling-Rate Moving Horizon Control of a Class of Linear Systems with Input Saturation and Plant Uncertainty[1]

A. Brockwell[2], E. Polak[3], R. Evans[4], D. Ralph[5]

Abstract. Moving horizon control is a type of sampled-data feedback control in which the control over each sampling interval is determined by the solution of an open-loop optimal control problem. We develop a dual-sampling-rate moving horizon control scheme for a class of linear, continuous-time plants with strict input saturation constraints, in the presence of plant uncertainty and input disturbances. Our control scheme has two components: a slow-sampling moving horizon controller for a nominal plant, and a fast-sampling state-feedback controller whose function is to force the actual plant to emulate the nominal plant. The design of the moving horizon controller takes into account the non-negligible computation time required to compute the optimal control trajectory.

We prove local stability of the resulting feedback system and illustrate its performance with simulations. In these simulations, our dual-sampling-rate controller exhibits performance that is considerably superior to its single-sampling-rate moving horizon controller counterpart.

[1]This work has been supported by the Australian Research Council, the Melbourne Node of the Cooperative Research Centre for Sensor Signal and Information Processing, the National Science Foundation under Grants DMS-9819950, IIS-0083148, NSF-INT-9725220 and ECS-9900985, and also by the US Office of Naval Research under Contract N00014-92-J-1524.

[2]Visiting Assistant Professor, Department of Statistics, Carnegie Mellon University, Pittsburgh, Pennsylvania.

[3]Professor, Department of Electrical Engineering and Computer Science, The University of California at Berkeley, Berkeley, California.

[4]Professor, Department of Electrical and Electronic Engineering, The University of Melbourne, Parkville, Australia.

[5]University Lecturer, The Judge Institute of Management,University of Cambridge, Trumpington Street, Cambridge, UK.

Key Words. moving horizon control, robust, fast sampling, stability, linear systems, input constraints

1 Introduction

Moving horizon control, also referred to as model predictive control, or receding horizon control, is a form of state-feedback sampled-data control which is distinguished from classical sampled-data control primarily by the way in which the control is computed. Instead of using a simple feedback law, the control is re-computed at periodic intervals as the solution of an optimal control problem, each time using an up-to-date measurement of the current state. As a result, moving horizon control can account for control saturation effects, system nonlinearities, and bounds on state excursions as no classical feedback law can. Hence moving horizon control has become particularly popular in the process industry, where the system dynamics are sufficiently slow for the computation time needed to solve the optimal control problem not to be an issue. An excellent survey of the state-of-the-art and literature in moving horizon control can be found in Ref. 1. Here we see that closed-loop stability of the moving horizon control feedback system can be ensured in at least half a dozen of ways by inserting certain features in the open-loop optimal control problem that must be solved at each sampling time. These range from formulating the optimal control problem in such a way that its value function becomes an optimality function for the closed-loop system, Refs. 2-4, to a contraction technique used in Refs. 5-7 which is based on ideas first introduced in Ref. 8. More recently, the contraction technique of Refs. 5-7 has also been used in Ref. 9.

Conventional moving horizon control systems are sometimes more sensitive to plant modeling errors and disturbances than classical feedback systems, because the solutions of open-loop optimal control problems do not take into account these errors and disturbances. Hence robustness and disturbance rejection remain important issues in moving horizon control. An analysis of the robustness of particular schemes can be found in Refs. 4-7, among others cited in Ref. 1. Also, in Ref. 1 we find a description of several attempts to deal with the robustness issue by modifying the optimal control problem being solved at each sampling point, ranging from very conservative worst-case (min-max) optimization, see, e.g. Refs. 4,10, to more sophisticated schemes, e.g., Refs. 11-13.

In this paper we present a dual-sampling-rate moving horizon control scheme for a class of linear plants, with input saturation and input disturbances, in the presence of modeling errors. In our scheme, the available control action is divided into two components. The first component is allocated to a slow-sampling contractive moving horizon control law, using nominal plant dynamics, as in Refs. 5-7, and Ref. 9. This law ensures stability for the overall feedback system by means of a pair of Lyapunov-like contractive inequalities in the open-loop optimal problem which it solves during each sampling period. In addition, it takes into account the time needed to solve the open-loop optimal control problem.

The second control component is allocated to a fast-sampling feedback law which forces the actual plant to emulate the nominal plant dynamics. Variations of this feedback law, which is based on the idea of using feedback for pole placement, have been studied in Refs. 14-17. However, the asymptotic stability results obtained in Refs. 16-17 cannot be applied directly to the problem considered in this paper for two main reasons. Firstly, in this paper the performance of the fast-sampling controller, on finite (possibly small) intervals, is of critical importance, and secondly, in this paper the fast-sampling controller is used to force the state to track a reference trajectory, instead of simply driving it towards the origin.

In fact, the control scheme we propose can be regarded as a special type of feedback moving horizon control. Other more general forms of such controllers can be found in Ref. 18, where the control is broken into a pre-stabilizing component and an open-loop optimal component, and in Refs. 19-20, where the control is divided into an open-loop optimal component and a state-feedback component.

Although the ideas on which our moving horizon control scheme are based are relatively simple, proving that it actually works has turned out to be quite difficult. In Section 2, we formulate the control problem to be addressed. In Section 3, we introduce our proposed control law, defining each of its two primary components in detail. We also motivate the construction of the fast-sampling law for the feedback part of the control law. In Section 4, we analyze closed-loop stability of the system under our proposed dual-sampling-rate control law. Towards the end of the section, the main results are stated. In Section 5, simulation results are presented which demonstrate the effectiveness of our control scheme, and, finally, some concluding remarks are presented in Section 6.

2 Problem Formulation

Suppose that the plant to be controlled satisfies the differential equation

$$y^{(p)}(t) = a_{p-1}y^{(p-1)}(t) + a_{p-2}y^{(p-2)}(t) + \ldots + a_0 y(t) + d(t) + u(t), \tag{1}$$

where $y(t) \in \mathbb{R}$, $y^{(j)}(t)$ is the jth derivative with respect to time of $y(t)$ evaluated at time t, and p is the order of the system. The function $d : \mathbb{R} \to \mathbb{R}$ is some unknown disturbance which satisfies the Lipschitz condition

$$|d(t) - d(s)| \leq \gamma |t - s|, \quad \forall s, t \in \mathbb{R} \tag{2}$$

for some γ such that $0 \leq \gamma < \infty$ (this condition (2) can be motivated by the fact that the plant acts as a low-pass filter and hence we can assume that the disturbance does not contain components of arbitrarily high frequency), and is bounded by

$$|d(t)| \leq D \quad \forall t \in \mathbb{R}, \quad 0 \leq D < \infty. \tag{3}$$

Let

$$U_\delta^x := \{u(\cdot) : u(s) = \sum_{j=0}^{\infty} u_j I_{[j\delta,(j+1)\delta)}(s),\ |u_j| \leq x\ \forall j \geq 0\} \tag{4}$$

for $\delta > 0$, where $I_A(\cdot)$ is the indicator function for the set A, that is,

$$I_A(s) := \begin{cases} 1, & s \in A, \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

(U_δ^x is the set of controls which are piecewise constant on intervals $[0,\delta), [\delta, 2\delta), \ldots$. and bounded in magnitude by x.) The control $u(t)$ is broken into two components:

$$u(t) = u_s(t) + u_c(t),$$

with the requirement

$$u_s(t) \in U_\delta^S \quad \textit{and} \quad u_c(t) \in U_\delta^\infty, \tag{6}$$

for some positive constant S.

The "open-loop-optimal control component" $u_s(t)$, is bounded above in magnitude by the constant S, and is to be computed by solving an open-loop optimal control problem which imposes a set of state constraints. The main purpose of these constraints is to ensure contractive stability, as well as to prevent the state from entering undesirable regions. (The specific form of these constraints is described in the next section.)

The "compensating control" $u_c(t)$ will be used to compensate for unmodelled plant behavior, and at this stage, is not bounded in magnitude. However, later we will see that for the state in a suitably defined neighborhood or the origin, when $u_c(t)$ is chosen according to the control law we propose, its magnitude will indeed be bounded uniformly in time. Thus to bound the magnitude of the control $u(t)$, it will suffice to choose S slightly smaller than the intended bound on total control magnitude, to allow some leeway for the control action contributed by $u_c(t)$.

The sampling interval δ is to be regarded as a design parameter, that is, it must be chosen carefully in order to ensure good closed-loop system performance. Later in this paper, we show that under certain assumptions, one can always find such a δ.

In order to deal with issues of robustness, we assume that the parameters $a_0, \ldots, a_{p-1}, d$ are not known exactly. Rather, a set of assumed parameters $\tilde{a}_0, \ldots, \tilde{a}_{p-1}, \tilde{d}$ is used in conjunction with the "assumed plant model"

$$\tilde{y}^{(p)}(t) = \tilde{a}_{p-1}\tilde{y}^{(p-1)}(t) + \tilde{a}_{p-2}\tilde{y}^{(p-2)}(t) + \ldots + \tilde{a}_0\tilde{y}(t) + \tilde{d} + u_s(t), \tag{7}$$

with $|\tilde{d}| \leq D$, to generate the open-loop control law. The constant $\tilde{d}$ is to be chosen as a rough guess of the unknown input offset $d(t)$. In theory this guess could be updated periodically as the plant operates, but we will simply regard it as a fixed constant.

For the sake of conciseness, we will usually write (1) and (7) in the forms

$$\dot{x}(t) = Ax(t) + b[d(t) + u(t)] \tag{8}$$

and

$$\dot{\tilde{x}}(t) = \tilde{A}\tilde{x}(t) + b[\tilde{d} + u_s(t)] \tag{9}$$

respectively, where

$$A = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \\ a_0 & a_1 & a_2 & \dots & a_{p-1} \end{bmatrix}, \tilde{A} = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \\ \tilde{a}_0 & \tilde{a}_1 & \tilde{a}_2 & \dots & \tilde{a}_{p-1} \end{bmatrix} \tag{10}$$

$x(t) = \left(y(t), \dots, y^{(p-1)}(t)\right)^T$, $\tilde{x}(t) = \left(\tilde{y}(t), \dots, \tilde{y}^{(p-1)}(t)\right)^T$, and $b = (0, \dots, 0, 1)^T$. Thus $x(t)$ is the actual state of the system at time t, and $\tilde{x}(t)$ is an approximation to $x(t)$ based on the use of the model (9).

Our object is to drive the (actual) system state near the origin (i.e. regulate it) using a control law which satisfies the constraints given in the next section and minimizes some optimization criterion.

Note that we impose the relatively strong assumption that b is known exactly, and we do not explicitly impose a bound on the difference between A and $\tilde{A}$. It will become apparent later that as the magnitude of $A - \tilde{A}$ increases, the magnitude of the compensating control $u_c(t)$ will also increase.

3 Control Laws

The controller which we propose consists of two components:

1. A slow-sampling, open-loop optimal, controller, with associated slow-sampling interval Δ and computing time $T_C < \Delta$. For $i = 1, 2, 3, ...$, at time $i\Delta - T_C$, this controller measures the plant state, computes an estimate of the plant state at time $i\Delta$, and computes an optimal control $u_i^*(t)$ and associated state trajectory $\tilde{x}_i^*(t)$, for the model plant (9), by solving an open-loop optimal control problem defined on the interval $[i\Delta, i\Delta + T_h]$, with $\Delta \leq T_h < \infty$, for the model plant

(9). A pair of contractive inequalities in the optimal control problem ensure that, in the presence of no modeling errors, the resulting feedback system is stable in the sense of Lyapunov.

2. A fast-sampling, sampled-data state-feedback controller, with associated sampling time $\delta = \Delta/N$ (N a positive integer) which measures the state at the times $k\delta$, $k = 0, 1, 2, 3, ...$ and computes the piecewise constant control $u_c(t)$ to ensure that, even in the face of plant uncertainty and external disturbances $d(t)$, the plant follows (closely) the model plant trajectory computed by the open loop controller.

3.1 Slow-Sampling Open-Loop-Optimal Controller

Using the measured plant state $x(i\Delta - T_C)$ and a precomputed nominal trajectory $\tilde{x}^*_{i-1}(t)$ (which is a solution $\tilde{x}(t) = \tilde{x}^*_{i-1}(t)$ of (9) for some choice of $u(\cdot)$, for $t \in [(i-1)\Delta, i\Delta]$), the plant state at time $i\Delta$ is predicted to be

$$\hat{x}(i\Delta) = e^{HT_C}[x(i\Delta - T_C) - \tilde{x}^*_{i-1}(i\Delta - T_C)] + \tilde{x}^*_{i-1}(i\Delta), \tag{11}$$

where H is as in (25) and (26), to be discussed later.

Let $\tilde{x}(t; x_0, t_0, v)$ denote the solution of (9) with $u(t) = v(t)$ and $\tilde{x}(t_0) = x_0$. The fixed horizon, open-loop optimal control problem, $\mathbf{P}(\hat{x}(i\Delta), i\Delta)$, to be solved in the time interval $[i\Delta - T_C, i\Delta]$, with initial state $\hat{x}(i\Delta) \in \mathbb{R}^p$, initial time $i\Delta$, and final time $i\Delta + T_h$, $\Delta \leq T_h < \infty$, takes the form

$$\mathbf{P}(\hat{x}(i\Delta), i\Delta), \qquad \min_{u \in U^S_\delta} \quad \begin{array}{l} \{g^0(u) | g^i(u) \leq 0, \; i = 1, 2, \ldots, l_1, \\ \max_{s \in [i\Delta, i\Delta + T_h]} \phi^j(u, s) \leq 0, \; j = 1, 2, \ldots, l_2\}, \end{array}$$

where the cost and constraint functions are defined by

$$\begin{aligned} g^i(u) &:= h^i(\tilde{x}(i\Delta + T_h; \hat{x}(i\Delta), i\Delta, u)), \; i = 0, 1, \ldots, l_1 - 1, & (12)\\ g^{l_1}(u) &:= \|\tilde{x}((i+1)\Delta; \hat{x}(i\Delta), i\Delta, u)\|^2 - \alpha^2 \|\hat{x}(i\Delta)\|^2, & (13)\\ \phi^j(u, s) &:= \hat{h}^j(\tilde{x}(s; \hat{x}(i\Delta), i\Delta, u), s), \; j = 1, \ldots, l_2 - 1, & (14)\\ \phi^{l_2}(u, s) &:= \|\tilde{x}(s; \hat{x}(i\Delta), i\Delta, u)\|^2 - \beta^2 \|\hat{x}(i\Delta)\|^2. & (15) \end{aligned}$$

Note that although we wrote the cost function as depending only on the terminal state $\tilde{x}(i\Delta + T_h; \hat{x}(i\Delta), i\Delta, u)$, we could have equally well used a cost function in sum form (as we do in our simulations), without affecting the validity of the results to follow.

The requirements that

$$g^{l_1}(u) \leq 0 \tag{16}$$

and

$$\phi^{l_2}(u, s) \le 0, \tag{17}$$

with $\alpha \in (0, 1)$ and $\beta \in [1, \infty)$, are used to ensure stability. (For more discussion on selection of α and β, see Ref. 5, where the same kind of open-loop optimal control problem is used.) The functions h^i and $\hat{h}^j$, $i = 0, \ldots, l_1 - 1$ and $j = 1, \ldots, l_2 - 1$, are convex, locally Lipschitz, continuously differentiable, and can be used to ensure that various performance requirements (other than those required for stability) are satisfied.

The construction of the optimal control problem requires some care. The major consideration is to make sure that the optimal control problem has a solution, for otherwise the control becomes undefined[6].

There are two major cases to consider. The first is when the plant model is strictly stable and there is no need to enforce state-space constraints exactly. In this case, there is a natural decay constant $\alpha < 1$ (provided we use a quadratic norm based on a Lyapunov function), for which (13) will be satisfied with the control set to zero, and β is easy to pick. The state-space constraints can be bricked into the cost function as a penalty function, and the desire for a faster decay than that given by α can also be dealt with by an appropriate term in the cost function. Then the optimal control problem will have a solution for any initial state.

The second case is when either the plant model is unstable, or one needs to enforce the state-space constraints rigorously (e.g., sometimes the state must be prevented from venturing into regions, such as high temperature regions, where serious damage to the system or product may result). In this case, the best one can hope for is that there is a ball of initial states around the origin for which a stabilizing control satisfying state-space constraints can be found.

3.2 Fast-Sampling Feedback Controller

Standard contractive moving horizon control schemes such as those described in Refs. 5-7, and Ref. 9, use an open-loop-optimal controller, and introduce feedback into the system by re-computing the control at successive sampling instants. Because of this feedback, they show some tolerance to plant uncertainty and input disturbances. However, plant uncertainty and input disturbances cause the true state trajectory to differ from the trajectory computed by the open-loop controller, and hence the impact of plant uncertainty and input disturbances can be quite noticeable in such systems. (This problem is illustrated later in the paper, in Section 5, Example 2.)

[6] Actually, even when the optimal control problem does not have a solution because the constraints cannot be satisfied, many optimal control algorithms (see, e.g., Ref. 21) will produce a control that tends to minimize constraint violation.

Our fast-sampling feedback controller is designed to reduce this problem by ensuring that the true plant state follows the required state trajectory more closely than in systems that use standard contractive moving horizon controllers.

We begin our derivation of the fast-sampling feedback control law by writing down explicit solutions

$$x((j+1)\delta) = e^{\delta A}x(j\delta) + \int_0^\delta e^{(\delta-s)A}b[u(j\delta) + d(j\delta+s)]ds \tag{18}$$

and

$$\tilde{x}((j+1)\delta) = e^{\delta\tilde{A}}\tilde{x}(j\delta) + \int_0^\delta e^{(\delta-s)\tilde{A}}b(u_s(j\delta) + \tilde{d})ds \tag{19}$$

of (8) and (9) over a single time interval $t \in (j, (j+1)\delta)$, where $\delta = \Delta/N$ is the fast-sampling period ($N > 1$ is an integer). Note that $u_s(t)$ and $u_c(t)$ (the two components of $u(t)$) are constant over the interval $t \in [j\delta, (j+1)\delta)$. Observe also that for small δ,

$$e^{\delta A} \simeq I + \delta A \tag{20}$$

and

$$\int_0^\delta e^{(\delta-s)A}ds \simeq \delta I. \tag{21}$$

Substituting these approximations into (18) and (19) and making the additional observation that for small s, $d(j\delta+s)$ is approximately equal to $d(j\delta)$ (since $d(\cdot)$ is Lipschitz-smooth), we can obtain the approximating systems, distinguished by the subscript a,

$$x_a((j+1)\delta) = (I + A\delta)x_a(j\delta) + b\delta(u_s(j\delta) + u_{a,c}(j\delta) + d(j\delta)), \tag{22}$$

and

$$\tilde{x}_a((j+1)\delta) = (I + \tilde{A}\delta)\tilde{x}_a(j\delta) + b\delta(u(j\delta) + \tilde{d}). \tag{23}$$

These are, in fact, the Euler approximations to the continuous-time systems (8) and (9), respectively, with $x_a(j\delta)$ approximating $x(j\delta)$ and $\tilde{x}_a(j\delta)$ approximating $\tilde{x}(j\delta)$. We also define the tracking error $e_a(j\delta)$ for the approximating systems by

$$e_a(j\delta) := x_a(j\delta) - \tilde{x}_a(j\delta). \tag{24}$$

In (22), (23), the control $u_s(j\delta)$ is the actual control determined by the slow sampling controller, while the control $u_{a,c}(j\delta)$ is only an approximation to the actual compensating control $u_c(j\delta)$.

We will proceed in two steps. First we will derive a stabilizing state feedback law for determining a control $u_{a,c}(j\delta)$ for the approximating error equation (24). Then we will make certain modifications to the expression for $u_{a,c}(\cdot)$ to obtain a feedback law $u_c(\cdot)$ for the actual error dynamics.

Thus, our first objective is to obtain a feedback law for determining the controls $u_{a,c}(j\delta),\ j = 0, 1, 2, \ldots$, such that the approximating error dynamics satisfy the equation

$$e_a((j+1)\delta) = (I + H\delta)e_a(j\delta), \tag{25}$$

where

$$H = \begin{bmatrix} 0 & 1 & 0 & \ldots & 0 \\ 0 & 0 & 1 & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \ldots & 1 \\ h_0 & h_1 & h_2 & \ldots & h_{p-1} \end{bmatrix} \tag{26}$$

is a matrix whose eigenvalues all have negative real components. This matrix can be thought of as describing the "desired error behavior"; the fact that its eigenvalues have negative real components reflects our desire for the tracking error to converge to zero as time goes by.

Now

$$\begin{aligned} e_a((j+1)\delta) &= (I + A\delta)x_a(j\delta) + b\delta(d(j\delta) - \tilde{d} + u_{a,c}(j\delta)) - (I + \tilde{A}\delta)\tilde{x}_a(j\delta) \\ &= e_a(j\delta) + \delta A x_a(j\delta) - \delta\tilde{A}x_a(j\delta) + \tilde{A}\delta e_a(j\delta) + b\delta(d(j\delta) - \tilde{d} + u_{a,c}(j\delta)) \\ &= (I + \tilde{A}\delta)e_a(j\delta) + \delta(A - \tilde{A})x_a(j\delta) + b\delta(d(j\delta) - \tilde{d} + u_{a,c}(j\delta)) \end{aligned} \tag{27}$$

Equating this with the right side of (25), we obtain

$$He_a(j\delta) = \tilde{A}e_a(j\delta) + (A - \tilde{A})x_a(j\delta) + b(d(j\delta) - \tilde{d} + u_{a,c}(j\delta)). \tag{28}$$

The first $p-1$ components of this equation are always satisfied, regardless of the choice of $u_{a,c}(j\delta)$. The pth component can be written as

$$h^T e_a(j\delta) = \tilde{a}^T e_a(j\delta) + (a - \tilde{a})^T x_a(j\delta) + (d(j\delta) - \tilde{d} + u_{a,c}(j\delta)), \tag{29}$$

where $a := (a_0, a_1, \ldots, a_{p-1})^T$, $\tilde{a} := (\tilde{a}_0, \tilde{a}_1, \ldots, \tilde{a}_{p-1})^T$ and $h := (h_0, h_1, \ldots, h_p)^T$. This leads to the *desired* control equation

$$u_{a,c}(j\delta) = h^T e_a(j\delta) - [\tilde{a}^T e_a(j\delta) + (a - \tilde{a})^T x_a(j\delta) + d(j\delta) - \tilde{d}]. \tag{30}$$

Unfortunately, the second term on the right side of (30) cannot be included in a valid control law, since neither the vector a nor the quantity $d(j\delta)$ is known exactly. It is, however, possible to come up with an estimate of this term, which relies only on continuity of $e(t)$, $x(t)$ and $d(t)$. Observe that the pth component of the vector equation (27) can be written in the form

$$(a - \tilde{a})^T x_a(j\delta) + \tilde{a}^T e_a(j\delta) + (d(j\delta) - \tilde{d}) = b^T[e_a((j+1)\delta) - e_a(j\delta)]/\delta - u_{a,c}(j\delta). \tag{31}$$

The term on the right side of (31) cannot be used in (30) to compute $u_{a,c}(j\delta)$, since it relies on knowledge of the tracking error $e_a((j+1)\delta)$ as well as knowledge of $u_{a,c}(j\delta)$ itself. Hence we construct the estimate

$$\begin{aligned}&(a-\tilde{a})^T x_a(j\delta)+\tilde{a}^T e_a(j\delta)+(d(j\delta)-\tilde{d})\\ &\quad\simeq\; b^T[e_a(j\delta)-e_a((j-1(\delta))]/\delta-u_{a,c}((j-1)\delta),\end{aligned} \tag{32}$$

which is obtained by subtracting one from the index j on the right side of (31). Since $e(t)$,$x(t)$ and $d(t)$ have continuous sample paths, it seems reasonable to expect that as δ becomes small, this estimate approaches the quantity it is estimating. Substituting (32) into the desired control equation (30), we have

$$u_{a,c}(j\delta)=u_{a,c}((j-1)\delta)-b^T[e_a(j\delta)-e_a((j-1)\delta)]/\delta+h^T(e_a(j\delta)), \tag{33}$$

for $j\geq 1$. Since $u^a_{c,-1}$ is not defined, we initialize the control sequence with $u^a_{c,0}=0$, and in order to translate (33) into a control law for the true system, we simply replace the states of the approximating systems with the corresponding states of the true continuous-time systems to obtain the control law

$$u_c(j\delta)=\begin{cases} u_c((j-1)\delta)-b^T[x(j\delta)-x((j-1)\delta)-\tilde{x}(j\delta)+\tilde{x}((j-1)\delta)]/\delta & \\ \quad +h^T(x(j\delta)-\tilde{x}(j\delta), & j>0,\\ 0, & j=0.\end{cases} \tag{34}$$

(Recall that $u(\cdot)$ is piecewise-constant, with $u(t)=u(j\delta)$ for $t\in[j\delta,(j+1)\delta)$, so (34) completely determines $u(\cdot)$.)

Quite obviously, the approach we used in constructing this feedback control law is derived from a standard technique for pole-placement via feedback, and hence on an intuitive level one expects it to "work" for sufficiently small values of δ. However, as we shall see later, because of the nature of the approximations we have used (in particular (32)), proving that it "works" takes quite a lot of effort.

3.3 Dual-Sampling-Rate Controller

Now suppose that we want to force $x(t)$ to track a trajectory $x^*_i(t)$ (generated by solving an open loop control problem with assumed plant model (9) for $t\in[i\Delta,(i+1)\Delta)$. In view of the discussion in the previous subsection, this can be achieved by choosing the compensating controller according to (34) with $\tilde{x}(j\delta)$ replaced by $x^*_i(j\delta)$, that is,

$$u_c(t)=u_c(j\delta) \text{ for } t\in[j\delta,(j+1)\delta), \quad iN\leq j<(i+1)N,$$

where

$$u_c(j\delta)=\begin{cases} 0, & j=iN,\\ u_c((j-1)\delta)-b^T[e(j\delta)-e((j-1)\delta)]/\delta+h^Te(j\delta), & iN<j<(i+1)N,\end{cases} \tag{35}$$

with

$$e(j\delta) = x(j\delta) - x_i^*(j\delta), \quad j = iN, iN+1, \ldots, (i+1)N-1.$$

Furthermore, if we assume that $u(t) = 0$ for $t \in [0, \Delta)$ and that for $t \geq \Delta$, the compensating controller is forcing tracking error $e(t)$ to satisfy approximately $\dot{e}(t) = He(t)$, a natural choice of predictor of $x((i+1)\Delta)$ at time

$$\tau_i := (i+1)\Delta - T_C, \tag{36}$$

is

$$\hat{x}((i+1)\Delta) = \begin{cases} e^{\tilde{A}T_C} x(\tau_0) + \int_{\tau_0}^{\Delta} e^{\tilde{A}(\Delta - s)} b\tilde{d} \cdot ds, & i = 0, \\ \hat{x}((i+1)\Delta) = e^{HT_C}[x(\tau_i) - x_i^*(\tau_i)] + x_i^*((i+1)\Delta), & i > 0. \end{cases} \tag{37}$$

With these preliminaries completed, we are now able to state our dual-sampling algorithm.

Algorithm 3.1.

Step 0 (Design parameters). Parameters $\alpha \in (0,1)$, $\beta \geq 1$, $C > 0$, open-loop-sampling interval Δ, open-loop control horizon $T_h \geq \Delta$, fast-sampling interval $\delta = \Delta/N$, with $N \in \mathbb{Z}$, computation time $T_C = N_C\delta < \Delta$, and a matrix H of the form (26) whose eigenvalues have strictly negative real parts.

Step 1: Initialization. Set $i = 0$ and $u_0^*(t) \equiv 0$.

Step 2: Application of Control. At time $i\Delta$, begin applying the control $u(t) = u_s(t) + u_c(t)$ to the system. Use $u_s(t) = u_i^*(t)$. If $i = 0$ set $u_c(t) = 0$, otherwise choose $u_c(t)$ according to equation (35) above.

Step 3: Prediction. At time τ_i (see (36)), measure the state $x(\tau_i)$ and compute the predicted state $\hat{x}((i+1)\Delta)$ using the formula (37).

Step 4: Computation of Slow Control. Compute the optimal control $u_{i+1}^*(t)$ and the corresponding optimal trajectory $x_{i+1}^*(t)$ for the problem $\mathbf{P}(\hat{x}((i+1)\Delta), (i+1)\Delta)$, defined in section 3.1.

Step 5. Wait until time $(i+1)\Delta$. Replace i by $i+1$ and go back to Step 2.

Steps 0 and 1 simply define and initialize parameters, and Step 2 states how the control is applied over the interval $[i\Delta, (i+1)\Delta)$. Steps 3 and 4 compute the open loop control for the interval $[(i+1)\Delta, (i+2)\Delta)$, and are carried out concurrently with the application of the control as prescribed by Step 2.

Remark 3.1. If we were to set $u_c(t) \equiv 0$ for every iteration and replace Step 3 by

Step 3*. At time $i\Delta$, measure the state $x(i\Delta)$ and compute the predicted state $\hat{x}((i+1)\Delta)$ using the formula

$$\hat{x}((i+1)\Delta) = e^{\tilde{A}\Delta}x(i\Delta) + \int_0^{\Delta} e^{\tilde{A}(\Delta-s)}b(u_i^*(s+i\Delta) + \tilde{d})ds, \qquad (38)$$

then Algorithm 3.1 would be functionally equivalent to a sampled-data version of Algorithm 2.1 of Ref. 5, applied to the true and assumed systems (8) and (9).

Qualitatively, the differences between Algorithm 3.1 above and Algorithm 2.1 in Ref. 5 are as follows. Firstly, and most importantly, in Algorithm 3.1 the control computed by solving the open-loop optimal problem is not applied directly to the system. Rather, it is combined with the control produced by the fast-sampling controller which forces the system to track the state trajectory $x^*(t) = x_i^*(t),\ t \in [i\Delta, (i+1)\Delta),\ i = 1, 2, \ldots$. Secondly, the state prediction carried out in Step 3 is carried out using the state measurement at time τ_i instead of at time $i\Delta$, as in Ref. 5. The time τ_i is the last possible moment at which we can guarantee that the open-loop controller will still be able to compute its control $u_{i+1}^*(t)$ in time for the next interval. By making the prediction using the latest possible state measurement, robustness of the controller to modeling errors and disturbances is improved.

4 Stability of the Dual-Sampling-Rate Moving Horizon Control System

In this section, we establish conditions for closed-loop stability of the system (8) under Control Algorithm 3.1. We give an overview of the analysis before resolving the details.

4.1 Overview

We begin by defining the tracking error $e(t)$ as the difference between the actual state-trajectory $x(t)$ and the target state-trajectory obtained by patching together the functions $x_i^*(\cdot)$ obtained in Step 4 of Algorithm 3.1. (This target trajectory is the state-trajectory which the open-loop component of the controller expects the system to follow in the absence of noise or modelling error.) The differential equation describing the evolution of the tracking error depends on $u_c(t)$ but not on $u_s(t)$. Substituting the control law (35) into this equation, we obtain a closed-loop equation for the tracking error on each time interval $[i\Delta, (i+1)\Delta)$, $i = 1, 2, \ldots$, which is of the form

$$\begin{aligned} \dot{e}(t) &= He(t) + A_1[e(t) - e(t - f_1(t))] + A_2[e(t - f_1(t)) - e(t - f_2(t))] \\ &\quad + b[r_i(t) - r_i(t - f_2(t))] \end{aligned} \tag{39}$$

for some matrices A_1 and A_2 and functions $f_1(\cdot)$, $f_2(\cdot)$, where

$$0 \le f_1(t) \le \delta, 0 \le f_2(t) - f_1(t) \le \delta, \tag{40}$$

and $r_i(\cdot)$ is Lipschitz continous on $[i\Delta, (i+1)\Delta)$. For $0 \le s < \Delta$, the solution of (39) can be written in the form

$$e(i\Delta + s) = e^{Hs}e(i\Delta) + O(i\Delta + s, \delta). \tag{41}$$

To show that the residual term $O(t, \delta)$ goes to zero as δ goes to zero, *uniformly* in $t \in [i\Delta, (i+1)\Delta)$, we show that $e(\cdot)$ satisfies a Lipschitz condition on the interval $[i\Delta, (i+1)\Delta)$, and that the Lipschitz constant depends only on $x_i^*(i\Delta)$ and $e(i\Delta)$. While showing that $e(\cdot)$ is Lipschitz-continuous on any particular interval $[j\delta, (j+1)\delta)$ is trivial, showing that it is Lipschitz on the entire interval $[i\Delta, (i+1)\Delta)$, and obtaining an expression for the Lipschitz constant, is considerably more difficult.

At this point we conclude that by choosing appropriate H and δ, we can make the tracking error arbitrarily small, on any particular interval $[i\Delta, (i+1)\Delta)$, i.e., we can make the true state trajectory arbitrarily close to the target trajectory $x_i^*(t)$ on such an interval.

The next step is to combine this result with the contraction constraints (16) and (17), to obtain a difference inequality for the norms of the vectors $\{(x_i^*(i\Delta), e(i\Delta))^T,\ i = 1, 2, \ldots$. Using this difference inequality, it is a relatively straightforward matter to establish closed-loop stability of the system, in the sense that the norm of the state (and consequently also the norm of the tracking error) is asymptotically bounded by a quantity which is $O(\delta)$. The main results are stated at the end of section 4.2 in Theorem 4.1 and Corollary 4.1.

For the sake of readability, some proofs are relegated to the Appendix.

4.2 Stability Analysis

In what follows, we adopt the convention that terms c_j and k_j, $j = 1, 2, \ldots$ will be used to denote positive constants. We also write $v_1 < v_2$ for vectors v_1 and v_2 of equal dimension, when each element of v_1 is less than the corresponding element of v_2.

First, as before, we define the tracking error by

$$e(t) = x(t) - x_i^*(t), \quad t \in [i\Delta, (i+1)\Delta), \; i = 1, 2, \ldots, \tag{42}$$

and observe that since $x_i^*(t)$ is a solution to (9) for $t \in [i\Delta, (i+1)\Delta)$, with $u_s(t) = u_i^*(t)$, we also have that

$$\dot{x}_i^*(t) = \tilde{A}x_i^*(t) + b[\tilde{d} + u_s(t)]. \tag{43}$$

Differentiating both sides of (42) and substituting (8) and (43), we get

$$\begin{aligned} \dot{e}(t) &= Ax(t) - \tilde{A}x_i^*(t) + b[u_c(t) + d(t) - \tilde{d}] \\ &= Ae(t) + (A - \tilde{A})x_i^*(t) + b[u_c(t) + d(t) - \tilde{d}]. \end{aligned} \tag{44}$$

Now for $t \in [j\delta, (j+1)\delta)$ and $j = iN+1, iN+2, \ldots, (i+1)N-1$, we can write (35) in the form

$$u_c(t) = h^T e(j\delta) - f(t), \tag{45}$$

where

$$f(t) = -u_c(t-\delta) + b^T(e(j\delta) - e((j-1)\delta)/\delta. \tag{46}$$

Substituting (45) into (44), we get

$$\begin{aligned} \dot{e}(t) &= Ae(t) + (A - \tilde{A})x_i^*(t) + bh^T e(j\delta) - bf(t) + b(d(t) - \tilde{d}) \\ &= (A - B)e(t) + Be(t) + bh^T e(t) - bh^T e(t) + bh^T e(j\delta) \\ &\quad +(A - \tilde{A})x_i^*(t) - bf(t) + b(d(t) - \tilde{d}) \\ &= He(t) + ba^T x(t) + bh^T(e(j\delta) - e(t)) - b\tilde{a}^T x_i^*(t) + b(d(t) - \tilde{d}) - bf(t) \end{aligned} \tag{47}$$

Next, using (44) and (46), it follows from the mean value theorem that, for $t \in [j\delta, (j+1)\delta)$ and some

$$s_j \in [(j-1)\delta, j\delta] \tag{48}$$

(using the fact that $u_c(\cdot)$ is piecewise-constant, so that $u_c(t-\delta) = u_c(s_j)$),

$$\begin{aligned} f(t) &= -u_c(t-\delta) + b^T \dot{e}(s_j) \\ &= -u_c(t-\delta) + b^T[Ae(s_j) + (A - \tilde{A})x_i^*(s_j) + b(u_c(s_j) + d(s_j) - \tilde{d})] \\ &= b^T Ae(s_j) + (a - \tilde{a})^T x_i^*(s_j) + d(s_j) - \tilde{d}. \end{aligned} \tag{49}$$

Substituting this back into (47) gives us, for $t \in [j\delta, (j+1)\delta)$ and $j = iN+1, iN+2, \ldots, (i+1)N-1$,

$$\begin{aligned} \dot{e}(t) &= He(t) + ba^T e(t) + bh^T(e(j\delta) - e(t)) + b(a - \tilde{a})^T x_i^*(t) \\ &\quad +b(d(t) - \tilde{d}) - bb^T Ae(s_j) - b(a - \tilde{a})x_i^*(s_j) - b(d(s_j) - \tilde{d}) \end{aligned}$$

$$
\begin{aligned}
&= He(t) + ba^T(e(t) - e(s_j)) + b(a - \tilde{a})^T(x_i^*(t) - x_i^*(s_j)) \\
&\quad + bh^T(e(j\delta) - e(t)) + b(d(t) - d(s_j)) \\
&= He(t) + b(a - h)^T(e(t) - e(j\delta)) + ba^T(e(j\delta) - e(s_j)) \\
&\quad + b(a - \tilde{a})^T(x_i^*(t) - x_i^*(s_j)) + b(d(t) - d(s_j)) \\
&= He(t) + b(a - h)^T(e(t) - e(j\delta)) + ba^T(e(j\delta) - e(s_j)) + b(r_i(t) - r_i(s_j)) \qquad (50)
\end{aligned}
$$

where

$$
r_i(t) = (a - \tilde{a})^T x_i^*(t) + d(t). \tag{51}
$$

The following two results follow from (50), and are proved in the Appendix. The first is that the tracking error $e(t)$ is locally Lipschitz continuous, with Lipschitz constants that can be determined for each interval $[i\Delta, (i+1)\Delta)$, $i = 1, 2, ...$

Lemma 4.1. There exist finite positive constants k_1, k_2, k_3, k_4 such that for each integer $i \geq 1$, the tracking error satisfies the Lipschitz condition

$$
\|e(t) - e(s)\| \leq \bar{\lambda}_i |t - s| \text{ for all } s, t \in [i\Delta, (i+1)\Delta), \tag{52}
$$

where

$$
\bar{\lambda}_i = (1 + \sqrt{2} k_3 e^{k_4 \Delta})(\lambda_i^* + \|e(i\Delta)\|) + k_2 \Delta k_3 e^{k_4 \Delta} \xi_i, \tag{53}
$$

with

$$
\lambda_i^* = k_1(\|e(i\Delta)\| + \|x_i^*(i\Delta)\| + 1), \tag{54}
$$

and

$$
\xi_i = \|a - \tilde{a}\|[\|\tilde{A}\|\beta\|x_i^*(i\Delta)\| + C + D] + \gamma, \tag{55}
$$

where C is as in (6), D is as in (3) and γ is as in (2).

The next result (proved in the Appendix) follows from Lemma 4.1. It shows that as δ becomes small, the error $e(t)$ becomes close to $e^{H(T-i\Delta)}e(i\Delta)$. That is, the compensating control $u_c(t)$ is indeed forcing the error dynamics to behave approximately like a solution of the differential equation $\dot{e}(t) = He(t)$.

Lemma 4.2. There exist finite positive constants k_5, k_6 and k_7 such that for each positive integer i,

$$
\|e(t) - e^{H(t-i\Delta)}e(i\Delta)\| \leq \delta\,(k_5\|e(i\Delta)\| + k_6\|x_i^*(i\Delta)\| + k_7) \tag{56}
$$

for $t \in [i\Delta + \delta, (i+1)\Delta)$.

We will also need the following lemma.

Lemma 4.3. There exist positive constants m and w such that

$$
\|e^{Ht}\| \leq m e^{-wt} \tag{57}
$$

for $t \geq 0$.

Proof. This is a standard result in linear systems theory for matrices whose eigenvalues lie strictly in the left half-plane. See, e.g., Ref. 22, pp. 185-186. □

We are now in a position to analyze the asymptotic behavior of the tracking error and to state our main results.

By definition,

$$\begin{aligned} e((i+1)\Delta) &= x((i+1)\Delta) - x_{i+1}^*((i+1)\Delta) \\ &= x((i+1)\Delta) - \hat{x}((i+1)\Delta). \end{aligned} \tag{58}$$

We also have, by continuity of $x(\cdot)$, that

$$x((i+1)\Delta) = \lim_{t \uparrow (i+1)\Delta} x(t) = x_i^*((i+1)\Delta) + \lim_{t \uparrow (i+1)\Delta} e(t). \tag{59}$$

Also, by construction (see (37)), we have

$$\hat{x}((i+1)\Delta) = x_i^*((i+1)\Delta) + e^{HT_C} e(\tau_i). \tag{60}$$

Subtracting (60) from (59), we get

$$\begin{aligned} e((i+1)\Delta) &= \lim_{t \uparrow (i+1)\Delta} e(t) - e^{HT_C} e(\tau_i) \\ &= \lim_{t \uparrow (i+1)\Delta} e(t) - e^{HT_C} \left[e(\tau_i) - e^{H(\Delta - T_C)} e(i\Delta) + e^{H(\Delta - T_C)} e(i\Delta) \right] \\ &= \lim_{t \uparrow (i+1)\Delta} e(t) - e^{H\Delta} e(i\Delta) - e^{HT_C} \left[e(\tau_i) - e^{H(\Delta - T_C)} e(i\Delta) \right]. \end{aligned} \tag{61}$$

Taking norms on both sides and making use of Lemma 4.2, we get

$$\begin{aligned} \|e((i+1)\Delta)\| &\leq \lim_{t \uparrow (i+1)\Delta} \|e(t) - e^{H\Delta} e(i\Delta)\| + \|e^{HT_C}\| \|e(\tau_i) - e^{H(\Delta - T_C)} e(i\Delta)\| \\ &\leq \delta(1 + \|e^{HT_C}\|)[k_5 \|e(i\Delta)\| + k_6 \|x_i^*(i\Delta)\| + k_7] \\ &\leq \delta(1 + m)[k_5 \|e(i\Delta)\| + k_6 \|x_i^*(i\Delta)\| + k_7], \end{aligned} \tag{62}$$

where m is the constant defined in Lemma 4.3. We can also obtain an expression for $x_{i+1}^*((i+1)\Delta)$ in terms of $x_i^*(i\Delta)$ and $e(i\Delta)$. Observe that

$$\begin{aligned} x_{i+1}^*((i+1)\Delta) &= \hat{x}((i+1)\Delta) = x_i^*((i+1)\Delta) + e^{HT_C} e(\tau_i) \\ &= x_i^*((i+1)\Delta) + e^{H\Delta} e(i\Delta) + e^{HT_C}[e(\tau_i) - e^{H(\Delta - T_C)} e(i\Delta)]. \end{aligned} \tag{63}$$

Taking norms on both sides, using the fact that, by construction,

$$\|x_i^*((i+1)\Delta)\| \leq \alpha \|x_i^*(i\Delta)\|,$$

and applying Lemma 4.2, we get

$$\begin{aligned} \|x_{i+1}^*((i+1)\Delta)\| &\leq \alpha \|x_i^*(i\Delta)\| + \|e^{H\Delta}\| \cdot \|e(i\Delta)\| \\ &\quad + \delta \|e^{HT_C}\| \left[k_5 \|e(i\Delta)\| + k_6 \|x_i^*(i\Delta)\| + k_7 \right] \\ &\leq \alpha \|x_i^*(i\Delta)\| + m \|e(i\Delta)\| \\ &\quad + \delta m \left[k_5 \|e(i\Delta)\| + k_6 \|x_i^*(i\Delta)\| + k_7 \right]. \end{aligned} \tag{64}$$

Defining $\eta_{1,i} = \|x_i^*(i\Delta)\|$ and $\eta_{2,i} = \|e(i\Delta)\|$, and combining the inequalities (62) and (64), we get, for $i = 1, 2, \ldots$,

$$\begin{bmatrix} \eta_{1,i+1} \\ \eta_{2,i+1} \end{bmatrix} \leq M_2(\delta) \begin{bmatrix} \eta_{1,i} \\ \eta_{2,i} \end{bmatrix} + \delta \begin{bmatrix} k_8 \\ k_8 \end{bmatrix}, \tag{65}$$

where

$$M_2(\delta) = \begin{bmatrix} \alpha + \delta k_8 & m + \delta k_8 \\ \delta k_8 & \delta k_8 \end{bmatrix} \tag{66}$$

and $k_8 = (1 + m)(k_5 + k_6 + k_7)$.

Note that the system of inequalities (65) relies implicitly on the assumption that a solution to the optimal control problem $\mathbf{P}(x_j^*(j\Delta), 0)$ exists for $j = 1, 2, \ldots, i$.

It is not difficult to show that $\eta_{1,1}$ and $\eta_{2,1}$ must satisfy

$$\eta_{1,1} + \eta_{2,1} \leq k_9\|x(0)\| + k_{10}D \tag{67}$$

for some positive constants k_9 and k_{10}. (This follows from equations (2),(8) and (37), and the fact that $u(t)$ and $u_c(t)$ are equal to zero for all $t \in [0, \Delta)$.)

We will need the following assumption.

Assumption 4.1. Let $\alpha \in (0, 1)$ and $\beta \in [1, \infty)$ be specified. We assume that there exists a positive constant ρ_0 such that for every $x \in \mathbb{R}^p$, with $\|x\| < \rho_0$, the optimal control problem $\mathbf{P}(x, 0)$ of Section 3.1 has a solution.

We now state our main result.

Theorem 4.1. Suppose that Assumption 4.1 is satisfied and that Algorithm 3.1 is applied to the system (8). Then there exist finite positive constants ρ_1, ρ_2, k_{11} and k_{12} for which the following property holds. Given any $\epsilon > 0$ such that $\alpha + \epsilon < 1$, there exists a "critical value" $\delta_0(\epsilon)$ such that if $\|x(0)\| \leq \rho_1$, $D \leq \rho_2$, and $\delta < \delta_0(\epsilon)$, then solutions exist for each control problem $\mathbf{P}(\hat{x}(i\Delta), i\Delta)$, $i = 1, 2, \ldots$ solved in Step 1 of the algorithm, and the closed-loop system is stable, in the sense that for $i = 1, 2, \ldots$,

$$\|x(t)\| \leq \bar{X}_i \tag{68}$$

and

$$|u_c(t)| \leq \bar{X}_i + \|d(t) - \tilde{d}\| \tag{69}$$

for all $t \in [i\Delta, (i + 1)\Delta)$, where

$$\bar{X}_i = k_{11}(\alpha + \epsilon)^{i-1}\rho_0 + \delta k_{12}. \tag{70}$$

Proof. First note that $M_2(\delta)$, in (66) can be factored as follows:

$$M_2(\delta) = P_2(J_2 + \delta K_2)P_2^{-1}, \tag{71}$$

where

$$P_2 = \begin{bmatrix} -m/\alpha & 1 \\ 1 & 0 \end{bmatrix}, \quad J_2 = \begin{bmatrix} 0 & 0 \\ 0 & \alpha \end{bmatrix} \quad \text{and} \quad K_2 = P_2^{-1} \begin{bmatrix} k_8 & k_8 \\ k_8 & k_8 \end{bmatrix} P_2. \tag{72}$$

Hence for non-negative integers j,

$$M_2(\delta)^j = P_2(J_2 + \delta K_2)^j P_2^{-1}. \tag{73}$$

Taking norms on both sides gives

$$\begin{aligned} \|M_2(\delta)^j\| &\leq \|P_2\| \|(J_2 + \delta K_2)^j\| \|P_2^{-1}\| \\ &\leq c_{10}(\|J_2\| + \delta \|K_2\|)^j \\ &\leq c_{10}(\alpha + \delta c_{11})^j, \end{aligned} \tag{74}$$

for appropriately defined positive constants c_{10}, c_{11}. Now let $\eta_{3,i} := \|(\eta_{1,i}, \eta_{2,i})\|$, so that

$$\|x_i^*(i\Delta)\| \leq \eta_{3,i}, \quad \|e(i\Delta)\| \leq \eta_{3,i} \tag{75}$$

and

$$\|x_i(i\Delta)\| \leq \|x_i^*(i\Delta)\| + \|e(i\Delta)\| \leq 2\eta_{3,i}. \tag{76}$$

Now from (67), we have

$$\eta_{3,1} \leq k_9 \|x(0)\| + k_{10} D. \tag{77}$$

Also, iterating (65), taking norms on both sides and using (74), we get, for $i \geq 1$,

$$\begin{aligned} \eta_{3,i+1} &\leq c_{10}(\alpha + \delta c_{11})^i \eta_{3,1} + \sum_{j=0}^{i-1} \delta c_{10} (\alpha + \delta c_{11})^j \cdot 2k_8 \\ &\leq c_{10}(\alpha + \delta c_{11})^i \eta_{3,1} + \delta c_{12}, \end{aligned} \tag{78}$$

with $c_{12} = 2\delta c_{10} k_8/(1 - \alpha - \delta c_{11})$, as long as $\delta c_{11} < 1 - \alpha$ and

$$\eta_{1,j} \leq \rho_0 \text{ for } j = 1, \ldots, i. \tag{79}$$

(Condition (79) ensures that whenever the optimal control problem needs to be solved, a solution exists).

Examining (78), it is clear that there exist $\delta_0' > 0$ and $\rho_0' \leq \rho_0$ such that if $\eta_{3,1} \leq \rho_0'$ and $\delta < \delta_0'$, then $\delta c_{11} < 1 - \alpha$ and

$$\eta_{3,2} \leq c_{10}(\alpha + \delta c_{11}) \eta_{3,1} + \delta c_{12} \leq \rho_0. \tag{80}$$

Then (78) also applies for $i = 2$, and we have

$$\eta_{3,3} \leq c_{10}(\alpha + \delta c_{11})^2 \eta_{3,1} + \delta c_{12} \leq \rho_0. \tag{81}$$

Hence (78) applies for $i = 3$.

This argument can be repeated inductively to show that if $\eta_{3,1} \leq \rho_0'$ and $\delta < \delta_0'$, then (78) applies for *all* positive integers i. If we impose the further restriction that $\delta < \delta_0(\epsilon) := \min(\delta_0', \epsilon/c_{11})$, then (78) becomes

$$\eta_{3,i+1} \leq c_{10}(\alpha + \epsilon)^i \eta_{3,1} + \delta c_{12}, \tag{82}$$

Furthermore, it follows from (77) that there must exist positive constants ρ_1 and ρ_2 such that if $\|x(0)\| \leq \rho_1$ and $D \leq \rho_2$, then we have $\eta_{3,1} \leq \rho_0'$.

Consequently, when $\|x(0)\| \leq \rho_1$, $D \leq \rho_2$ and $\delta \leq \delta_0$, the inequality (82) is valid for all positive integers i. It follows directly that under these conditions,

$$\eta_{3,i} \leq c_{10}(\alpha + \delta\epsilon)^{i-1}\rho_0 + \delta c_{12} \tag{83}$$

for all positive integers i. For $t \in [i\Delta, (i+1)\Delta)$, we have, by construction (see (15) and (17))

$$\|x_i^*(t)\| \leq \beta \|x_i^*(i\Delta)\| = \beta \eta_{1,i}, \tag{84}$$

and, by Lemmas 4.2 and 4.3

$$\begin{aligned} \|e(t)\| &\leq \delta[k_5\|e(i\Delta)\| + k_6\|x_i^*(i\Delta)\| + k_7] + \|e^{H(t-i\Delta)}\| \cdot \|e(i\Delta)\| \\ &\leq \delta[k_5\eta_{2,i} + k_6\eta_{1,i} + k_7] + m\eta_{2,i}. \end{aligned} \tag{85}$$

So for $t \in [i\Delta, (i+1)\Delta)$, we have

$$\begin{aligned} \|x_i^*(t)\| + \|e(t)\| &\leq (\beta + \delta k_6)\eta_{1,i} + (m + \delta k_5)\eta_{2,i} + \delta k_7 \\ &\leq [\beta + m + \delta(k_5 + k_6)]\eta_{3,i} + \delta k_7 \\ &\leq [\beta + m + \delta(k_5 + k_6)][c_{10}(\alpha + \epsilon)^{i-1}\rho_0 + \delta c_{12}] + \delta k_7 \\ &\leq c_{13}(\alpha + \epsilon)^{i-1}\rho_0 + \delta c_{14}. \end{aligned} \tag{86}$$

for appropriately defined positive constants c_{13} and c_{14}.

Also, from (45) and (49), for $t \in [j\delta, (j+1)\delta)$ and $iN < j < (i+1)N$,

$$\begin{aligned} u_c(t) &= h^T e(j\delta) - f(t) \\ &= h^T e(j\delta) - [b^T A e(s_j) + (a - \tilde{a})^T x_i^*(s_j) + d(s_j) - \tilde{d}] \\ &= h^T e(j\delta) - [b^T A e(s_j) + (a - \tilde{a})^T x_i^*(s_j) + d(s_j) - d(t) + d(t) - \tilde{d}]. \end{aligned} \tag{87}$$

Next, we use the fact that

$$\|e(j\delta)\| \leq \max_{t \in [i\Delta, (i+1)\Delta)} \|e(t)\| \quad \text{and} \quad \|e(s_j)\| \leq \max_{t \in [i\Delta, (i+1)\Delta)} \|e(t)\|$$

to obtain the bound

$$
\begin{aligned}
|u_c(t)| &\le \max_{t\in[i\Delta,(i+1)\Delta)} [(\|h\| + \|a\|)\cdot\|e(t)\| + \|a-\tilde{a}\|\cdot\|x_i^*(t)\|] + \delta\gamma + \|d(t)-\tilde{d}\| \\
&\le (\|h\| + \|a\| + \|a-\tilde{a}\|)[c_{13}(\alpha+\epsilon)^{i-1}\rho_0 + \delta c_{14}] + \delta\gamma + \|d(t)-\tilde{d}\| \\
&\le c_{15}(\alpha+\epsilon)^{i-1}\rho_0 + \delta c_{16} + \|d(t)-\tilde{d}\|. \qquad (88)
\end{aligned}
$$

Setting $k_{11} = \max(c_{13}, c_{15})$ and $k_{12} = \max(c_{14}, c_{16})$, the result of Theorem 4.1 follows directly from the inequalities (86) and (88) and the fact that for $t \in [i\Delta, (i+1)\Delta)$, $\|x(t)\| = \|x_i^*(t) + e(t)\| \le \|x_i^*(t)\| + \|e(t)\|$. □

Remark 4.1. Loosely speaking, Theorem 4.1 states that there exists a ball centered at the origin, such that if the initial state $x(0)$ is inside this ball and if the system does not drift too far from the origin in the initial time interval $[0, \Delta]$ (when both controllers are inactive), one can always find a fast-sampling interval δ for which the overall closed-loop system is stable in the sense that the norm of the state is bounded above by a quantity which converges exponentially (in time) to a quantity which is $O(\delta)$.

Remark 4.2. We can also see from the inequality (70) that when the conditions of Theorem 4.1 are satisfied, the magnitude of the compensating control $u_c(t)$ is bounded asymptotically by a quantity which consists of the magnitude of the difference between $d(t)$ and $\tilde{d}$, and another term which is $O(\delta)$.

Remark 4.3. The difference between the true and assumed plant models is important. Theorem 4.1 holds regardless of this difference, but as a general rule, the magnitude of the fast-sampling control $u_c(t)$ increases as $\|A - \tilde{A}\|$ and $|d(t) - \tilde{d}|$ increase. This means that in order to guarantee closed-loop stability, either the radius ρ_1 of the required initial neighborhood for $x(0)$ must be smaller or the allowed magnitude of $u_c(t)$ must be larger.

In view of Theorem 4.1, the above remarks, and the fact that $\bar{X}_i$ is uniformly bounded above by $k_{11}\rho_0 + \delta k_{12}$ (see (70)) we are able to state our main result in the following form.

Corollary 4.1. Suppose that Assumption 4.1 is satisfied and that Algorithm 3.1 is applied to the system (8) and that the total control $u_s(t) + u_c(t)$ must satisfy the bound

$$|u_s(t)| + |u_c(t)| \le S + C, \ \forall t \ge 0, \qquad (89)$$

where $C \ge D + |\tilde{d}| + \kappa$, for some $\kappa > 0$. Then there exist finite positive constants ρ_1, ρ_2, k_{11} and k_{12} for which the following property holds. Given any $\epsilon > 0$ such that $\alpha + \epsilon < 1$, there exists a "critical value" $\delta_0(\epsilon)$ such that if $\|x(0)\| \le \rho_1$, $D \le \rho_2$, and $\delta < \delta_0(\epsilon)$, then (89) holds and the closed-loop system is stable in the sense that for $i = 1, 2, \ldots$,

$$\|x(t)\| \le \bar{X}_i \quad \text{for } t \in [i\Delta, (i+1)\Delta), \qquad (90)$$

where $\bar{X}_i$ is as in (70).

Theorem 4.1 and Corollary 4.1 state asymptotic stability results for the closed-loop system which allow for the input constraint $|u_s(t) + u_c(t)| < S + C$. Essentially, the total allowable magnitude of the control is partitioned into a component S for $|u_s(\cdot)|$ and another component C for $|u_c(\cdot)|$. The results do not, however, guarantee satisfaction of strict state-constraints. The open-loop component of the controller, without the compensating controller, would satisfy the state constraints if the true and assumed models were the same. The difference between the true and assumed model, as well as the incorporation of $u_c(\cdot)$ into the control, cause system behaviour to be somewhat different from that which the open-loop controller expects. On the other hand, the role of the compensating controller is to force the actual state trajectory to track the trajectory expected by the open-loop controller. Thus it is reasonable to expect that in the majority of cases, the compensating controller will indeed ensure that the state-constraints are satisfied.

5 Simulations

We now illustrate the performance of Algorithm 3.1 by simulation. As a basis for comparison, we will first generate simulations of systems under a saturated infinite-horizon LQR controller with integral action, and under the sampled-data version of Algorithm 2.1 of Ref. 5 specified in Remark 3.1, which we will hereafter refer to simply as "Algorithm 2.1".

All simulations are carried out using programs written in C and C++. Continuous-time processes are approximated by discrete-time Euler approximations with step size of length 0.001.

In Ref. 23, the problem of controlling the azimuth and elevation of a large antenna structure is considered. The problem was motivated by the need to design quick and accurate tracking controllers for 22m, 26m and 64m radio-astronomy and telecommunication antennae across Australia.

One component of the antenna structure is a motor whose control action $u(t)$ affects the azimuth angle position $y(t)$ of the antenna structure. For relatively small control action, this relation can be approximated very well by the differential equation

$$y^{(3)}(t) + 2\zeta\omega\ddot{y}(t) + \omega^2\dot{y}(t) = u(t), \tag{91}$$

where ω represents a dominant structural resonance and ζ is connected with an associated damping factor. Note that this is effectively a second-order system with the control acting on the system through an integrator.

In practice the usual goal is to drive $y(t)$ to some specified target point, subject to input constraints, and possibly minimizing some cost function. This is essentially the same problem as driving $y(t)$ to zero, and can be solved simply by shifting the coordinate system before applying our algorithm. Therefore, without loss of generality, we restrict attention to the problem of driving $y(t)$ to zero.

As an added complication, in order to test our algorithm a little more, we will assume that there is a structural failure in the system at time $t = 21$. We will model this by allowing the parameters ζ and ω to change at time 21. It will also be realistic to allow the true system model to include an offset $d(t)$ in the control action. In practice, such an offset is often present due to various aspects of the motor electronics, and usually takes the form of a slowly time-varying signal.

Rewriting (91) in state-space form with $x(t) = (y(t), \dot{y}(t), \ddot{y}(t))^T$, using the values $(\zeta(t), \omega(t)) = (0.025, 2)$ for $t < 21$ and $(\zeta(t), \omega(t)) = (-0.0447, 2.2361)$ for $t \geq 21$, and incorporating the disturbance $\{d(t)\}$, our system is specified by

$$\dot{x}(t) = A(t)x(t) + b[d(t) + u(t)] \tag{92}$$

(c.f. (8)) with

$$A(t) = \begin{cases} \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -4 & -0.1 \end{bmatrix}, & 0 \leq t < 21, \\ \\ \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -5 & 0.2 \end{bmatrix}, & t \geq 21. \end{cases} \tag{93}$$

(Note that the dynamics after time $t = 21$ are unstable.)

For purposes of simulation and evaluation of our control algorithm, we will suppose that the system dynamics really are specified exactly by (92) and (93).

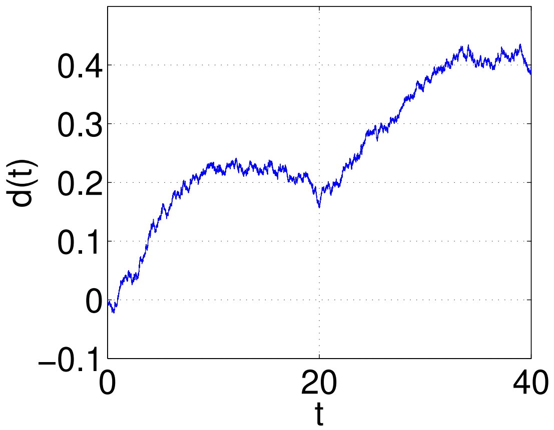

Figure 1: Disturbances $\{d(t),\ t \in [0, 40]\}$

We will use the disturbance process shown in Figure 1, which was generated as follows. The discrete-time skeleton $\{d(0.001j),\ j = 0, \ldots, 40000\}$ was obtained as a realization of a random walk with small positive drift, and the process $\{d(t), t \in [0, 40]\}$ was then obtained for points $t \neq 0.001j$ by linear interpolation. (It is not difficult to see that such a process is Lipschitz-smooth on any compact time-interval.)

We will also suppose that the controller does not have exact information about the model parameters. Specifically, we will take (9) to represent the controller's assumed model, with

$$\tilde{A} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -5 & -0.2 \end{bmatrix}$$

and $\tilde{d} = 0$. The assumed model will remain the same even after the structural failure at time $t = 21$.

The optimal control problem $\mathbf{P}(x(i\Delta), i\Delta)$ (see Section 3.1) will be to choose $u(t) \in U_\delta^6$ (see (6)) so as to minimize the quadratic cost

$$\int_{i\Delta}^{(i+3)\Delta} [x(t)^T Q x(t) + u(t)^2] dt \tag{94}$$

with

$$Q = \begin{bmatrix} 10 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

subject to the constraints (13) and (15), with $\alpha = 0.8$ and $\beta = 2$. We also choose $\Delta = 2.5$, $T_h = 7.5$ and $\delta = 0.001$.

We start with a simulation of the system under a simple saturated LQR controller with integral action, and then simulate the system when Algorithm 2.1 is used. Finally, we apply our algorithm to the system with the disturbances of Figure 1. The algorithm appears to compensate extremely well for both the unmodelled structural failure and the time-varying control offset $d(t)$, and also performs very well in terms of cost.

In all simulations, we use initial states $y(0) = 4$, $\dot{y}(0) = 2$, and $\ddot{y}(0) = 0$.

5.1 Example 1

For our first test, we construct a state-feedback controller which is updated at times $0, \Delta, 2\Delta, \ldots$ and is chosen as the solution to a discrete-time infinite-horizon LQR problem. In order to improve its ability to handle the disturbance $d(t)$, we augment the state-space with the integral of the position and incorporate this integral into the cost. To be more precise, we use the assumed model

$$\dot{\tilde{x}}(t) = A^* \tilde{x}(t) + b[u_s(t)],$$

where $x(t) \in \mathbb{R}^4$ and

$$A^* = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -5 & -0.2. \end{bmatrix}$$

Since the control will be changing only at times $0, 2.5, 5, \ldots$, we use the discrete-time representation

$$\tilde{x}((k+1)\Delta) = \exp(2.5A^*)\tilde{x}(k\delta) + b^* u_s(k\delta),$$

where $b^* = \int_0^{2.5} \exp(A^*(2.5 - s))bds$. For this system, we compute (by solving the appropriate discrete-time algebraic Riccati equation) an optimal state-feedback law which minimizes

$$\sum_{k=0}^{\infty} [\tilde{x}(k\Delta)^T Q^* \tilde{x}(k\Delta) + u_s(k\delta)^2],$$

where

$$Q^* = \begin{bmatrix} 1 & 0 \\ 0 & Q \end{bmatrix},$$

(This is a discrete-time infinite-horizon analog of the cost function (94), but it includes a penalty for the integral of position as well), obtaining the control law

$$u_s(k\Delta) = -(0.3357, 2.1446, 0.1778, 0.3431)\tilde{x}(k\Delta). \tag{95}$$

(We hold the control constant over the time interval $[k\Delta, (k+1)\Delta)$.) In order to satisfy the input constraint, we saturate $u_s(\cdot)$ at ± 6.

Applying our simple state-feedback law (95) to the system (92),(93), we obtain the state and control trajectories shown in Figure 2.

Although the state appears to be approaching the origin at the beginning, the feedback law is clearly not robust enough to deal with the unstable system dynamics after the structural failure.

5.2 Example 2

Now we use the sampled-data version of Algorithm 2.1 in Ref. 5, with parameters as described above, to control the antenna motor with the (unmodelled) disturbances $\{d(t)\}$ of Figure 1. The time T_C required for computation of the open-loop control is 1.0 (which, as required, is less than $\Delta = 2.5$).

The state and control trajectories obtained are shown in Figure 3. Performance looks tolerable, although the disturbance prevents the state from approaching the origin. The total cost in this case is

$$J = \int_0^{40} [x^T(t)Qx(t) + u(t)^2]dt \simeq 943.81$$

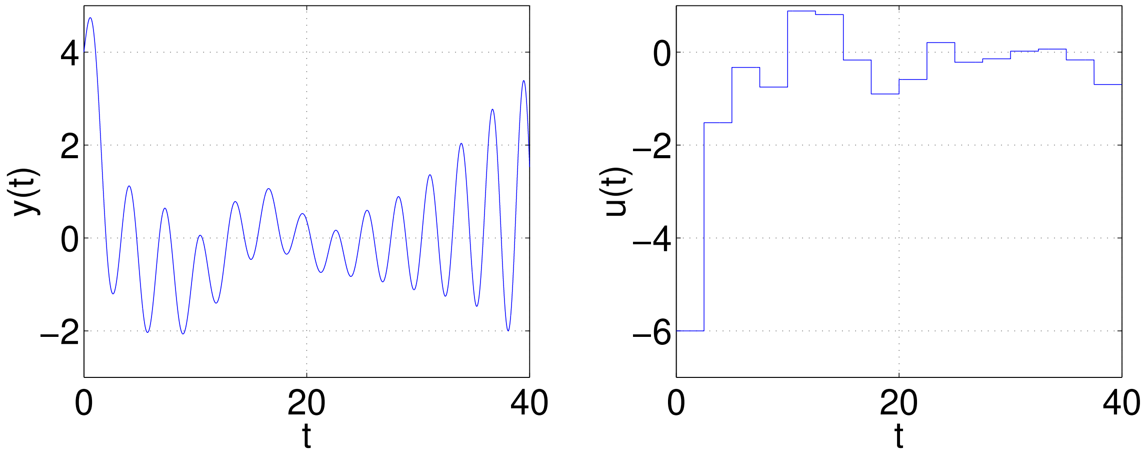

Figure 2: State ($y(t)$) and control ($u(t)$) trajectories for the saturated LQR controller of Example 1

5.3 Example 3

Finally, we examine the performance of Algorithm 3.1 when applied to the system. System parameters are the same as in the previous case. The disturbance $d(t)$ is again as shown in Figure 1.

Additional parameters required for the fast-sampling component of the controller are as follows. The third-order error reference dynamics are specified by the matrix

$$H = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -8 & -12 & -6 \end{bmatrix}. \tag{96}$$

The fast-sampling interval is $\delta = 0.001$.

As in Examples 1 and 2, we want to restrict the magnitude of the control to 6 - that is, we want $u_s(t) + u_c(t) \in U_\delta^6$. To do this, we saturate the total control $u(t) = u_s(t) + u_c(t)$ at ± 6.

The closed-loop state and control trajectories are shown in Figure 4. We can see that the disturbance is extremely well-compensated for by the fast-sampling component $u_c(t)$ of the control. In fact, as t becomes large, $u_c(t) \simeq -d(t)$ (except for $t \in [iN\delta, (iN+1)\delta)$, where $u_c(t) = 0$). System stability appears to have improved dramatically (relative to Example 2 above).

In addition, the total cost

$$\int_0^{40} [x^T(t)Qx(t) + (u_s(t) + u_c(t))^2]dt \tag{97}$$

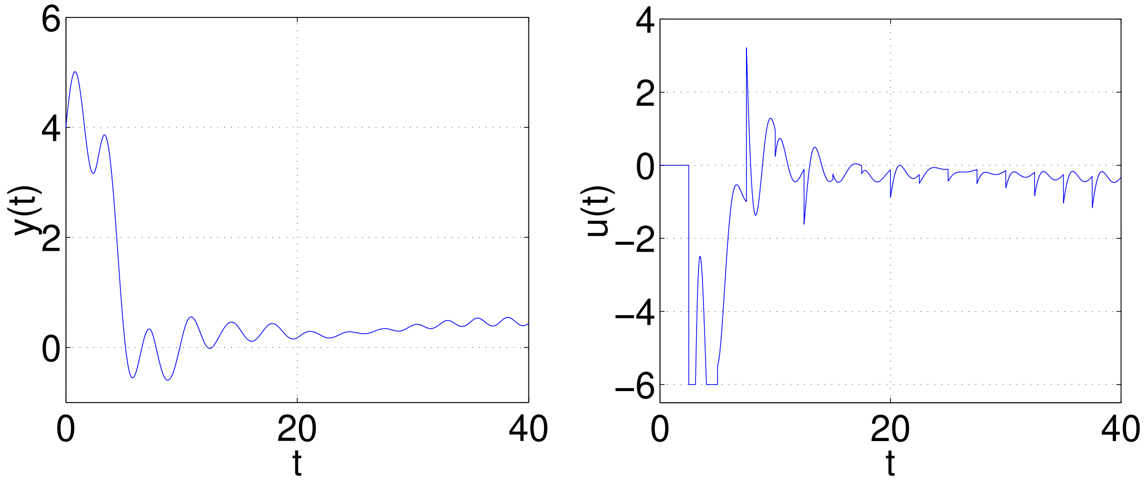

Figure 3: State ($y(t)$) and control ($u(t)$) trajectories for Example 2

associated with these trajectories is approximately 860.05.

6 Concluding Remarks

We have shown that it is possible to improve the performance of a moving horizon controller by "mixing" it with a simple state-feedback controller. The resulting hybrid controller is not difficult to implement, and exhibits many of the "nice" properties of both moving horizon control laws and simple state-feedback control laws.

As mentioned in the introduction, it has been shown in Ref. 16 that the fast-sampling component of the controller can be used to stabilize stochastic systems with nonlinear dynamics. It has also been shown in Ref. 17 that the fast-sampling controller is robust to unmodelled non-minimum phase effects on the dynamics induced by the presence of a term depending on the derivative of the control itself. In view of this, we feel that the results of this paper can be extended in a straightforward manner to the case where the system is nonlinear, non-minimum phase, and perturbed by random noise.

7 Appendix: Additional Proofs

In this section, we give proofs of Lemmas 4.1 and 4.2.

To obtain bounds on the tracking error, we will use a recursive method to show that

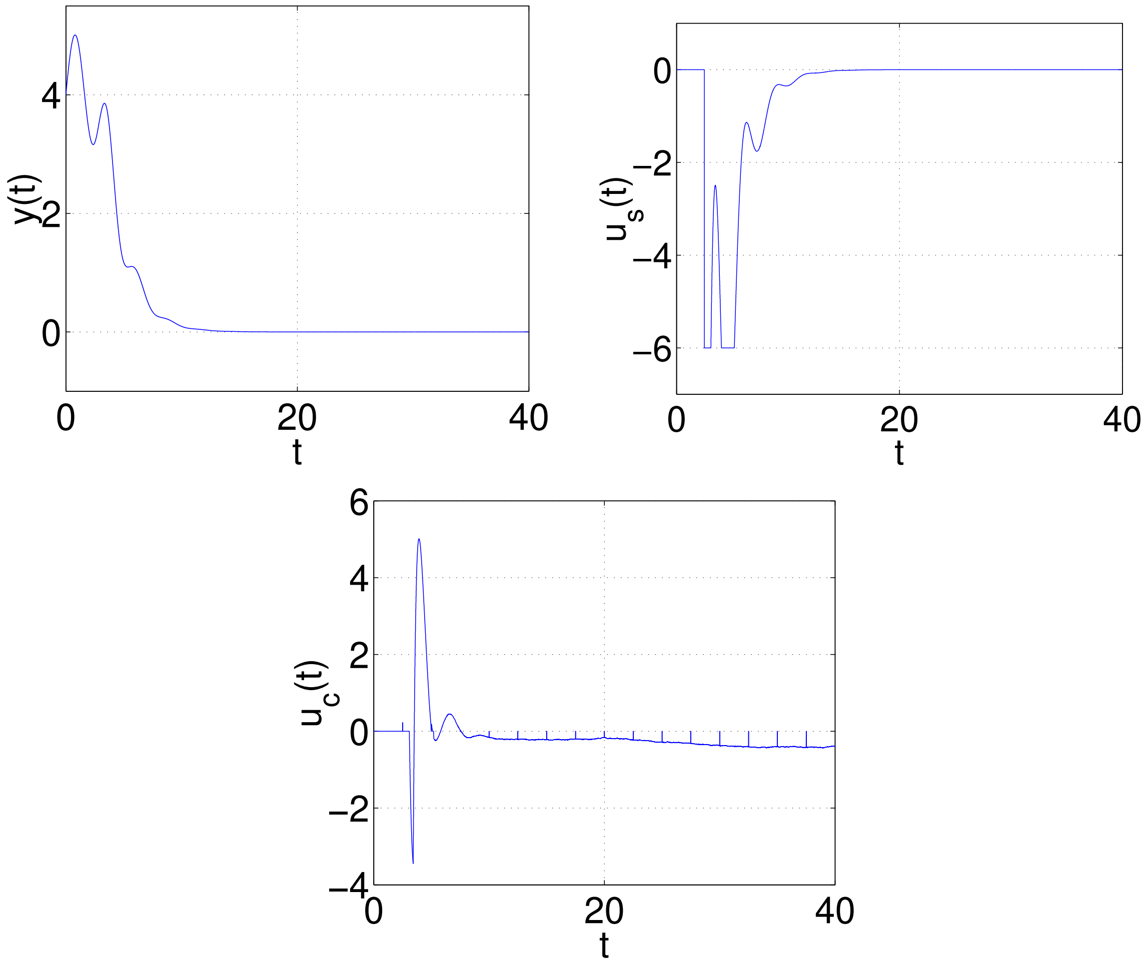

Figure 4: Trajectories of the state $y(t)$, control $u_s(t)$, and compensating control $u_c(t)$ for Example 3

it is Lipschitz continuous on each interval $[i\Delta,(i+1)\Delta)$ and obtain an expression for the Lipschitz constants. Our first task is to initialize the recursion process by obtaining an expression for a Lipschitz constant for the error for $t \in [i\Delta, i\Delta+\delta)$, with $i = 1, 2, 3, ...$, and then to use a recursion to compute a Lipschitz constant for the error for $t \in [j\delta,(j+1)\delta)$, with $j = iN+1, iN+2, \ldots, (i+1)N-1$.

We begin by rewriting (50) in a form that is valid over the entire time interval $[i\Delta,(i+1)\Delta)$, $i = 1, 2, \ldots$. We use notation defined in sections 2, 3.2, and 4.2. In particular, note that from section 4.2, s_j is defined as in (48) and $r_i(t)$ is defined by (51).

Defining

$$s_1(t) = j\delta,\ t \in [j\delta,(j+1)\delta),\ j = 0, 1, 2, \ldots \tag{98}$$

and

$$s_2(t) = s_j,\ \ t \in [j\delta, (j+1)\delta,\ \ j = iN+1, \dots, (i+1)N-1, \tag{99}$$

we can write (50) in the form

$$\dot{e}(t) = He(t) + b(a-h)^T[e(t) - e(s_1(t))] + ba^T[e(s_1(t)) - e(s_2(t))] + b[r_i(t) - r_i(s_2(t))], \tag{100}$$

which applies for $t \in [i\Delta + \delta, (i+1)\Delta)$. We will also make use of the equivalent form of (100) given by

$$\dot{e}(t) = Ae(t) + bh^T e(s_1(t)) - ba^T e(s_2(t)) + b[r_i(t) - r_i(s_2(t))]. \tag{101}$$

We now state a few preliminary results.

Lemma 7.1. Let A be a square matrix. Then for all $t \geq 0$

$$\|e^{At}\| \leq e^{\|A\|t} \tag{102}$$

and

$$\|e^{At} - I\| \leq e^{\|A\|t} - 1. \tag{103}$$

Proof. To show that (102) holds, observe that

$$\|e^{At}\| = \|\sum_{j=0}^{\infty}(At)^j/j!\| \leq \sum_{j=0}^{\infty}\|(At)^j\|/j! \leq \sum_{j=0}^{\infty}\|At\|^j/j! = e^{\|A\|t}. \tag{104}$$

For the second part, note that

$$\|e^{At} - I\| = \|\sum_{j=1}^{\infty}(At)^j/j!\| \leq \sum_{j=1}^{\infty}\|(At)^j\|/j! \leq \sum_{j=1}^{\infty}\|At\|^j/j! = e^{\|A\|t} - 1. \tag{105}$$

□

Lemma 7.2. Let i be some positive integer. Then for all s and t in the interval $[i\Delta, (i+1)\Delta)$, $r_i(\cdot)$ satisfies the Lipschitz condition

$$\|r_i(t) - r_i(s)\| \leq \xi_i|t-s|, \tag{106}$$

where

$$\xi_i = \|a - \tilde{a}\|[\|\tilde{A}\|\beta\|x_i^*(i\Delta)\| + S + D] + \gamma, \tag{107}$$

where S is as in (6), D is as in (3) and γ is as in (2).

Proof. We make use of the fact that a differentiable function whose derivative is bounded in norm must satisfy a Lipschitz condition. Taking norms on both sides of (43) we have, for $t \in [i\Delta, (i+1)\Delta)$,

$$\|\dot{x}_i^*(t)\| \leq \|\tilde{A}\|\|x_i^*(t)\| + S + D. \tag{108}$$

By construction, $\|x_i^*(t)\| \leq \beta \|x_i^*(i\Delta)\|$ for $t \in [i\Delta, (i+1)\Delta)$. Hence

$$\|\dot{x}_i^*(t)\| \leq \|\tilde{A}\|\beta\|x_i^*(i\Delta)\| + S + D. \tag{109}$$

It follows that $x_i^*(\cdot)$ must satisfy a Lipschitz condition with Lipschitz constant equal to the right-hand side of (109). From (51) we have

$$r_i(t) - r_i(s) = (a - \tilde{a})[x_i^*(t) - x_i^*(s)] + b[d(t) - d(s)]. \tag{110}$$

Taking norms on both sides and making use of the Lipschitz properties of $x_i^*(\cdot)$ and of $d(t)$ (see (2)) leads to the desired result. □

Now we are in a position to obtain the following two results.

Lemma 7.3. There exists a constant $k_1 \in (0, \infty)$ such that for each integer $i \geq 1$, the tracking error satisfies the Lipschitz condition

$$\|e(t) - e(s)\| \leq \lambda_i^* |t - s| \text{ for all } s, t \in [i\Delta, i\Delta + \delta], \tag{111}$$

where

$$\lambda_i^* = k_1(\|e(i\Delta)\| + \|x_i^*(i\Delta)\| + 1). \tag{112}$$

In what follows, we will make use of the fact that $\|e^{tA}\| \leq e^{c_1 t}$ for $t \geq 0$, where $c_1 = \|A\|$. (This is a direct consequence of Lemma 7.1.)

Proof. For $t \in [i\Delta, i\Delta + \delta)$, $u_c(t)$ is zero. Differentiating both sides of (42) and substituting (8) and (43) gives us

$$\dot{e}(t) = Ae(t) + (A - \tilde{A})x_i^*(t) + b[d(t) - \tilde{d}]. \tag{113}$$

The solution to (113) satisfies

$$e(t) = e^{A(t - i\Delta)}e(i\Delta) + \int_{i\Delta}^{t} e^{A(t-\tau)}[(A - \tilde{A})x_i^*(\tau) + b(d(\tau) - \tilde{d})]d\tau. \tag{114}$$

Taking norms on both sides and making use of Lemma 7.1 gives

$$\|e(t)\| \;\; \leq \;\; e^{c_1\delta}\|e(i\Delta)\| + (e^{c_1\delta} - 1)c_1^{-1}[\|A - \tilde{A}\|\beta\|x_i^*(i\Delta)\| + 2D]. \tag{115}$$

The solution to (113) also satisfies, for $i\Delta \leq s \leq t < i\Delta + \delta$,

$$e(t) - e(s) = (e^{A(t-s)} - I)e(s) + \int_{s}^{t} e^{A(t-\tau)}[(A - \tilde{A})x_i^*(\tau) + b(d(\tau) - \tilde{d})]d\tau. \tag{116}$$

Taking norms on both sides, we have

$$\begin{aligned}
&\|e(t) - e(s)\| \\
&\leq \;\; \|e^{A(t-s)} - I\| \cdot \|e(s)\| + \int_{s}^{t} \|e^{A(t-u)}\| \left[\|A - \tilde{A}\|\beta\|x_i^*(i\Delta)\| + 2D\right] du \\
&\leq \;\; (e^{\alpha(t-s)} - 1)\|e(s)\| + (e^{c_1(t-s)} - 1)c_1^{-1}(2D + \|A - \tilde{A}\|\beta\|x_i^*(i\Delta)\|).
\end{aligned} \tag{117}$$

Now since $|t-s| \leq \delta$, we have

$$e^{c_1(t-s)} - 1 \leq c_2(t-s) \tag{118}$$

for some $c_2 > c_1$. Consequently,

$$\begin{aligned} \|e(t)-e(s)\| &\leq c_2\|e(s)\||t-s| + (c_2/c_1)|t-s|(2D + \|A-\tilde{A}\|\beta\|x_i^*(i\Delta)\|) \\ &\leq c_2 \max_{s\in[i\Delta, i\Delta+\delta)} \|e(s)\| \cdot |t-s| \\ &\quad +(c_2/c_1)|t-s|(2D + \|A-\tilde{A}\|\beta\|x_i^*(i\Delta)\|). \end{aligned} \tag{119}$$

Finally, substituting (115) into (119) gives us, for appropriately defined k_1, the Lipschitz bound (111). ◻

Lemma 7.4. There exists a constant $k_2 \in (0, \infty)$ such that if for some integer $i \geq 1$ and integer $j \in \{iN, iN+1, \ldots, (i+1)N-2\}$, the tracking error $e(\cdot)$ satisfies the Lipschitz condition

$$\|e(t)-e(s)\| \leq \lambda_j|t-s| \text{ for all } s,t \in [j\delta, (j+1)\delta), \tag{120}$$

then the tracking error also satisfies

$$\|e(t)-e(s)\| \leq \lambda_{j+1}|t-s| \text{ for all } s,t \in [(j+1)\delta, (j+2)\delta), \tag{121}$$

where

$$\lambda_{j+1} = k_2(\|e(j\delta)\| + \delta\lambda_j + \delta\xi_i), \tag{122}$$

with ξ_i defined as in (107).

Proof. This proof is very similar to the proof of Lemma 7.3. For $t \in [(j+1)\delta, (j+2)\delta)$, the solution to (101) satisfies

$$e(t) = e^{A(t-(j+1)\delta)}e((j+1)\delta) + \int_{(j+1)\delta}^{t} e^{A(t-\tau)}b[h^T e((j+1)\delta) - a^T e(s_{j+1}) + r_i(\tau) - r_i(s_{j+1})]d\tau. \tag{123}$$

From the assumption (120), we have

$$\|e((j+1)\delta)\| \leq \|e(j\delta)\| + \delta\lambda_j \quad \text{and} \quad \|e(s_{j+1})\| \leq \|e(j\delta)\| + \delta\lambda_j. \tag{124}$$

Taking norms on both sides of (123) and making use of (124), we get, for $t \in [(j+1)\delta, (j+2)\delta)$,

$$\begin{aligned} \|e(t)\| &\leq e^{c_1\delta}(\|e(j\delta)\| + \delta\lambda_j) + \int_{(j+1)\delta}^{t} e^{c_1(t-u)}[(\|h\|+\|a\|)(\|e(j\delta)\|+\delta\lambda_j) + 2\delta\xi_i]du \\ &\leq e^{c_1\delta}(\|e(j\delta)\| + \delta\lambda_j) + (e^{c_1\delta}-1)c_1^{-1}[(\|h\|+\|a\|)(\|e(j\delta)\|+\delta\lambda_j) + 2\delta\xi_i] \\ &\leq (1+c_3\delta)\|e(j\delta)\| + c_4\delta\lambda_j + c_5\delta\xi_i \end{aligned} \tag{125}$$

for appropriately defined constants c_3,c_4 and c_5. Using a similar argument, it can be shown that the solution to (101) satisfies, for $s,t \in [(j+1)\delta,(j+2)\delta)$,

$$\|e(t)-e(s)\| \leq (e^{c_1(t-s)}-1)\|e(s)\|+(e^{c_1(t-s)}-1)c_1^{-1}\left[(\|h\|+\|a\|)(\|e(j\delta)\|+\delta\lambda_j)+2\delta\xi_i\right]. \tag{126}$$

Let c_2 be as defined in (118). Then we have

$$\|e(t)-e(s)\| \leq c_2|t-s|\cdot\|e(s)\|+(c_2/c_1)|t-s|\left[(\|h\|+\|a\|)(\|e(j\delta)\|+\delta\lambda_j)+2\delta\xi_i\right] \tag{127}$$

Substituting (125) into this inequality leads to the inequality (121). This completes the proof. □

We can now use an inductive argument based on the results of Lemmas 7.3 and 7.4 to obtain the system of inequalities

$$\begin{bmatrix} \lambda_{j+1} \\ \|e((j+1)\delta)\| \end{bmatrix} \leq M_1(\delta)\begin{bmatrix} \lambda_j \\ \|e(j\delta)\| \end{bmatrix}+\delta k_2\begin{bmatrix} \xi_i \\ 0 \end{bmatrix} \tag{128}$$

for $j=iN,iN+1,\ldots,(i+1)N-2$, where

$$M_1(\delta)=\begin{bmatrix} \delta k_2 & k_2 \\ \delta & 1 \end{bmatrix}, \tag{129}$$

with k_2 as in Lemma 4.5, and ξ_i was defined in (107). As in Section 4, the inequality (128) is to be interpreted as meaning that each element in the vector on the left is less than or equal to the corresponding element in the vector on the right.

In order to analyze this system of inequalities further, we will require the following result.

Lemma 7.5. There exist finite positive constants k_3 and k_4 such that for all non-negative integers j,

$$\|M_1(\delta)^j\| \leq k_3(1+\delta k_4)^j. \tag{130}$$

Proof. The matrix $M_1(\delta)$ can be factorized as follows

$$M_1(\delta)=P_1(J_1+\delta K_1)P_1^{-1}, \tag{131}$$

where

$$P_1=\begin{bmatrix} 1 & \kappa \\ 0 & 1 \end{bmatrix}, J_1=\begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \text{ and } K_1=\begin{bmatrix} 0 & 0 \\ 1 & \kappa \end{bmatrix}. \tag{132}$$

Raising both sides of (131) to the power j, taking norms, and making use of the triangle inequality, we get

$$\begin{aligned} \|M_1(\delta)^j\| &\leq \|P_1\|\cdot\|J_1+\delta K_1\|^j\cdot\|P_1^{-1}\| \\ &\leq \|P_1\|\cdot(\|J_1\|+\delta\|K_1\|)^j\cdot\|P_1^{-1}\| \\ &\leq \|P_1\|\cdot(1+\delta\|K_1\|)^j\cdot\|P_1^{-1}\|. \end{aligned} \tag{133}$$

Choosing $k_3 = \|P_1\| \cdot \|P_1^{-1}\|$ and $k_4 = \|K_1\|$ gives the desired result. $\square$

Next we will prove Lemma 4.1, which we restate in abbreviated form for the sake of convenience, as follows:

Lemma 4.1. For each integer $i \geq 1$, the tracking error satisfies the Lipschitz condition

$$\|e(t) - e(s)\| \leq \bar{\lambda}_i |t - s| \quad \text{for all} \quad s, t \in [i\Delta, (i+1)\Delta), \tag{134}$$

where

$$\bar{\lambda}_i = (1 + \sqrt{2} k_3 e^{k_4 \Delta})(\lambda_i^* + \|e(i\Delta)\|) + k_2 \Delta k_3 e^{k_4 \Delta} \xi_i, \tag{135}$$

with k_2 as in Lemma 7.4, k_3, k_4 as in Lemma 7.5, λ_i^* as in (111), and ξ_i as in (107).

Proof. Since the elements in the vectors and matrix in (128) are all non-negative, an inductive argument shows that for $k = iN+1, \ldots, (i+1)N-1$,

$$\begin{bmatrix} \lambda_k \\ \|e(k\delta)\| \end{bmatrix} \leq M_1(\delta)^{k-iN} \begin{bmatrix} \lambda_i^* \\ \|e(i\Delta)\| \end{bmatrix} + \delta k_2 \sum_{j=0}^{k-iN-1} M_1(\delta)^j \xi_i. \tag{136}$$

Since $\lambda_k \leq \left\| \begin{bmatrix} \lambda_k \\ \|e(k\delta)\| \end{bmatrix} \right\|$, it follows that

$$\lambda_k \leq \left\| M_1(\delta)^{k-iN} \right\| \left\| \begin{bmatrix} \lambda_i^* \\ \|e(i\Delta)\| \end{bmatrix} \right\| + \delta k_2 \xi_i \sum_{j=0}^{k-iN-1} \|M_1(\delta)^j\|. \tag{137}$$

Making use of Lemma 7.5, we get

$$\begin{aligned}
\lambda_k &\leq k_3(1+\delta k_4)^{k-iN} \left\| \begin{bmatrix} \lambda_i^* \\ \|e(i\Delta)\| \end{bmatrix} \right\| + \delta k_2 \xi_i (k - iN) k_3 (1+\delta k_4)^{k-iN} \\
&\leq k_3(1+\delta k_4)^{\Delta/\delta} \left\| \begin{bmatrix} \lambda_i^* \\ \|e(i\Delta)\| \end{bmatrix} \right\| + \delta k_2 \xi_i (\Delta/\delta) k_3 (1+\delta k_4)^{\Delta/\delta} \\
&\leq \sqrt{2} k_3 e^{k_4 \Delta} (\lambda_i^* + \|e(i\Delta)\|) + k_2 \xi_i \Delta k_3 e^{k_4 \Delta} \\
&< \bar{\lambda}_i.
\end{aligned} \tag{138}$$

Recall that for $s, t \in [k\delta, (k+1)\delta]$, $e(t)$ satisfies the Lipschitz condition $\|e(t) - e(s)\| \leq \lambda_k |t - s|$. Since $\lambda_k \leq \bar{\lambda}_i$ for $k = iN, \ldots, (i+1)N-1$, $e(t)$ also satisfies the Lipschitz condition $\|e(t) - e(s)\| \leq \bar{\lambda}_i |t - s|$ for $s, t \in [k\delta, (k+1)\delta)$ and $k = iN, \ldots, (i+1)N-1$. Hence it must satisfy (134). This completes the proof. $\square$

Note that the result of Lemma 4.1 holds regardless of the choice of δ. This property is useful in the proof of Lemma 4.2, which we restate below for the sake of convenience.

Lemma 4.2. There exist finite positive constants k_5, k_6 and k_7 such that for each positive integer i,

$$\|e(t) - e^{H(t-i\Delta)} e(i\Delta)\| \leq \delta \left(k_5 \|e(i\Delta)\| + k_6 \|x_i^*(i\Delta)\| + k_7 \right) \tag{139}$$

for $t \in [i\Delta + \delta, (i+1)\Delta)$.

Proof. For $t \in [i\Delta, i\Delta + \delta)$, $u_c(t)$ is zero. Differentiating both sides of (42), substituting (8) and (43) and solving the resulting differential equation gives us

$$e(i\Delta+\delta) = e^{A\delta}e(i\Delta)+\int_0^\delta e^{A(\delta-\tau)}\left[(A-\tilde{A})x_i^*(i\Delta+\tau)+b(d(i\Delta+\tau)-\tilde{d})\right]d\tau. \quad (140)$$

Then using (100), for $t \in [i\Delta + \delta, (i+1)\Delta)$, we obtain

$$\begin{aligned} e(t) &= e^{H(t-i\Delta-\delta)}e(i\Delta+\delta)+\int_{i\Delta+\delta}^t e^{H(t-\tau)}b\{(a-h)^T[e(\tau)-e(s_1(\tau))] \\ &\quad +a^T[e(s_1(\tau))-e(s_2(\tau))]+r_i(\tau)-r_i(s_2(\tau))\}d\tau \\ &= e^{H(t-i\Delta-\delta)}e^{A\delta}e(i\Delta)+\zeta_i(t) \\ &= e^{H(t-i\Delta)}e^{(A-H)\delta}e(i\Delta)+\zeta_i(t), \end{aligned} \quad (141)$$

where

$$\begin{aligned} \zeta_i(t) &= e^{H(t-i\Delta-\delta)}\int_0^\delta e^{A(\delta-\tau)}[(A-\tilde{A})x_i^*(i\Delta+\tau)+b(d(i\Delta+\tau)-\tilde{d})]d\tau \\ &\quad +\int_{i\Delta+\delta}^t e^{H(t-\tau)}b\{(a-h)^T[e(\tau)-e(s_1(\tau))]+a^T[e(s_1(\tau))-e(s_2(\tau))] \\ &\quad +r_i(\tau)-r_i(s_2(\tau))\}d\tau. \end{aligned} \quad (142)$$

From Lemma 4.3, $\max_{0\le s\le\Delta}\|e^{Hs}\| \le m$. Taking norms on both sides of (142), we get

$$\begin{aligned} \|\zeta_i(t)\| &\le m\int_0^\delta e^{(\delta-\tau)\|A\|}\left[\|A-\tilde{A}\|\cdot\|x_i^*(i\Delta+\tau)\|+2D\right]d\tau \\ &\quad +\int_{i\Delta+\delta}^t \|e^{H(t-\tau)}\|\{\|a-h\|\cdot\|e(\tau)-e(s_1(\tau))\|+\|a\|\cdot\|e(s_1(\tau))-e(s_2(\tau))\| \\ &\quad +\|r_i(\tau)-r_i(s_2(\tau))\|\}. \end{aligned} \quad (143)$$

Since $|\tau - s_1(\tau)| < \delta$, $|s_1(\tau) - s_2(\tau)| < \delta$ and $|\tau - s_2(\tau)| < \delta$, it then follows from Lemmas 4.1, 4.3, 7.1 and 7.2 that

$$\begin{aligned} \|\zeta_i(t)\| &\le m\left(\frac{e^{\delta c_1}-1}{c_1}\right)[\|A-\tilde{A}\|\cdot\max_{\tau\in[0,\delta)}\|x_i^*(i\Delta+\tau)\|+2D] \\ &\quad +\delta c_6(\bar{\lambda}_i+\xi_i)\int_{i\Delta+\delta}^t\|e^{H(t-\tau)}\|d\tau \\ &\le m\cdot\frac{\delta c_2}{c_1}\left[\|A-\tilde{A}\|\beta\|x_i^*(i\Delta)\|+2D\right]+\delta c_6(\bar{\lambda}_i+\xi_i)\int_{i\Delta+\delta}^t\|e^{H(t-\tau)}\|d\tau, \end{aligned} \quad (144)$$

where $c_6 = \|a - h\| + \|a\| + 1$. It follows from (57) that

$$\int_{i\Delta+\delta}^{t} \|e^{H(t-u)}\| du \leq \int_0^\infty m e^{-wt} < \infty. \tag{145}$$

Substituting this inequality into (144), we get

$$\|\zeta_i(t)\| \leq \delta c_7 \|x_i^*(i\Delta)\| + \delta c_8 + \delta c_9 (\bar{\lambda}_i + \xi_i) \tag{146}$$

for appropriately defined positive constants c_7, c_8 and c_9 and ξ_i as in (107). Now (141) can be rewritten in the form

$$e(t) - e^{H(t-i\Delta)} e(i\Delta) = e^{H(t-i\Delta)} (e^{(A-H)\delta} - I) e(i\Delta) + \zeta_i(t). \tag{147}$$

Taking norms, substituting the inequality (146), applying Lemma 7.1, and defining the constants k_5, k_6 appropropriately, we get the desired result. □

References

1. MAYNE, D. Q., RAWLINGS, J. B., RAO, C. V., and SCOKAERT, P. O. M. *Constrained Model Predictive Control: Stability and Optimality.* Automatica, Vol. 36, pp. 789–814, 2000.

2. MAYNE, D. and MICHALSKA, H. *Receding Horizon Control of Nonlinear Systems.* IEEE Transactions on Automatic Control, Vol. 35, pp. 814–824, 1990.

3. KEERTHI, S. S. and GILBERT, E. G. *Optimal, Infinite Horizon Feedback Laws for a General Class of of Constrained Discrete-Time Sysems: Stability and Moving Horizon Approximations.* Journal of Optimization Theory and Applications, Vol. 57, pp. 265–293, 1988.

4. MICHALSKA, H. and MAYNE, D. Q. *Robust Receding Horizon Control of Constrained Nonlinear Systems.* IEEE Transactions on Automatic Control, Vol. 38, pp. 995–1006, 1993.

5. POLAK, E. and YANG, T. *Moving Horizon Control of Linear Systems with Input Saturation and Plant Uncertainty, Part 1: Robustness.* International Journal of Control, Vol. 58, pp. 613–638, 1993.

6. POLAK, E. and YANG, T. *Moving Horizon Control of Linear Systems with Input Saturation and Plant Uncertainty, Part 2: Disturbance Rejection and Tracking.* International Journal of Control, Vol. 58, pp. 639–663, 1993.

7. YANG, T. and POLAK, E. *Moving Horizon Control of Nonlinear Systems with Input Saturation, Disturbances and Plant Uncertainty.* International Journal of Control, Vol. 58, pp. 875–903, 1993.

8. POLAK, E. and MAYNE, D. Q. *Design of Nonlinear Feedback Controllers.* IEEE Transactions on Automatic Control, Vol. 26, pp. 730–733, 1981.

9. KOTHARE, S. L. D. O. and MORARI, M. *Contractive model predictive control for constrained nonlinear systems.* IEEE Transactions on Automatic Control, Vol. 45, pp. 1053–1071, 2000.

10. CHEN, H., SCHERER, C., and ALLGOWER, F. *A Game Theoretic Approach to Nonlinear Robust Receding Horizon Control of Constrained Systems.* In Proceedings of the American Control Conference, Albuquerque, New Mexico, pp. 3073–3077, 1997.

11. MAYNE, D. Q. *Nonlinear Model Predictive Control: An Assessment.* In Proceedings of the 5th International Conference on Chemical Process Control, edited by Kantor, J. C., Garcia, C. E., and Carnahan, B., pp. 217–231, 1997.

12. SCOKAERT, P. O. M. and MAYNE, D. Q. *Min-max Feedback Model Predictive Control for Constrained Nonlinear Systems.* IEEE Transactions on Automatic Control, Vol. 43, pp. 1136–1169, 1998.

13. KOTHARE, M. V., BALAKRISHNAN, V., and MORARI, M. *Robust Constrained Model Predictive Control using Linear Matrix Inequalities.* Automatica, Vol. 32, pp. 1361–1379, 1996.

14. MAREELS, I., PENFOLD, H., and EVANS, R. *Controlling Nonlinear Time-Varying Systems Via Euler Approximations.* Automatica, Vol. 28, pp. 681–696, 1992.

15. BROCKWELL, A., BOROVKOV, K., and EVANS, R. *Regulation of Partially Known Nonlinear Stochastic Systems Using Fast Identification.* In Proceedings of the 36th IEEE Conference on Decision and Control, pp. 3964–3969, 1997.

16. BROCKWELL, A., BOROVKOV, K., and EVANS, R. *Stability of an Adaptive Regulator for Partially Known Nonlinear Stochastic Systems.* SIAM Journal on Control and Optimization, Vol. 37, pp. 1553–1567, 1999.

17. BROCKWELL, A., EVANS, R., and MAREELS, I. *Stability of a Trajectory-Based Control Law for an Unknown Nonlinear Non-Minimum Phase System.* Asian Journal of Control, Vol. 3, pp. 45–49, 2001.

18. ROSSITER, J., KOUVARITAKIS, B., and RICE, M. *A Numerically Robust State-Space Approach to Stable-Predictive Control Strategies.* Automatica, Vol. 34, pp. 65–73, 1998.

19. LEE, Y. and KOUVARITAKIS, B. *Robust Receding Horizon Control.* Automatica, Vol. 36, pp. 1497–1504, 2000.

20. MAYNE, D. and LANGSON, W. *Robustifying Model Predictive Control of Constrained Linear Systems.* Electronics Letters, Vol. 37, pp. 1422–1423, 2001.

21. POLAK, E. *Optimization: Algorithms and Consistent Approximations.* Springer-Verlag, 1997.

22. CALLIER, F. and DESOER, C. *Linear System Theory.* Springer, 1991.

23. EVANS, R., MAREELS, I., SCIACCA, L., COOPER, D., MIDDLETON, R., BETZ, R., and KENNEDY, R. *Adaptive Servo Control of Large Antenna Structure.* In Essays on Adaptive Control. Springer-Verlag, 2000.