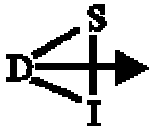
Dipartimento di Sistemi e Informatica



Dottorato di Ricerca in Ingegneria Informatica e dell'Automazione - X Ciclo

Reference Governors: On-Line Set-Point Optimization Techniques for Constraint Fulfillment

by

Alberto Bemporad

A dissertation submitted in partial fulfillment
of the requirements for the degree of
"Dottore di Ricerca"
(Doctor of Philosophy)
in "Ingegneria Informatica e dell'Automazione"
(Control and Computer Engineering)
Università di Firenze, Italy
October 1997

Ph.D. Program Coordinator: Prof. Edoardo Mosca

Advisor: Prof. Edoardo Mosca

Abstract

This dissertation presents a control technique to cope with set-point tracking problems in the presence of input and/or state constraints. The main idea consists of feeding to a conventional controller artificial set-points, which are calculated in real-time by manipulating the desired reference trajectory. For this reason, the resulting control tool is called *reference governor* (RG). Set-point manipulation is performed on-line through an optimization procedure. This attempts at minimizing a performance index, which is evaluated by predicting the future evolution of the system. The RG is a high-level intelligent control module which supervises conventional controller operation, by "smoothing out" the reference trajectory when abrupt set-point changes would lead to constraint violation.

The proposed control scheme is computationally light and easily implementable on low-cost hardware, and is general enough to cope systematically with different constrained tracking problems. We develop here theory and present simulation results of reference governors for linear, nonlinear, uncertain, robotic, and teleoperated control systems.

For my nieces Sara and Tamara

Acknowledgments

I wish to express my gratitude for the guidance provided by Professor Edoardo Mosca, who inspired this research, collaborated with me in the material presented in this dissertation, and gave me invaluable advise.

Special thanks to: Professor Tzyh-Jong Tarn of Washington University, St.Louis, for his collaboration and advise, which has been fundamental for the results presented in Chapter 6, and for having kindly hosted me in his research group during the academic year 1996/97, providing me full access to his laboratory equipment; Dr. Ning Xi, for assisting me in the experimental sessions with the robots; several graduate students in the Department of Systems Science and Mathematics at Washington University, including Andrea Serrani, Alpay Ozcan, Ben Schwartz, Dave Lockman, and Kevin Brady; and Professor Brian D.O. Anderson, for his constructive comments about the introduction in Chapter 1.

Also, I would like to reserve special acknowledgment for: Giovanni Torrini, for the enjoyable time we spent together as graduate students in our Department; Andrea Garulli, for the same reason and for having shared with me discussions and formulation of the ideas presented in Chapter 4; and Professor Alberto Tesi, for his priceless academic, professional, and personal advise and encouragement.

Finally, I wish to remember the memory of my brother Giorgio A. Bemporad, PhD, missed just after I started my graduate studies, who has always been for me a reference example during my life.

List of Figures

List of Tables

Contents

Chapter 1

Introduction

1.1 Motivation

In almost all industrial applications the design of feedback control systems is complicated by the presence of physical constraints. Just a few examples: Electrical motors have voltage, current, and power limits; in every production plant, temperatures and pressures have to fulfill safety margins; in chemical reactors emission of pollutants is regulated by environmental laws; computer numerically controlled (CNC) machines must satisfy manufacturing tolerances, i.e. tracking error constraints; valves, compressors, pumps, have a limited capacity.

In order to exploit the maximum production capacity of plants while minimizing the costs, control systems that are able to cope with constraints are of paramount importance, to avoid dangerous situations for both equipment (to limit the repair costs) and humans. Although critical constraints could be handled by supervisory systems generating emergency maneuvers according to signals from alarm devices, this would lead to an economic loss, because of discontinuous production cycles. An easy solution to constraint problems could be oversizing certain plant components (e.g. actuators, pipelines, tanks, etc.), or downsizing the throughput, but these correspond again to higher production costs, and consequently products which will be less competitive on the market [APR96, ABKL94, Lau94].

One of the main reasons which led to the Chernobyl nuclear power plant disaster in 1986 was that the speed at which the control rods could move

in and out of the nuclear reactor core was limited. This prevented the slowing down the reaction, because the control action could not be fast enough [Kot97, Ste89]. In Chapter 2 a simulation study on the AFTI-16 aircraft will show how neglecting the elevator and flaperon angles bounds leads to instability. The reason for these phenomena is that when the controller's output is clamped to satisfy input constraints, the control loop is broken between the controller and the actuator. Hence, any closed loop stability warranty is lost.

Constraints can be divided in two classes: *input constraints*, which derive from the presence of *saturating actuators*, and *state-dependent constraints* (or *input/state constraints*), which depend on plant variables that cannot be directly manipulated. Clearly, the first class is easier to deal with, since the command input is artificially generated, while state dependent variables are affected by the dynamics of the plant and disturbances. Because of this intrinsic difficulty, sometimes engineers and researchers distinguish between *hard constraints*, which cannot be violated (e.g. motor voltage limits), and *soft constraints*, whose violation is instead allowed (e.g. temperature bounds), even if strongly penalized.

Many systematic control design techniques are nowadays available to solve stabilization, tracking, robustness, noise reduction problems, both for linear (e.g. PID, lead/lag networks, LQG, H_∞) and nonlinear (e.g. sliding mode, feedback linearization, gain scheduling, nonlinear H_∞) systems. However, none of these is able to cope with constraints. Hence, in control engineering practice, it is common to tune heuristically some of the design parameters (for instance, PID gains), until a satisfactory performance is reached, usually after extensive simulation sessions, leading to an overly conservative linear design. Alternatively, inputs are saturated before being commanded to the actuators, and extra structures (*anti-reset windup* devices) are inserted to avoid pernicious effects. These structures however depend greatly on the particular application at hand. For state-dependent constraints, how to handle saturation is often an unsolved issue.

Because of its natural capability to specify desired performance and handle hard constraints on inputs and state variables in a systematic manner, in recent years *predictive control*—also known as *generalized predictive control* (GPC)–has been investigated and successfully employed in many industrial

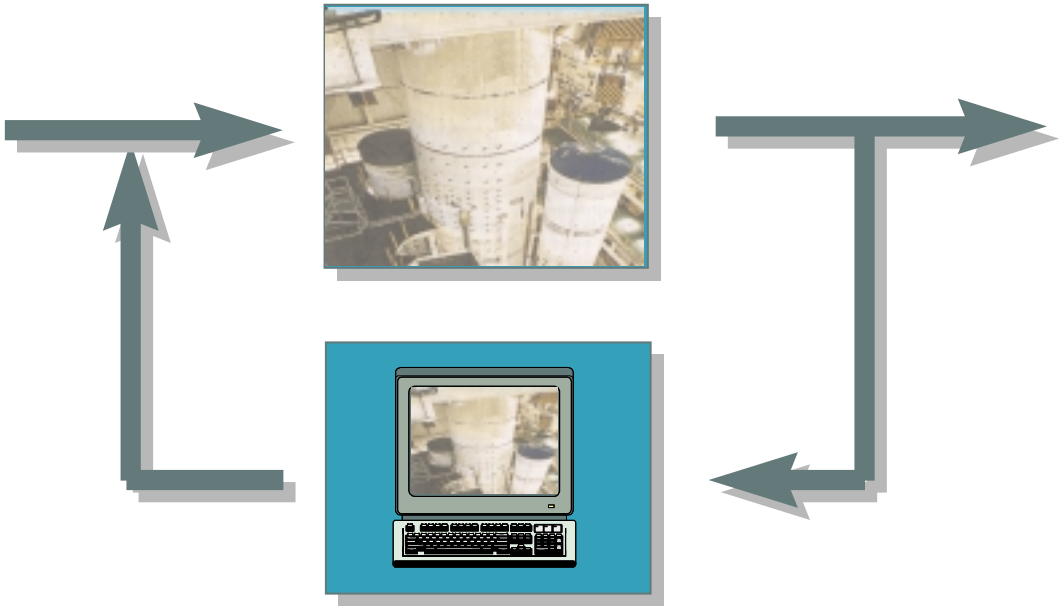


Figure 1.1: Predictive control.

applications [Ric93, Cla94, KG88, MM90, Mos95, RM93]. In predictive control, the main idea is to use a *model* of the plant to *predict* the future evolution of the system and, accordingly, select the command input (Fig. 1.1). For this reason, in the literature it is also referred to as *model based predictive control* (MBPC). Prediction is handled according to the so called *receding horizon* (RH) philosophy: a sequence of future control actions is chosen, by predicting the future evolution of the system, and applied to the plant until new measurements are available. Then, a new sequence is evaluated so as to replace the previous one. Each selected sequence is the result of an optimization procedure which takes into account two objectives: (i) maximize the tracking performance, and (ii) guarantee that the constraints are—and *will be*—fulfilled, i.e., no "blind-alley" is entered. RH is a natural concept, which is applied in many different contexts whenever a *future evolution* can be hypothesized. Consider for instance what happens during a chess game. At each turn, a good player looks at the chessboard (i.e. measures the "state of the process"), and chooses a sequence of moves, according to a prediction of the future evolution of the game (the "dynamics" are the opponent's behavioral features) which fulfils some constraints (the motion of the chessmen must satisfy precise rules) and optimizes a goal (the "cost function", for instance obtaining a checkmate). Then, the first move of this sequence is the actual move of the player. When

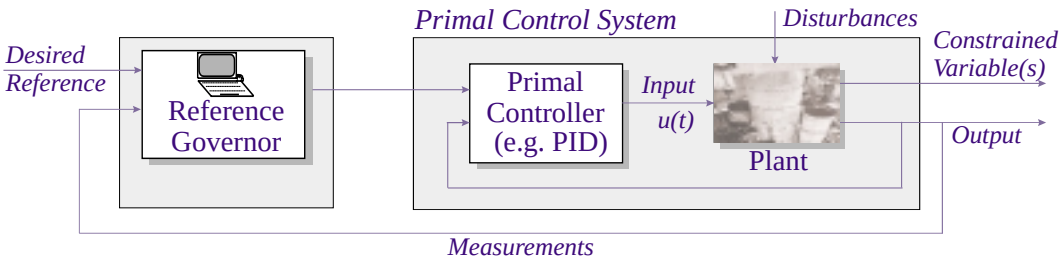


Figure 1.2: Control scheme with reference governor.

it is his turn again, the complete selection process is repeated. This ensures robustness against unpredicted opponent's moves (i.e. "model errors" and "disturbances").

It is clear that the predictive model should be sufficiently accurate to catch the main dynamics of the system, but at the same time simple enough to efficiently compute predictions. However, even in the presence of simple low-order linear models, the computational burden can be huge. For this reason, the scope of applicability of MBPC has been traditionally restricted to slow processes (typically chemical/petrolchemical), for which on-line optimization can be solved within a sampling time by using standard hardware. Although cheaper and more powerful electronic technology is becoming more and more available, the development of computationally simpler schemes has initially motivated our research, with the objective of making predictive control techniques appealing in new application areas.

1.2 The Reference Governor (RG)

Predictive control has the property of being able to take into account the future evolution of the *reference* during the selection of the control action. Such an evolution can be: known in advance, as in applications where repetitive tasks are executed, e.g. industrial robots; predicted, if a dynamic model for the reference is given; or planned in real time. This last instance is a peculiar and important feature of predictive control. In fact, taking into account the current value of both the state vector and the reference, a *virtual* reference evolution can be designed on line so as to possibly make the related input and state responses fulfill pointwise-in-time inequality constraints. However, this mode of operation, whereby the reference is made state-dependent,

introduces an extra feedback loop that complicates the stability analysis of the overall control system. This has been one of the reasons for which on-line reference design, though advocated for a long time as one of the key potential advantages of predictive control [San76b, Ric93, Mos95], had received so far rare consideration in industrial applications.

In most cases, as mentioned above, predictive control computations amount to numerically solving on-line high-dimensional programming problems. In order to lighten computations, it is important to know when and how it is possible to borrow from predictive control the concept of on-line reference management so as to tackle constrained control problems by schemes requiring a lighter computational burden. The main goal of the present research is to address this issue by laying down guidelines for synthesizing *reference governors* (RG) based on predictive control ideas. A RG is a predictive device whose output is a reference trajectory rather than a control input, and it is added to a primal compensated control system, as depicted in Fig. 1.2. We assume that the RG operates at a higher hierarchical level, and hence that a primal controller has already been designed to stabilize the plant and provide nice tracking and disturbance attenuation properties *in the absence of constraints*. The constraint fulfillment task is left to the RG. Whenever necessary, the RG modifies the reference supplied to the primal control system so as to enforce the fulfillment of the constraints. In the sequel, some of the proposed RG schemes will be shown to be only able to slowing down the primal control system. In this case, the primal controller should be designed to provide fast closed-loop response. This corresponds to have large violations of the constraints, which then can be enforced by the RG.

In contrast with other predictive control approaches [KBM96, ZM95], which attempt to solve stabilization/tracking and constraint fulfillment at the same time, we introduce a new philosophy, based on a *separation* of these two requirements: First, stabilize the system, by using one of the many available control techniques, for the unconstrained case; then, impose constraint fulfillment by adding an extra feedback loop. The RG operates in accordance with the receding horizon strategy mentioned above, by selecting *on-line* optimal reference input sequences which, in order to substantially reduce the computational complexity, are described by a few parameters, for instance a scalar quantity.

Preliminary studies along these lines have appeared in [BM94b]. For problems involving saturating actuators, [KAS88, KAS90] developed error governor schemes for continuous-time linear systems. Simultaneously and independently of our research, Gilbert's group at University of Michigan has developed similar control schemes [GK95a, GKT95] for linear systems. Rather than by predictive control ideas, their work was in fact inspired by the output maximal admissible set theory developed in [GT91].

We recently discovered that reference management ideas have already found industrial application in heuristic supervising intelligent control schemes. In fact, [Chi97] reports the following about Yokogawa Electric's temperature controllers: "...fuzzy logic is used to determine artificial set-points that are fed to a conventional PID controller. As the fuzzy module detects impending overshoots, it 'fools' the PID controller by telling the PID controller to aim for a temperature value that is some-what lower than the actual set-point". This further motivates our aim at developing a systematic tool for reference management, easy to implement but with guaranteed stability and constraint fulfillment properties.

1.3 Thesis Outline

The RG is a rather general idea, not dependent on the particular nature of plant under consideration. However, stability and performance issues strongly depend on the nature of the primal control system, this leading to the need of different theoretical developments and computational schemes. In Chapter 2, we present theory and simulative examples for linear systems in the ideal deterministic case. Chapter 3 deals with linear systems in the presence of model uncertainty, expressed in terms of impulse/step-response uncertainty ranges. In Chapter 4, we extend the RG to an arbitrary number of degrees of freedom, and cope with linear systems in the presence of unknown but bounded disturbances and partial state information, by adopting set-membership state estimation techniques. Chapter 5 is dedicated to extend the theory to nonlinear systems, for which a computationally light RG scheme is proposed. In Chapter 6, we specialize the idea to robotic systems, by presenting the *path governor* (PG), a variation of the RG which allows the tracking of given desired paths for manipulators subject to hard constraints. Finally, in Chapter 7

we describe an application of the RG—and, more in general, of predictive control—to teleoperation problems in the presence of arbitrarily large delays, such as those which may arise on the Internet.

Chapter 2

Linear Systems

A method based on conceptual tools of predictive control is described for solving set-point tracking problems wherein pointwise-in-time input and/or state inequality constraints are present. It consists of adding to a primal compensated linear system a nonlinear device called reference governor (RG) whose action is based on the current state, set-point and prescribed constraints. The RG selects at any time a virtual sequence amongst a family of linearly parameterized reference sequences by solving a convex constrained quadratic optimization problem, and feeds the primal system according to a receding horizon control philosophy. The overall system is proved to fulfill the constraints, be asymptotically stable, and exhibit an offset-free tracking behavior, provided that an admissibility condition on the initial state is satisfied. Though the RG can be tailored for the application at hand by appropriately choosing the available design knobs, the required on-line computational load for the usual case of affine constraints is well tempered by the related relatively simple convex quadratic programming problem.

2.1 Introduction

In recent years there have been substantial theoretical advancements in the field of feedback control of dynamic systems with input and/or state-related constraints. For an account of pertinent results see [MP93, SSY94] which also include relevant references.

Most of the work in the field of feedback control of dynamic systems with input and/or state-related constraints has addressed the pure regulation

problem with *time-invariant* constraint sets, particularly input saturation constraints. This chapter aims at studying constrained tracking problems wherein the reference to be tracked is possibly time-varying. In some cases, such problems can be recast as pure regulation problems subject to *time-varying* constraint sets. However, such a time dependence severely limits in practice the potential of many of the existing approaches.

A convenient framework to deal with constrained tracking problems in the presence of time-varying references is the predictive control methodology described in Chapter 1. In most cases, predictive control computations amount to numerically solving on-line a high-dimensional convex quadratic programming problem. Though this can be tackled with existing software packages [BB91], it is a quite formidable computational burden if, as in predictive control, on-line solutions are required. In order to lighten computations, it is important to know when and how it is possible to borrow from predictive control the concept of on-line reference management so as to tackle constrained control problems by schemes requiring a lighter computational burden. The main goal of this chapter is to address this issue by laying down guidelines for synthesizing *reference governors* (RG) for linear systems in the deterministic case. As mentioned in Chapter 1, a RG is a nonlinear device which is added to a primal compensated control system. The latter, in the absence of the RG, is designed so as to perform satisfactorily in the absence of constraints. Whenever necessary, the RG modifies the input to the primal control system so as to avoid violation of the constraints. Hence, the RG action is finalized to let the primal control system operate linearly within a wider dynamic range than that which would result with no RG. Preliminary studies along these lines have already appeared in [BM94b, BM94a, BM95b]. For RGs approached from different perspectives the reader is referred to [KAS88, GKT95].

The chapter is organized as follows. Sect. 2 presents the problem formulation, and defines the RG based on the concept of a virtual reference sequence. Some of the RG stability and performance features are also considered in Sect. 2. Sect. 3 discusses solvability aspects related to the RG optimization problem, and addresses the important practical issue of reducing to a fixed and off-line computable finite prediction-horizon the infinite time-interval over which the fulfilment of constraints has to be checked. In

particular, in Sect. 3 Theorem 2.1 summarizes the main properties of the RG. Simulation examples are presented in Sect. 4 so as to exhibit the results achievable by the method.

2.2 Problem Formulation

Consider the following linear time-invariant system

$$\begin{cases} x(t+1) &= \Phi x(t) + Gg(t) \\ y(t) &= Hx(t) \\ c(t) &= H_c x(t) + Dg(t) \end{cases} \tag{2.1}$$

In (2.1): $t \in \mathbb{Z}_+ \triangleq \{0, 1, \ldots\}$; $x(t) \in \mathbb{R}^n$ is the state vector; $g(t) \in \mathbb{R}^p$ the manipulable reference input which, if no constraints were present, would essentially coincide with the output reference $r(t) \in \mathbb{R}^p$; $y(t) \in \mathbb{R}^p$ the output which is required to track $r(t)$; and $c(t) \in \mathbb{R}^{n_c}$ the constrained vector which has to fulfill the pointwise-in-time set-membership constraint

$$c(t) \in \mathcal{C}, \ \forall t \in \mathbb{Z}_+ \tag{2.2}$$

with $\mathcal{C} \subset \mathbb{R}^{n_c}$ a prescribed constraint set. The problem is to design a memoryless device

$$g(t) \triangleq \underline{g}(x(t), r(t)) \tag{2.3}$$

in such a way that, under suitable conditions, the constraints (2.2) are fulfilled and possibly $y(t) \approx r(t)$. It is assumed that

$$(\mathbf{A.1.}) \begin{cases} \mathbf{1}. \ \Phi \text{ is a stability matrix, i.e. all its eigenvalues} \\ \quad \text{are in the open unit disk;} \\ \mathbf{2}. \ \text{System (2.1) is offset-free, i.e. } H(I-\Phi)^{-1}G = I_p \end{cases}$$

One important instance of (2.1) consists of a linear plant under stabilizing linear state-feedback control. In this way, the system is compensated so as to satisfy stability and performance requirements, regardless of the prescribed constraints. In order to enforce the constraints, the RG (2.3) is added to the primal compensated system (2.1).

It is convenient to adopt the following notations for the equilibrium solution of (2.1) to a constant reference $g(t) \equiv w$

$$\begin{cases} x_w & \triangleq & (I-\Phi)^{-1}Gw \\ y_w & \triangleq & Hx_w \\ c_w & \triangleq & H_c x_w + Dw = \left[H_c(I-\Phi)^{-1}G + D\right] w \end{cases} \tag{2.4}$$

It is further assumed that:

$$(\mathbf{A.2.}) \begin{cases} \mathbf{1}. & \mathcal{C} \text{ is bounded;} \\ \mathbf{2}. & \mathcal{C} = \{c \in \mathbb{R}^{n_c} : q_j(c) \leq 0,\ j \in \underline{n}_q\}, \text{ with} \\ & \underline{n}_q \triangleq \{1, 2, ..., n_q\} \text{ and } q_j : \mathbb{R}^{n_c} \to \mathbb{R} \text{ continuous and convex;} \\ \mathbf{3}. & \mathcal{C} \text{ has a non-empty interior.} \end{cases}$$

(A.2) implies that $\mathcal{C}$ is compact and convex.

Consider a θ-parameterized family $\mathcal{V}_\Theta$ of sequences

$$\mathcal{V}_\Theta = \{v(\cdot,\theta) :\ \theta \in \Theta \subset \mathbb{R}^{n_\theta}\}, \quad v(\cdot,\theta) \triangleq \{v(k,\theta)\}_{k=0}^{\infty} \tag{2.5}$$

with the property of closure w.r.t. left time-shifts, viz. $\forall \theta \in \Theta$, there exist $\bar{\theta} \in \Theta$ such that

$$v(k+1,\theta) = v(k,\bar{\theta}),\ \forall k \in \mathbb{Z}_+ \tag{2.6}$$

Suppose temporarily that $v(\cdot,\theta)$ is used as an input to (2.1) from the state x at time 0. The latter will be referred to as the event $(0,x)$. Assume that

$$c(\cdot,x,\theta) \triangleq \{c(k,x,\theta)\}_{k=0}^{\infty} \subset \mathcal{C} \tag{2.7}$$

In (2.7), $c(k,x,\theta)$ denotes the c-response at time k to $v(\cdot,\theta)$ from the event $(0,x)$. If the inclusion (2.7) is satisfied for some $\theta \in \Theta$, x is said to be *admissible*, (x,θ) an *executable* pair, and $v(\cdot,\theta)$ a *virtual* reference sequence for the state x. Notice that (2.6) ensures that

$$(x,\theta) \text{ is executable} \Longrightarrow \exists \bar{\theta} \in \Theta : (\bar{x},\bar{\theta}) \text{ is executable} \tag{2.8}$$

provided that $\bar{x} = \Phi x + Gv(0,\theta)$. In fact, from (2.6) it follows that $c(k+1,x,\theta) = c(k,\bar{x},\bar{\theta})$. Then, any state is admissible along the trajectory corresponding to a virtual reference sequence $v(\cdot,\theta)$. Consequently, no danger occurs of being trapped into a blind alley if (2.1) is driven by a virtual reference sequence or its input switched from one to another virtual reference sequence.

For reasons which will appear clear soon, it is convenient to introduce the following sets for a given $\delta > 0$:

$$\mathcal{C}_\delta \triangleq \{c \in \mathcal{C} : \mathcal{B}_\delta(c) \subset \mathcal{C}\}, \quad \text{with } \mathcal{B}_\delta(c) \triangleq \{\bar{c} \in \mathbb{R}^{n_c} : \|c - \bar{c}\| \leq \delta\} \tag{2.9}$$

$$\mathcal{W}_\delta \triangleq \{w \in \mathbb{R}^p : c_w \in \mathcal{C}_\delta\} \tag{2.10}$$

Henceforth, we shall assume that there exists a possibly vanishingly small $\delta > 0$ such that

$$(\mathbf{A.3}.) \left\{ \qquad \mathcal{W}_\delta \text{ is non-empty} \right.$$

From the foregoing definitions and (A.3), it follows that $\mathcal{W}_\delta$ is closed and convex. In the developments that follow we shall consider the family $\mathcal{V}_\Theta$ where

$$v(k, \theta) = \gamma^k \mu + w, \tag{2.11}$$

$$\theta \triangleq [\mu' \;\; w']' \in \Theta \quad \triangleq \mathbb{R}^p \times \mathcal{W}_\delta \tag{2.12}$$

where $\gamma \in [0, 1)$ and the prime denotes transpose. The rationale for (2.11)-(2.12) hinges upon the requirement that $\bar{c}(k)$, as given next in (2.20), be in $\mathcal{C}$, $\forall k \in \mathbb{Z}_+$, and the following result whose straightforward proof is omitted.

Lemma 2.1 *The family of reference sequences $v(\cdot, \theta) = \{\gamma(k)\mu + w\}_{k=0}^{\infty}$, with θ as in (2.11)-(2.12) and $\gamma(\cdot)$ a real-valued asymptotically vanishing nonnegative sequence, owns the property of closure w.r.t. left time-shifts (2.6) if and only if*

$$\gamma(k) = \gamma^k, \qquad \gamma \in [0, 1) \tag{2.13}$$

In such a case, (2.6) is satisfied with

$$\bar{\theta} = [\gamma\mu' \;\; w']'. \tag{2.14}$$

□

We consider next the c-response $c(\cdot, x, \theta)$ to the reference sequence (2.11)-(2.12). By straightforward manipulations we find

$$c(k) \;\; \triangleq \;\; c(k, x, \theta) \tag{2.15}$$

$$= \;\; \hat{c}(k) + H_c \Phi^k \left[x - x_{\mu+w}\right] + \tilde{c}(k) \tag{2.16}$$

$$\hat{c}(k) \;\; \triangleq \;\; \gamma^k c_{\mu+w} + (1 - \gamma^k) c_w \tag{2.17}$$

$$\tilde{c}(k) \;\; \triangleq \;\; (1 - \gamma) H_c \sum_{i=0}^{k-1} \Phi^i \gamma^{k-1-i} x_\mu \tag{2.18}$$

In order to establish the existence of admissible c-responses $c(\cdot, x, \theta)$, consider the special case $x = x_{\bar{w}}$ with $\bar{w} \in \mathcal{W}_\delta$. Thus, we can make $x - x_{\mu+w} = 0$ by the choice $\mu = \bar{w} - w$. Accordingly,

$$c(k) = \bar{c}(k) + \tilde{c}(k) \tag{2.19}$$

$$\bar{c}(k) = \gamma^k c_{\bar{w}} + (1 - \gamma^k) c_w \tag{2.20}$$

By the convexity of $\mathcal{C}_\delta$, it follows that $\bar{c}(k) \in \mathcal{C}_\delta$, $\forall k \in \mathbb{Z}_+$. Then, $c(k)$ belongs to $\mathcal{C}$ provided that $\|\tilde{c}(k)\|$ is sufficiently small for all $k \in \mathbb{Z}_+$. In this connection, by stability of (2.1) and given $\gamma \in [0, 1)$, there are two positive reals M and λ, $\lambda \in [0, 1)$ with $\lambda \neq \gamma$, such that for each $x \in \mathbb{R}^n$ one has that $\|\Phi^k x\| \leq M\lambda^k \|x\|$, $\forall k \in \mathbb{Z}_+$. Then, it is possible to show that there are $\mu = \bar{w} - w$, $\|\bar{w} - w\| > 0$, such that $c(\cdot, x, [(\bar{w} - w)' \;\; w']') \subset \mathcal{C}$. In fact, the following inequality holds for all $k \in \mathbb{Z}_+$

$$\|\tilde{c}(k)\| \leq \frac{(1-\gamma)}{|\gamma - \lambda|} \bar{\sigma}(H_c) M \|x_{\bar{w}} - x_w\| \tag{2.21}$$

with $\bar{\sigma}(H_c)$ the maximum singular value of H_c. Recalling that $x_w = (I - \Phi)^{-1} G w$, from (2.21) it follows that $\|\tilde{c}(k)\| \leq \delta$, $\forall k \in \mathbb{Z}_+$, provided that $\|x_{\bar{w}} - x_w\| \leq \delta|\gamma - \lambda|/[(1-\gamma)\bar{\sigma}(H_c)M]$, or $\|\bar{w} - w\| \leq \delta(1-\lambda)|\gamma - \lambda|/[(1 - \gamma)\bar{\sigma}(H_c)\bar{\sigma}(G)M^2]$. The foregoing analysis holds true if the initial state $x_{\bar{w}}$ is additively perturbed by $\tilde{x}$, $0 < \|\tilde{x}\| \leq \varepsilon$, with ε sufficiently small. In this case, the perturbed constrained vector $c(k)$ is such that $c(k) - \bar{c}(k) = H_c \Phi^k \tilde{x} + \tilde{c}(k)$. The condition $\|c(k) - \bar{c}(k)\| \leq \delta$, $\forall k \in \mathbb{Z}_+$ can be ensured, e.g., by requiring that $\|x_{\bar{w}} - x_w\| \leq \frac{1}{2}\delta|\gamma - \lambda|/[(1-\gamma)\bar{\sigma}(H_c)M]$, and $\|\tilde{x}\| \leq \frac{1}{2}\delta/[\bar{\sigma}(H_c)M]$ The conclusion is that starting sufficiently close to an equilibrium state $x_{\bar{w}}$, $\bar{w} \in \mathcal{W}_\delta$, in a finite time one can arrive as close as desired to any state x_w, $w \in \mathcal{W}_\delta$, at a nonzero, though possibly small, distance from $x_{\bar{w}}$. Then, we can move out from any admissible state $x(0)$ to reach asymptotically x_w, any $w \in \mathcal{W}_\delta$, by concatenating a finite number of virtual reference sequences by switching from one to another, the last switching taking place at a finite, though possibly large, time. This result, which by adopting the terminology of [Aub91] will be referred to as a viability property, is summarized in the following Proposition 2.1.

Proposition 2.1 (Viability property). *Consider the system (2.1) along with the family of reference sequences (2.11)-(2.12). Let the assumptions (A.1)-(A.3) be fulfilled and the initial state $x(0)$ of (2.1) be admissible. Then, there*

exists a concatenation of a finite number of virtual reference sequences $v(\cdot,\theta_i)$, $\theta_i = [\mu_i' \quad w_i']'$, $\theta_i \in \Theta$, *with finite switching times, capable of asymptotically driving the system state from* $x(0)$ *to* x_w, *any* $w \in \mathcal{W}_\delta$.

□

Remark 1 We leave to the reader the simple task of specializing the analysis and the results of this chapter to the case of a family $\mathcal{V}_\Theta$ of constant sequences

$$v(k,\theta) = \theta \in \Theta \triangleq \mathcal{W}_\delta \subseteq \mathbb{R}^p.$$

□

Hereafter, we shall address the problem of how to select appropriate virtual reference sequences, and when to switch from one to another. To this end, consider the quadratic selection index

$$J(x(t),r(t),\theta) \triangleq \|\mu\|^2_{\Psi_\mu} + \|w - r(t)\|^2_{\Psi_w} + \sum_{k=0}^{\infty} \|y(k,x(t),\theta) - w\|^2_{\Psi_y} \quad (2.22)$$

where θ is as in (2.12), $\|x\|^2_\Psi \triangleq x'\Psi x$, $\Psi_\mu = \Psi_\mu' > 0$, $\Psi_w = \Psi_w' > 0$, $\Psi_y = \Psi_y' \geq 0$, and $y(k,x(t),\theta)$ the output response at time k to the reference $v(k,\theta) = \gamma^k\mu + w$ from the event $(0,x(t))$. It is easy to see that (2.22) has a unique unconstrained minimum $\theta(t) \in \mathbb{R}^{2p}$ for every $x(t) \in \mathbb{R}^n$ and $r(t) \in \mathbb{R}^p$. Let $\mathcal{V}(x)$ be the set of all $\theta \in \Theta$ such that (x,θ) is executable

$$\mathcal{V}(x) \triangleq \{\theta \in \Theta : \ c(\cdot,x,\theta) \subset \mathcal{C}\} \quad (2.23)$$

Assume that $\mathcal{V}(x(t))$ is non-empty, closed and convex for every $t \in \mathbb{Z}_+$. This implies that the following minimizer exists uniquely

$$\begin{aligned} \theta(t) &\triangleq \arg\min_{\theta\in\Theta} \{J(x(t),r(t),\theta) : \ c(\cdot,x(t),\theta) \subset \mathcal{C}\} \\ &= \arg\min_{\theta\in\mathcal{V}(x(t))} J(x(t),r(t),\theta) \end{aligned} \quad (2.24)$$

Proposition 2.1 ensures that $\mathcal{V}(x(t))$ non-empty implies that $\mathcal{V}(x(t+1))$ non-empty if $(x(t),\theta)$ is executable and $x(t+1) = \Phi x(t) + Gv(0,\theta)$. Further, the concatenation mechanism embedded in the viability property of Proposition 2.1 naturally suggests that we can select the actual RG action according to the following receding horizon control strategy if $\theta(t)$ is as in (2.24):

$$g(t) = v(0,\theta(t)) = \mu(t) + w(t) \quad (2.25)$$

Remark 2 If the computational delay is not negligible w.r.t. the sampling interval, we can modify (2.24) as follows

$$\theta((i+1)\tau) = \arg \min_{\theta \in \mathcal{V}(x(i\tau))} J(x(i\tau), r(i\tau), \theta)$$

$i \in \mathbb{Z}_+$, and set for $k = 0, 1, \ldots, \tau - 1$

$$g((i+1)\tau + k) = v(k, \theta((i+1)\tau))$$

This amounts to using an "open-loop" reference sequence over intervals made up by τ steps. While the results proved in the remaining part of this section and in Sect. 3 can be easily extended to cover this case, a tracking performance degradation typically results from a significant computational delay. □

Remark 3. As elaborated in some detail in next Example 2, the weighting matrices Ψ_μ and Ψ_w can be made $r(t)$-dependent so as to force the direction of the selected vector $g(t) = \mu(t) + w(t)$ to be as close as possible to that of $r(t)$, compatibly with the constraints. This can be a qualitatively important requirement in some MIMO applications. □

We defer the proof that $\mathcal{V}(x(t))$ is closed and convex to Sect. 3. A question we wish to address now is whether the foregoing RG yields an overall stable offset-free control system. Assume that the reference is kept constant, $r(t) \equiv r$ for all $t \geq t^*$, and $\mathcal{V}(x(t))$ is non-empty, closed and convex at each $t \in \mathbb{Z}_+$. Consider the following candidate Lyapunov function

$$V(t) \triangleq J(x(t), r, \theta(t)) \tag{2.26}$$

If $x(t+1) = \Phi x(t) + Gv(0, \theta(t))$, it results that $J(x(t+1), r, [\gamma\mu'(t) \;\; w'(t)]') \geq V(t+1)$. In fact, $(x(t+1), [\gamma\mu'(t) \;\; w'(t)]')$ is executable, but $[\gamma\mu'(t) \;\; w'(t)]'$ need not be the minimizer for $J(x(t+1), r, \theta)$. It follows that along the trajectories of the system

$$V(t) - V(t+1) \geq (1-\gamma^2)\|\mu(t)\|^2_{\Psi_\mu} + \|y(t) - w(t)\|^2_{\Psi_y} \geq 0 \tag{2.27}$$

Hence, $V(t)$, being nonnegative monotonically non increasing, has a finite limit $V(\infty)$ as $t \to \infty$. This implies $\lim_{t\to\infty}[V(t) - V(t+1)] = 0$, and by (2.27)

$$\lim_{t\to\infty} \mu(t) = 0_p \tag{2.28}$$

$$\lim_{t\to\infty} \|y(t) - w(t)\|_{\Psi_y} = 0 \tag{2.29}$$

Lemma 2.2 *Consider the system (2.1) controlled by the RG (2.24)-(2.25). Assume that (A.1)-(A.3) are satisfied. Let $x(0)$ be admissible and $\mathcal{V}(x(t))$ closed and convex at each $t \in \mathbb{Z}_+$. Let $r(t) \equiv r,\ \forall t \geq t^* \in \mathbb{Z}_+$. Then,*

$$V(\infty) \triangleq \lim_{\tau\to\infty} V(\tau) \leq V(t+1) \leq V(t),\ \forall t \geq t^* \tag{2.30}$$

(2.28) and (2.29) hold, and the RG output exhibits asymptotically vanishing variations

$$\lim_{t\to\infty} [w(t+1) - w(t)] = 0_p \tag{2.31}$$

Further,

$$\lim_{t\to\infty} [x(t) - x_{w(t)}] = 0_p \tag{2.32}$$

where $x_{w(t)} \triangleq (I - \Phi)^{-1} G w(t)$, and

$$V(\infty) = \lim_{t\to\infty} \|w(t) - r\|^2_{\Psi_w} \tag{2.33}$$

Proof. Eq. (2.30) has already been proved under the stated assumptions. It follows by strict positivity of Ψ_μ and Ψ_w that the RG output sequence $g(\cdot) = \mu(\cdot) + w(\cdot)$ is bounded. Hence, by (2.28) and stability of (2.1), the system state evolution $x(\cdot)$ remains bounded as well. Let $\bar{\theta}$ be defined in terms of θ as in (2.14). Then, along the trajectories of the system for each $t \geq t^*$

$$V(t) = J(x(t+1), r, \bar{\theta}(t)) + (1 - \gamma^2)\|\mu(t)\|^2_{\psi_\mu} + \|y(t) - w(t)\|^2_{\psi_y}$$

and, by (2.8), (2.24), and convexity of J,

$$V(t+1) = J(x(t+1), r, \theta(t+1)) \leq \underline{J}(t, \alpha) \leq J(x(t+1), r, \bar{\theta}(t)) \tag{2.34}$$

where $\alpha \in [0, 1]$ and

$$\underline{J}(t, \alpha) \triangleq J(x(t+1), r, \bar{\theta}(t) + \alpha[\theta(t+1) - \bar{\theta}(t)]) \tag{2.35}$$

Now, taking into account (2.22), it is easy to see that

$$\underline{J}(t, \alpha) = \alpha^2 \|\theta(t+1) - \bar{\theta}(t)\|^2_\Psi + \alpha c_1(t) + c_2(t) \tag{2.36}$$

with $\Psi = \Psi' > 0$, and $c_1(\cdot)$ and $c_2(\cdot)$ bounded real-valued sequences. Then, because $\lim_{t\to\infty} \underline{J}(t,1) = \lim_{t\to\infty} \underline{J}(t,0) = V(\infty)$, from (2.34) and (2.36) it follows that

$$\lim_{t\to\infty} \underline{J}(t,\alpha) = V(\infty), \forall \alpha \in [0,1]. \tag{2.37}$$

Consequently, $\lim_{t\to\infty}[\theta(t+1) - \bar{\theta}(t)] = 0$, and hence (2.31) from (2.14). Eq. (2.1), (2.25), (2.28) and (2.31) imply that $\lim_{t\to\infty}[x(t+1) - x(t)] = 0_p$. Hence, $\lim_{t\to\infty}[x(t) - \Phi x(t) + Gw(t)] = 0_p$ from which (2.32) follows. To show (2.33), consider that $y(k, x(t), \theta(t)) = y(k, x_{w(t)}, [0_p' \;\; w'(t)]') + y(k, \tilde{x}(t), [\mu'(t) \;\; 0_p']') = w(t) + y(k, \tilde{x}(t), [\mu'(t) \;\; 0_p']')$, $\tilde{x}(t) \triangleq x(t) - x_{w(t)}$. Then, $\sum_{k=0}^{\infty} \|y(k, x(t), \theta(t)) - w(t)\|^2_{\Psi_y} = \|[\tilde{x}'(t) \;\; \mu'(t)]'\|^2_Q$ for some symmetric nonnegative-definite matrix Q. Because of (2.28) and (2.32), last quantity goes to zero as $t \to \infty$. This proves (2.33). □

We are now ready to prove that, under the conditions stated after (2.23), the output of the system controlled by the RG converges to the best possible approximation to the reference.

Proposition 2.2 *Under the same assumptions as in Lemma 2.2, the prescribed constraints are satisfied at every $t \in \mathbb{Z}_+$, and*

$$\lim_{t\to\infty} y(t) = \lim_{t\to\infty} g(t) = w_r \tag{2.38}$$

$$w_r \triangleq \arg\min_{w\in\mathcal{W}_\delta} \|w - r\|^2_{\Psi_w} \tag{2.39}$$

Proof. Because (2.33) implies that $V(\infty) \geq V_r \triangleq \min_{w\in\mathcal{W}_\delta} \|w - r\|^2_{\Psi_w}$, (2.38) is proven if we show that $V(\infty) = V_r$. Assume to the contrary that $V_r < V(\infty)$. Under this assumption, we show that, along the trajectory of the system controlled by the RG, for t large enough, we can find a virtual reference sequence $v(\cdot, \theta_\alpha(t))$ such that $J(x(t), r, \theta_\alpha(t)) < V(\infty)$. Because $V(t) \leq J(x(t), r, \theta_\alpha(t))$ and (2.30), the previous inequality contradicts the assumption. For $\alpha \in (0, 1]$ let

$$\theta_\alpha(t) \triangleq \begin{bmatrix} 0_p \\ w_\alpha(t) \end{bmatrix}, \quad w_\alpha(t) \triangleq (1-\alpha)w(t) + \alpha w_r \tag{2.40}$$

Because $x(t) - x_{w(t)} \to 0$, by Proposition 2.1 there is a time $t_1 \geq t^*$ and a positive real $\alpha_1 \in (0, 1]$ such that $\forall t \geq t_1$, $\forall \alpha \in [0, \alpha_1]$, $(x(t), \theta_\alpha(t))$ is

executable. Look next at $J(x_{w(t)}, r, \theta_\alpha(t))$. It can be found that

$$\begin{aligned} J(x_{w(t)}, r, \theta_\alpha(t)) &= (1-\alpha)^2 \|w(t) - w_r\|^2_{\Psi_w} + 2(1-\alpha)(w_r - r)'\Psi_w(w(t) - w_r) + \\ &\quad \alpha^2 \|w(t) - w_r\|^2_{\mathcal{K}} + \|r - w_r\|^2_{\Psi_w} \end{aligned} \tag{2.41}$$

where $\mathcal{K} = \mathcal{K}' \geq 0$ is such that $\sum_{k=0}^{\infty} \|y(k, x_{w(t)}, \theta_\alpha(t)) - w_\alpha(t)\|^2_{\Psi_y} = \alpha^2 \|w(t) - w_r\|^2_{\mathcal{K}}$. Such a matrix equals $\mathcal{K} = G'(I-\Phi)^{-T}\mathcal{L}(I-\Phi)^{-1}G$ where $X^{-T} \triangleq (X^{-1})'$ and $\mathcal{L} = \mathcal{L}' \geq 0$ is the observability Gramian of the pair $(\Phi, \Psi_y^{\frac{1}{2}} H)$. From (2.41), we find that

$$J(x_{w(t)}, r, \theta_\alpha(t)) < \|w(t) - r\|^2_{\Psi_w} \tag{2.42}$$

provided that

$$\frac{\alpha}{2} \|w(t) - w_r\|^2_{(\Psi_w + \mathcal{K})} < \|w(t) - w_r\|^2_{\Psi_w} + (w_r - r)'\Psi_w(w(t) - w_r) \tag{2.43}$$

By convexity of $\mathcal{W}_\delta$, the right-most term of (2.43) is nonnegative. Hence, for every $\alpha \in (0, \bar{\alpha}(t))$

$$\bar{\alpha}(t) \triangleq \frac{2\|w(t) - w_r\|^2_{\Psi_w}}{\|w(t) - w_r\|^2_{\Psi_w + \mathcal{K}}}$$

(2.43) is satisfied. Then, because $x(t) - x_{w(t)} \to 0$ and of continuity of J w.r.t. x, there is a time $t_2 \geq t_1$ such that for every $t \geq t_2$ and $\alpha \in (0, \underline{\alpha}_2)$, $0 < \underline{\alpha}_2 < \min\{\alpha_1, \bar{\alpha}(t)\}$, $(x(t), \theta_\alpha(t))$ is executable and $J(x(t), r, \theta_\alpha(t)) < V(\infty)$. □

2.3 Solvability and Computability

It remains to find existence conditions for the minimizer (2.24). Further, even if solvability is guaranteed, (2.24) embodies an infinite number of constraints. For practical implementation, we must find out if and how these constraints can be reduced to a finite number of constraints whose time locations be determinable off-line. To this end, it is convenient to introduce some extra notation. We express the response of (2.1) from an event $(0, x)$ to the reference sequence (2.11)-(2.12) as follows

$$\left\{ \begin{array}{rcl} z(k+1) &=& Az(k), \text{ with } z(0) = \left[\begin{array}{c} x \\ \theta \end{array} \right] = \left[\begin{array}{c} x \\ \mu \\ w \end{array} \right] \in \mathbb{R}^n \times \Theta, \\ \\ c(k) &\triangleq& c(k, x, \theta) \\ &=& E_c z(k) \end{array} \right. \tag{2.44}$$

where

$$A = \begin{bmatrix} \Phi & G & G \\ 0_{p\times n} & \gamma I_p & 0_{p\times p} \\ 0_{p\times n} & 0_{p\times p} & I_p \end{bmatrix}, \quad E_c = [H_c \;\; D \;\; D] \tag{2.45}$$

For $i \in \mathbb{Z}_1 \triangleq \{1, 2, 3, ...\}$, consider the following sets

$$\mathcal{Z}_i \triangleq \{z \in \mathbb{R}^n \times \Theta : \; q_j(E_c A^{k-1} z) \leq 0, \; j \in \underline{n}_q, \; k \in \underline{i}\}. \tag{2.46}$$

$$\mathcal{Z} \triangleq \bigcap_{i=0}^{\infty} \mathcal{Z}_i \tag{2.47}$$

$\mathcal{Z}_i$ are the sets of initial states z with $w \in \mathcal{W}_\delta$ which give rise to evolutions fulfilling the constraints over the first i-th time steps $k = 0, 1, ..., i-1$, while $\mathcal{Z}$ is the set of all executable pairs (x, θ). $\mathcal{Z}_{i+1} \subset \mathcal{Z}_i$, $\forall i \in \mathbb{Z}_1$, and under (A.2), all the $\mathcal{Z}_i$'s, and hence $\mathcal{Z}$, are closed and convex. Moreover, by the viability property of Proposition 2.1 $\mathcal{Z}$ is non-empty. The lemma that follows can be proved as in [GT91] taking into account (2.44)-(2.46).

Lemma 2.3

$$\mathcal{Z}_i = \mathcal{Z}_{i+1} \Longrightarrow \mathcal{Z}_i = \mathcal{Z}.$$

□

Consider next the "slice" of $\mathcal{Z}$ along x

$$\mathcal{V}(x) \triangleq \{ \begin{bmatrix} \mu \\ w \end{bmatrix} \in \Theta : \begin{bmatrix} x \\ \mu \\ w \end{bmatrix} \in \mathcal{Z}\}. \tag{2.48}$$

If x is admissible for some $\theta \in \Theta$, $\mathcal{V}(x)$ is non-empty. In addition, it is closed being the intersection of two closed sets, $\mathcal{V}(x) = \mathcal{Z} \bigcap \{\{x\} \times \Theta\}$. $\mathcal{V}(x)$ is also convex because the "slicer" operator is convexity-preserving. Then, existence and uniqueness of the minimizer (2.24) follows, provided that the initial state of (2.1) be admissible.

Practical implementation of the RG requires an effective way to solve the optimization problem (2.24). Notice in fact that there might be no algorithmic procedure capable of computing the exact minimizer, unless $\mathcal{Z}$ is finitely determined, *viz.* $\mathcal{Z} = \mathcal{Z}_i$ for some $i \in \mathbb{Z}_+$. In what follows, we shall show

that only a finite number of pointwise-in-time constraints suffices to determine $\mathcal{Z}$. To this end, let (A_o, E_{co}), with $A_o \in \mathbb{R}^{n_o \times n_o}$, $n_o \leq n+2p$, be an observable subsystem obtained via a canonical observability decomposition of (A, E_c). Then

$$c(k) = E_{c_o} A_o^k z_o(0) \tag{2.49}$$

with $z_o = P_o z$, P_o defined by the observability decomposition. Consequently, define the following sets

$$\mathcal{Z}_i^o \triangleq \{P_o z \in \mathbb{R}^{n_o} : z \in \mathcal{Z}_i\}, \quad \mathcal{Z}^o \triangleq \bigcap_{i=0}^{\infty} \mathcal{Z}_i^o \tag{2.50}$$

It is easy to see that $\mathcal{Z}_i^o$ and $\mathcal{Z}^o$ own the same properties shown to hold for $\mathcal{Z}_i$ and, respectively, $\mathcal{Z}$. In particular, they are non-empty, closed and convex. Moreover, the following result holds.

Proposition 2.3 *Let (A.1)-(A.3) be fulfilled. Then, $\mathcal{Z}_i^o$, $\forall i \geq n_o$ is compact and convex. Moreover, there exists an integer $i_o \geq n_o$ such that $\mathcal{Z}_{i_o} = \mathcal{Z}$.*

Proof. Let $z_o \in \mathcal{Z}_{n_o}^o$. Because (A_o, E_{c_o}) is an observable pair, $\Theta'\Theta$ is a nonsingular matrix, where $\Theta \triangleq \left[E_{c_o}' | (E_{c_o} A_o)' | ... | (E_{c_o} A_o^{n_o-1})'\right]'$. It follows that, $z_o = (\Theta'\Theta)^{-1}\Theta' R$, with $R = [c'(0, x, \theta),\ \ldots,\ c'(n_o - 1, x, \theta)]'$. Then, being $\mathcal{C}$ bounded, $\mathcal{Z}_{n_o+l}^o$, $\forall l \in \mathbb{Z}_+$, is bounded as well, because $\mathcal{Z}_{n_o+l}^o \subset \mathcal{Z}_{n_o}^o$, $\forall l \in \mathbb{Z}_+$. In order to show that $\mathcal{Z}$ is finitely determined, note that $\lim_{i\to\infty} c(i, x, \theta) = c_w$. Now

$$\begin{aligned} c(i, x, \theta) - c_w &= E_c M^i (z - z_w) \\ &= E_{co} M_o^i (z_o - z_{wo}) \end{aligned}$$

where $M = \begin{bmatrix} \Phi & G & 0_{n\times p} \\ 0_{p\times n} & \gamma I_p & 0_{p\times p} \\ 0_{p\times n} & 0_{p\times p} & 0_{p\times p} \end{bmatrix}$, $z_w = \begin{bmatrix} x_w \\ 0 \\ w \end{bmatrix}$, M_o is obtained from M in the same way as A_o from A, and $z_o = P_o z$ and $z_{wo} = P_o z_w$. Then,

$$\|c(i, x, \theta) - c_w\| \leq \bar{\sigma}\left(E_{co} M_o^i\right) (\|z_o\| + \|z_{wo}\|)$$

Because $z_{wo} \in \mathcal{Z}^o$, $\|z_o\| + \|z_{wo}\|$ is bounded for all $z_o \in \mathcal{Z}^o$. Therefore, the existence of an integer i_o such that

$$i \geq i_o \Longrightarrow \|c(i, x, \theta) - c_w\| \leq \delta, \quad \forall z \in \mathcal{Z}$$

follows from asymptotic stability of M. □

It follows that $\mathcal{Z}^o$, and hence $\mathcal{Z}$ as well, is finitely determined, that is it suffices to check the constraints over the initial i_o-th time steps in order to ensure constraint fulfillment over $\mathbb{Z}_+$. Consequently, problem (2.24) is equivalent to the following finite dimensional convex constrained optimization problem at each $t \in \mathbb{Z}_+$:

$$\theta(t) = \begin{bmatrix} \mu(t) \\ w(t) \end{bmatrix} \triangleq \begin{array}{l} \arg\min_{\mu\in\mathbb{R}^p,\ \ w\in\mathcal{W}_\delta} J(x(t), r(t), \theta) \\ \qquad \text{subject to } q_j(c(i-1, x(t), \theta) \le 0,\ j \in \underline{n}_q,\ i \in \underline{i}_o \end{array} \tag{2.51}$$

The Gilbert and Tan algorithm [GT91] can be adapted to the present case to find $i_o = \min_{i\ge n_o}\{i \mid \mathcal{Z}_i^o = \mathcal{Z}^o\}$. To this end, let

$$\begin{array}{rcl} G_i(j) & \triangleq & \max_{x\in\mathbb{R}^n,\theta\in\Theta}\{q_j(c(i, x, \theta)\},\ j \in \underline{n}_q, i = 1, 2, ... \\ & & \text{subject to } q_j(c(k-1, x, \theta) \le 0,\ j \in \underline{n}_q,\ k \in \underline{i} \end{array} \tag{2.52}$$

Then, i_o can be computed off-line via the following algorithm:

$$\left.\begin{array}{ll} 1. & i \leftarrow n_o; \\ 2. & \text{Solve } G_i(j),\ \forall j \in \underline{n}_q; \\ 3. & \text{If } G_i(j) \le 0,\ \forall j \in \underline{n}_q, \text{ let } i_o = i \text{ and stop;} \\ 4. & \text{Otherwise } i \leftarrow i + 1, \text{ and go to 2.} \end{array}\right\} \tag{2.53}$$

Notice that step 2. in (2.53) is well posed because, according to Proposition 2.3, the implied maximization is carried out over a compact and convex set. In conclusion, we have found that our initial optimization problem having an infinite number of constraints is equivalent to a convex constrained optimization problem with a finite number of constraints.

Theorem 2.1 *Let (A.1)-(A.3) be fulfilled. Consider the system (2.1) with the RG (2.24)-(2.25), and let $x(0)$ be admissible. Then:*

i. *The J-minimizer (2.24) uniquely exists at each $t \in \mathbb{Z}_+$ and can be obtained by solving a convex constrained optimization problem with inequality constraints $q_j(c(i-1, x(t), \theta)) \le 0,\ j \in \underline{n}_q$, limited to a finite number i_o of time-steps, viz. $i = 1, ..., i_o$;*

ii. *The integer i_o can be computed off-line via (2.53);*

iii. *The overall system satisfies the constraints, is asymptotically stable and off-set free in that the conclusions of Proposition 2.2 hold.*

Remark 4 Theorem 2.1 assumes $x(0)$ admissible. The set $\mathcal{X}$ of all admissible initial states $x(0)$ corresponds to the projection of $\mathcal{Z}$ on the x-space. Moreover, by the property of closure w.r.t. left time-shifts (2.6), $\mathcal{X}$ is an invariant set for system (2.1), (2.24)-(2.25).

2.4 Computations

We first concentrate on finding the analytical form of the command selection index (2.22) in terms of the vectors μ and w for the system (2.1). It is easy to show that

$$J(x(t), r(t), \theta) = \mu'\psi_\mu\mu + (w - r(t))'\psi_w(w - r(t)) + [x'(t) - x'_w \ \mu']\mathcal{L}\begin{bmatrix} x(t) - x_w \\ \mu \end{bmatrix}$$

where $\mathcal{L} = \mathcal{L}'$ solves the Lyapunov equation

$$\mathcal{L} = A'\mathcal{L}A + C'\psi_y C \tag{2.54}$$

Matrix $\mathcal{L}$ can be partitioned in the form

$$\mathcal{L} = \begin{bmatrix} \mathcal{L}_{11} & \mathcal{L}_{12} \\ \mathcal{L}'_{12} & \mathcal{L}_{22} \end{bmatrix}, \ \dim \mathcal{L}_{11} = \dim \Phi$$

and computed by the following relations

$$\begin{aligned} \mathcal{L}_{11} &= \Phi'\mathcal{L}_{11}\Phi + H'\psi_y H, \\ (I - \gamma\Phi')\mathcal{L}_{12} &= \Phi'\mathcal{L}_{11}G, \\ (1 - \gamma^2)\mathcal{L}_{22} &= G'\mathcal{L}_{11}G + \gamma(\mathcal{L}'_{12}G + G'\mathcal{L}_{12}) \end{aligned}$$

Finally,

$$J(\mu, w) = [\mu' \ w']A_J \begin{bmatrix} \mu \\ w \end{bmatrix} + 2B_J \begin{bmatrix} \mu \\ w \end{bmatrix} + C_J \tag{2.55}$$

where

$$\begin{aligned} A_J &= \begin{bmatrix} \psi_\mu + \mathcal{L}_{22} & -\mathcal{L}'_{12}(I - \Phi)^{-1}G \\ -G'(I - \Phi)'\mathcal{L}_{12}^{-1} & \psi_w + G'(I - \Phi)'^{-1}\mathcal{L}_{11}(I - \Phi)^{-1}G \end{bmatrix} \\ B_J &= x'(t)\begin{bmatrix} \mathcal{L}_{12}, & -\mathcal{L}_{11}(I - \Phi)^{-1}G \end{bmatrix} - r'(t)\begin{bmatrix} 0 & \psi_w \end{bmatrix}, \\ C_J &= x'(t)\mathcal{L}_{11}x(t) + r'(t)\psi_w r(t) \end{aligned}$$

The on-line computational complexity required to express $J(\mu, w)$ derives from the evaluation of B_J ($p(n+p)$ multiplications and $p(n+p-2)$ additions).

Notice that if the constraints are non active and $\psi_y = 0$, then $\mathcal{L} = 0$ and

$$\begin{bmatrix} \mu^* \\ w^* \end{bmatrix} = -A_J^{-1} B_J = \begin{bmatrix} 0 \\ r(t) \end{bmatrix}$$

which means that the RG does not perturb the original system dynamics.

The constraints $c(k, x, \theta) \in \mathcal{C}$ are expressed in terms of μ and w by the $n_q \cdot i_o$ linear constraints

$$A_c(D_c\gamma^k + \sum_{i=0}^{k-1} H_c\Phi^i G\gamma^{k-1-i})\mu + A_c(D_c + \sum_{i=0}^{k-1} H_c\Phi^i G)w \leq B_c - H_c\Phi^k x(0), \tag{2.56}$$
$$k = 0, \ldots, i_o$$

and the one which derives from the constraint $w \in W_\delta$

$$A_c H_c (I - \Phi)^{-1} G w \leq B_c - \delta \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}$$

2.5 Reducing the Complexity by Scalar Optimization

In some applications the QP minimization in $\mathbb{R}^{2p}$ could turn out to be too much time consuming. In order to cope with this situation, here we propose a slight modification to our algorithm, which drastically reduce the computational complexity. First, we restrict the parameter domain to $\mathbb{R}^p$ by setting

$$\mu = 0 \tag{2.57}$$

and

$$J(w) = \|w - r(t)\|^2_{\psi_w}$$

Then, we impose

$$w = g(t-1) + \beta[r(t) - g(t-1)], \tag{2.58}$$

where β is the scalar optimization parameter. Hence (2.22) can be rewritten as

$$J(\beta) = c(1-\beta)^2$$

where $c = [r(t) - g(t-1)]'\psi_w[r(t) - g(t-1)]$. Moreover, the constraints (2.56) assume the form

$$\beta_k^- \leq \beta \leq \beta_k^+$$

Since the unconstrained optimizer is $\beta = 1$, by setting

$$\beta^- = \max_{k \leq i_o} \beta_k^-, \ \beta^+ = \min_{k \leq i_o} \beta_k^-$$

the constrained minimizer β has the following form

$$\beta = \begin{cases} 1 & \text{if} \quad \beta^- \leq 1 \leq \beta^+ \\ \beta^- & \text{if} \quad \beta^- > 1 \\ \beta^+ & \text{if} \quad \beta^+ < 1 \end{cases}$$

In order to assess the computational benefits provided by this simplified scheme, next Table 2.1 reports the number of flops measured in both vector and scalar optimization approaches. However, due to diminished degrees of freedom in the choice of the command input, this computational improvement is at the expenses of a loss of tracking performance.

Table 2.1: Comparison between vector and scalar optimization (in both cases $i_o = 10$)

Method	*Flops for step*	*Example*
vector	7000	2×2 system
scalar	640	$\dim x = 6$, $n_q = 4$
vector	1915	SISO system
scalar	720	$\dim x = 8$, $n_q = 4$

2.6 Comparison with other methods

We shall briefly compare our RG with different schemes appeared in the literature.

Without referring to predictive control ideas, [GKT95] have developed a different reference governor based on the following ideas. The actual reference $g(t)$ is obtained by successive increments toward the desired reference trajectory $r(t)$

$$g(t+1) = g(t) + K(t)(r(t) - g(t)) \tag{2.59}$$

where $K(t) \in [0,1]$ is chosen so as to have constraint fulfillment. This method can be reformulated in our framework by unit-delaying the reference $r(t)$ and setting $\gamma = 0$, $w = \beta[w(t-1) - r(t-1)] + r(t-1)$, $\Psi_y = 0_p$, where $\beta \in [0,1]$ is a new free parameter. Because of the further constraints imposed on w, this formulation gives less degrees of freedom in the choice of the reference. For example, in a MIMO context, each reference trajectory increases toward its desired value of the same fraction β. On the other hand, a fast scalar optimization is performed.

The method of [BM94b] is closer in spirit with the one adopted in the present chapter in that it uses the virtual reference philosophy. In fact, [BM94b] considers a virtual reference as either

$$g(k, \lambda(t)) = \lambda^{k+1}(t) g(k, \lambda(t-1)) + (1 - \lambda^{k+1}(t)) r(t), \ \lambda \in [0,1] \tag{2.60}$$

or

$$g(k, \lambda(t)) = \begin{cases} \lambda^{k+1}(t) y(t) + (1 - \lambda^{k+1}(t)) w(t), & \text{if } \exists \lambda \in [0,1] \text{ feasible} \\ g(k, \lambda(t-1)), & \text{otherwise} \end{cases} \tag{2.61}$$

and in both instances λ is selected as the smallest real making admissible the system response to the designed reference sequence. Here, if high performance is required, on-line selection of the λ-parameter becomes time consuming even for SISO systems, and stability and performance analysis of (2.61) turns out not to be so immediate and transparent as for (2.60) and the case of the present chapter.

2.7 Simulation studies

We investigate in some detail how to tune the free parameters of the RG, with direct reference to two different examples. The simulation results reported hereafter were obtained under Matlab 4.0 + Simulink 1.2 on a 486 DX2/66 personal computer, with no particular care of code optimization. The standard Matlab `QP.M` routine was used for quadratic optimization.

Example 1 Consider the following nonminimum-phase SISO system

$$y(t) = \frac{-0.8935z + 1.0237}{z^2 - 1.5402z + 0.6703} g(t) \tag{2.62}$$

The unit step response of (2.62) is depicted in Fig. 2.1a (thin line). The task

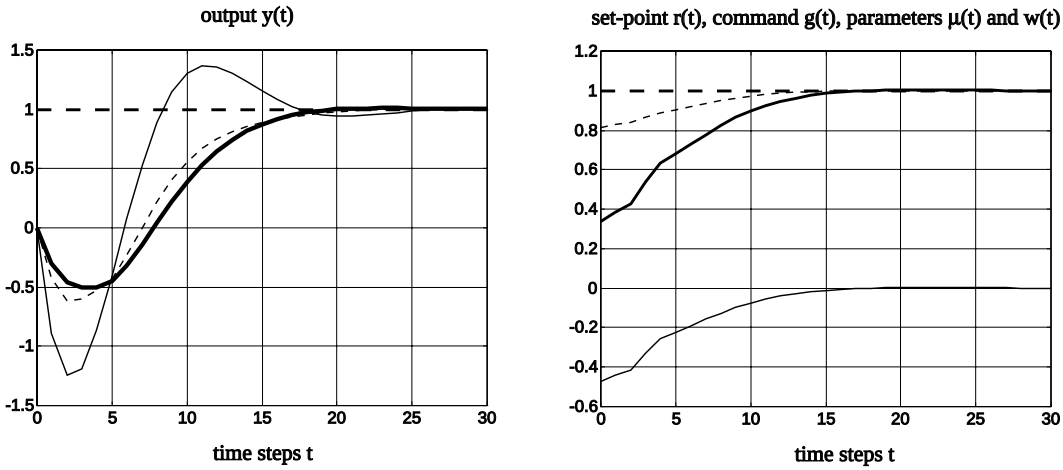


Figure 2.1: Example 1: (a) [left] Unit step response with no RG (thin line) and with RG ($\delta = 0.05$; $\gamma = 0.9$; $\Psi_\mu/\Psi_w = 0.1$; $\Psi_y/\Psi_w = 0.01$; thick line) for the nominal plant (2.62); Response with RG for the perturbed plant (2.63) (dashed line). (b) [right] Reference trajectory $r(t)$ (thick dashed line); Generated reference trajectory $g(t)$ (thick line); minimizing parameters $\mu(t)$ (thin solid line) and $w(t)$ (thin dashed line).

of the RG is to bound the output between -0.5 and 5. Accordingly, $c(t) = y(t)$ and $\mathcal{C} = [-0.5, 5]$. The RG has the following free parameters: δ; γ; Ψ_μ; Ψ_w; Ψ_y. They will be referred to as RG *design knobs*. The δ and γ knobs affect the resulting number of constraints involved in the on-line optimization. This number, which in general is given by the minimal constraint horizon i_o in (2.53) minus the delay between $c(t)$ and $g(t)$, equals here $i_o - 1$. Fig. 2.2 shows i_o as a function of δ and γ. For small values of δ, which are the ones of practical interest, i_o is larger at intermediate values of γ. Thus, in this respect, it is convenient to restrict γ close either to 1 or 0. Another item that can be affected by the choice of γ is the set of admissible states. For $\delta = 0.05$, this set is depicted in Fig. 2.3 for two candidate values of γ, viz. $\gamma = 0.1$ and 0.9. For intermediate values of γ, the set of admissible states is approximatively comprised within the two depicted sets. The conclusion is that here γ affects the size and the shape of the set only slightly. The value of γ has also an influence on the output transient response, as described in Fig 2.4 to give an example. We select $\gamma = 0.9$ because of the more regular behaviour. Before choosing either $\gamma = 0.9$ or 0.1, we focussed on the remaining tuning knobs. The choice $\Psi_\mu/\Psi_w \leq 0.1$ turned out to be an appropriate one in that, in

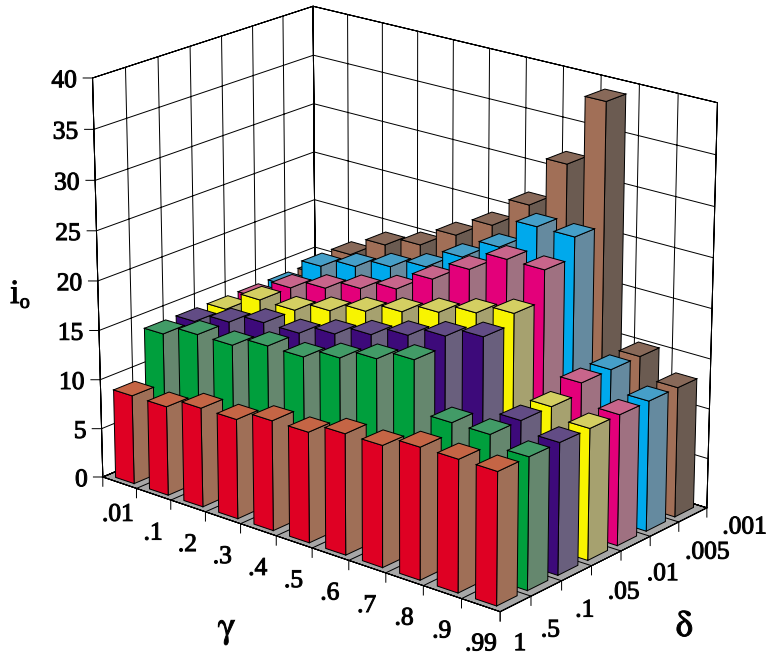


Figure 2.2: Example 1. Minimal constraint horizon i_o computed via (2.53) as a function of γ and δ.

practice, it entails no limitation on the values that $\mu(t)$ can take on. Choosing $\Psi_\mu/\Psi_w = 0.1$, we considered the constrained unit step response as a function of Ψ_y/Ψ_w for both the candidate values of γ. Globally, the shape of this response turned out to suggest the choice $\gamma = 0.9$. Fig. 2.5 depicts the constrained unit step response for $\delta = 0.05$, $\gamma = 0.9$, $\Psi_w = 10$, $\Psi_\mu = 1$, as a function of Ψ_y. As can be expected, a nonzero Ψ_y slows down the response of the overall system. As shown in Fig. 2.5, Ψ_y has the effect of slowing down the dynamical behaviour of the overall system, should the primal system be such to exhibit too fast responses with excessive under/overshoots of the constrained vector. This can be explained by considering the quadratic selection index (2.22). In fact, the third term in (2.22) induces the system to track w as more promptly as large is Ψ_y, thus exalting the undesired transient effects of the constrained vector (in our example, the output undershoot). For this reason, the RG is forced to slow down the evolution of $w(t)$ toward $r(t)$, despite the opposite action induced by the first term weighted by Psi_w. As a result, the more the ratio Ψ_y/Ψ_w is large the more the system is slowed down. Notice that for $\Psi_y = 0$ the dynamics of the system is unchanged when the constraints are inactive, because the RG sets $g(t) = r(t)$. Taking into account the foregoing

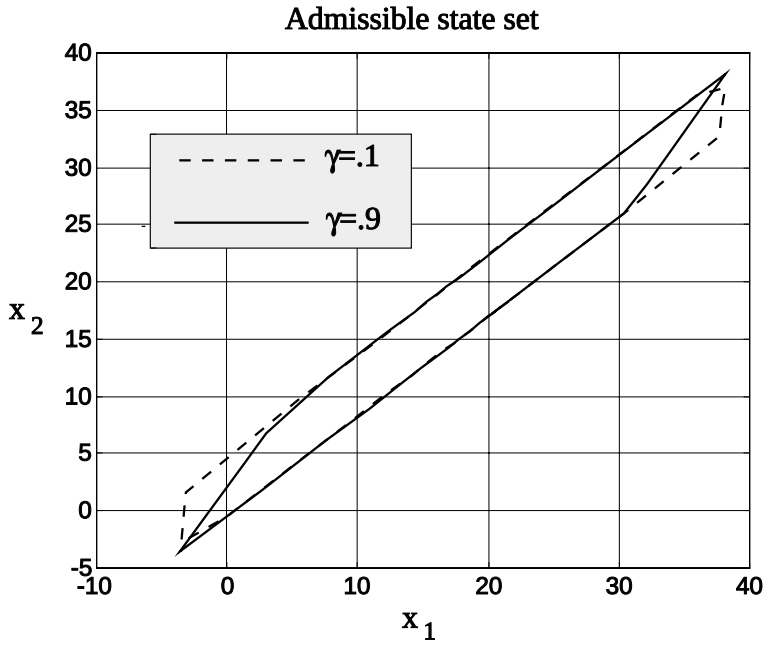


Figure 2.3: Example 1. Set of admissible states x for different values of γ ($\delta = 0.05$).

simulation analysis, the final selected tuned knobs were $\delta = 0.05$, $\gamma = 0.9$, $\Psi_\mu = 1$, $\Psi_w = 10$, $\Psi_y = 0.1$. Algorithm (2.53) was executed on the above machine and took 3.8s to give $i_o = 14$. The related constrained unit step response is shown in Fig. 2.1a (thick line). This was computed in $0.15s$ per time step. Fig. 2.1b depicts the generated reference trajectory $g(t)$ (thick line), the reference trajectory $r(t)$ (thick dashed line), and minimizing parameters $\mu(t)$ (thin line) and $w(t)$ (thin dashed line).

In order to consider the effects of model uncertainties, the same RG as the one designed for the nominal plant (2.62) was used with the plant

$$y(t) = \frac{-1.2517z + 1.4352}{z^2 - 1.4657z + 0.6492}u(t) \tag{2.63}$$

It has been noticed that the RG better behaves when supplied by an estimate $\hat{x}(t)$ rather than the actual $x(t)$, being the estimate carried out from the output $y(t)$ with an observer based on the model itself. Fig. 2.1a exhibits the related output response (thick dashed line). The prescribed lower bound is slightly violated.

Example 2 The RG is applied to the AFTI-16 aircraft modelled in continuous-

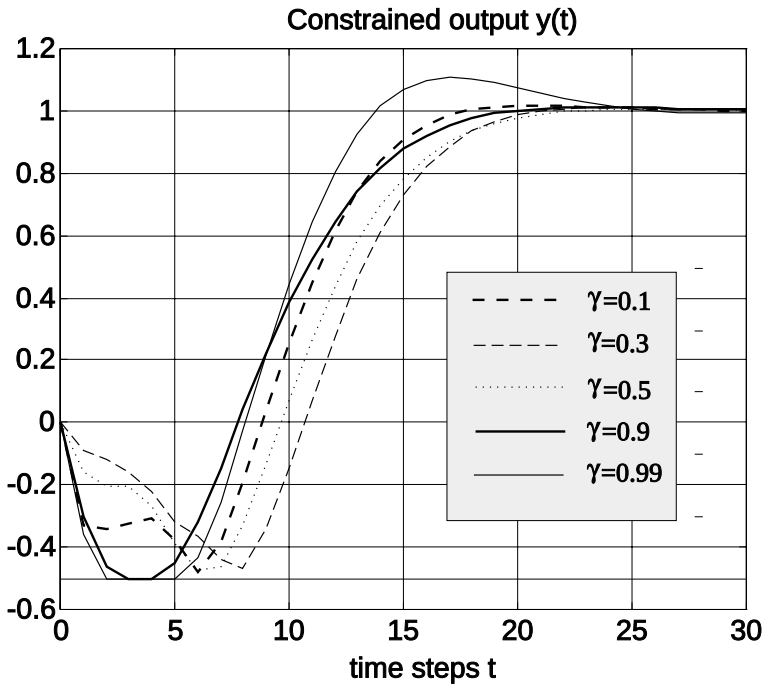


Figure 2.4: Example 1. Constrained variable (output) response for different values of parameter γ.

time as in [KAS90]

$$\begin{cases} \dot{x}(\tau) &= \begin{bmatrix} -.0151 & -60.5651 & 0 & -32.174 \\ -.0001 & -1.3411 & .9929 & 0 \\ .00018 & 43.2541 & -.86939 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} x(\tau) + \begin{bmatrix} -2.516 & -13.136 \\ -.1689 & -.2514 \\ -17.251 & -1.5766 \\ 0 & 0 \end{bmatrix} u(\tau) \\ y(\tau) &= \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} x(\tau) \end{cases} \tag{2.64}$$

The elevator and the flaperon angles are the inputs u to the plant. They are subject to the physical constraints $|u_i| \leq 25^o$, $i = 1, 2$. Then, $c = u$. The attack and the pitch angles are the outputs y. The task is to get zero offset for piecewise-constant references, while avoiding input saturations. The continuous-time model in [KAS90] is sampled every $T_s = .05s$ and a zero-order hold is used at the input. The following linear compensator

$$u(t) = \begin{bmatrix} 0.00005 & 1.25601 & -0.17872 & 0.55620 \\ -0.00043 & 13.71101 & 4.06960 & -0.37350 \end{bmatrix} x(t) + \begin{bmatrix} 1.93476 & -0.55618 \\ -21.18923 & 0.37351 \end{bmatrix} g(t) \tag{2.65}$$

was designed, with no concern of the constraints, so as to obtain both adequate

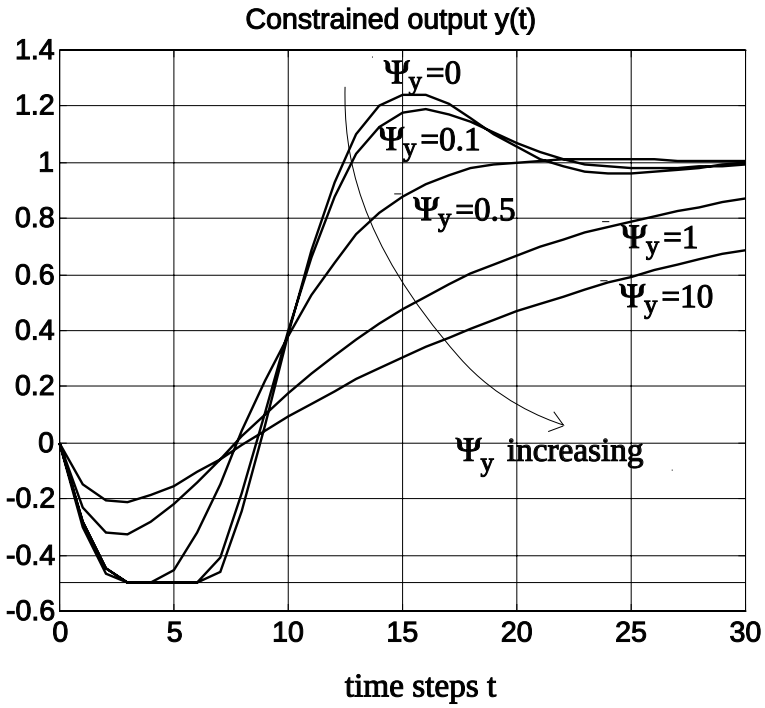


Figure 2.5: Example 1. Constrained variable (output) response for different values of parameter Ψ_y.

dynamic decoupling and fast transient response. Fig. 2.6 shows the response of the compensated linear system with no RG. Note that the constraints are violated. This is a general design rule: the compensator should make the system excessively fast and produce inadmissible values, since then the RG, when required, will slow down the loop. It can be shown that, if the linear compensator outputs given by (2.65) are saturated so as to fulfill the constraints, the system becomes unstable. Fig. 2.8 and Fig. 2.9 depict the trajectories resulting when the RG is activated so as to constrain the two plant inputs within the prescribed bounds. To this end, after some simulation analysis, we tuned the RG design knobs as follows: $\gamma = 0.9$; $\delta = 0.1$; $\Psi_\mu = 10I_2$; $\Psi_w = I_2$; $\Psi_y = 0$, the last choice was made in order to leave unchanged the dynamics with inactive constraints. Under these choices, algorithm (2.53) finds $i_o = 140$. Simulations were carried out in $55s$ using standard QP optimization routine with a computational time of 0.91s per step. Heuristically, it was found that, for the reference in Fig. 2.6, indistinguishable results can be obtained with a constraint horizon equal to 5 in 0.13s per time step. Though these computational times exceed the sampling interval T_s, the simulation results

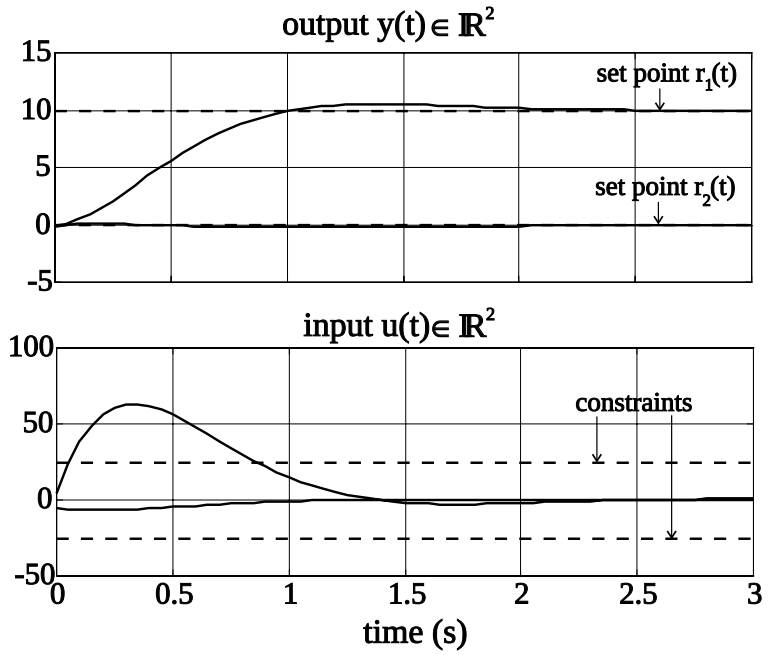


Figure 2.6: Example 2. Compensated AFTI-16 response with no RG.

indicate the performance achievable by using faster processors with software specifically optimized for the application at hand. (indistinguishable results were also obtained in $8s$ by reducing the constraint horizon to 5 steps). Because of vector optimization, the reference is filtered both in modulus and direction. This explains the coupling between the two outputs. In order to let the direction of $g(t)$ be as close as possible to that of $r(t)$, Ψ_μ and Ψ_w were modified by penalizing at each time t also the component of $g(t)$ orthogonal to $r(t)$. This is accomplished by adding to Ψ_μ and Ψ_w the weighting matrix

$$100(I - \frac{r(t)r'(t)}{r'(t)r(t)}) = \begin{bmatrix} 100 & 0 \\ 0 & 0 \end{bmatrix} \tag{2.66}$$

This modification does not affect the analysis in Sect. 2.2 where $r(t)$ is assumed to become constant. The trajectories related to the modification (2.66), as depicted in Fig. 2.10, exhibit a reduced crosscoupling at the cost of longer settling times. Fig. 2.11 shows the performance of the system with the same RG when the reference exhibits time-variations in such a way that transients take place also from non-equilibrium states, if by an equilibrium state we mean a vector x_w, $w \in \mathcal{W}_\delta$, as in (2.4).

Finally, the behaviour of the reference governor in the presence of an output zero-mean white Gaussian sensor noise with covariance $0.2I_2$ was simulated

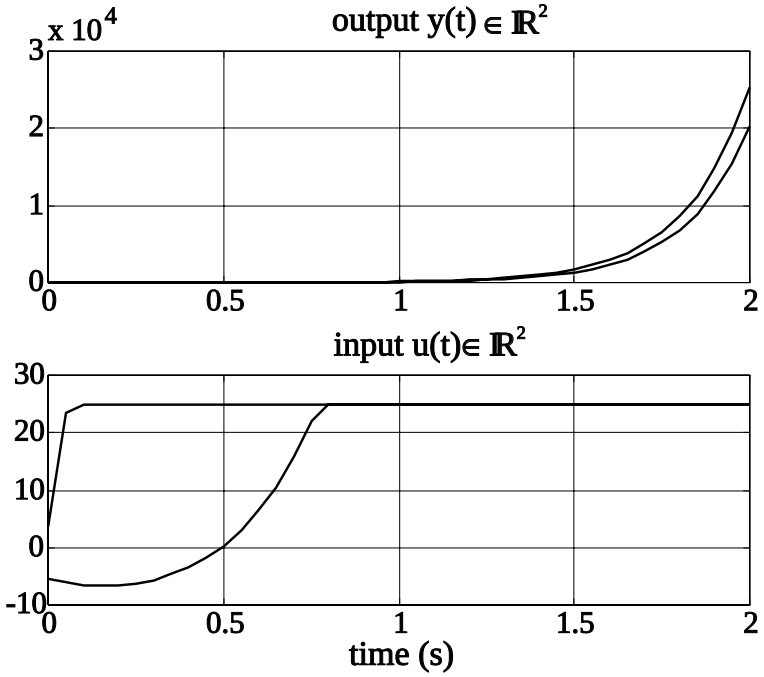


Figure 2.7: Example 2. Linear controller with saturating actuators.

under the same GC knob choice as in Fig. 2.10. This behaviour is depicted in Fig. 2.12. In this case, the state $x(t)$ used by the RG is replaced by an estimate $\hat{x}(t)$ provided by the state Kalman observer. The state $x(t)$ is estimated from the output by means of a Kalman filter

$$\hat{x}(t+1) = \Phi\hat{x}(t) + Gg(t) + L[y(t) - H_y\hat{x}(t)] \tag{2.67}$$

with

$$L = \begin{bmatrix} -2.2613 & -3.6758 \\ 0.1696 & 0.2060 \\ 1.1640 & 1.4113 \\ 0.2060 & 0.2574 \end{bmatrix} \tag{2.68}$$

A Gaussian white output sensor noise with covariance $0.2I_2$ is present. In this case, constraint fulfilment is still longer guaranteed by the RG as both the prediction inherent in the minimization and the feedback are performed by the same estimated state vector. This is not true when the constrained vector $c(t)$ is intrinsically dependent on $x(t)$, see e.g. the plant output. In this case constraint fulfilment is no longer guaranteed. Notice that the constraints can be still fulfilled.

By performing the scalar parameterization (2.57)-(2.58), we obtain the

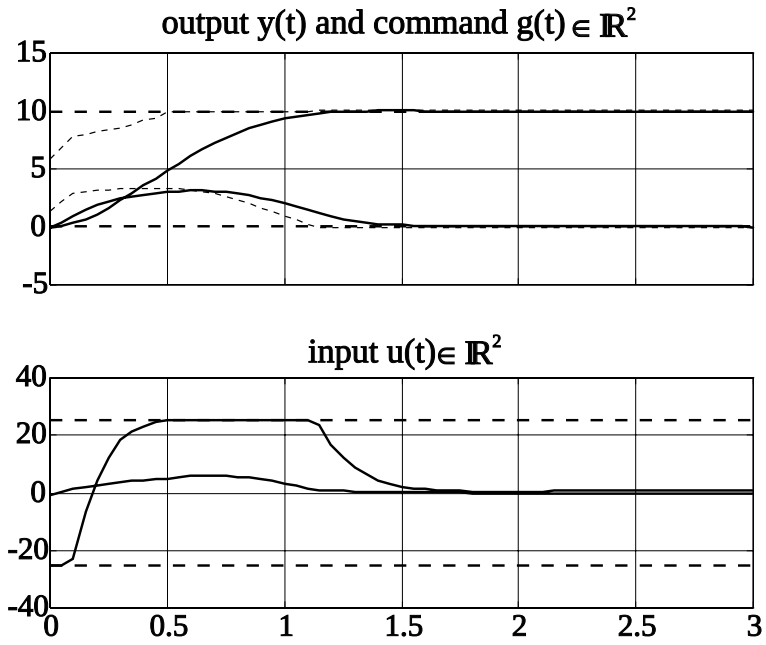


Figure 2.8: Example 2. Response with the RG: output $y(t)$ (solid line), reference $g(t)$ (thin dashed line), and reference $r(t)$ (thick dashed line).

trajectories depicted in Fig. 2.12. It is clear that if the tracking performance is mesaured in terms of settling times, this scalar reduction exhibits a poorer performance.

Finally, we summarize the effects of the RG design knobs in Table 2.2.

Table 2.2: RG design knobs (— means no effect)

Knob	*Performance*	*Steady-state*	*Complexity*	*Suggested*
δ	—	If $r \notin W_\delta$	Affects i_o	$\mathcal{C} \approx \mathcal{C}_\delta$
γ	y-shape, if $\Psi_y = 0$	—	Affects i_o	$\gamma \approx 1$
Ψ_y/Ψ_w	The larger, the slower y	—	—	$\Psi_y \approx 0$
Ψ_μ/Ψ_w	Yes, but no general rule	—	—	$\leq 10\%$

2.8 Concluding Remarks

The RG problem, viz. the one of on-line designing a reference input in such a way that a primal compensated linear control system can operate in a stable way with satisfactory tracking performance and no constraint violation, has been addressed by exploiting some ideas originating from predictive control. In

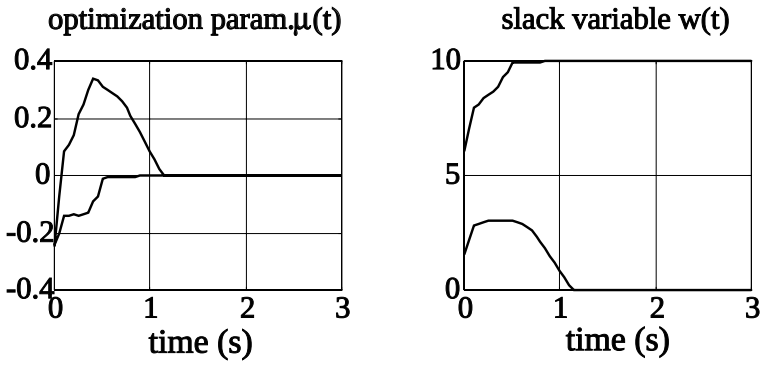


Figure 2.9: Example 2. Response with the RG: minimization parameters $\mu(t)$ and $w(t)$

this connection, the concept of a "virtual" reference sequence is instrumental to synthesize a RG having the stated properties along with a moderate computational burden. This is achieved by: first, linearly parameterizing the virtual reference sequences by a vector of twice the dimension of the reference, and defining the functional form of the sequence so as to ease stability analysis; second, choosing at each sampling time the free parameter vector as the one minimizing a constrained quadratic selection index.

It has been shown how to use off-line an iterative algorithm so as to restrict to a fixed finite integer the infinite number of time-instants over which the prescribed constraints must be checked in order to decide admissibility of virtual reference sequences. A stability analysis based on a Lyapunov function argument shows that, if the reference becomes constant, the system output asymptotically converges to the closest admissible approximation to the reference. Simulations have shown the effectiveness of the RG when applied to systems with input and/or state-related constraints.

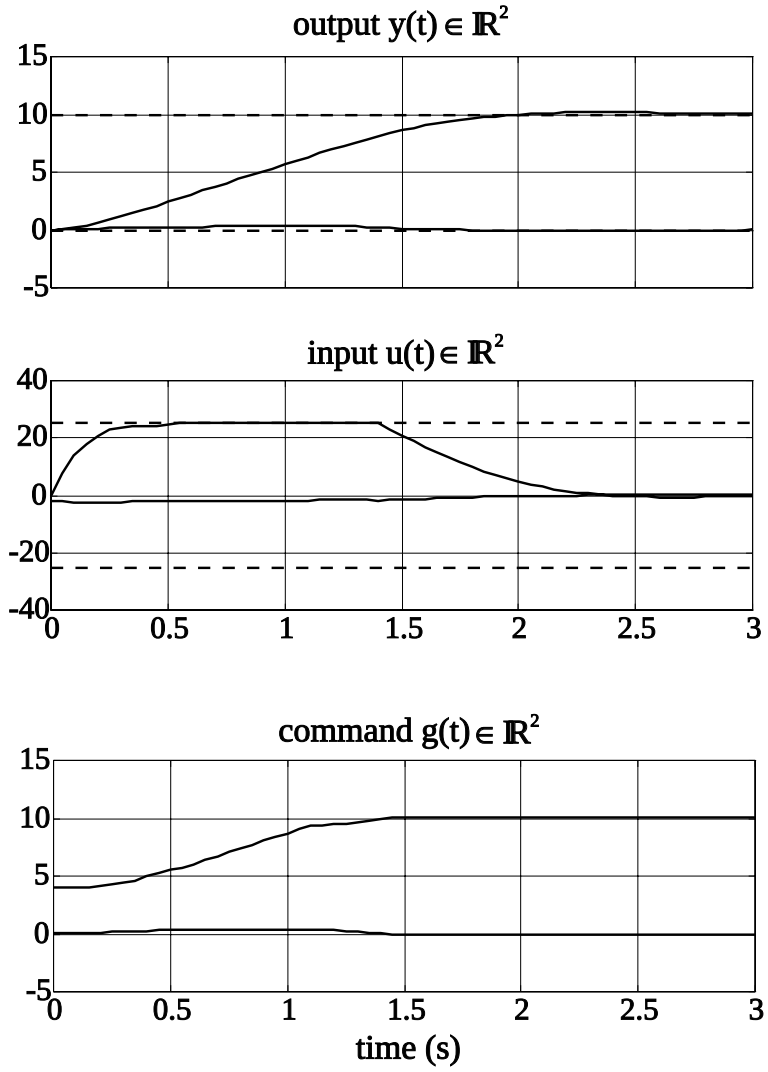


Figure 2.10: Example 2. Response with the RG penalizing the component of μ and w orthogonal to $r(t)$.

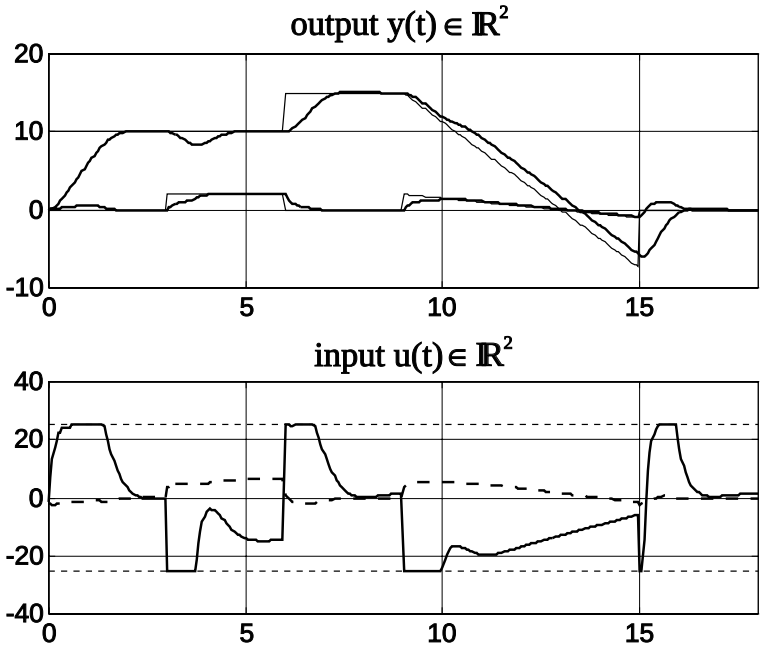


Figure 2.11: Example 2. Output $y(t)$ (thick line) and reference $r(t)$ (thin line); input $u(t)$ (solid and dashed line)

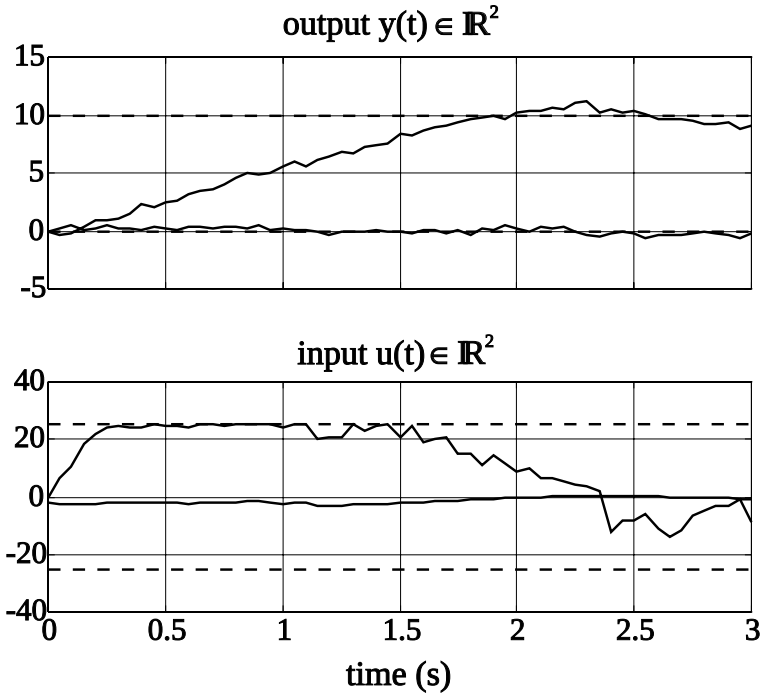


Figure 2.12: Example 2. Response with the RG and output measurement noise.

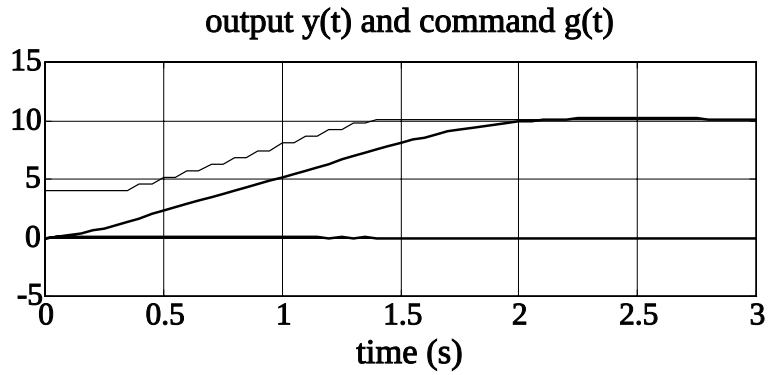


Figure 2.13: Example 2. Response with a scalar parameterized reference: output $y(t)$ (thick line) and reference $g(t)$ (thin line)

Chapter 3

Uncertain Linear Systems

A method based on conceptual tools of predictive control is described for tackling tracking problems of uncertain linear systems wherein pointwise-in-time input and/or state inequality constraints are present. The method consists of adding to a primal compensated system a nonlinear device called Robust Reference Governor (RRG) which manipulates the desired reference in order to fulfill the prescribed constraints. Provided that an admissibility condition on the initial state is satisfied, the control scheme is proved to fulfill the constraints, as well as stability and set-point tracking requirements, for all systems whose impulse/step responses lie within given uncertainty ranges.

3.1 Introduction

Most of the work in the field of feedback control of dynamic systems subject to constraints—in particular input saturation—has addressed the constrained regulation problem under the hypothesis that the plant model is exactly known. The main goal of the present chapter is to address the constrained tracking problem for systems affected by *model uncertainties*, by using tools from *predictive control*.

Other predictive control approaches [KBM96] try to solve robust constraint fulfillment and stabilization, with a consequent heavy computational burden. In this chapter, the constraint fulfilment problem is separated from stability, set-point tracking, and disturbance attenuation requirements, which in turn—in the absence of constraints—are supposed to be taken care of by a formerly designed robust compensator. The aim of the present chapter is to lay down

guidelines for synthesizing *Robust Reference Governors (RRG)* for systems whose impulse and step response are uncertain in that they are only known to be comprised within given interval ranges. Whenever necessary, the filter alters on line the input to the primal control system so as to avoid constraint violation, according to a worst-case criterion.

This chapter is arranged as follows. Sect. 3.2 introduces the problem and presents the RRG algorithm for the class of systems under consideration. A description of the adopted model uncertainty is given in Sect. 3.3. Sect. 3.4 studies how to reduce the infinite number of constraints involved in the problem formulation into a finite number. Stability, tracking and other properties of the PRF are investigated in Sect. 3.5, while Sect. 3.6 is devoted to computational aspects. Finally, a simulative example of application of the PRF, which provides some design guidelines, is described in Sect. 3.7.

3.2 Problem Formulation

Consider a family $\mathcal{S}$ of linear asymptotically stable systems. Each member Σ of $\mathcal{S}$ has a state-space description of the form

$$\Sigma : \left\{ \begin{array}{rcl} x(\tau+1) & = & \Phi x(\tau) + G g(\tau) \\ y(\tau) & = & H x(\tau) + D g(\tau) \\ c(\tau) & = & H_c x(\tau) + D_c g(\tau), \ \Sigma \in \mathcal{S} \end{array} \right. \tag{3.1}$$

where: $\tau \in \mathbb{Z}_+ \triangleq \{0, 1, \ldots\}$; $x(\tau) \in \mathbb{R}^{n_\Sigma}$ is the state vector; $g(\tau) \in \mathbb{R}^p$ the command input, which in the absence of constraints would coincide with the desired output reference $r(\tau)$; $y(\tau) \in \mathbb{R}^p$ the output which is required to track $r(\tau)$; and $c(\tau) \in \mathbb{R}^q$ the vector to be constrained within a given set $\mathcal{C}$, which satisfies the following

Assumption 3.1 *$\mathcal{C}$ is a convex polyhedron with nonempty interior.*

Without loss of generality, we assume that $\mathcal{C}$ has the form

$$\mathcal{C} = \{c \in \mathbb{R}^q : \ c \leq B_c\} \tag{3.2}$$

In fact, a generic polyhedron described by inequalities of the form $A_c c \leq B_c$ can be rewritten in the form (3.2) by defining a new vector $c^* = A_c c$ and, accordingly, new matrices $\hat{H}_c^* = A_c \hat{H}_c$, $\hat{D}_c^* = A_c \hat{D}_c$. Typically, (3.1) consists

of an uncertain linear system under robustly stabilizing control. We take into account model uncertainties by assuming the set-membership inclusion $\Sigma \in \mathcal{S}$, where $\mathcal{S}$ will be characterized in Sect. 3.3. Furthermore, inside the $\mathcal{S}$, we choose a particular $\hat{\Sigma}$ called *nominal* system

$$\hat{\Sigma}: \left\{ \begin{array}{rcl} \hat{x}(\tau+1) & = & \hat{\Phi}\hat{x}(\tau) + \hat{G}g(\tau) \\ c(\tau) & = & \hat{H}_c\hat{x}(\tau) + \hat{D}_c g(\tau) \end{array} \right. \tag{3.3}$$

with $\hat{x} \in \mathbb{R}^n$.

The aim of this chapter is to design a *robust reference governor* (RRG), a device finalized to transform the desired reference $r(\tau)$ to the command vector $g(\tau)$ so as to possibly enforce the prescribed constraints $c(\tau) \in \mathcal{C}$, $\tau \in \mathbb{Z}_+$, for all possible systems in $\mathcal{S}$, and make the tracking error $y(\tau) - r(\tau)$ small. The filtering action operates in a predictive manner: at time τ a *virtual* command sequence $\{g(\tau), g(\tau+1), \ldots\}$ is selected in such a way that, for all systems $\Sigma \in \mathcal{S}$, the corresponding predicted c-evolution lies within $\mathcal{C}$. Then, according to a *receding horizon* strategy, only the first sample of the virtual sequence is applied at time τ, a new virtual command sequence being recomputed at time $\tau+1$. Several criteria [BCM97], [BM94b], [GKT95] can be used to select the class of virtual commands. For reasons that will be clearer soon, we restrict our attention to the class of constant command sequences introduced in [GKT95]. This class is parameterized by the scalar β, and each of its members defined by

$$g(t+\tau|\tau,\beta) \triangleq g(\tau-1) + \beta[r(\tau) - g(\tau-1)], \ \forall t \in \mathbb{Z}_+. \tag{3.4}$$

At each time τ, the free parameter β is selected by the RRG via the optimization criterion

$$\beta(\tau) \triangleq \left\{ \begin{array}{l} \arg\max\limits_{\beta\in[0,1]} \beta \\ \text{subject to} \left\{ \begin{array}{l} c(t+\tau|\tau,\Sigma,x(\tau),\beta) \in \mathcal{C}, \\ \forall t \geq 0, \forall \Sigma \in \mathcal{S} \end{array} \right. \end{array} \right. \tag{3.5}$$

where $c(t+\tau|\tau,\Sigma,x(\tau),\beta)$ denotes the predicted c-evolution at time $t+\tau$ which results by applying the constant input $g(t+\tau|\tau,\beta)$ to Σ from state $x(\tau)$ at time τ onward. Then, according to the receding horizon strategy, at each time τ the RRG selects

$$g(\tau) = g(\tau|\tau,\beta(\tau))$$

Notice that requiring $\beta(\tau)$ as close as possible to 1 corresponds to minimizing $\|g(\tau) - r(\tau)\|$, and, consequently, at least in steady-state when the reference become constant, the norm of the tracking error $\|y(\tau) - r(\tau)\|$. A scalar β, or a constant command $g \in \mathbb{R}^p$ satisfying the constraints in (3.5) will be referred to as *admissible* at time τ.

Assumption 3.2 (Feasible Initial Condition) *There exists a vector* $g(-1) \in \mathbb{R}^p$ *such that at time* $\tau = 0$ *the virtual command* $g(t|0,0) = g(-1)$, $\forall t \in \mathbb{Z}_+$, *is admissible.*

For instance, Assumption 3.2 is satisfied for $x(0) = (I - \Phi)^{-1}Gg(-1)$, $H_c x(0) + D_c g(-1) \in \mathcal{C}$. Assumption 3.2, the particular structure of (3.4), and (3.5) ensure that $\beta = 0$ is admissible, and therefore the optimization problem (3.5) admits feasible solutions, at each time $\tau \in \mathbb{Z}_+$.

3.3 Model Uncertainty Description

Uncertainty of dynamic systems models can be described in various ways. In the case at hand, frequency domain descriptions are not convenient because of the time-domain RRG design logic (3.4)-(3.5). Furthermore, if uncertainties involving state-space realizations are adopted, the effect of matrix perturbations on the predicted evolutions become cumbersome to compute. Consider for instance a free response of the form $(\hat{\Phi} + \tilde{\Phi})^t x(0)$: this gives rise to prediction perturbations which are nonlinear in the uncertain parameter $\tilde{\Phi}$. On the contrary, uncertainties on the step-response or impulse-response samples provide a practical description in many applications, as they can be easily determined by experimental tests, and allow a reasonably simple way to compute predictions. Seemingly, step-response and impulse-response are equivalent, and one could be tempted to use either one or the other without distinction to describe model uncertainties. However, when used individually, both exhibit drawbacks. To show this, consider Fig. 3.1, which depicts perturbations expressed only in terms of the impulse response. The resulting step-response uncertainty turns out to be very large as $t \to \infty$. However, this is not the case when each Σ, for instance, contains an integrator in the feedback loop, which yields a unity DC-gain, and consequently vanishing step-response perturbations as $t \to \infty$. Conversely, as depicted in Fig. 3.2,

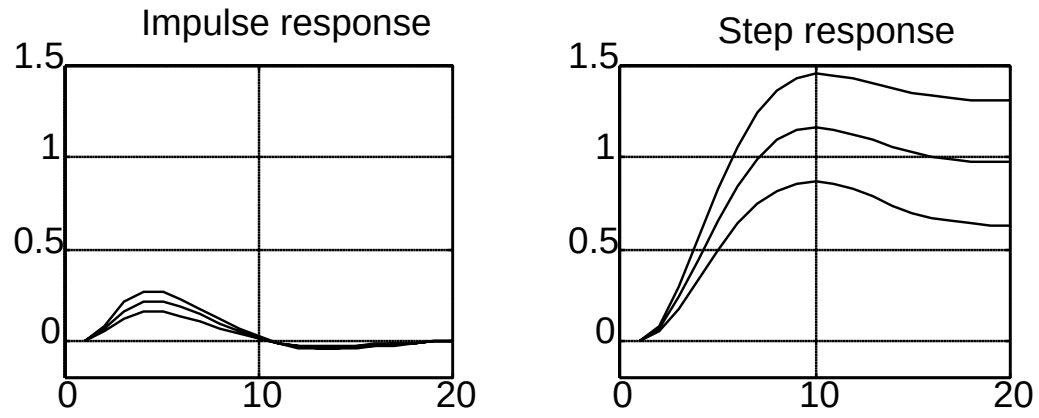


Figure 3.1: Step-response interval ranges (right) arising from an impulse-response description (left).

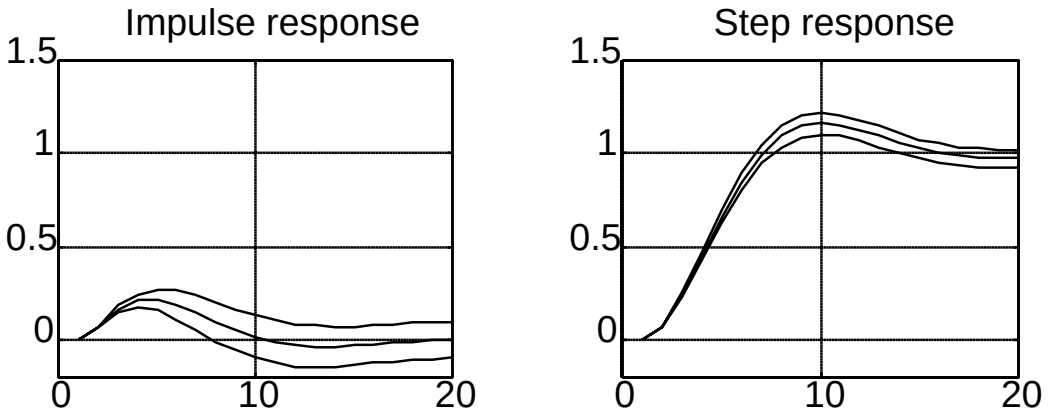


Figure 3.2: Impulse-response interval ranges (left) arising from a step-response description (right).

uncertainty expressed only in terms of the step response could lead to nonzero impulse-response samples at large values of t, for instance when the DC-gain from g to c is uncertain; therefore, any a priori information about asymptotic stability properties would be wasted. In order to minimize the conservatism of the approach (3.5), it is clear that the set Σ should be as small as possible. For this reason, in this chapter we will jointly consider both step-response and impulse-response in order to describe model uncertainty.

Assumption 3.3 *Let $\Sigma \in \mathcal{S}$ and H_t^{ij} be the (i,j)-entry of the impulse response H_t from g to c. Then, there exist a matrix $M \in \mathbb{R}^{q\times p}$ and a scalar λ,*

$0 \leq \lambda < 1$, *such that, for all systems* $\Sigma \in \mathcal{S}$,

$$|H_t^{ij}| \leq M^{ij}\lambda^t,\ t \in \mathbb{Z}_+,\ \forall i = 1, \ldots, q,\ \forall j = 1, \ldots, p \tag{3.6}$$

Notice that, although in (3.1) we are considering asymptotically stable systems, condition (3.6) characterizes only stability properties of the subspace which is observable from c.

The impulse response H_t can be expressed as the sum of a nominal impulse response

$$\hat{H}_t \triangleq \begin{cases} \hat{H}_c\hat{\Phi}^t\hat{G} & \text{if} \quad t > 0 \\ \hat{D}_c & \text{if} \quad t = 0 \end{cases}$$

and an additive perturbation $\tilde{H}_t$. We describe the range intervals of $\tilde{H}_t$ as

$$\begin{cases} \tilde{H}_t^{ij} \in [H_t^{-ij}, H_t^{+ij}] & \text{if} \quad t = 0, 1, \ldots, N-1 \\ |\tilde{H}_t^{ij}| \leq E^{ij}\lambda^t & \text{if} \quad t \geq N \end{cases} \tag{3.7}$$

where $E \in \mathbb{R}^{q \times p}$, N is a fixed integer, and $i = 1, \ldots, q$, $j = 1, \ldots, p$. In the same way, the step-response from g to c can be expressed as the sum of a nominal response

$$\hat{W}_t \triangleq \sum_{k=0}^{t-1} \hat{H}_c\hat{\Phi}^k\hat{G} + \hat{D}_c$$

and an additive perturbation $\tilde{W}_t$

$$\begin{cases} \tilde{W}_t^{ij} \in [W_t^{-ij}, W_t^{+ij}] & \text{if} \quad t = 0, 1, \ldots, N-1 \\ |\tilde{W}_t^{ij} - \tilde{W}_{t-1}^{ij}| \leq E^{ij}\lambda^t & \text{if} \quad t \geq N. \end{cases} \tag{3.8}$$

3.4 Reduction to a Finite Number of Constraints

Since the RRG operates over a semi-infinite prediction horizon, the optimization criterion in (3.5) involves an infinite number of constraints. In order to effectively solve (3.5), we need to reduce this infinite number to a finite one. This will be achieved by borrowing techniques presented in [GT91], as follows. Under some assumptions on the desired reference r and the past command inputs $g(-\tau)$ applied before the RRG was switched on at time $\tau = 0$,

Lemma 3.1 will show that the command sequence $g(\tau)$ generated by the RRG is bounded. A new constraint on β will be introduced, which ensures that in steady-state the predicted c-evolution stays away from the border of $\mathcal{C}$. Then, Theorem 3.1 will prove the existence of a finite *constraint horizon*, the shortest prediction interval over which constraints must be checked in order to assert admissibility of a given virtual command sequence. Finally, an algorithm to find such a constraint horizon will be provided.

Assumption 3.4 (Set-Point Conditioning) *The reference signal $r(\cdot)$ satisfies $r(\tau) \in \mathcal{R}$ for all $\tau \in \mathbb{Z}_+$, where $\mathcal{R}$ is compact and convex.*

This amounts to assuming that either the class of references to be tracked is bounded, or a clamping device is inserted in the RRG mechanism. In practice, this is not a restriction since bounds on the reference are often dictated by the physical application.

Assumption 3.5 *For all $\tau > 0$, $g(-\tau) \in \mathcal{R}$.*

Lemma 3.1 *Provided that Assumptions 3.4 and 3.5 are satisfied, $g(\tau) \in \mathcal{R}$, $\forall \tau \in \mathbb{Z}_+$.*

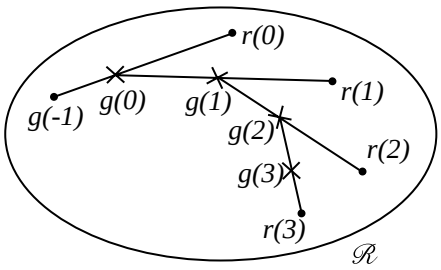


Figure 3.3: Reference set.

Proof. By (3.5), $\beta(\tau) \in [0, 1]$. Then, as depicted in Fig 3.3, $g(\tau)$ lies on the segment whose vertices are $g(\tau - 1)$, $r(\tau)$. By convexity of $\mathcal{R}$, the result straghtforwardly follows by induction. □

In order to proceed further, we impose an additional constraint on the optimization (3.5). By (3.7), we can define $\tilde{W}_\infty \triangleq \lim_{t\to\infty} \tilde{W}_t$, along with the relation

$$\tilde{W}_\infty \in [W_\infty^-, W_\infty^+], \tag{3.9}$$

wherein $W_\infty^- \triangleq W_{N-1}^- - \frac{\lambda^N}{1-\lambda}E$, $W_\infty^+ \triangleq W_{N-1}^+ + \frac{\lambda^N}{1-\lambda}E$, and, by Assumption 3.3, $\hat{W}_\infty \triangleq \lim_{t\to\infty} \hat{W}_t$.

Notice that (3.9) is not over-conservative in that there exist systems $\Sigma \in \mathcal{S}$ for which $\tilde{W}_\infty = W_\infty^-$ or W_∞^+. For an arbitrarily small $\delta > 0$, consider the following set

$$\begin{aligned} \mathcal{G}_\delta \;=\; \Big\{ & g \in \mathcal{R} :\; (\hat{W}_\infty + \tilde{W}_\infty) g \le B_c - \underline{\delta} \\ & \forall \tilde{W}_\infty \in [W_\infty^-, W_\infty^+],\; \underline{\delta} = \begin{bmatrix} \delta \\ \vdots \\ \delta \end{bmatrix} \in \mathbb{R}^q \Big\} \end{aligned} \tag{3.10}$$

For all constant command inputs $g \in \mathcal{G}_\delta$, the corresponding steady-state constrained vector $c_g \triangleq W_\infty g$ is located in $\mathcal{C}$ at a distance from the border greater than or equal to a fixed quantity, which depends on δ. The constraint $g \in \mathcal{G}_\delta$ adds the following q additional constraints

$$(\hat{W}_\infty + \tilde{W}_\infty) g \le B_c - \underline{\delta},\; \forall \tilde{W}_\infty \in [W_\infty^-, W_\infty^+]. \tag{3.11}$$

Hereafter, the constraints (3.11) will be added in (3.5) to determine $\beta(\tau)$.

In order to simplify the notation, in the next theorem we consider $\tau = 0$. Moreover, we replace $x(0)$ with the sequence of past commands $\mathcal{X}_- \triangleq \{g(k)\}_{k=-1}^{\infty}$, and define $\mathcal{R}^* \triangleq \{\mathcal{X}_- :\; \mathcal{X}_- \subseteq \mathcal{R}\}$. Accordingly, $c(t + 0|0, \Sigma, x(0), g)$ is denoted by $c(t, \Sigma, \mathcal{X}_-, g)$, or, when the remaining arguments will be clear from the context, by $c(t)$.

Theorem 3.1 *Suppose that Assumptions 3.4 and 3.5 hold. Then there exists a finite time t^* such that*

$$\begin{aligned} c(t) \in \mathcal{C},\; \forall t \le t^* \;\Leftrightarrow\; c(t) \in \mathcal{C},\; \forall t \in \mathbb{Z}_+ \\ \forall \Sigma \in \mathcal{S},\; \forall \mathcal{X}_- \in \mathcal{R}^*,\; \forall g \in \mathcal{R}, \end{aligned} \tag{3.12}$$

Proof. Consider the c-evolution at a generic time $t \ge N$ to commands $g(-t) \in \mathcal{R}$, $g(t) \equiv g \in \mathcal{R}$, $\forall t \in \mathbb{Z}_+$,

$$c(t) = \sum_{k=-\infty}^{-1} H_{t-k} g(k) + \hat{W}_\infty g + \tilde{W}_\infty g - \sum_{k=t+1}^{\infty} H_t g$$

Since $\mathcal{R}$ is bounded, all references $r \in \mathcal{R}$ satisfy inequalities of the form $|r^j| \leq \bar{r}^j$, $\forall j = 1, \ldots, p$, where r^j denotes the j-th component of r. Therefore, for $t \geq N-1$,

$$|c^i(t) - [\hat{W}_\infty + \tilde{W}_\infty g]^i| \leq 2\frac{\lambda^{t+1}}{1-\lambda}\sum_{j=1}^{p} M^{ij}\bar{r}^j$$

By setting

$$t_i = \frac{\log[\frac{\delta}{2}(1-\lambda)\sum_{j=1}^{p} M^{ij}\bar{r}^j]}{\log\lambda} - 1 \tag{3.13}$$

and

$$t^* = \max_{i=1,\ldots,q} t_i \tag{3.14}$$

one has

$$|c^i(t) - \hat{c}_g^i - [\tilde{W}_\infty g]^i| \leq \delta,\ \forall t \geq t^*,\ \forall i = 1, \ldots, q \tag{3.15}$$

Since $g \in \mathcal{G}_\delta$, it follows that $c(t) \in \mathcal{C}$, $\forall t > t^*$.

□

Definition 3.1 (Constraint Horizon)

$$t_o = \min_{t^*}\{t^* :\ \textit{(3.12) holds}\} \tag{3.16}$$

Theorem 3.1 provides an existence result, as it proves that t_o in (3.16) is well defined. However, in general, t_o is smaller than the quantity t^* estimated by Theorem 3.1 (see (3.13)-(3.14) for details). In order to find out an algorithm which enables us to compute the minimal t_o, define $\mathcal{Q}_t \triangleq \{[\Sigma, \mathcal{X}_-, g] \in \mathcal{S}\times\mathcal{R}^*\times\mathcal{R} : c(k, \Sigma, \mathcal{X}_-, g) \leq B_c, \forall k = 0, \ldots, t, \lim_{j\to\infty} c(j, \Sigma, \mathcal{X}_-, g) \leq B_c - \underline{\delta}\}$. where, clearly, $\mathcal{Q}_{t+k} \subseteq \mathcal{Q}_t$.

Theorem 3.2 *Suppose that Assumptions 3.4 and 3.5 hold. Then,*

$$\mathcal{Q}_t \subseteq \mathcal{Q}_{t+1} \ \Rightarrow\ \mathcal{Q}_{t+k} = \mathcal{Q}_t,\ \forall k \geq 0 \tag{3.17}$$

and

$$t_o = \min_t\{t :\ \mathcal{Q}_t = \mathcal{Q}_{t+1}\}$$

Proof. Let t such that (3.17) holds, and consider a generic $[\Sigma, \mathcal{X}_-, g] \in \mathcal{Q}_{t+1}$. By shifting $\mathcal{X}_- = \{g(-1), g(-2), \ldots\}$ in the new sequence $\mathcal{X}_-^* \triangleq \{g, g(-1), g(-2), \ldots\}$, it follows that $c(t+1, \Sigma, \mathcal{X}_-, g) = c(t, \Sigma, \mathcal{X}_-^*, g)$

Moreover, by Lemma 3.1, $\mathcal{X}_-^* \in \mathcal{R}^*$. Then, $[\Sigma, \mathcal{X}_-^*, g] \in \mathcal{Q}_t$, and being $\mathcal{Q}_t \subseteq \mathcal{Q}_{t+1}$, $[\Sigma, \mathcal{X}_-^*, g] \in \mathcal{Q}_{t+1}$, which implies $[\Sigma, \mathcal{X}_-, g] \in \mathcal{Q}_{t+2}$. Hence, $\mathcal{Q}_{t+1} \subseteq \mathcal{Q}_{t+2}$, or $\mathcal{Q}_{t+2} = \mathcal{Q}_{t+1} = \mathcal{Q}_t$. By induction, $\mathcal{Q}_{t+k} = \mathcal{Q}_t$, $\forall k \geq 0$. Let now $t_m \triangleq \min_t\{t : \ \mathcal{Q}_t = \mathcal{Q}_{t+1}\}$. If $[\Sigma, \mathcal{X}_-, g] \in \mathcal{Q}_{t_o}$, then $[\Sigma, \mathcal{X}_-, g] \in \mathcal{Q}_t$, $\forall t \geq t_o$, and hence $t_o \geq t_m$. If, by contradiction, $t_m \leq t_o$, then, by minimality of t_o, there exists $[\Sigma, \mathcal{X}_-, g] \in \mathcal{Q}_{t_m}$ such that $c(t_m + 1, \Sigma, \mathcal{X}_-, g) \notin \mathcal{C}$, which implies $\mathcal{Q}_{t_m} \notin \mathcal{Q}_{t_m+1}$. □

Following an approach similar to the one used in [BCM97] and [GT91], t_o can be determined by the following algorithm:

Algoritm 1 (Determination of t_o)

1. $t \leftarrow -1$
2. $\mathcal{Q}_{-1} \leftarrow \{[\Sigma, \mathcal{X}_-, g] : c(-1, \Sigma, \mathcal{X}_-, g) \leq B_c, \lim_{j\to\infty} c(j, \Sigma, \mathcal{X}_-, g) \leq B_c - \underline{\delta}, \Sigma \in \mathcal{S}, \mathcal{X}_- \in \mathcal{R}^*, g \in \mathcal{R}\}$
3. $m_t^i \leftarrow \max_{[\Sigma, \mathcal{X}_-, g] \in \mathcal{Q}_t} \{c^i(t+1, \Sigma, \mathcal{X}_-, g) - B_c^i\}$, $i = 1, \ldots, q$
4. If $m_t^i \leq 0$, for all $i = 1, \ldots, q$, go to 7
5. $t \leftarrow t + 1$
6. Go to 3
7. $t_o \leftarrow t$
8. Stop

Observe that Algorithm 1 involves optimizations with respect to an infinite dimensional vector which contains $\mathcal{X}_-$ and the impulse (or step) response coefficients of system Σ. However, by virtue of Assumption 3.3 (asymptotic stability), these can be approximated with arbitrary precision by finite dimensional optimizations. In fact, once a precision ϵ_p has been fixed, we can express the evolution of vector c as

$$c(t, \Sigma, \mathcal{X}_-, g) = W_t g + \sum_{k=t+1}^{M} H_k g(t-k) + \sum_{k=M+1}^{\infty} H_k g(t-k) \tag{3.18}$$

with M such that

$$|\sum_{k=M+1}^{\infty}\sum_{j=1}^{p} H_k^{ij} g^i(t-k)| \leq \epsilon_p^i,$$
$$\forall i = 1,\ldots,q,\ \forall \Sigma \in \mathcal{S},\ \forall \mathcal{X}_- \in \mathcal{R}^*$$

This allows one to implement Algorithm 1 by solving quadratic programs with quadratic constraints with respect to $[\{g(-1),\ldots,g(-M)\},\{\tilde{H}_0,\ldots,\tilde{H}_M\},g]$. Notice that M, and, consequently, the complexity of Algorithm 1, is related to the estimate λ in (3.6).

3.5 Main Properties of RRG

We investigate how the RRG affects system stability when the reference to be tracked becomes constant. Since all systems $\Sigma \in \mathcal{S}$ are asymptotically stable, next Lemma 3.2 first guarantees system stability by simply showing that $g(\tau)$ converges to a vector g_∞. By using the viability result of Lemma 3.4, Lemma 3.5 will prove that, amongst the admissible command inputs, g_∞ is the vector which is closest to r. Lemma 3.6 will show that this limit is reached in a finite time. Finally, Theorem 3.3 will summarize the overall properties of the RRG.

Lemma 3.2 *Suppose $r(\tau) = r$ for all $\tau \geq \tau_0$. Then there exists $g_\infty = \lim_{\tau\to\infty} g(\tau)$.*

Proof. If $g(\tau_0) = r$, it follows that $\beta(\tau) = 1$ is admissible for all $\tau \geq \tau_0$, and therefore $g_\infty = r$. Suppose $g(\tau_0) \neq r$. Since, by (3.4), $g(\tau_0 + k)$ lies on a segment whose vertices are $g(\tau_0)$ and r, by setting $d(\tau) \triangleq \|g(\tau) - r\|$ one has $g(\tau) = r + \frac{d(\tau)}{\|g(\tau_0)-r\|}[g(\tau_0) - r]$ and $0 \leq d(\tau) \leq d(\tau - 1)$. Hence, since $d(\tau)$ is monotonically non increasing and lowerbounded, there exists $d_\infty \triangleq \lim_{\tau\to\infty} d(\tau)$, and, as a consequence, $g_\infty = r + \frac{d_\infty}{\|g(\tau_0)-r\|}[g(\tau_0) - r]$. □

Lemma 3.3 *Convexity of $\mathcal{C}$ implies convexity of the set $\mathcal{G}_\delta$ defined in (3.10), $\forall \delta > 0$.*

Proof. Consider δ such that $\mathcal{G}_\delta \neq \emptyset$ ($\mathcal{G}_\delta = \emptyset$ is trivially convex), two set-points $g_0,\ g_1 \in \mathcal{G}_\delta$, and $g_\alpha \triangleq g_0 + \alpha[g_1 - g_0]$, $0 < \alpha < 1$. Define $W_\infty \triangleq \hat{W}_\infty + \tilde{W}_\infty$.

Being $\mathcal{C}$ convex, the set $\mathcal{C}_\delta = \{c \in \mathbb{R}^q : \ c \leq B_c - \underline{\delta}\}$ is convex as well. Since $W_\infty g_\alpha = W_\infty g_0 + \alpha[W_\infty g_1 - W_\infty g_0]$, it follows that $W_\infty g_\alpha \in \mathcal{C}_\delta$, $\forall \Sigma \in \mathcal{S}$. $\square$

We show now that, given an admissible set-point $g_0 \in \mathcal{G}_\delta$ and a new set-point $g_1 \in \mathcal{G}_\delta$, there always exists a finite settling time after which a new admissible set point g_γ is found by moving from g_0 towards g_1. In other words, as $x(\tau)$ approaches the equilibrium state respective to g_0, a new set-point in the direction of g_1 becomes admissible.

Lemma 3.4 (Viability) *Let g_0, $g_1 \in \mathcal{G}_\delta$. At each time τ there exist two positive reals γ and ϵ such that the command $g_\gamma = g_0 + \gamma(g_1 - g_0)$ is admissible for all the past input sequences $\mathcal{X}_- = \{g(\tau-1), g(\tau-2), \ldots\} \in \mathcal{R}^*$ satisfying the condition*

$$|g^i(\tau - t) - g_0^i| \leq \epsilon \lambda^{-t/2}, \ \forall t > 0, \ \forall i = 1, \ldots, p \tag{3.19}$$

where g^i denotes the i-th component of g.

Proof. Without loss of generality, assume $\tau = 0$. Define $\tilde{g}(t) \triangleq g(t) - g_0$ and consider the predicted evolution of vector c obtained by supplying system Σ with the costant command $g(t) = g_\gamma$, $\forall t \in \mathbb{Z}_+$

$$\begin{aligned} c(t) &= \sum_{k=-\infty}^{-1} H_{t-k}[g_0 + \tilde{g}(k)] + \sum_{k=0}^{t} H_{t-k} g_\gamma \\ &= (\hat{W}_\infty + \tilde{W}_\infty)[g_0 + \gamma(g_1 - g_0)] + \\ &\quad \sum_{k=-\infty}^{-1} H_{t-k}[\tilde{g}(k) - \gamma(g_1 - g_0)] \end{aligned} \tag{3.20}$$

By Lemma 3.3, $g_\gamma \in \mathcal{G}_\delta$. In order to prove that g_γ is admissible, we must show that $c(t) \in \mathcal{C}$, $\forall \Sigma \in \mathcal{S}$, and $\forall t \geq 0$, or, equivalently, by (3.20), $\sum_{k=-\infty}^{-1} H_{t-k}[\tilde{g}(k) - \gamma(g_1 - g_0)] \leq \underline{\delta}$, $\forall t \geq 0$. Then,

$$\sum_{k=-\infty}^{-1} H_{t-k}[\tilde{g}(k) - \gamma(g_1 - g_0)] = \sum_{k=t+1}^{N-1} H_k \tilde{g}(k) + \sum_{k=\max\{N,t+1\}}^{\infty} H_k \tilde{g}(k) + \gamma \sum_{k=t+1}^{\infty} H_k (g_1 - g_0)$$

where the first sum in the second term is equal to zero for $t \geq N-1$. By letting

$$\begin{aligned}
\alpha_1^j &\triangleq \begin{cases} \max\limits_{\substack{i=1,\dots,q \\ t=0,\dots,N-2}} \left\{ \begin{array}{l} |\hat{W}_{N-1}^i - \hat{W}_t| + \\ \max\{|W_{N-1}^{+ij} - W_t^{-ij}|, \\ |W_{N-1}^{-ij} - W_t^{+ij}|\} \end{array} \right\}, \\ \qquad\qquad \text{if } 0 \leq t \leq N-2 \\ 0, \quad \text{if } t \geq N-1 \end{cases} \\
\alpha_2^j &\triangleq \max_{i=1,\dots,q} \frac{\lambda^{N/2}}{1-\lambda} E^{ij}
\end{aligned}$$

it follows that

$$c(t) \leq B_c - \underline{\delta} + \sum_{h=1}^{2} \sum_{j=1}^{p} \alpha_h^j [\epsilon + (g_1^j - g_0^j)\gamma] \leq B_c$$

for

$$\begin{cases} \epsilon \leq \dfrac{\delta}{2 \displaystyle\sum_{h=1}^{2} \sum_{j=1}^{p} \alpha_h^j} \\ \gamma \leq \dfrac{\delta}{2 \displaystyle\sum_{h=1}^{2} \sum_{j=1}^{p} \alpha_h^j (g_1^j - g_0^j)} \end{cases} \tag{3.21}$$

□

Lemma 3.5 *Suppose $r(\tau) = r \in \mathcal{R}$, $\forall \tau \geq \tau_0$. Then $g_\infty = g_r$, where*

$$g_r = \begin{cases} \arg\min\limits_{d \in \mathbb{R}} \|g_d - r\|^2 \\ \text{subject to } g_d \triangleq g(\tau_0 - 1) + d[r - g(\tau_0 - 1)] \in \mathcal{G}_\delta. \end{cases} \tag{3.22}$$

Proof. Suppose by contradiction that $g_\infty \neq g_r$. We can apply Lemma 3.4 to the pair g_∞, g_r. In fact, since $\mathcal{R}$ is bounded and $g(\tau) \in \mathcal{R}$, $\forall \tau \in \mathbb{Z}$, then there exist ϵ, τ_1 such that $|g^i(\tau_1 - k) - g_\infty| \leq \epsilon \lambda^{-k/2}$, $\forall k \geq 0$, with ϵ given by (3.21). By defining γ as in (3.21), by Lemma 3.4 the command $g_\gamma = g_\infty + \gamma(g_r - g_\infty)$ is admissible. Since $d_\gamma < d_\infty$ and $d(\tau)$ is monotonically not increasing, $d(\tau) \leq d(\tau_1) = d_\gamma < d_\infty$, $\forall \tau \geq \tau_1$, which contradicts $d_\infty = \lim_{\tau \to \infty} d(\tau)$. □

Lemma 3.6 (Finite Stopping Time) *If $r(\tau) = r \in \mathcal{R}$, $\forall \tau \geq \tau_0$, then there exists a finite stopping time τ_s such that $g(\tau) = g_r$, $\forall \tau \geq \tau_s$, with g_r as in (3.22).*

Proof. By Lemma 3.5, $\lim_{\tau\to\infty} g(\tau) = g_r$, and therefore Lemma 3.4 can be applied to the pair $g(\tau_1)$, g_r, with ϵ given by (3.21) and τ_1 such that $\gamma = 1$ satisfies (3.21). □

Next Theorem 3.3 summarizes the properties of RRG.

Theorem 3.3 (RRG Properties) *Suppose that Assumptions 3.1–3.5 hold, and that $r(\tau) = r \in \mathcal{R}$, $\forall\tau \geq \tau_0$. Then, once the integer t_o is computed off-line via Algorithm 1, the optimization problem*

$$\beta(\tau) = \begin{cases} \arg\max_{\beta\in[0,1]} \beta \\ \quad \textit{subject to} \begin{cases} c(t+\tau|\tau, \Sigma, x(\tau), \beta) \in \mathcal{C}, \\ \quad \forall t \geq t_o, \forall\Sigma \in \mathcal{S} \\ g(\tau-1) + \beta[r(\tau) - g(\tau-1)] \in \mathcal{G}_\delta \end{cases} \end{cases} \tag{3.23}$$

can be solved for all $\tau \geq 0$. By setting

$$g(\tau) = g(\tau-1) + \beta(\tau)[r(\tau) - g(\tau-1)],\ \forall\tau \geq 0,$$

the constraints $c(\tau) \in \mathcal{C}$ are fulfilled for all $\tau \geq 0$, and for all $\Sigma \in \mathcal{S}$. Moreover, after a finite stopping time τ_s,

$$g(\tau) = g_r,\ \forall\tau \geq \tau_s$$

where g_r is defined in (3.22). In particular, if $r \in \mathcal{G}_\delta$, the RRG behaves as an all-pass filter for all $\tau \geq \tau_s$.

Remark 1 Since after a finite time $g(\tau) = g_r$, the asymptotical properties of the original system 3.1 remain unaltered, in particular $x(\tau) \to (I-\Phi)^{-1}Gg_r$ as $\tau \to \infty$.

3.6 Predictions and Computations

To lighten the notation, assume $\tau = 0$, and consider the predicted evolution $c(t)$ determined by a past command sequence $\mathcal{X}_- \subseteq \mathcal{R}$ and a future constant command $g(t) = g,\ \forall t \geq 0$, $c(t) = \sum_{k=0}^{\infty} H_k g(t-k) = \hat{c}(t) + \tilde{c}(t)$, $\hat{c}(t) \triangleq \sum_{k=0}^{\infty} \hat{H}_k g(t-k)$, $\tilde{c}(t) \triangleq \sum_{k=0}^{\infty} \tilde{H}_k g(t-k)$. Equivalently, the nominal prediction $\hat{c}(t)$ can be expressed in the computationally more preferable form,

$$\hat{c}(t) = \hat{H}_c \hat{\Phi}^t \hat{x}(0) + \hat{W}_t g, \tag{3.24}$$

for a consistent initial state $\hat{x}(0)$. For the sake of simplicity, suppose that the quantity N in (3.7) is such that $N > t_o$, where t_o is the constraint horizon computed via Algorithm 1. Then, the prediction error $\tilde{c}(t)$ can be rewritten as $\tilde{c}(t) = \tilde{W}_t g + \sum_{k=t+1}^{N-1} \tilde{H}_k g(t-k) + \tilde{c}_p(t)$, $\tilde{c}_p(t) \triangleq \sum_{k=N}^{\infty} \tilde{H}_k g(t-k)$, $0 \leq t \leq t_o < N$.

We wish to obtain a recursive formula to determine the range of $\tilde{c}_p(t)$ without requiring the storage of all past commands $g(t-k)$. Consider now a generic time $\tau \in \mathbb{Z}_+$ and define

$$\begin{aligned} \bar{c}_p^i(t+\tau|\tau) &\triangleq \max_{\{\tilde{H}_k \in [-E\lambda^k, E\lambda^k]\}_{k=N}^{\infty}} \tilde{c}_p^i(t+\tau|\tau) \\ &= \sum_{j=1}^{p} \sum_{k=N}^{\infty} E^{ij} \lambda^k |g^j(t-k+\tau)| \end{aligned}$$

At time $\tau+1$, at the same prediction step t, one has

$$\begin{aligned} \bar{c}_p^i(t+(\tau+1)|\tau+1) &= \lambda \bar{c}_p^i(t+\tau|\tau) + \\ &\sum_{j=1}^{p} E^{ij} \lambda^N |g^j(t-N+\tau+1)| \end{aligned} \tag{3.25}$$

Assuming that at time $\tau = 0$ system (3.1) is in an equilibrium condition, corresponding to $g(-t) \equiv g_0$ for all $t > 0$, (3.25) can be initialized with

$$\bar{c}_p^i(t|0) = \frac{\lambda^N}{1-\lambda} \sum_{j=1}^{p} E^{ij} |g_0^j| \tag{3.26}$$

The constraint $c(t+\tau|\tau, \Sigma, x(\tau), g) \in \mathcal{C}$ can be finally expressed as the following constraint on g

$$\begin{aligned} (\hat{W}_t + \tilde{W}_t) g \leq B_c - \hat{H}_c \hat{\Phi}^k \hat{x}(\tau) - \tilde{c}_p(t+\tau|\tau) - \\ \sum_{k=t+1}^{N-1} (\tilde{W}_k - \tilde{W}_{k-1}) g(t-k+\tau) \end{aligned} \tag{3.27}$$

or equivalently, by (3.4), as the following constraint on β

$$(a_t + y_t)\beta \leq b_t + x_t, \ t = 0, \ldots, t_o \tag{3.28}$$

where $a_t \triangleq \hat{W}_t[r(\tau) - g(\tau-1)]$, $b_t \triangleq B_c - \hat{H}_c \hat{\Phi}^t \hat{x}(\tau) - \hat{W}_t g(\tau-1)$, $x_t \triangleq \tilde{c}_p(t+\tau|\tau) - \sum_{k=t+2}^{N-1} (\tilde{W}_k - \tilde{W}_{k-1}) g(t-k+\tau) - \tilde{W}_{t+1} g(\tau-1)$, and $y_t \triangleq \tilde{W}_t[r(\tau) - g(\tau-1)]$. In the same manner, constraints (3.11) can be

rearranged in the form (3.28). While vectors a_t and b_t are known, vectors x_t and y_t are linear functions of the uncertainties $\tilde{W}_k$. Observe that the particular form (3.4) has allowed to make x_t, y_t independent; in fact, y_t is a function of $\tilde{W}_t$, while x_t depends on $\{\tilde{W}_k\}_{k=t+1}^{N-1}$. Therefore, $x_t^i \in [x_t^{-i}, x_t^{+i}]$, $y_t^i \in [y_t^{-i}, y_t^{+i}]$, for each component $i = 1, \ldots, q$, and for all $t = 0, \ldots, t_o$, with

$$\begin{aligned} x_t^{-i} &= \min \left\{ \tilde{c}_p^i(t) - \sum_{k=t+2}^{N-1} \sum_{j=1}^{p} (\tilde{W}_k^{ij} - \tilde{W}_{k-1}^{ij}) g^j(t-k) - \right. \\ &\quad \left. \sum_{j=1}^{p} \tilde{W}_{t+1}^{ij} g^j(-1) \right\} \\ &\quad \text{subject to } \left\{ \begin{array}{l} \tilde{W}_k^{ij} \in [W_k^{-ij}, W_k^{+ij}] \\ \tilde{W}_k^{ij} - \tilde{W}_{k-1}^{ij} \in [H_k^{-ij}, H_k^{+ij}] \end{array} \right. \qquad (3.29) \\ y_t^{-i} &= \min \left\{ \sum_{j=1}^{p} \tilde{W}_k^{ij} [r^j(0) - g^j(t-k)] \right\} \\ &\quad \text{subject to } \tilde{W}_k^{ij} \in [W_k^{-ij}, W_k^{+ij}] \qquad (3.30) \end{aligned}$$

and x_t^{+i}, y_t^{+i} defined analogously.

3.6.1 β-Parameter Selection

The constraints involved in the optimization problem (3.23) can be rewritten as

$$\begin{aligned} &(a_t^i + y_t^i)\beta \leq b_t^i + x_t^i, \ \forall x_t^i \in [x_t^{-i}, x_t^{+i}], \forall y_t^i \in [y_t^{-i}, y_t^{+i}] \\ &t = 0, \ldots, t_o + 1, \ i = 1, \ldots, q \end{aligned} \qquad (3.31)$$

where x_t^{-i}, x_t^{+i}, y_t^{-i}, y_t^{+i} are obtained by (3.29)-(3.30) for $t \leq t_o$, and the index $t = t_o + 1$ has been defined for the constraints deriving by (3.11). Consider the generic constraint

$$(a + y)\beta \leq b + x \qquad (3.32)$$

By defining

$$f(x, y) \triangleq \frac{b + x}{a + y}, \ x \in [x^-, x^+], \ y \in [y^-, y^+] \setminus \{a\}$$

and recalling the result [BS79, pp. 101, 473], $f(x, y)$ assumes global minima and maxima on the extreme points of its domain. Consequently, it is easy to

show that (3.28) has the following solution

$$\begin{cases} \beta \leq \min\{\frac{b+x^-}{a+y^-}, \frac{b+x^-}{a+y^+}\} & \text{if} \quad y^- > -a \\ \frac{b+x^-}{a+y^-} \leq \beta \leq \frac{b+x^-}{a+y^+} & \text{if} \quad y^- < -a < y^+ \\ \beta \geq \min\{\frac{b+x^-}{a+y^-}, \frac{b+x^-}{a+y^+}\} & \text{if} \quad y^+ < -a \end{cases} \tag{3.33}$$

Constraints (3.31) can be summarized as

$$\max_{\substack{i=1,\ldots,q \\ t=0,\ldots,t_o+1}} \beta_t^{-i} \leq \beta \leq \min_{\substack{i=1,\ldots,q \\ t=0,\ldots,t_o+1}} \beta_t^{+i}$$

where, possibly, $\beta_t^{-i} = -\infty$, $\beta_t^{+i} = +\infty$.

Remark 2. Recalling (3.29), the bounds x_t^{-i}, x_t^{+i} are evaluated as solutions of linear programs. The computational burden can be hugely lightened if uncertainty ranges are given only on $\tilde{W}_t$, $t = 0, \ldots, N-1$. In this case, (3.29) can be solved as trivially as (3.30) without the need of linear programs, at the cost of a more conservative description of system uncertainty, as pointed out in Sect. 3.3.

Remark 3. In the present formulation, the state $\hat{x}(\tau)$ is calculated in an open-loop manner by iterating (3.3). Apparently, this would lead to high sensitivity w.r.t. sensor noise, which in this chapter is not taken into account. In fact, no feedback from Σ takes part in the selection of $g(\tau)$. However, one should remind that system Σ, in general, represents a feedback linear loop. For example, for a constant reference trajectory $r(\tau) \equiv r \in \mathcal{G}_\delta$, after a finite stopping time τ_s system Σ receives $g(\tau) \equiv r$. Then, $y(\tau)$ will track r with the error rejection properties deriving from the primal linear controller.

Remark 4. If the reference sequence $\{r(\tau)\}_{\tau=0}^{\infty}$ is known a priori the whole trajectory $\{g(\tau)\}_{\tau=0}^{\infty}$ can be calculated in batch mode and stored into a memory device. Then, when determining the minimum sample time achievable for a given hardware configuration, the computational delays caused by the optimization algorithm can be ignored.

3.7 An Example

The RRG is applied in connection with the position servomechanism schematically described in Fig. 3.4. This consists of a DC-motor, a gear-box, an elastic shaft and an uncertain load. Technical specifications involve

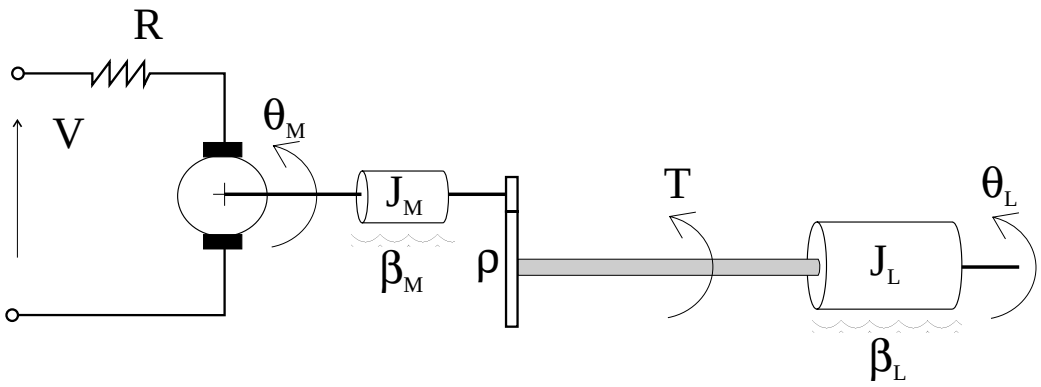


Figure 3.4: Servomechanism model.

bounds on the shaft torsional torque T as well as on the input voltage V. System parameters are reported in Table 3.1. Model uncertainties originate

Table 3.1: Model parameters

Symbol	Value (MKS)	Meaning
L_S	1.0	shaft length
d_S	0.02	shaft diameter
J_S	negligible	shaft inertia
J_M	0.5	motor inertia
β_M	0.1	motor viscous friction coefficient
R	20	resistance of armature
K_T	10	motor constant
ρ	20	gear ratio
k_θ	1280.2	torsional rigidity
$\hat{J}_L$	$20J_M$	nominal load inertia
β_L	25	load viscous friction coefficient
T_s	0.1	sampling time

from the moment of inertia J_L of the load, which is determined by the specific task. Denoting with θ_M, θ_L respectively the motor and the load angle, and by setting $x_p \triangleq [\theta_L \ \dot{\theta}_L \ \theta_M \ \dot{\theta}_M]'$, the model can be described by the following

state-space form

$$\begin{aligned}
\dot{x}_p &= \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{k_\theta}{J_L} & -\frac{\beta_L}{J_L} & \frac{k_\theta}{\rho J_L} & 0 \\ 0 & 0 & 0 & 1 \\ \frac{k_\theta}{\rho J_M} & 0 & -\frac{k_\theta}{\rho^2 J_M} & -\frac{\beta_M + k_T^2/R}{J_M} \end{bmatrix} x_p + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{k_T}{R J_M} \end{bmatrix} V \\
\theta_L &= \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} x_p \\
T &= \begin{bmatrix} k_\theta & 0 & -\frac{k_\theta}{\rho} & 0 \end{bmatrix} x_p
\end{aligned}$$

Since the steel shaft has finite shear strength, determined by a maximum admissible $\tau_{adm} = 50N/mm^2$, the torsional torque T must satisfy the constraint

$$|T| \leq 78.5398 \ Nm \tag{3.34}$$

Moreover, the input DC voltage V has to be constrained within the range

$$|V| \leq 220 \ V \tag{3.35}$$

The model is transformed in discrete time by sampling every $T_s = 0.1s$ and using a zero-order holder on the input voltage. A robust digital controller is designed by pole-placement techniques, and has the following transfer function from $e = (r - \theta_L)$ to V

$$G_c(z) = 1000 \frac{9.7929z^3 - 2.1860z^2 - 7.2663z + 2.5556}{10z^4 - 2.7282z^3 - 3.5585z^2 - 1.3029z - 0.0853}, \tag{3.36}$$

The resulting closed-loop system exhibits a very fast response but inadmissible voltage inputs and torsional torques for the references of interest, as shown in Fig. 3.5 for a set-point $r = 30 \ deg$.

The RRG is applied in order to fulfill (3.34), (3.35) for the uncertainty range which derives by an unknown load J_L, $10J_M \leq J_L \leq 30J_M$ (the corresponding impulse/step response uncertainty limits were obtained by maximizing $\pm H_t^{ij}, \pm W_t^{ij}$ with respect to J_L).

Fig. 3.6 shows the resulting uncertainty set for both the impulse and step responses. The constraint horizon is $t_o = 15$. A nominal load inertia $\hat{J}_L = 20J_M$ is selected, along with $N = 17$, $E = 1000[1 \ 1 \ 1 \ 1]'$, $\lambda = 0.8$, $\delta = 10^{-6}$. As a design rule of thumb, in order to have a description of the family of plant $\mathcal{S}$ as less conservative as possible, N should be approximately equal to the "length" of the impulse response in terms of time-steps: sufficiently

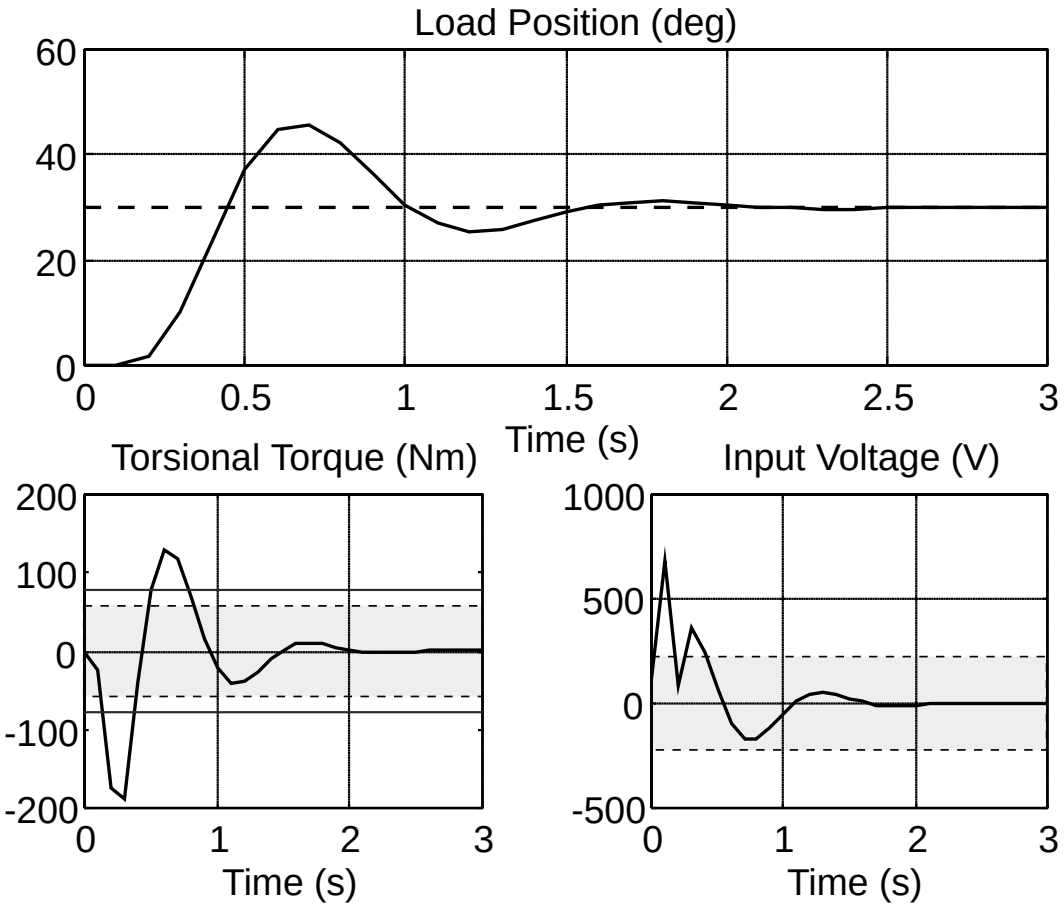


Figure 3.5: Unconstrained linear response. The shadowed area represents the admissible range.

large to describe accurately the range of variation of each sample when this is perceptibly nonzero, but also small enough to minimize computational complexity. Fig. 3.7 shows the resulting trajectories for a set-point $r = 30\ deg$ and a load $J_L = 25J_M$. This were obtained in $112s$ by using Matlab 4.2 on a 486 DX2/66 personal computer, with no particular care of code optimization. The standard Matlab `LP.M` routine was used to solve linear programs. Fig. 3.8 describe the effect of the width of the uncertainty interval. The larger the uncertainty range, the more conservative the RRG action, and hence the slower the output response. In order to make comparisons, constraint fulfilment is also achieved by linear control. The gain of controller (3.36) is reduced by a factor 16.9802 in order to have a maximum admissible set-point of 180 deg for the nominal plant $\hat{J}_L = 20J_M$ (with such a gain the linear loop reaches the maximum admissible torque T during the transient). The resulting trajectories

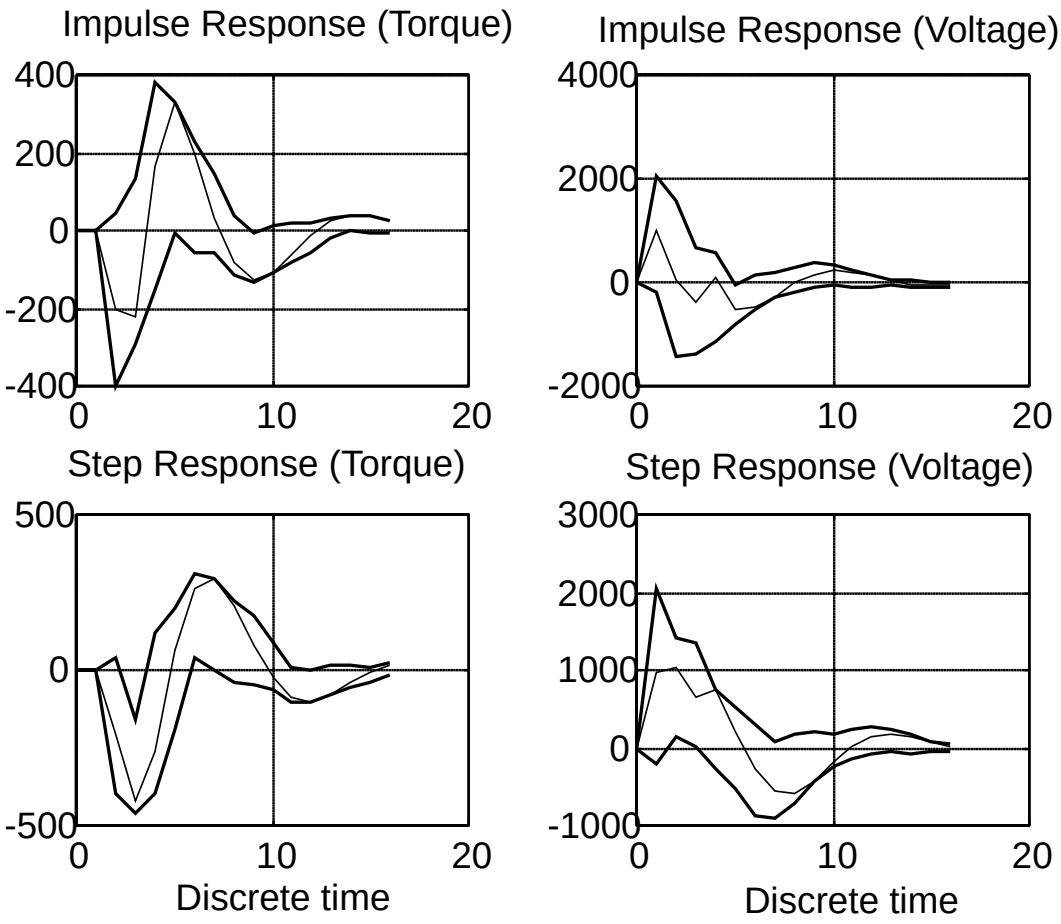


Figure 3.6: Uncertainty ranges for $10J_M \leq J_L \leq 30J_M$ (thick lines) and nominal $\hat{J}_L = 20J_M$ response (thin lines).

are depicted in Fig. 3.9 (thin line). In the same figure, the trajectories produced by applying the RRG together with the fast controller (3.36) are shown (thick lines). Fig. 3.9 shows also the resulting trajectories for $r = 90$ *deg*. Whilst for the linear closed loop the rise-time is the same for both $r = 180$ *deg* and $r = 90$ *deg* (thin lines), this is no more true for responses obtained by applying the nonlinear RRG (thick lines). In general, even if a linear controller gives the same performance of the RRG *for the maximum admissible set-point*, when the desired reference sequence is nonconstant a better tracking is provided by using a nonlinear reference filter. Finally, Fig. 3.10 describes the effect of RRG for a nonconstant desired trajectory $r(t)$ (dashed line). In order to fulfill the constraints for the uncertainty set $J_L \in [10J_M, 30J_M]$, the filtered command input $g(t)$ (thin line) is generated, obtaining the depicted output behaviour (thick line).

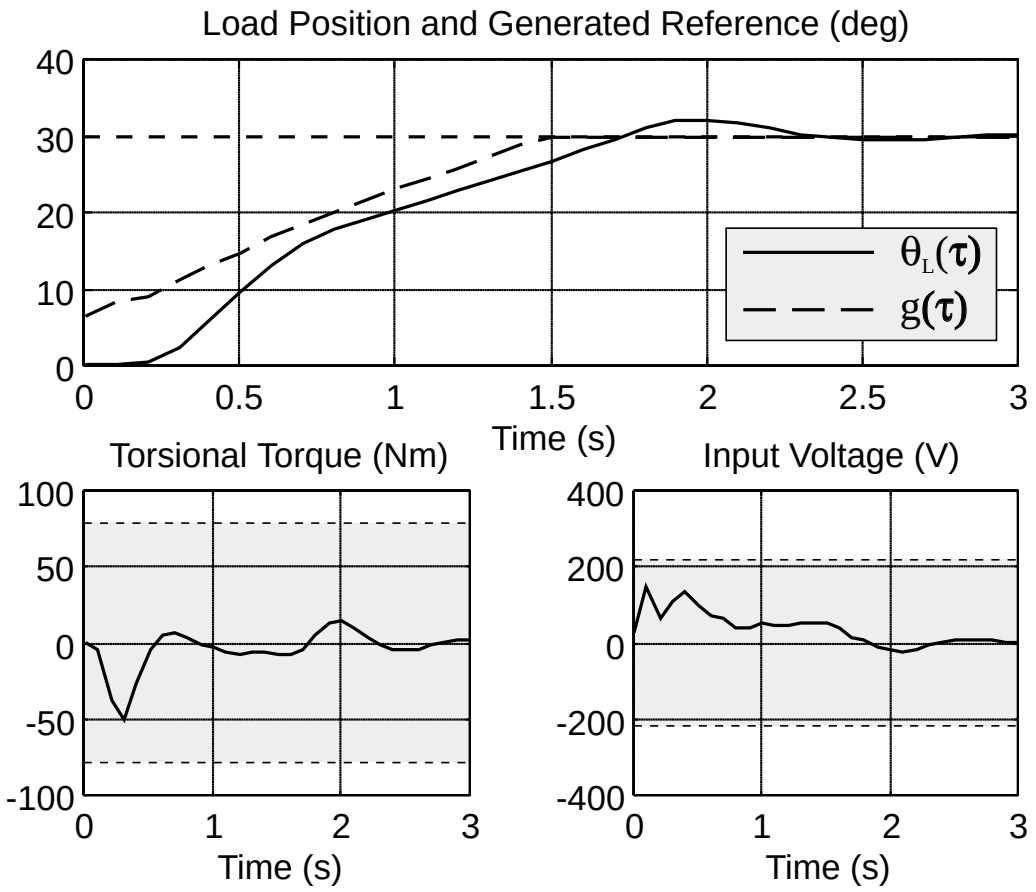


Figure 3.7: Response for $\hat{J}_L = 20J_M$, $10J_M \leq J_L \leq 30J_M$, and a real $J_L = 25J_M$.

Remark 5. As a general rule of thumb to design controllers which will be used in connection with a RRG, in order to maximize the properties of tracking one should select a robust controller which provides fast closed-loop response for all the systems of the considered family. This usually corresponds to large violations of the constraints, which therefore can be enforced by inserting a RRG. On the other hand, this cannot improve poor tracking properties of the original system because, as observed in Remark 1, the RRG becomes an all-pass filter when the constraints are inactive.

3.8 Concluding Remarks

This chapter has addressed the robust reference governor (RRG) problem, viz. the one of filtering the desired reference trajectory in such a way that an uncertain primal compensated control system can operate in a stable way with

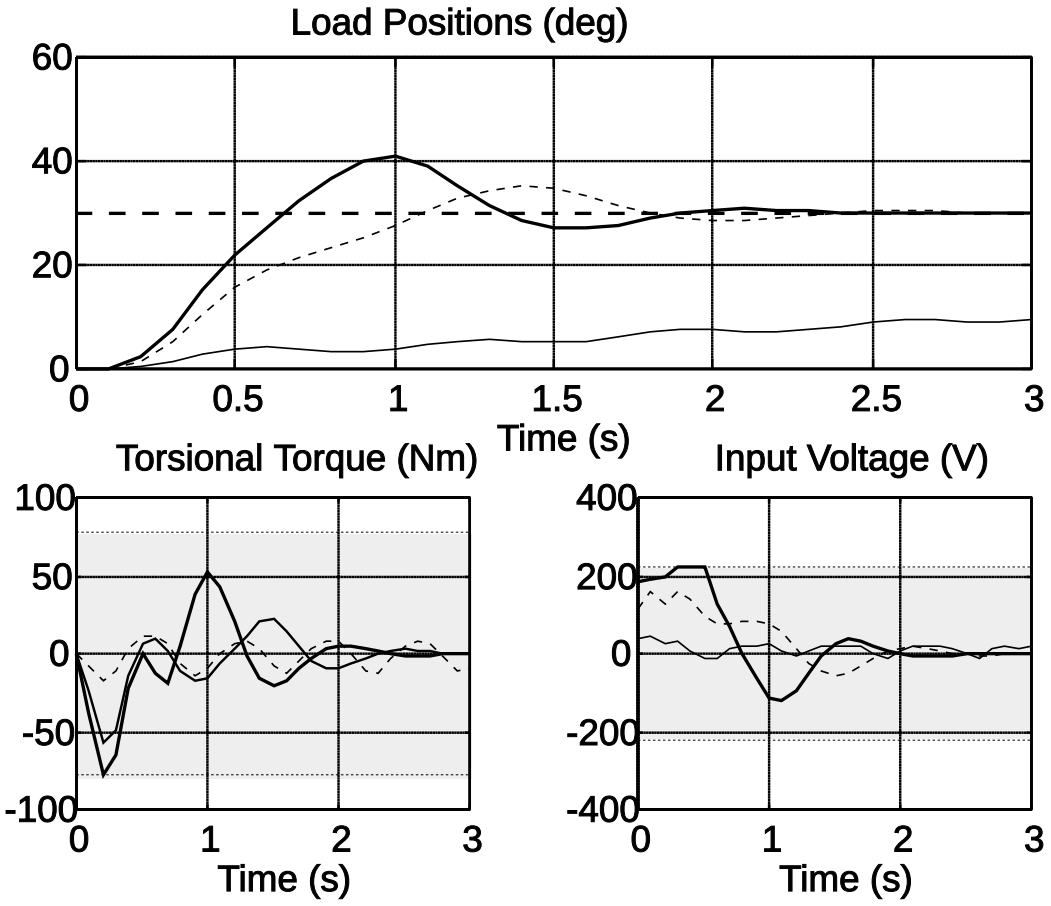


Figure 3.8: Response for $\hat{J}_L = J_L = 20J_M$, and different uncertainty ranges: no uncertainty (thick solid line), $[15J_M, 25J_M]$ (dashed line), and $[2J_M, 40J_M]$ (thin solid line).

satisfactory tracking performance and no constraint violation in the face of plant impulse/step responses uncertainties. The computational burden turns out to be moderate because of the underlying simple constrained optimization problem. Moreover, computations can be drastically reduced if only either step or impulse response uncertainty descriptions are given, an attractive feature when the filter has to be operated on line.

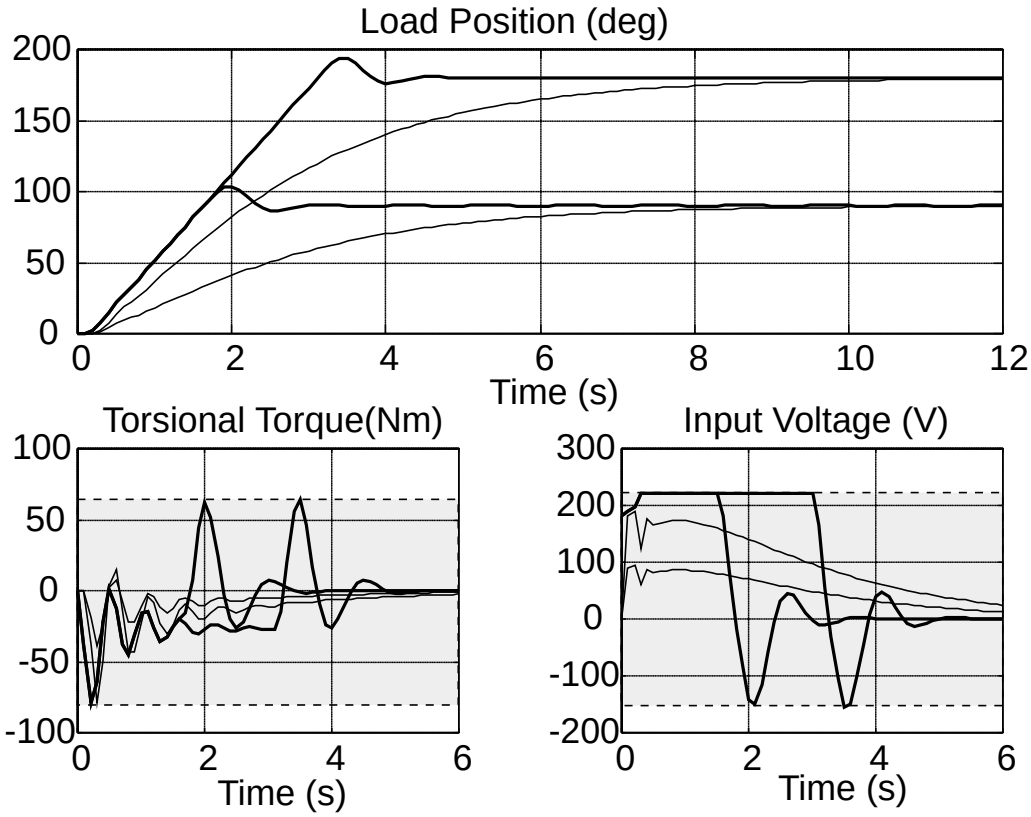


Figure 3.9: Set-point $r = 90,\ 180\ deg$, $J_L = 20J_M$, no uncertainty. Fast controller + RRG (thick lines) linear controller (thin lines).

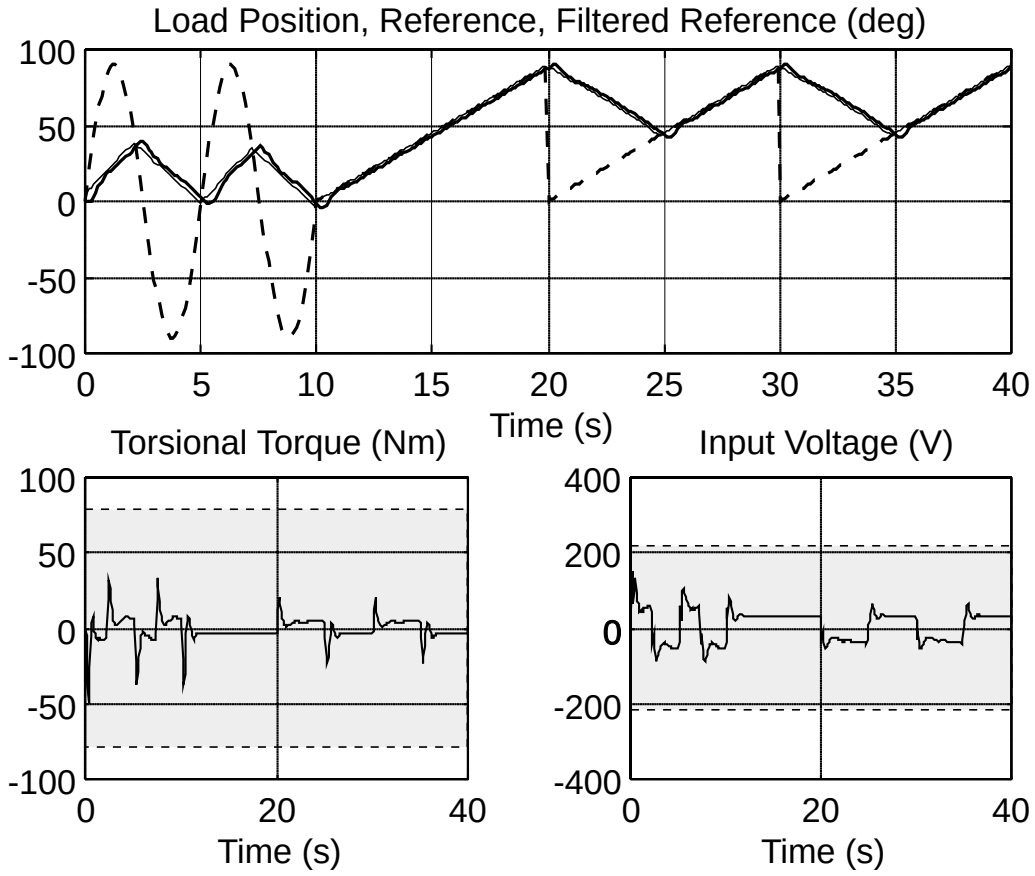


Figure 3.10: Nonconstant reference trajectory and same parameters of Fig. 3.7: $r(t)$ (dashed line), $g(t)$ (thin line), $\theta_L(t)$ (thick line).

Chapter 4

Disturbances and Output-Feedback

In this chapter we combine predictive control and set-membership state estimation techniques for input/state hard constraints fulfillment. Linear systems with unknown but bounded disturbances and partial state information are considered. The adopted worst-case approach guarantees that the constraints are satisfied for all the states which are compatible with the available information and for all the disturbances within the given bounds. A stability result and simulative studies are reported for N-DOF (degrees of freedom) reference governors.

4.1 Introduction

Two features frequently arise in many practical control problems: the necessity of satisfying input/state constraints and the presence of disturbances. The former are dictated by physical limitations of the actuators and by the necessity to keep some plant variables within safe limits; the latter by model inaccuracy and unmeasurable noises. In recent years, industry has been attracted by predictive controllers, based on the so called *receding horizon* strategy mentioned in Chapter 1. This consists in determining a *virtual control input sequence* that optimizes an open-loop *performance function*, according to a prediction of the system evolution from the current time t, over a semi-infinite prediction horizon. Then, the sequence is actually applied to the system, until

another sequence based on more recent data is newly computed. The involved prediction depends on the current state, the future state disturbances, and the selected control input. Several strategies, which have been developed for deterministic frameworks (for instance [KG88, MM90, MLZ90, CS91, ZM95]) can be applied by neglecting the presence of the state disturbances over the prediction horizon. However, this does not guarantee that state related constraints are actually satisfied.

Very recently, [GK95a] have developed a reference governor for constraint fulfillment guaranteed in the presence of state disturbances. However, this technique requires full state measurements. When these are not available, it is common practice to provide the predictor with an estimate generated by a state observer, *e.g.* a Kalman filter. Again, no guarantee of constraint fulfillment holds, due to a mismatch between the predicted evolution and the actual behavior of the system.

This chapter copes with both the presence of state disturbances and full state information unavailability. We assume that the uncertainties acting on the system (state disturbances and output noises) are unknown but bounded. We adopt a worst-case approach, which consists of: *i)* considering the effect of the worst state disturbance sequence over the prediction horizon; *ii)* handling state unavailability by using the so-called *set-membership* state estimation, which was first introduced in [Sch68, BR71]. By collecting the maximum information available about the current state at time t, this approach to state estimation considers the *state uncertainty set*, i.e. the set of all the state vectors compatible with the model equations, the initial uncertainty, the disturbance bounds and the available output measurements up to time t. Due to the tremendous amount of calculations required by the updating of the state uncertainty sets, many recursive approximation algorithms, based on simple regions in the state space, like ellipsoids [Sch68, BR71, Che80, FH82, FKSV94], or limited complexity polytopes [BS90], have been proposed in the literature. In this chapter, we adopt the minimum volume parallelotopic approximation developed in [VZ96, CGZ19]. The resulting set-membership estimation algorithm is particularly appealing for predictive control, as it presents both good approximation capabilities and reasonable computational complexity.

The chapter is organized as follows. In Section 2 we formulate the worst-case predictive control problem and give the basic assumptions. Section 3

deals with asymptotic properties of the overall feedback loop and shows how the posed infinite horizon optimization problem can be solved by considering only a finite number of constraints. The computations involved in the optimization algorithm are investigated in Section 4. Section 5 characterizes the relationships between the existence of feasible solutions and the set-membership approximation. Finally, simulation results are reported in Section 6 and some conclusions in Section 7.

4.2 Problem Formulation and Assumptions

Consider the following linear discrete-time time-invariant system

$$\begin{cases} x(t+1) &= Ax(t) + Bu(t) + \xi(t) \\ y(t) &= Cx(t) + Du(t) + \zeta(t) \\ c(t) &= Ex(t) + Fu(t) \end{cases} \tag{4.1}$$

along with a desired output reference $r(t) \in \mathbb{R}^p$, where the state $x(t) \in \mathbb{R}^n$ is supposed not to be directly measurable, $u(t) \in \mathbb{R}^m$ is the control input, $y(t) \in \mathbb{R}^p$ the measured output, $\xi(t) \in \mathbb{R}^n$ the state disturbance, $\zeta(t) \in \mathbb{R}^p$ the output noise, and $t \in \mathbb{Z}_+ = \{0, 1, \ldots\}$. We assume that both $\xi(t)$ and $\zeta(t)$ are unknown but bounded

$$\xi_i^- \le \xi_i(t) \le \xi_i^+, \qquad i = 1, \ldots, n \tag{4.2}$$

$$\zeta_i^- \le \zeta_i(t) \le \zeta_i^+, \qquad i = 1, \ldots, p \tag{4.3}$$

$$\forall t \in \mathbb{Z}_+$$

or, in a more compact form,

$$\xi(t) \in \Xi, \quad \zeta(t) \in \mathcal{Z}.$$

and Ξ and $\mathcal{Z}$ are obviously defined and given . We assume that system (4.1) satisfies the following

Assumption 4.1 *A is asymptotically stable.*

This assumption is justified if one thinks (4.1) to represent a precompensated feedback system.

Assumption 4.2 *The dc-gain matrix $H \triangleq C(I - A)^{-1}B$ has full rank,* $\operatorname{rank} H = \min\{m, p\}$.

The problem is to generate the control input $u(t)$ so as to constrain the vector $c(t) \in \mathbb{R}^l$

$$c(t) \in \mathcal{C}, \tag{4.4}$$

where $\mathcal{C}$ is the convex polyhedron

$$\mathcal{C} = \left\{ c \in \mathbb{R}^l : \; A_c c \leq B_c \right\}, \; B_c \in \mathbb{R}^q, \tag{4.5}$$

without affecting the original tracking properties of the system. W.l.o.g., we consider

$$c(t) = \left[\begin{array}{c} x(t) \\ u(t) \end{array} \right],$$

and rewrite (4.5) as

$$\mathcal{C} = \left\{ [x' \; u']' \in \mathbb{R}^{n+m} : \; A_c^x x + A_c^u u \leq B_c \right\} \tag{4.6}$$

where $'$ denotes transposition. Hereafter, we shall assume that

Assumption 4.3 *$\mathcal{C}$ is bounded*

According to the above setting, at a generic time t the available information on the state vector $x(t)$ is given by the model equation (4.1), the bounds on the state disturbances (4.2) and the output noise (4.3), and the observed measurements $y(k)$, $k = 0, 1, \ldots, t$. Let us denote by $\mathcal{X}^*(t_1|t_2)$ the *state uncertainty set* of all state vectors at time t_1, compatible with the information available at time t_2. If the *a priori information set* $\mathcal{X}^*(0|-1)$ is bounded and contains the initial state $x(0)$, then the state uncertainty sets are provided by the following recursion

$$\begin{array}{rcl} \mathcal{X}^*(t|t) & = & \mathcal{X}^*(t|t-1) \bigcap \mathcal{X}_y^*(t) \\ \mathcal{X}^*(t+1|t) & = & A\mathcal{X}^*(t|t) \oplus \{Bu(t)\} \oplus \Xi \end{array}$$

where

$$\mathcal{X}_y^*(t) = \{ x \in \mathbb{R}^n : \; y(t) - Cx - Du(t) \in \mathcal{Z} \}$$

is the set of states compatible with the single measurement $y(t)$, and $\oplus$ denotes the vector sum of sets. It is clear that the complexity of $\mathcal{X}^*(t|t)$ and $\mathcal{X}^*(t+1|t)$ grows with t, and therefore it is common practice to approximate these sets by simpler regions, the so-called *set-valued estimates* $\mathcal{X}(t|t)$ and $\mathcal{X}(t|t+1)$ respectively. In the following, the set-valued estimate involved in

the optimization procedure at time t will be $\mathcal{X}(t|t-1)$. This means that the input at time t will be computed on the basis of the available information up to time $t-1$, so that the required computations can be performed over one full sample interval.

In order to use efficient optimization procedures, we adopt the strategy proposed in [RM93] by limiting to N_v the number of control degrees of freedom. This consists of defining the finite dimensional optimization vector

$$\mathcal{V} \triangleq \begin{bmatrix} v(N_v-1) \\ \vdots \\ v(0) \end{bmatrix} \in \mathbb{R}^{N_v m}$$

and then considering, as virtual control input sequence, its *constant extension* over a semi-infinite horizon

$$S_{\mathcal{V}} \triangleq \{v(k)\}_{k=0}^{\infty},$$

which is obtained by setting

$$v(k) \equiv v(N_v-1), \forall k \geq N_v-1,\ k \in \mathbb{Z}_+ \ . \tag{4.7}$$

Notice that when (4.1) represents a closed-loop system, u represents the reference input, and hence, by setting $N_v = 1$, we obtain a Reference Governor.

In the following, we shall indicate with $c(t+k, x, \mathcal{V}, \mathcal{K}_k)$ the constrained vector at time $t+k$, predicted at time t, according to model (4.1), initial state $x \in \mathcal{X}(t|t-1)$, future state disturbances $\mathcal{K}_k \in \Xi_k$,

$$\begin{aligned} \mathcal{K}_k &\triangleq \begin{bmatrix} \xi(t+k-1) \\ \vdots \\ \xi(t) \end{bmatrix} \\ \Xi_k &\triangleq \left\{\mathcal{K}_k \in \mathbb{R}^{kn} :\ \mathcal{K}_{k,i} \in [\xi_i^-, \xi_i^+]\right\}, \end{aligned}$$

and by setting $u(t+k) = v(k)$, $\forall k \in \mathbb{Z}_+$.

We wish to select a performance function $J(t, \mathcal{V})$ in such a way that the minimization of $J(t, \mathcal{V})$ w.r.t. $\mathcal{V}$ ensures good tracking properties. In the predictive control literature, it is common use to weight the sum of the predicted tracking error squares over a semi-infinite horizon, which makes the performance function depend on the current state $x(t)$. On the other hand, we

have assumed that $x(t)$ is not available. In the present worst-case formulation, a possible solution consists in defining

$$J(t,\mathcal{V}) = \max_{\substack{x\,\in\,\mathcal{X}(t|t-1)\\ \mathcal{K}_\infty\,\in\,\Xi_\infty}} \left\{ \sum_{k=0}^{\infty} \|y(t+k,x,\mathcal{V},\mathcal{K}_\infty) - r(t)\|^2_{\Upsilon_y} + \|\mathcal{V}\|^2_{\Upsilon_v} \right\},$$

where $\Upsilon_y,\ \Upsilon_v > 0$, and $\|y\|^2_{\Upsilon_y} = y'\Upsilon_y y$. However, a control law based on such a cost function would require the solution of a very complex constrained min-max optimization problem at each time step t. In order to sidestep these difficulties, we adopt the following function

$$J(t,\mathcal{V}) = \sum_{k=0}^{N_v-2} \|v(k) - v(N_v-1)\|^2_{\Upsilon_1} + \|v(N_v-1) - H^{\#} r(t)\|^2_{\Upsilon_2}, \Upsilon_1 > 0, \Upsilon_2 > 0, \tag{4.8}$$

where

$$H^{\#} \triangleq \begin{cases} (H'H)^{-1}H & \text{if} \quad m \le p \\ H'(HH')^{-1} & \text{if} \quad m > p \end{cases}$$

Notice that no feedback term is present in (4.8). Since as explained later the constraints involved in the minimization of (4.8) depend on the current set-valued state estimate $\mathcal{X}(t|t-1)$, feedback will be present only when the constraints are active. This should not be considered as a drawback, since noise and unmodeled dynamics effects rejection can be achieved by designing a precompensator and then labeling as (4.1) the resulting closed-loop system.

At each time t, the selection of the optimal vector $\mathcal{V}_t$ proceeds as follows. Denote by $\Omega(t)$ the set of all vectors $\mathcal{V}$ leading to feasible evolutions of the constrained vector,

$$\Omega(t) = \left\{ \mathcal{V} \in \mathbb{R}^{N_v m} :\ c(t+k,x,\mathcal{V},\mathcal{K}_k) \in \mathcal{C}, \forall x \in \mathcal{X}(t|t-1),\ \forall \mathcal{K}_k \in \Xi_k,\ \forall k \in \mathbb{Z}_+ \right\}. \tag{4.9}$$

If $\Omega(t)$ is nonempty, define

$$\mathcal{V}_t^* = \arg\,\min_{\mathcal{V}\in\Omega(t)} J(t,\mathcal{V}) \tag{4.10}$$

Then, denoting by $\mathcal{V}_t^1$ the extension of the previous optimal vector $\mathcal{V}_{t-1}$, i.e.

$$\mathcal{V}_{t-1} = \begin{bmatrix} v_{t-1}(N_v-1) \\ v_{t-1}(N_v-2) \\ \vdots \\ v_{t-1}(1) \\ v_{t-1}(0) \end{bmatrix},\ \mathcal{V}_t^1 = \begin{bmatrix} v_{t-1}(N_v-1) \\ v_{t-1}(N_v-1) \\ v_{t-1}(N_v-2) \\ \vdots \\ v_{t-1}(1) \end{bmatrix},$$

we set

$$\mathcal{V}_t = \begin{cases} \mathcal{V}_t^* & \text{if } \Omega(t) \neq \emptyset \text{ and } J(t, \mathcal{V}_t^*) < J(t, \mathcal{V}_t^1) - \epsilon(t) \\ \mathcal{V}_t^1 & \text{otherwise} \end{cases} \tag{4.11}$$

where $\epsilon(t) \triangleq \min\{\rho_1 J(t, \mathcal{V}_t^1), \rho_2)$, and ρ_1, ρ_2 are fixed arbitrarily small positive scalars. Then, according to the receding horizon strategy described above, we set

$$u(t) = v_t(0). \tag{4.12}$$

The entire procedure is then repeated at time $t + 1$.

Finally, in order to complete the above scheme, we make the following hypothesis on $\mathcal{X}(0|-1)$.

Assumption 4.4 *For the a priori information set $\mathcal{X}(0|-1)$ there exists a finite input sequence $\mathcal{V}_{-1}$ such that $\mathcal{V}_{-1}^1 \in \Omega(0)$.*

4.3 Main Results

The optimization problem (4.10) involves an *infinite* number of linear constraints. However, in order to be able to computationally solve (4.10) via standard Quadratic Programming (QP) tools, a *finite* number of constraints is desirable. Next Theorem 4.1 shows that this can be achieved by adding an extra linear constraint on $\mathcal{V}$.

Theorem 4.1 *There exist an index $k_o \geq N_v$ and $\delta > 0$ such that, if $\mathcal{V}$ satisfies*

$$\left[A_c^x (I - A)^{-1} B + A_c^u\right] v(N_v - 1) \leq B_c - \delta \, \underline{1} \tag{4.13}$$

where $\underline{1} = [1, \ldots, 1]'$, then $\mathcal{V} \in \Omega(t)$ iff

$$\begin{aligned} & c(t + k, x, \mathcal{V}, \mathcal{K}_k) \in \mathcal{C}, \ \forall x \in \mathcal{X}(t|t-1), \\ & \forall \mathcal{K}_k \in \Xi_k, \ \forall k = 0, \ldots, k_o, \end{aligned} \tag{4.14}$$

Proof. W.l.o.g. set $t = 0$. Let $k \geq N_v$, $x \in \mathcal{X}(0|-1)$, $\mathcal{V} \in \Omega(0)$ such that (4.13) is satisfied, and $c(h, x, \mathcal{V}, \mathcal{K}_h) \in \mathcal{C}$, $\forall h = 0, \ldots, N_v$. Consider the prediction of the state at time k

$$x(k, x, \mathcal{V}, \mathcal{K}_k) = A^{k-N_v} x(N_v, x, \mathcal{V}, \mathcal{K}_{N_v}) + \left[\sum_{i=0}^{k-N_v-1} A^i B\right] v(N_v-1) + \sum_{i=0}^{k-N_v-1} A^i \xi(k-1-i)$$

where

$$A_c^x x(N_v, x, \mathcal{V}, \mathcal{K}_{N_v}) + A_c^u v(N_v - 1) \leq B_c.$$

By Assumption 4.3, there exist constants $\Delta_{\mathcal{C}}^x$ and $\Delta_{\mathcal{C}}^u$ such that

$$\|x\|_\infty \leq \Delta_{\mathcal{C}}^x, \|u\|_\infty \leq \Delta_{\mathcal{C}}^u, \ \forall \begin{bmatrix} x \\ u \end{bmatrix} \in \mathcal{C}.$$

By letting M and λ such that

$$\|A^k\|_\infty \leq M\lambda^k,$$

where $\|A\|_\infty$ denotes the ∞-induced matrix norm, one gets

$$\begin{aligned} \| \sum_{i=0}^{k-N_v-1} A^i \xi(i) \|_\infty &\leq \sum_{i=0}^{k-N_v-1} M\lambda^i \|\xi(i)\|_\infty \\ &\leq \sum_{i=0}^{k-N_v-1} M\lambda^i \bar{\xi} \leq \frac{M}{1-\lambda}\bar{\xi} \end{aligned}$$

where $\bar{\xi} = \max_{i=1,\ldots,n}\{\max\{|\xi_i^-|, |\xi_i^+|\}\}$. Being

$$A^{k-N_v} x(N_v, x, \mathcal{V}, \underline{\xi}_{N_v}) + \sum_{i=0}^{k-N_v-1} A^i B v(N_v - 1) - (I - A)^{-1} B v(N_v - 1) =$$

$$A^{k-N_v} x(N_v, x, \mathcal{V}, \mathcal{K}_{N_v}) + \sum_{i=0}^{k-N_v-1} A^i B v(N_v - 1) - \sum_{i=0}^{\infty} A^i B v(N_v - 1) =$$

$$A^{k-N_v}[x(N_v, x, \mathcal{V}, \mathcal{K}_{N_v}) - (I - A)^{-1} B v(N_v - 1)],$$

it results

$$\begin{aligned} &\left\| A_c^x \left[x(k, x, \mathcal{V}, \mathcal{K}_k) - (I - A)^{-1} B v(N_v - 1) \right] \right\|_\infty \leq \\ &\leq M\|A_c^x\|_\infty \lambda^{k-N_v} (\Delta_{\mathcal{C}}^x + \|(I - A)^{-1} B\|_\infty \Delta_{\mathcal{C}}^u) + \frac{M\|A_c^x\|_\infty}{1-\lambda}\bar{\xi} \leq \delta, \end{aligned} \quad (4.15)$$

for

$$\delta \geq \frac{2M\|A_c^x\|_\infty \bar{\xi}}{1-\lambda} \quad (4.16)$$

$$k \geq k^* = N_v + \log_\lambda \frac{\delta}{2M\|A_c^x\|_\infty (\Delta_{\mathcal{C}}^x + \|(I - A)^{-1} B\|_\infty \Delta_{\mathcal{C}}^u)} \quad (4.17)$$

Hence, by (4.13) and (4.15),

$$\begin{aligned} &A_c^x x(k, x, \mathcal{V}, \mathcal{K}_k) + A_c^u v(k) \\ &= A_c^x \left[x(k, x, \mathcal{V}, \mathcal{K}_k) - (I - A)^{-1} B v(N_v - 1) \right] + A_c^x (I - A)^{-1} B v(N_v - 1) + A_c^u v(N_v - 1) \\ &\leq A_c^x \left[x(k, x, \mathcal{V}, \mathcal{K}_k) - (I - A)^{-1} B v(N_v - 1) \right] + B_c - \delta \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} \leq B_c \end{aligned}$$

or equivalently

$$\begin{bmatrix} x(k, x, \mathcal{V}, \mathcal{K}_k) \\ v(k) \end{bmatrix} \in \mathcal{C},\ \forall k \geq k^*$$

Then, there exist integers $k_o \leq k^*$ such that (4.14) is satisfied. □

Notice that (4.13) imposes that the predicted steady-state constrained vector, corresponding to the constant input level $v(N_v - 1)$, lies inside $\mathcal{C}$ by at least a fixed distance away from the border. Then, by virtue of Assumption 4.1, this implies that, after a finite transient, all the trajectories of the constrained vector will be inside $\mathcal{C}$.

In the following we will assume that the set defined by the inequality (4.13), with δ as in (4.16), is nonempty, and hence the existence of feasible solutions $\mathcal{V}$ is allowed.

Next Theorem 4.2 describes the asymptotical behavior of the overall control scheme.

Theorem 4.2 *Consider system (4.1) and a sequence of approximated state uncertainty sets* $\{X(t|t-1)\}_{t=0}^{\infty}$. *Let* $r(t) \equiv r$, $\forall t \geq t_r \in \mathbb{Z}_+$. *Then, the control strategy (4.10)-(4.12), based on the optimization of the performance function (4.8) in the presence of constraints (4.9) and (4.13), guarantees stability of the overall control loop.*

Proof. Let $\mathcal{L}(t) = J(t, \mathcal{V}_t)$, with $\mathcal{V}_t$ as in (4.11). Then, by (4.8), $\mathcal{L}(t)$ is a monotonically decreasing nonnegative sequence

$$\mathcal{L}(t-1) - \mathcal{L}(t) \geq \mathcal{L}(t-1) - J(t, \mathcal{V}_t^1) \geq \|v_{t-1}(0) - v_{t-1}(N_v - 1)\|_{\Upsilon_1}^2 \geq 0 \quad (4.18)$$

and converges as $t \to \infty$. Moreover,

$$\lim_{t\to\infty} \|v_{t-1}(0) - v_{t-1}(N_v - 1)\|_{\Upsilon_1} = 0 \quad (4.19)$$

Let

$$t_\epsilon \triangleq \sup_{t \geq t_r} \left\{ t :\ J(t, \mathcal{V}_t^*) < J(t, \mathcal{V}_t^1) - \epsilon(t) \right\} \quad (4.20)$$

If $t_\epsilon < +\infty$, then by (4.11)

$$\mathcal{V}_t = \mathcal{V}_t^1,\ \forall t \geq t_\epsilon,$$

and therefore

$$u(t) = v(0) \equiv v_{t_\epsilon}(N_v - 1),\ \forall t \geq t_\epsilon + N_v - 1$$

By Assumption 4.1, stability of the overall system follows. Assume that $t_\epsilon = +\infty$, and define a subsequence $\{t_k\}_{k=0}^{\infty}$ such that $J(t_k, \mathcal{V}_{t_k}^*) < J(t_k, \mathcal{V}_{t_k}^1) - \epsilon(t_k)$. By (4.11),

$$\mathcal{L}(t_k - 1) - \mathcal{L}(t_k) \geq \|v_{t_k-1}(0) - v_{t_k-1}(N_v - 1)\|_{\Upsilon_1}^2 + \epsilon(t_k) \geq 0.$$

As $k \to \infty$, since $\mathcal{L}(t_k)$ converges and by (4.19), it results $\epsilon(t_k) = \rho_1 J(t_k, \mathcal{V}_{t_k}^1) \to 0$. Then,

$$\mathcal{L}(t_k) = J(t_k, \mathcal{V}_{t_k}) < J(t_k, \mathcal{V}_{t_k}^1) - \epsilon(t_k) \to 0$$

and, being $\mathcal{L}(t)$ monotonic, it follows $\lim_{t\to\infty} \mathcal{L}(t) = 0$, or equivalently

$$\lim_{t\to\infty} \mathcal{V}_t = [v_r' \ \dots \ v_r']'$$

where $v_r \triangleq H^{\#} r$. In particular, $\lim_{t\to\infty} u(t) = v_r$. Observe that (4.1) is converging-input converging-state stable [Son95]. In fact, let $\tilde{x}(t) \triangleq x(t) - (I - A)^{-1} r$, $\tilde{u}(t) = u(t) - v_r$ and consider the system $\tilde{x}(t+1) = A\tilde{x}(t) + B\tilde{u}(t)$. By Assumption 4.1, there exist constants $\lambda < 1$, $k_1 > 0$, $k_2 > 0$ such that

$$\|\tilde{x}(t)\| \leq k_1 \lambda^{t-t_0} \|\tilde{x}(t_0)\| + k_2 \|\tilde{u}(t, t_0)\|_\infty$$

where $\tilde{u}(t, t_0) \triangleq \tilde{u}(t)$ for $t \geq t_0$ and 0 otherwise. Clearly, $\|\tilde{u}(t, t_0)\|_\infty \to 0$ as $t_0 \to \infty$. Let $\alpha > 0$, t_1 such that $k_2 \|\tilde{u}(t, t_0)\|_\infty \leq \frac{\alpha}{2}$, $\forall t_0 \geq t_1$, and $t_\alpha \geq t_1$ such that $k_1 \lambda^{t-t_1} \|\tilde{x}(t_0)\| \leq \frac{\alpha}{2}$, $\forall t \geq t_\alpha$. Then, $\|\tilde{x}(t)\| \leq \alpha$, $\forall t \geq t_\alpha$. This proves that $\lim_{t\to\infty} x(t) = (I - A)^{-1} v_r$, $\lim_{t\to\infty} y(t) = r$. □

Remark 1. Despite Theorem 4.2 guarantees the convergence of the state, zero-offset set-point tracking in steady-state holds only when in (4.20) $t_\epsilon = +\infty$. The verification of this condition is influenced by the constraints and the disturbance bounds Ξ, $\mathcal{Z}$. In fact, because of the adopted worst-case approach, too stringent constraints and/or too large Ξ, $\mathcal{Z}$ can prevent the control input $u(t)$ to reach the desired value v_r. In this case, the cost function (4.8) tends to pull $u(t)$ toward the safe input level which is closest to v_r.

4.4 Constrained Optimization Algorithm

In this section, we derive the solution of the constrained optimization problem posed in Section 4.2. The control algorithm must perform two main tasks:

i) updating the approximated state uncertainty set $\mathcal{X}(t|t-1)$;

ii) performing the constrained optimization (4.10) with the additional constraint (4.13).

In this chapter, we will consider parallelotopes [VZ96] as approximating regions for the state uncertainty sets.

Definition 4.1 *Let a nonsingular matrix $T \in \mathbb{R}^{n\times n}$ and a vector $\hat{x} \in \mathbb{R}^n$ be given. Then*

$$\mathcal{P}(T, \hat{x}) = \{x : \ x = \hat{x} + T\alpha, \|\alpha\|_\infty \leq 1\}$$

defines a parallelotope *in $\mathbb{R}^n$, with center $\hat{x}$ and edges parallel to the column vectors of T.*

Recently, a recursive algorithm for the outer approximation of the uncertainty state set of a linear system through parallelotopic regions has been proposed in [CGZ19]. The recursive approximation is computed according to a minimum volume criterion. At a generic time t, the following two steps are performed

- *measurement update*: given the parallelotope $\mathcal{X}(t|t-1) = \mathcal{P}(t-1)$, compute the minimum volume parallelotope $\overline{\mathcal{P}}$ outbounding $\mathcal{P}(t-1) \bigcap \mathcal{X}_y^*(t)$;
- *time update*: compute the minimum volume parallelotope $\mathcal{P}(t)$ outbounding $A\overline{\mathcal{P}} \oplus \{Bu(t)\} \oplus \Xi$ and set $\mathcal{X}(t+1|t) = \mathcal{P}(t)$.

The iterations above are initialized by setting $\mathcal{X}(0|-1)$ equal to the given a priori information set $\mathcal{X}^*(0|-1)$. The computational complexity of the algorithm has been proved to be polynomial in the state dimension n.

In order to solve the optimization problem, we need to express the cost function (4.8), the set $\Omega(t)$ in (4.9), and the additional constraint (4.13) in terms of the optimization vector $\mathcal{V}$. By letting

$$\Psi \triangleq \left[\begin{array}{c|ccc} (N_v-1)\Upsilon_1 + \Upsilon_2 & -\Upsilon_1 & \cdots & -\Upsilon_1 \\ \hline -\Upsilon_1 & \Upsilon_1 & & \\ \vdots & & \ddots & \\ -\Upsilon_1 & & & \Upsilon_1 \end{array}\right]$$

the cost (4.8) can be rewritten as

$$J(t,\mathcal{V}) = \mathcal{V}'\Psi\mathcal{V} - 2r'(t)H^{\#\prime}\Upsilon_2 \left[\begin{array}{cccc} I_m & 0 & \cdots & 0 \end{array}\right]\mathcal{V} + r'(t)H^{\#\prime}\Upsilon_2 H^{\#} r(t) \tag{4.21}$$

By (4.6), the fulfillment of the constraints $c(t+k,x,\mathcal{V},\mathcal{K}_k) \in \mathcal{C}$, for every $x \in \mathcal{X}(t|t-1)$ and $\mathcal{K}_k \in \Xi_k$, over a finite horizon $k = 0,\ldots,k_o$, can be expressed as

$$\begin{array}{l} A_c^x x(t+k,x,\mathcal{V},\mathcal{K}_k) + A_c^u v(k) \leq B_c, \\ \forall x \in \mathcal{X}(t|t-1),\ \forall \mathcal{K}_k \in \Xi_k,\ \forall k = 0,\ldots,k_o \end{array} \tag{4.22}$$

where

$$x(t+k,x,\mathcal{V},\mathcal{K}_k) = A^k x + R_k^v M\mathcal{V} + R_k^\xi \mathcal{K}_k \tag{4.23}$$

and

$$\begin{array}{rcl} R_k^v & = & \left[\begin{array}{cccc} B & AB & \ldots & A^{k-1}B \end{array}\right], \\ R_k^\xi & = & \left[\begin{array}{cccc} I_n & A & \ldots & A^{k-1} \end{array}\right], \\ M & = & \left[\begin{array}{c|c} \begin{array}{c} I_m \\ \vdots \\ I_m \end{array} & 0_{m(k-N_v)\times m(N_v-1)} \\ \hline \multicolumn{2}{c}{I_{mN_v}} \end{array}\right]. \end{array}$$

According to (4.23), after some algebraic manipulations, (4.22) can be rewritten as

$$\begin{array}{l} \mathcal{A}^x x + \mathcal{A}^v \mathcal{V} + \mathcal{A}^\xi \mathcal{K}_{k_o} \leq \mathcal{B},\ \mathcal{B} \in \mathbb{R}^h, \\ \forall x \in \mathcal{X}(t|t-1),\ \forall \mathcal{K}_{k_o} \in \Xi_{k_o}, \end{array} \tag{4.24}$$

where $h = q(k_o+1)$, and $\mathcal{A}^x \in \mathbb{R}^{h\times n}$, $\mathcal{A}^v \in \mathbb{R}^{h\times mN_v}$, $\mathcal{A}^\xi \in \mathbb{R}^{h\times nk_o}$ are suitably defined matrices.

Next Lemma 4.1 shows how to express (4.24) as a set of linear inequalities on the optimization vector $\mathcal{V}$.

Lemma 4.1 *Let*

$$V = \left\{v \in \mathbb{R}^v :\ P_1 v \leq P_2\right\},\ P_1 \in \mathbb{R}^{h\times v},\ P_2 \in \mathbb{R}^h$$

be bounded and nonempty. Denote by $[P]^i$ *the* i*-th row of* P *and by*

$$\max_{v\in V} Pv \triangleq \begin{bmatrix} \max_{v\in V}[P]^1 v \\ \vdots \\ \max_{v\in V}[P]^h v \end{bmatrix},$$

Then, the following sets

$$\begin{aligned} &\mathcal{D} = \{w \in \mathbb{R}^w :\ P_3 v + P_4 w \le P_5,\ \forall v \in V\}, \\ &P_3 \in \mathbb{R}^{k\times v},\ P_4 \in \mathbb{R}^{k\times w},\ P_5 \in \mathbb{R}^k \end{aligned}$$

and

$$\overline{\mathcal{D}} = \left\{ w \in \mathbb{R}^w :\ P_4 w \le P_5 - \max_{v\in V} P_3 v \right\},$$

are equal.

Proof. Since

$$P_4 w \le P_5 - \max_{v\in V} P_3 v \ \Rightarrow\ P_4 w \le P_5 - P_3 v,\ \forall v \in V,$$

it follows that $\overline{\mathcal{D}} \subseteq \mathcal{D}$. Conversely, let

$$v_i = \arg\min_{v\in V} [P_3]^i v,\ i = 1, \ldots, k.$$

Then,

$$\begin{aligned} \mathcal{D} &\subseteq \{w \in \mathbb{R}^w :\ P_3 v_i + P_4 w \le P_5,\ i = 1, \ldots, k\} \\ &\subseteq \left\{w \in \mathbb{R}^w :\ [P_3]^i v_i + [P_4]^i w \le [P_5]^i,\ i = 1, \ldots, k\right\} \\ &= \overline{\mathcal{D}}. \end{aligned}$$

□

The above lemma proves that $\mathcal{V}$ satisfies the constraints (4.24) iff

$$\mathcal{A}^v \mathcal{V} \le \mathcal{B} - \max_{x\in\mathcal{X}(t|t-1)} \mathcal{A}^x x - \max_{\mathcal{K}_{k_o}\in\Xi_{k_o}} \mathcal{A}^\xi \mathcal{K}_{k_o}. \tag{4.25}$$

Notice that the second term in the RHS of (4.25) depends on the current approximated state uncertainty set $\mathcal{X}(t|t-1)$, and hence it provides feedback from new output measurements. On the other hand, the third term can be computed off-line. Therefore, at each time instant t one has to solve h

linear programming problems in order to compute the second term in the RHS of (4.25). Finally, the additional constraint (4.13) can be rewritten as

$$\left[\begin{array}{cccc} A_c^x(I-A)^{-1} & 0 & \cdots & 0 \end{array} \right] \mathcal{V} \leq B_c - \underline{\delta} \tag{4.26}$$

In summary, the optimal vector $\mathcal{V}_t^*$ is evaluated by minimizing the quadratic cost function (4.21) subject to the linear constraints (4.25)-(4.26), which can be performed by standard QP algorithms. Then, the input $u(t)$ is selected according to (4.11)-(4.12).

Remark 2 An additional bounded noise $\gamma(t)$ on the constrained quantity $c(t)$ in (4.1) could be taken into account without changing the solution presented so far. In fact, because of the adopted worst-case approach, by letting

$$c(t) = \left[\begin{array}{c} x \\ u \end{array} \right] + \gamma(t),\ \gamma^- \leq \gamma(t) \leq \gamma^+,$$

the problem can be simply recast by computing off line a new vector $\hat{B}_c \triangleq B_c - \max_{\gamma^- \leq \gamma \leq \gamma^+} A_c \gamma$ to replace B_c in the above scheme.

4.5 Feasibility and Set-Membership State Estimation

In this section, we study the conditions which have to be fulfilled by the approximated state uncertainty set $\mathcal{X}(t|t-1)$ in order to guarantee *feasibility*. We distinguish between two different definitions.

Definition 4.2 *A vector $\mathcal{V}$ (and its constant extension $S_{\mathcal{V}}$) is said to be* virtually admissible at time t *if it fulfils constraints (4.13) and (4.22) $\forall x \in \mathcal{X}(t|t-1)$, $\forall \mathcal{K}_k \in \Xi_k$, and $\forall k \in \mathbb{Z}_+$.*

Definition 4.3 *A vector $\mathcal{V}$ (and its constant extension $S_{\mathcal{V}}$) is said to be* actually admissible at time t *if, by applying the command input $\{u(t+k)\}_{k=0}^{\infty} = S_{\mathcal{V}}$ to system (4.1), the corresponding evolution of the constrained vector satisfies $c(t+k) \in \mathcal{C}$, $\forall k \in \mathbb{Z}_+$.*

It is worth pointing out the difference between virtual and actual admissibility. Whilst virtual admissibility is an analytical property of vector

$\mathcal{V}$, actual admissibility depends a posteriori on the specific state $x(t)$ and disturbance sequence realization $\{\xi(k)\}_{k=t}^{\infty}$. Intuitively, if the approximated uncertainty state set $\mathcal{X}(t|t-1)$ is too small, it can happen that the actual state vector $x(t) \notin \mathcal{X}(t|t-1)$, and hence an input $\mathcal{V}$ is virtually but not actually admissible. Conversely, when $\mathcal{X}(t|t-1)$ is too large, it may not exist a vector $\mathcal{V}$ that satisfies (4.22) for every $x \in \mathcal{X}(t|t-1)$. However, for the particular experiment, actually admissible vectors $\mathcal{V}$ may exist. The result presented in this section provide a theoretical ground for the intuitions above.

As the next theorem points out, the relationship between the true state uncertainty set $\mathcal{X}^*(t|t-1)$ and its approximation $\mathcal{X}(t|t-1)$ is a key factor for guaranteeing actual admissibility.

Theorem 4.3 *Suppose that $\mathcal{V}$ is virtually admissible at time t. Then*

i) $\mathcal{X}^*(t|t-1) \subseteq \mathcal{X}(t|t-1) \implies S_{\mathcal{V}}^j = \{v(k+j)\}_{k=0}^{\infty}$, *is actually admissible at each time $t+j$, $j \in \mathbb{Z}_+$.*

ii) $\mathcal{X}^*(t|t-1) \not\subset \mathcal{X}(t|t-1) \implies S_{\mathcal{V}}$ *is not guaranteed to be actually admissible at time t.*

Proof. i) If $\mathcal{V}$ is virtually admissible at time t, then it satisfies

$$A_c^x x(t+k, x, \mathcal{V}, \mathcal{K}_k) + A_c^u v(k) \leq B_c, \tag{4.27}$$

$\forall x \in \mathcal{X}(t|t-1)$, $\forall \mathcal{K}_k \in \Xi_k$, and $\forall k \in \mathbb{Z}_+$. Since the current state $x(t) \in \mathcal{X}^*(t|t-1) \subseteq \mathcal{X}(t|t-1)$, if we apply the input $\{u(t+k)\}_{k=0}^{\infty} = S_{\mathcal{V}}$, the actual state at time $t+k$, will coincide with the predicted state corresponding to some state disturbance realization $\overline{\mathcal{K}}_k \in \Xi_k$, i.e.

$$x(t+k) = x(t+k, x(t), \mathcal{V}, \overline{\mathcal{K}}_k).$$

Hence, from (4.27) we get

$$A_c^x x(t+k) + A_c^u u(t+k) \leq B_c \quad \forall k \in \mathbb{Z}_+$$

and therefore $S_{\mathcal{V}}$ is actually admissible at time t. Actual admissibility of $S_{\mathcal{V}}^j$ at time $t+j$, $\forall j \in \mathbb{Z}_+$, trivially follows.

ii) If $\mathcal{V}$ is virtually admissible at time t, then it satisfies (4.25). Let us assume that equality holds for at least one row i, i.e.

$$[\mathcal{A}^v]^i\mathcal{V} = [\mathcal{B}]^i - \max_{x\in\mathcal{X}(t|t-1)} [\mathcal{A}^x]^i x - \max_{\mathcal{K}_{k_o}\in\Xi_{k_o}} [\mathcal{A}^\xi]^i\mathcal{K}_{k_o}. \tag{4.28}$$

Now suppose that the actual disturbance sequence is

$$\overline{\mathcal{K}}_{k_o} = \arg\max_{\mathcal{K}_{k_o}\in\Xi_{k_o}} [\mathcal{A}^\xi]^i\mathcal{K}_{k_o}, \tag{4.29}$$

and that the current state $x(t)$ satisfies

$$[\mathcal{A}^x]^i x(t) > \max_{x(t)\in\mathcal{X}(t|t-1)} [\mathcal{A}^x]^i x(t). \tag{4.30}$$

Inequality (4.30) can hold if $x(t) \notin \mathcal{X}(t|t-1)$, which is possible since $\mathcal{X}^*(t|t-1) \not\subset \mathcal{X}(t|t-1)$. From (4.28)-(4.30), we get

$$[\mathcal{A}^v]^i\mathcal{V} > [\mathcal{B}]^i - [\mathcal{A}^x]^i x(t) - [\mathcal{A}^\xi]^i\overline{\mathcal{K}}_{k_o}.$$

which corresponds to the violation of at least one constraint (4.22) at time $t+k$, i.e.

$$[A_c^x]^h x(t+k, x(t), \mathcal{V}, \overline{\mathcal{K}}_k) + [A_c^u]^h v(k) > [B_c]^h,$$

for some k, $0 \le k \le k_o$ and for some h, $1 \le h \le q$. If the applied control input is $\{u(t+k)\}_{k=0}^{\infty} = S_{\mathcal{V}}$, the actual value of $x(t+k)$ is equal to the predicted value $x(t+k, x(t), \mathcal{V}, \overline{\mathcal{K}}_k)$, and hence *actual* constraint violation occurs at time $t+k$. □

An immediate consequence of the above theorem is that outer approximations of the state uncertainty set must be chosen in order to guarantee actual admissibility. In fact, actual admissibility of $\mathcal{V}(t)$, $\forall t \in \mathbb{Z}_+$, is equivalent to fulfil constraint (4.4). Then, an important consequence of the previous result is the following.

Corollary 4.1 *If $\mathcal{X}(t|t-1) \supseteq \mathcal{X}^*(t|t-1)$, $\forall t \in \mathbb{Z}_+$, the control strategy (4.8)-(4.13) guarantees that $c(t) \in \mathcal{C}$, $\forall t \in \mathbb{Z}_+$.*

Notice that the parallelotopic approach adopted in the present chapter provides an outer state uncertainty set approximation, and hence the previous corollary holds.

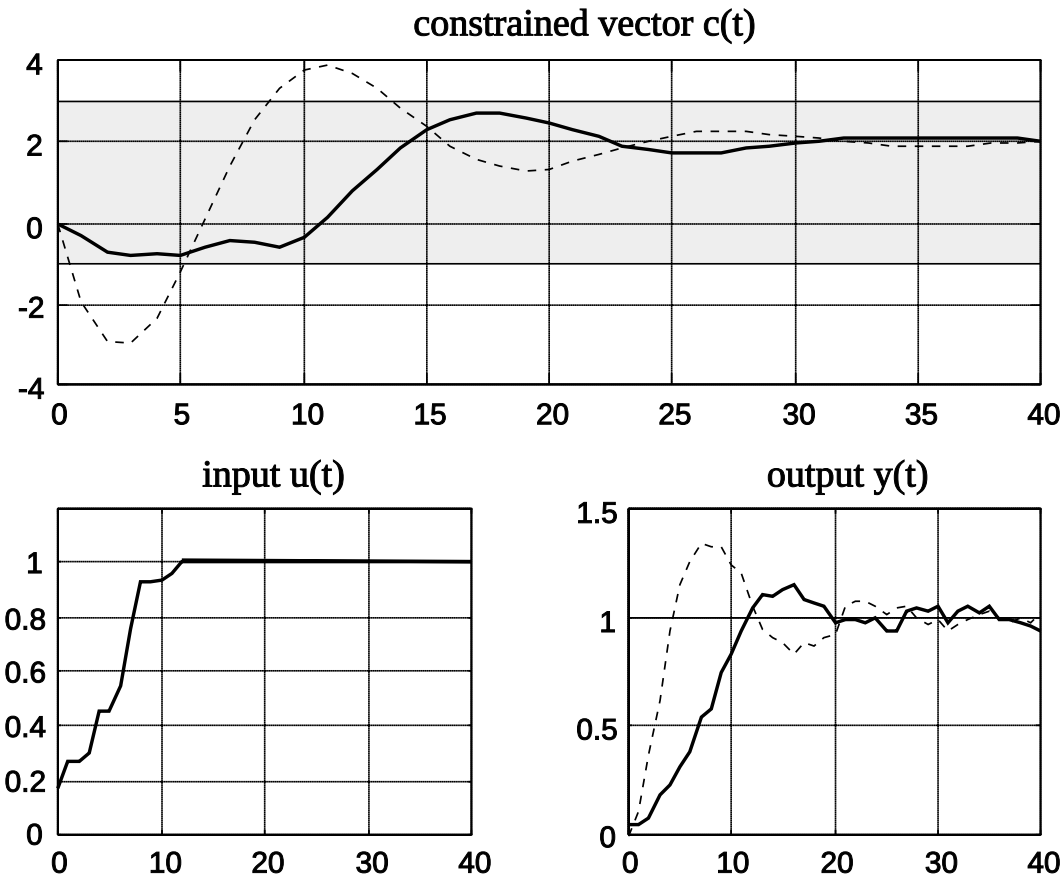


Figure 4.1: Closed loop behavior (thick lines) and unconstrained response (dashed lines) for $r(t) \equiv 1$.

4.6 Simulation Results

The proposed control strategy has been investigated by simulations on the following second order discrete time SISO system

$$\begin{aligned} x(t+1) &= \begin{bmatrix} 1.6463 & -0.7866 \\ 1 & 0 \end{bmatrix} x(t) + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u(t) + \xi(t) \\ y(t) &= \begin{bmatrix} 0.1404 & 0 \end{bmatrix} x(t) + \zeta(t) \\ c(t) &= \begin{bmatrix} -1.9313 & 2.2121 \end{bmatrix} x(t), \end{aligned} \tag{4.31}$$

whose y- and c-step responses are depicted in Fig. 4.1 (dashed lines). The transfer function from the input u to the constrained variable c is underdumped and nonminimum phase. In order to compress the dynamics of c within the range

$$\mathcal{C} = [-1, 3],$$

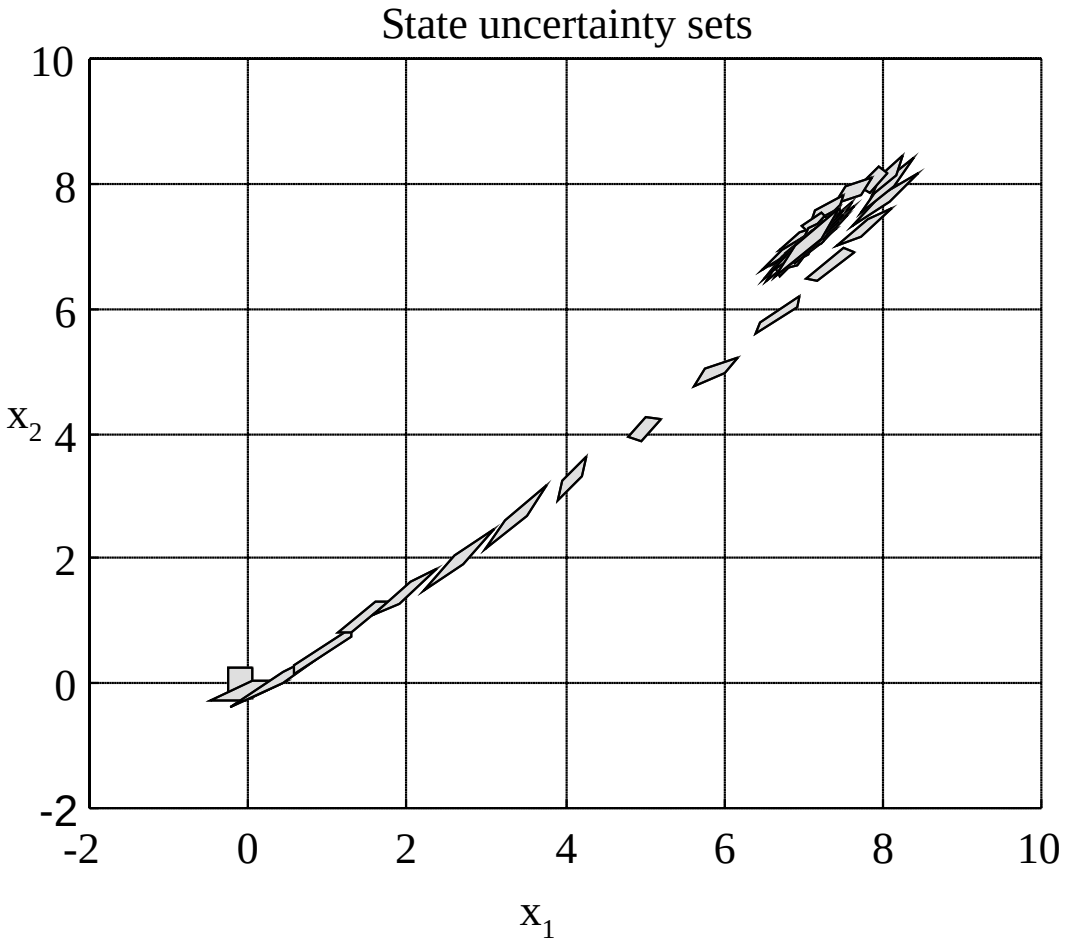


Figure 4.2: Evolution of the state uncertainty sets $\mathcal{X}(t|t)$.

and make the output y track the constant reference $r(t) \equiv 1$, we adopt the control law (4.8)-(4.13) along with the parameters $\Upsilon_1 = 1$, $\Upsilon_2 = 0.1$, $N_v = 2$, $\rho_1, \rho_2 \approx 0$. Also, we set $\delta \approx 0$ and $k_o = 16$, which have shown to guarantee a good constraint fulfillment, even if the conservative bounds (4.16)-(4.17) may not be satisfied. Fig. 4.1 shows the resulting trajectories (solid lines) when system (4.31) is affected by independent randomly generated disturbances $\|\xi(t)\|_\infty \leq 0.01$ and $|\zeta(t)| \leq 0.05$, for the a priori information set $\mathcal{X}(0|-1) = 0.25 \cdot [-1, 1] \times [-1, 1]$. Notice that the constraints are fulfilled at the price of a slower output response. Fig. 4.2 shows the evolution of the parallelotopic state uncertainty sets $\mathcal{X}(t|t)$, and Fig. 4.3 reports $\mathcal{X}(t|t)$, $\mathcal{X}(t|t-1)$ during the initial steps $t = 0, 1, 2$.

In Fig. 4.4 the effect of different state disturbance bounds is investigated. Due to the adopted worst-case approach, it results that as the size of the

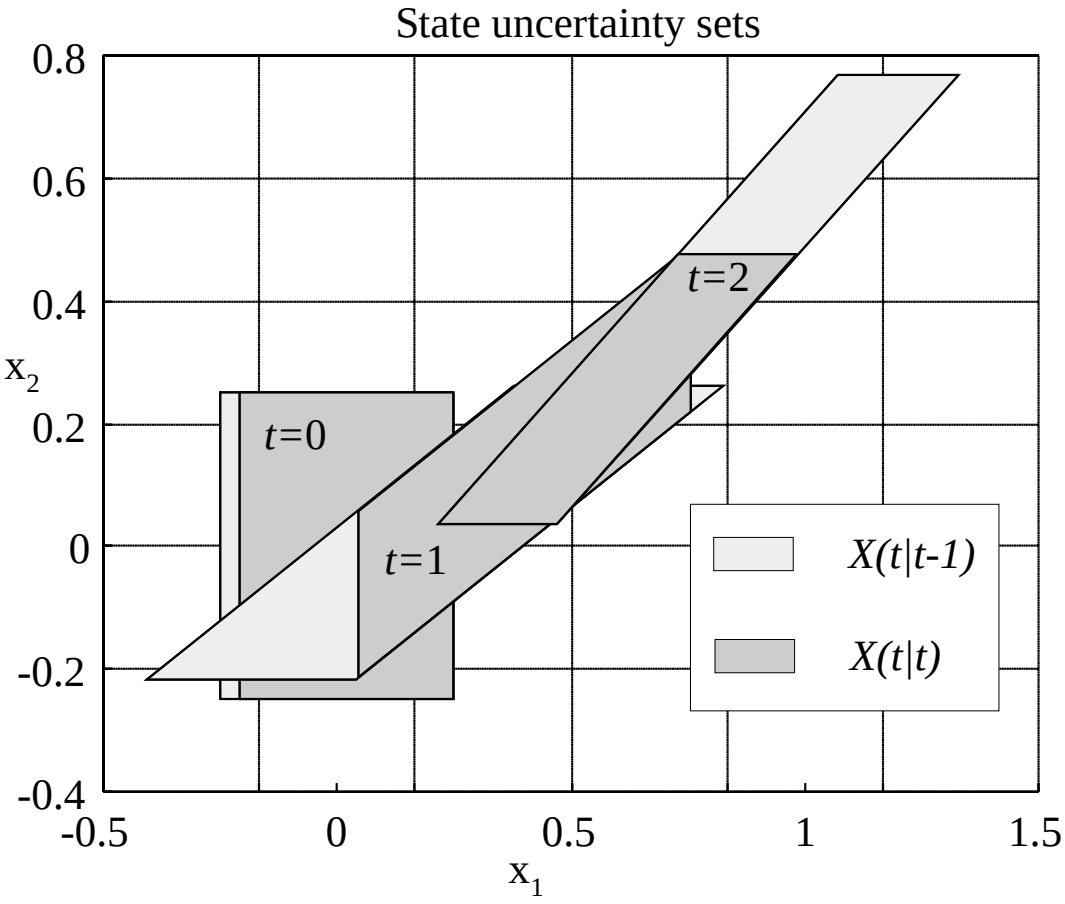


Figure 4.3: State uncertainty sets $\mathcal{X}(t|t)$, $\mathcal{X}(t|t-1)$, for $t = 0, 1, 2$.

disturbance increases, the constraints are fulfilled in a more conservative way, and the output dynamics gets slower.

4.7 Concluding Remarks

In this chapter we have combined a predictive control approach with a set-membership state estimation algorithm to satisfy input and/or state hard constraints. For the worst situation compatible with the available information, the resulting control law has been shown to guarantee constraint fulfillment, while preserving the asymptotical stability properties of the system. Encouraged by the promising simulation results, future research will concern tracking properties of predictive controllers based on set-valued observers.

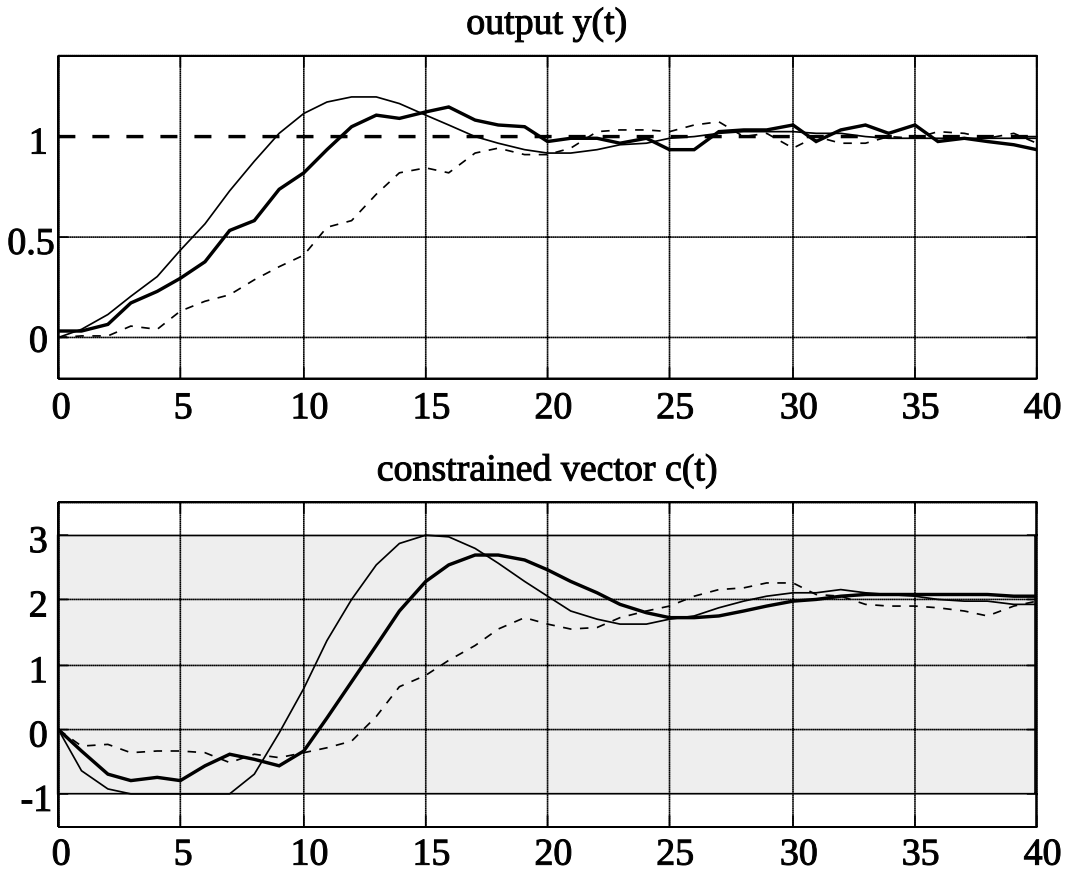


Figure 4.4: Effect of different disturbance intensities: no disturbance and known initial state $x(0)$ (thin line); $\|\xi(t)\|_\infty \leq 0.01$ (thick line); $\|\xi(t)\|_\infty \leq 0.04$ (dashed line).

Chapter 5

Nonlinear Systems

This chapter addresses the problem of satisfying pointwise-in-time input and/or state hard constraints in nonlinear control systems. The approach is based on conceptual tools of predictive control and consists of adding to a primal compensated nonlinear system a Reference Governor (RG). This is a discrete-time device which on-line handles the reference to be tracked, taking into account the current value of the state, in order to satisfy the prescribed constraints. The resulting hybrid system is proved to fulfill the constraints, as well as stability and tracking requirements.

5.1 Introduction

Predictive control, when applied to models described by nonlinear differential equations, requires the on-line solution of high dimensional nonlinear optimization problems. Then, the need of simpler schemes as the RGs presented in the previous chapters is of paramount importance. The RG operates in accordance with the receding horizon strategy mentioned in Chapter 1, by selecting on-line optimal reference input sequences which, in order to drastically reduce the required computational burden, are parameterized by a scalar quantity.

The present chapter extends these ideas developed in Chapter 2 to nonlinear continuous-time systems, and is organized as follows: In Sect. 5.2 we formulate the problem, specify the assumptions on the primal system, and present the RG strategy; Sect. 5.3 is devoted to the derivation of interesting properties of the RG; computational aspects are considered in Sect. 5.4; and

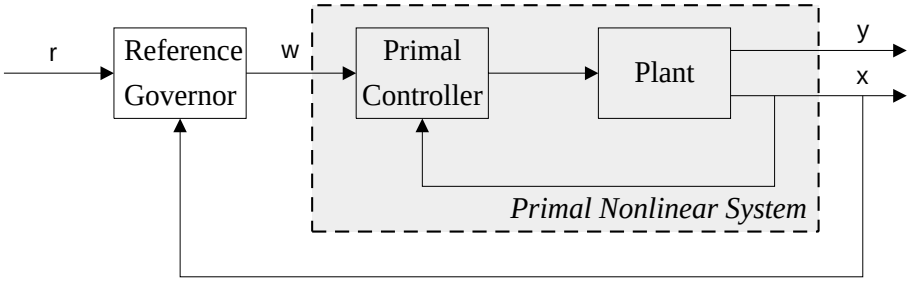


Figure 5.1: Reference governor for nonlinear systems.

a simulative example is reported in Sect 5.5.

5.2 Problem Formulation and Assumptions

Consider the following nonlinear system

$$\begin{cases} \dot{x}(t) &= \Phi(x(t), w(t)) \\ y(t) &= H(x(t), w(t)) \\ c(t) &= \begin{bmatrix} x(t) \\ w(t) \end{bmatrix} \end{cases} \tag{5.1}$$

representing in general a (nonlinear) plant under (nonlinear) feedback, where: $x(t) \in \mathbb{R}^n$ is the state vector, which collects both plant and controller states; $w(t) \in \mathbb{R}^p$ is the reference input, which in the absence of constraints would coincide with a desired reference $r(t) \in \mathbb{R}^p$; $y(t) \in \mathbb{R}^p$ is the output vector which shall track $r(t)$. Since input and/or state variables of the plant can be expressed as a function of $x(t)$ and $w(t)$, without loss of generality we define $c(t) \in \mathbb{R}^{n+p}$ as the vector to be constrained within a given set $\mathcal{C}$.

Assumption 5.1 *$\mathcal{C}$ is compact and has a nonempty interior.*

Compactness of $\mathcal{C}$ is non restrictive since in practice the desired references and state variables are bounded. Since we are interested in operating on vectors $[x'\ w']'$ in $\mathcal{C}$, we restrict the properties required by system (5.1) to the projections of $\mathcal{C}$ on the x-space

$$\mathcal{X} \triangleq \left\{ x \in \mathbb{R}^n : \ \exists w \in \mathbb{R}^p, \begin{bmatrix} x \\ w \end{bmatrix} \in \mathcal{C} \right\}$$

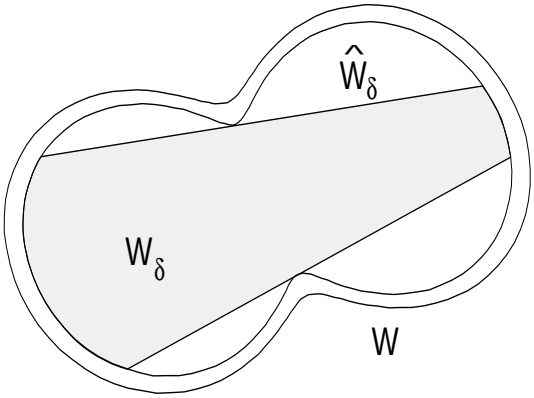


Figure 5.2: Sets $\mathcal{W}$, $\hat{\mathcal{W}}_\delta$, and $\mathcal{W}_\delta$.

and the projection on the w-space $\mathcal{W}$, which is defined analogously. It is easy to show that compactness of $\mathcal{C}$ implies that both $\mathcal{X}$ and $\mathcal{W}$ are compact. System (5.1) is required to fulfill the following assumptions.

Assumption 5.2 *$\forall w \in \mathcal{W}$, there exists a unique equilibrium state $x_w \in \mathcal{X}$.*

We denote by

$$X(\cdot) : \mathbb{R}^p \mapsto \mathbb{R}^n \tag{5.2}$$

the function implicitly defined by $\Phi(X(\cdot), \cdot) = 0$, and define $x_w \triangleq X(w)$, $c_w \triangleq [x_w' \; w']'$. Notice that in general $w \in \mathcal{W} \not\Rightarrow c_w \in \mathcal{C}$.

Assumption 5.3 *The mapping $\Phi(x, w) : \mathcal{X} \times \mathcal{W} \mapsto \mathbb{R}^n$ is continuous in (x, w).*

Consider now an arbitrarily small scalar $\delta > 0$, and define the following set

$$\widehat{\mathcal{W}_\delta} \triangleq \{w \in \mathcal{W} : \; B(c_w, \delta) \subseteq \mathcal{C}\}. \tag{5.3}$$

where $B(c_w, \delta)$ denotes the closed ball $\{c \in \mathbb{R}^{n+p} : \; \|c - c_w\| \leq \delta\}$. We restrict the set of reference inputs w which can be supplied by assuming, as depicted in Fig. 5.2, that

Assumption 5.4 *(Reference Input Conditioning) The class of reference inputs is restricted to a convex, nonempty, and compact set $\mathcal{W}_\delta \subseteq \widehat{\mathcal{W}_\delta} \subset \mathcal{W}$.*

Assumption 5.4 is needed to prevent that the border of $\mathcal{C}$ is approached in steady-state, and is required later to prove Theorem 5.2. The constraint

$c \in \mathcal{C}$ and the reference input conditioning can be summarized as the unique constraint

$$c \in \mathcal{C}_\delta \triangleq \mathcal{C} \bigcap (\mathcal{X} \times \mathcal{W}_\delta) \tag{5.4}$$

where $\mathcal{C}_\delta$ is compact. We fix $\delta > 0$ such that $\mathcal{C}_\delta$ is nonempty. In order to derive the properties proved in Sect. 5.3, system (5.1) is supposed to satisfy some extra assumptions.

Assumption 5.5 *For all piece-wise constant reference input signals $w(t) \in \mathcal{W}_\delta$, $t \in [0, +\infty)$, and for all initial states $x(0) \in \mathcal{X}$, there exists a unique solution $x(t, x(0), w(t))$ of (5.1) defined $\forall t \in [0, +\infty)$.*

In the following we shall denote by $x(t, x(0), w)$ the solution corresponding to a constant reference $w(t) \equiv w$, $\forall t \in [0, +\infty)$.

Assumption 5.6 *(Converging Input Converging State Stability) Let $w(t) \to w \in \mathcal{W}_\delta$, and each component of vector $w(t)$ be monotonically convergent. Then, $\forall x(0) \in \mathcal{X}$, $\lim_{t\to\infty} x(t, x(0), w(t)) = x_w$.*

In particular, Assumption 5.6 ensures that x_w is an asymptotically stable solution of $\dot{x}(t) = \Phi(x(t), w)$.

Assumption 5.7 *(Uniform-in-$\mathcal{W}_\delta$ Stability) Let $w(t) \equiv w \in \mathcal{W}_\delta$. Then, $\forall \lambda > 0$ there exists $\alpha(\lambda) > 0$ such that $\|x(0) - x_w\| \leq \alpha(\lambda) \Rightarrow \|x(t, x(0), w) - x_w\| \leq \lambda$, $\forall t \geq 0$, $\forall w \in \mathcal{W}_\delta$.*

The aim of this chapter is to design a *Reference Governor* (RG), a discrete-time device which, based on the current state $x(t)$ and desired reference $r(t)$, generates the reference input $w(t)$ so as to satisfy the constraint (5.4) and minimize the tracking error. As depicted in Fig. 5.1, the RG can be seen as a reference filter which modifies the desired reference r whenever this, if directly supplied to (5.1), causes constraint violation. Since the filtering action requires a finite computational time τ, the RG operates in discrete-time, in that it is applied every *RG period* T, $T \geq \tau$. The reference input $w(t)$ is generated on-line in a predictive manner: During the time interval $(t - \tau, t]$ a *virtual* reference input signal $\{w(t + \sigma)\}$, $\sigma \in (0, +\infty)$, is selected in such a way that the corresponding predicted evolution $c(t + \sigma, x(t), w(t + \sigma))$ lies within $\mathcal{C}_\delta$, $\forall \sigma > 0$. Then, according to a *receding horizon* strategy, the virtual signal is applied during the following interval $(t, t + T]$; at time $t + T$ a new selection

is performed. For the sake of notational simplicity, we shall consider hereafter $\tau = 0$. However, a significant delay $\tau > 0$ can be considered in the following developments by suitable changes.

For reasons that will be clearer soon, we restrict our attention to the class of virtual constant reference input signals, introduced by [GKT95], which are parameterized by the scalar β and defined by

$$\begin{cases} w(kT+\sigma,\beta) & \equiv & r(kT)+\beta[w((k-1)T)-r(kT)] \\ & \triangleq & w_\beta,\ \forall\sigma>0,\ k\in\mathbb{N}, \\ w(-T) & = & w_0 \end{cases} \tag{5.5}$$

where $\mathbb{N} = \{0, 1, \ldots\}$. At each time kT a parameter $\beta(kT) \in \mathbb{R}$, and the corresponding constant reference input $w_k \triangleq w_{\beta(kT)}$, are selected in accordance with the optimization criterion

$$\beta(kT) = \begin{cases} \arg\min_{\beta\geq 0}\beta^2 \\ \text{subj. to } c(kT+\sigma, x(kT), w(kT+\sigma,\beta)) \in \mathcal{C}_\delta, \\ \forall\sigma\in(0,+\infty) \end{cases} \tag{5.6}$$

and

$$w(t) \equiv w_k,\ \forall t \in (kT, (k+1)T]$$

Notice that by minimizing β^2 one attempts to minimize $\|w-r\|^2$, and therefore $\|y-r\|^2$. A parameter β, or a constant reference w, satisfying the constraints in (5.6) will be referred to as *admissible*.

Assumption 5.8 *(Feasible Initial Condition) The initial state $x(0)$ is such that there exists at least one admissible virtual constant reference input $w_0 \in \mathcal{W}_\delta$.*

For instance Assumption 5.8 is satisfied for an equilibrium states $x(0) = x_{w_0}$ corresponding to $w_0 \in \mathcal{W}_\delta$.

5.3 Main Results

Lemma 5.1 *Suppose that Assumptions 5.1–5.3 hold. Then the function $X : \mathcal{W} \mapsto \mathcal{X}$ defined in (5.2) is continuous.*

Proof. Consider a generic $w^* \in \mathcal{W}$. By contradiction, suppose $\exists \epsilon \geq 0$ such that $\forall i \in \mathbb{N}$ there exists a reference input $w_i \in \mathcal{W}$, $\|w_i - w^*\| \leq \frac{1}{i}$, and $\|x_{w_i} - x_{w^*}\| > \epsilon$, $x_{w_i} = X(w_i)$, $x_{w^*} = X(w^*)$. By Assumption 5.2, the sequence $\{x_{w_i}\} \subseteq \mathcal{X}$. By Assumption 5.1, $\mathcal{X}$ is compact, and hence there exists a subsequence $\{x_{w_j}\}$ converging to a point $\bar{x} \in \mathcal{X}$, with $\|\bar{x} - x_{w^*}\| > \epsilon$, or equivalently $\bar{x} \neq x_{w^*}$. Because of continuity of Φ, $0 = \lim_{j\to\infty} \Phi(x_{w_j}, w_j) = \Phi(\bar{x}, w^*)$, which contradicts Assumption 5.2. □

Next proposition shows that, for constant desired reference trajectories, the RG yields a converging reference input.

Proposition 5.1 *Suppose that $r(t) \equiv r$, $\forall t \geq 0$, and Assumptions 5.3–5.4 hold. Then there exists $\lim_{t\to\infty} w(t) \triangleq w_\infty \in \mathcal{W}_\delta$. In addition, each component of vector $w(t)$ is monotonically convergent.*

Proof. If $w_0 = r$, then $\beta(kT) = 0$ is admissible, $\forall k \in \mathbb{N}$. Therefore, $w(t) = r$, $\forall t > 0$, and $w_\infty = r$ (the RG behaves as an all-pass filter). Suppose $w_0 \neq r$. Since $\beta(kT) \geq 0$, $w_k = r + \frac{d_k}{\|w_0 - r\|}[w_0 - r]$, where $d_k \triangleq \|w_k - r\|$. By construction, at time $(k+1)T$, $\beta = 1$ is admissible, and hence $\beta((k+1)T) \leq 1$. Then, $0 \leq d_{k+1}^2 = \beta^2((k+1)T)d_k^2 \leq d_k^2$, $\forall k \in \mathbb{N}$, and hence there exists $d_\infty = \lim_{k\to\infty} d_k$. Consequently, $\lim_{t\to\infty} w(t) = w_\infty \triangleq r + \frac{d_\infty}{\|w_0 - r\|}[w_0 - r]$. By compactness of $\mathcal{W}_\delta$, $w_\infty \in \mathcal{W}_\delta$ follows. □

Next Lemma 5.2 and Prop. 5.2 show that w_∞ is the admissible reference input which is closest to r along the line segment $\rho w_0 + (1-\rho)r$, $\rho \in [0, 1]$.

Lemma 5.2 *Suppose that Assumptions 5.1–5.5 and 5.7 hold. Consider two reference inputs w_a, $w_b \in \mathcal{W}_\delta$, $w_a \neq w_b$. Let $x(kT) = x_{w_a} + \Delta x \in \mathcal{X}$, and let η such that $B(c_{w_a}, \eta) \subseteq \mathcal{C}$. Then there exists a $\bar{\gamma} > 0$, dependent on w_a and η, such that reference input $w_a + \gamma(w_b - w_a)$ is admissible for all $\|\Delta x\| \leq \frac{1}{2}\alpha(\eta/2)$, and for all $0 \leq \gamma \leq \bar{\gamma}$.*

Proof. Let $\alpha = \alpha(\eta/2)$ in accordance with Assumption 5.7. By continuity of the mapping $X(w)$ in w_a there exists a $\lambda = \lambda(w_a, \eta)$, $0 < \lambda < \|w_b - w_a\|$ such that, $\forall w \in \mathcal{W}_\delta$, $\|w_a - w\| \leq \lambda \Rightarrow \|x_{w_a} - x_w\| \leq \frac{\alpha}{2}$. Let $\bar{\gamma} \triangleq \frac{\lambda}{\|w_b - w_a\|}$ and γ such that $0 < \gamma \leq \bar{\gamma}$; by Assumption 5.4, the reference input $w_\gamma \triangleq w_a + \gamma(w_b - w_a)$ lies within $\mathcal{W}_\delta$. By taking $\|\Delta x\| \leq \frac{\alpha}{2}$, $\|x(t) - x_{w_\gamma}\| \leq \|x_{w_a} - x_{w_\gamma}\| + \|\Delta x\| \leq \alpha$, and by Assumption 5.7 $\|c(kT + \sigma, x(kT), w_\gamma) - c_{w_\gamma}\| = \|x(kT + \sigma, x(kT), w_\gamma) - x_{w_\gamma}\| \leq \eta$, $\forall \sigma > 0$. Therefore, each reference w_γ is admissible at time kT. □

Proposition 5.2 *Suppose that $r(t) \equiv r$, $\forall t \geq 0$, and Assumptions 5.1–5.8 hold. Then $\lim_{t\to\infty} w(t) = w_r \in \mathcal{W}_\delta$ with*

$$w_r = \arg \min_{\rho\in[0,1]} \begin{cases} \|w - r\| \\ \text{subject to } w = r + \rho[w_0 - r] \in \mathcal{W}_\delta \end{cases} \tag{5.7}$$

where $w_0 \in \mathcal{W}_\delta$ is an admissible reference input at time $t = 0$.

Proof. By Prop. 5.1 there exists $\lim_{t\to\infty} w(t) = w_\infty \in \mathcal{W}_\delta$, and the convergence is component-by-component monotonic. Suppose by contradiction $w_\infty \neq w_r$. By Assumption 5.6, there exists a time t_0 such that $\|x(t_0, x(0), w(t_0)) - x_{w_\infty}\| \leq \alpha\left(\frac{\delta}{2}\right)$. Hence, by Lemma 5.2, there exists a constant $\bar{\gamma} > 0$ such that $w_\gamma \triangleq w_\infty + \gamma(w_r - w_\infty)$ is admissible at time t_0, $\forall\gamma$ such that $0 < \gamma \leq \bar{\gamma}$. Then, $\|w(t) - r\| \leq \|w_\gamma - r\|$. Since r, $w(t)$, w_γ, w_∞ are collinear, it follows that $\|w(t) - w_\infty\| = \|w(t) - w_\gamma\| + \|w_\gamma - w_\infty\| \geq \gamma\|w_r - w_\infty\| > 0$, $\forall t \geq t_0$, which contradicts the hypothesis $\lim_{t\to\infty} w(t) = w_\infty$. □

Lemma 5.3 *Under the hypotheses of Prop. 5.2, there exists a stopping time t_s such that $w(t) = w_r$ for all $t \geq t_s$.*

Proof. Since by Prop. 5.2 $\lim_{k\to\infty} w_k = w_r$, by Assumptions 5.6–5.7 there exists an index M such that $\|x(MT, x(0), w(MT)) - x_{w_r}\| < \alpha(\delta)$ which implies $\|c(MT+\sigma, x(MT), w_r) - c_{w_r}\| = \|x(MT+\sigma, x(MT), w_r) - x_{w_r}\| \leq \delta$, $\forall\sigma \in \mathbb{R}_+$ or, equivalently, that w_r is admissible at time $t_s \triangleq MT$. □

Next Theorem 5.1 summarizes the previous results.

Theorem 5.1 *Suppose $r(t) \equiv r$, $\forall t \geq 0$, and Assumptions 5.1–5.8 hold. Then, after a finite time t_s the RG generates a constant reference input $w(t) \equiv w_r$, where w_r is given by (5.7). Consequently, system (5.1) is asymptotically driven from $x(0)$ to x_{w_r} with no constraint violation.*

Notice that, when $r \in \mathcal{W}_\delta$, the RG has no effect on the asymptotic behavior of $y(t)$, which instead depends on the original tracking properties of the primal system (5.1).

5.3.1 Finite Constraint Horizon

The optimization criterion (5.6) requires that the constraint $c(kT + \sigma, x(kT), w_\beta) \in \mathcal{C}_\delta$ is checked for all $\sigma > 0$. In this section, we show that it suffices to verify this condition over a finite prediction horizon $(0, T_\infty]$.

Definition 5.1 *(Constraint Horizon) The constraint horizon T_∞ is defined as the shortest prediction horizon such that $c(t+\sigma, x(t), w) \in \mathcal{C}_\delta,\ \forall \sigma > 0 \Leftrightarrow c(t+\sigma, x(t), w) \in \mathcal{C}_\delta,\ \forall 0 < \sigma \leq T_\infty,\ \forall x(t) \in \mathcal{X},\ \forall w \in \mathcal{W}_\delta$.*

In order to prove that such a T_∞ exists, we recall the following result [CL55, pp. 58-60] for time-invariant systems.

Result 5.1 *(Variation of Solutions w.r.t. Initial Conditions and Parameters) Consider generic $x^*(0) \in \mathcal{X}$ and $w^* \in \mathcal{W}_\delta$. Let $\eta > 0$, $\eta \leq \delta$, and D_η the set of all c satisfying $x \in \mathcal{X}$, $w \in B(w^*, \eta) \subseteq \mathcal{W}$. Suppose Φ continuous and bounded on D_η. Then, there exists a $\gamma > 0$ such that for all $x(0)$, w satisfying $\|x(0) - x^*(0)\| < \gamma,\ \|w - w^*\| < \gamma$ the solution $x(t, x(0), w)$ exists over any bounded interval $[0, T^*]$, and as $(x(0), w) \to (x^*(0), w^*)$, $x(t, x(0), w) \to x(t, x^*(0), w^*)$ uniformly over $[0, T^*]$.*

Note that Assumption 5.3 and compactness of $\mathcal{X}$ and $\mathcal{W}$ imply that $\Phi(x, w)$ is bounded on $\mathcal{X} \times \mathcal{W}$.

When $w(t) \equiv w$, the following Theorem 5.2 proves that, for a fixed scalar $\lambda > 0$, the state $x(t)$ converges to the ball $B(x_w, \lambda)$ in a finite time T which is not dependent of the initial state $x(0) \in \mathcal{X}$ and reference input $w \in \mathcal{W}_\delta$.

Theorem 5.2 *Let Assumptions 5.1, 5.3, and 5.5–5.7 be satisfied. Then for all $\lambda > 0$ there exists a finite time $T(\lambda)$ such that, $\forall c(0) = [x'(0)\ w']' \in \mathcal{C}_\delta$,*

$$\|x(t, x(0), w) - x_w\| \leq \lambda,\ \forall t \geq T(\lambda). \tag{5.8}$$

Proof. By Assumption 5.6 it is immediate to show that (5.8) is verified for some $T(\lambda, x(0), w)$. Suppose by contradiction that $\sup_{c(0) \in \mathcal{C}_\delta} T(\lambda, w, x(0)) = +\infty$. Then, there exists a sequence $\{c_i(0)\}_{i=0}^{\infty}$ such that $\lim_{i \to \infty} T(\lambda, x_i(0), w_i) = +\infty$. By compactness of $\mathcal{C}_\delta$, there exists a subsequence $\{c_j(0)\}_{j=0}^{\infty}$ converging to a point $c^*(0) \in \mathcal{C}_\delta$. By Assumption 5.7, there exists an $\alpha = \alpha(\lambda)$, independent of w, such that $\|x(t_0) - x_w\| \leq \alpha \Leftrightarrow \|x(t, x(t_0), w) - x_w\| \leq \lambda,\ \forall t \geq t_0$. Let $w \in \mathcal{W}_\delta$ such that $\|x_w - x_{w^*}\| < \alpha/3$, $T^* \triangleq T(\alpha/3, x^*(0), w^*)$ and define $x(t) \triangleq x(t, x(0), w)$, $x^*(t) \triangleq x(t, x^*(0), w^*)$. By Result 5.1, setting $\eta \triangleq \delta/2$, there exists a $\gamma = \gamma(T^*, \alpha/3)$ such that $\|x(0) - x^*(0)\| < \gamma$, $\|w - w^*\| < \gamma \Rightarrow \|x(t) - x^*(t)\| < \frac{\alpha}{3},\ \forall t \in [0, T^*]$. Then, $\|x(T^*) - x_w\| \leq$

$\|x(T^*) - x^*(T)\| + \|x^*(T) - x_{w^*}\| + \|x_{w^*} - x_w\| \leq \frac{\alpha}{3} + \frac{\alpha}{3} + \frac{\alpha}{3} \leq \alpha$, which implies $\|x(t) - x_w\| \leq \lambda$ for all $t \geq T^*$. Hence, $T(\lambda, x(0), w) \leq T^*$. In conclusion, there exists an index j_0 such that, $\forall j \geq j_0$, $\|x_j(0) - x^*(0)\| \leq \gamma$, $\|w_j - w^*\| \leq \gamma$, $\|x_{w_j} - x_{w^*}\| \leq \frac{\alpha}{3}$, and $T(\lambda, x_j(0), w_j) \leq T^*$. This contradicts the assumption $\lim_{j\to\infty} T(\lambda, x_j(0), w_j) = +\infty$ □

By (5.3) and Assumption 5.4, Theorem 5.2 proves that T_∞ exists and satisfies the inequality $T_\infty \leq T(\delta)$.

5.4 Computations

In order to implement the RG described in the previous sections, the optimization (5.6) is solved by using a bisection algorithm over the interval $[0, 1]$. Testing the admissibility of a given β requires the numerical integration of (5.1) from initial state $x(kT)$. The fulfillment of the constraints $c(kT + \sigma, x(kT), w_\beta) \in \mathcal{C}$ is checked at integration steps. Let N denote the number of parameters β which can be evaluated during one RG period T. For a given T, N is determined by both the desired integration accuracy and the constraint horizon T_∞. Since admissibility of $\beta = 0$ is always tried first, the optimal $\beta(kT)$ is evaluated with a worst case precision of $2^{-(N-1)}$. Because $\mathcal{C}$ is generic and the plant is nonlinear, no convexity properties of the set of admissible β can be invoked. Then, the adopted bisection algorithm only provides local minima. However, this does not affect the convergence results proved in Sect. 5.3. In fact, if at time t after N evaluations no admissible $\beta < 1$ is found, $\beta(t) = 1$ is selected, which is admissible by construction. Consequently, Prop. 5.1 still holds. By Lemma 5.2, an admissible interval $[1 - \bar{\gamma}, 1]$ can be found after a finite time. For N large enough, the bisection method can therefore find admissible $\beta < 1$, and hence the proof of Prop. 5.2 holds. Since $\beta = 0$ is always tested, Lemma 5.3 and Theorem 5.1 hold as well. It is clear that if global minimization procedures were adopted in selecting $\beta(t)$, better tracking properties might be achieved, at the expense of an increased computational effort.

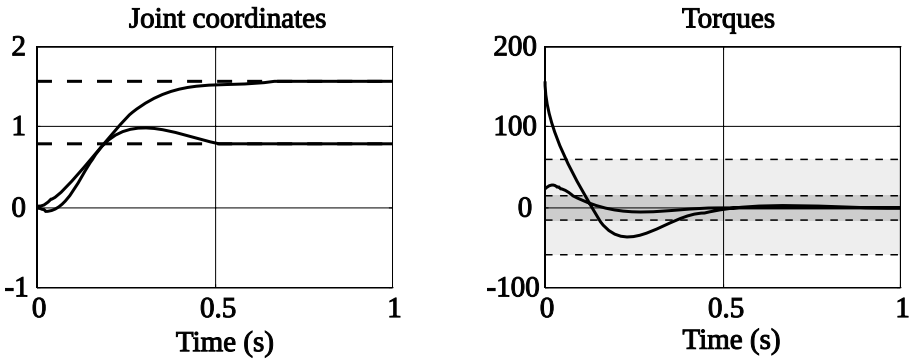


Figure 5.3: Response without RG.

5.5 An Example

The performance of the RG presented in the previous sections has been tested by computer simulations on a two link robot moving on a horizontal plane.

5.5.1 Nonlinear Model

Each joint is equipped with a motor for providing input torque, encoders and tachometers are used for measuring the joint positions θ_1, θ_2, and velocities $\dot{\theta}_1$, $\dot{\theta}_2$. By using Lagrangian equations, and by setting

$$x = \begin{bmatrix} \theta_1 \\ \dot{\theta}_1 \\ \theta_2 \\ \dot{\theta}_2 \end{bmatrix}, \; y = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix}, \; \mathcal{T} = \begin{bmatrix} \mathcal{T}_1 \\ \mathcal{T}_2 \end{bmatrix} \; w = \begin{bmatrix} \theta_{1d} \\ \theta_{2d} \end{bmatrix},$$

where θ_{1d}, θ_{2d} denote the desired values for joint positions and $\mathcal{T}_1$, $\mathcal{T}_2$ the motor torques, the dynamic model of the robot can be expressed as

$$H(x) \begin{bmatrix} \dot{x}_2 \\ \dot{x}_4 \end{bmatrix} + C(x) \begin{bmatrix} x_2 \\ x_4 \end{bmatrix} = \mathcal{T} \tag{5.9}$$

where

$$\begin{aligned} H(x) &= \begin{bmatrix} h_{11} & h_{12} \\ h_{12} & h_{22} \end{bmatrix} \\ h_{11} &= m_1 l_{c_1}^2 + I_1 + m_2[l_1^2 + l_{c_2}^2 + 2 l_1 l_{c_2} \cos(x_3)] + I_2 \\ h_{12} &= m_2 l_1 l_{c_2} \cos(x_3) + m_2 l_{c_2}^2 + I_2 \\ h_{22} &= m_2 l_{c_2}^2 + I_2 \end{aligned}$$

$$C(x) = m_2 l_1 l_{c_2} \sin(x_3) \begin{bmatrix} -x_4 & -x_2 - x_4 \\ x_2 & 0 \end{bmatrix}$$

Individual joint PD controllers

$$\mathcal{T} = - \begin{bmatrix} k_{p1}(x_1 - w_1) + k_{d1} x_2 \\ k_{p2}(x_3 - w_2) + k_{d2} x_4 \end{bmatrix} \tag{5.10}$$

provide reference tracking. As a general rule to design controllers to be used in connection with a RG, in order to maximize the properties of tracking one should try to select a primal controller which provides a fast closed-loop response of system (5.1). Usually this corresponds to large violations of the constraints, which therefore can be enforced by inserting a RG. In order to show that system (5.9)–(5.10) fulfills the required assumptions, consider the following function

$$\begin{aligned} V(x) &= \frac{1}{2} \begin{bmatrix} x_2 \\ x_4 \end{bmatrix}' H(x) \begin{bmatrix} x_2 \\ x_4 \end{bmatrix} + \\ &\quad \frac{1}{2} \begin{bmatrix} w_1 - x_1 \\ w_2 - x_3 \end{bmatrix}' K_p \begin{bmatrix} w_1 - x_1 \\ w_2 - x_3 \end{bmatrix}, \\ K_p &= \begin{bmatrix} k_{p1} & 0 \\ 0 & k_{p2} \end{bmatrix} > 0 \end{aligned}$$

which is a Lyapunov function for (5.9)–(5.10) [AS86]. Since its derivative along the trajectories of the system

$$\begin{aligned} \dot{V}(x) &= - \begin{bmatrix} x_2 \\ x_4 \end{bmatrix}' K_d \begin{bmatrix} x_2 \\ x_4 \end{bmatrix} \leq 0, \\ K_d &= \begin{bmatrix} k_{d1} & 0 \\ 0 & k_{d2} \end{bmatrix} > 0, \end{aligned}$$

and $V(x) = 0$ iff $x = [w_1 \ 0 \ w_2 \ 0]'$, Assumption 5.2 is satisfied. Moreover, in practice the reference input $w(t)$ is expressed by a finite numerical precision; therefore, if $w(t)$ monotonically tends toward w, after a finite time $w(t) \equiv w$, and hence Assumption 5.6 is verified as well. The fulfillment of Assumption 5.7 is proved as follows. By contradiction, suppose that there exists a $\lambda > 0$ such that, $\forall \alpha > 0$, there exists w and t_w with $\|x(0) - x_w\| \leq \alpha$ and $\|x(t_w, x(0), w) - x_w\| > \lambda$. Since $\gamma_1 I \leq H(x) \leq \gamma_2 I$ for some positive γ_1, γ_2, by denoting by

$\lambda_m(K_p)$, $\lambda_M(K_p)$ respectively the minimum and maximum eigenvalue of K_p, and by setting $\gamma_3 = \min\{\lambda_m(K_p), \gamma_1\}$, $\gamma_4 = \max\{\lambda_M(K_p), \gamma_3\}$, it follows that $\|x(t_w, x(0), w) - x_w\| \leq \frac{2}{\gamma_3} V(x(t_w)) \leq \frac{2}{\gamma_3} V(x(0)) \leq \frac{\gamma_4}{\gamma_3}\alpha$ for any arbitrary positive α, a contradiction.

5.5.2 Simulations

Simulations have been carried out with the system parameters reported in [KS94]. On-line optimization has been performed by using the bisection method mentioned in Sect. 5.4, and a standard fourth-order Runge-Kutta method with adaptive stepsize control has been adopted for numerical integration. Fig. 5.3 shows the closed loop system behavior for a constant desired reference $r_1(t) \equiv \frac{\pi}{2}$, $r_2(t) \equiv \frac{\pi}{4}$, $t \in \mathbb{R}_+$, in the absence of the RG. In order to bound the input torques within the range

$$|\mathcal{T}_1| \leq 60 \text{ Nm}, \ |\mathcal{T}_2| \leq 15 \text{ Nm}, \tag{5.11}$$

which has been represented by shadowed areas in Fig. 5.3, the RG is applied. The initial condition $\theta_1(0) = \theta_2(0) = 0$, $\dot{\theta}_1(0) = \dot{\theta}_2(0) = 0$ and $w_0 = [0\ 0]'$ satisfy Assumption 5.8. A RG period $T = 0.001$ s, a constraint horizon $T_\infty = 0.4$ s, $N = 10$ admissibility evaluations per period, and $\delta \approx 0$ are selected as parameters of the RG. The set $\mathcal{C}$ is determined by (5.11) and by further limiting the state and reference input in such a manner that only constraints (5.11) become active. The resulting trajectories are depicted in Fig. 5.4. In Fig. 5.5, the RG period is increased to $T = .05$ s, which causes a transient chatter on the input torques. The further constraint

$$|\theta_1 - \theta_2| \leq 0.2 \text{ rad}$$

is taken into account by the RG and the related simulated trajectories are depicted in Fig 5.6, with $r_1(t) = r_2(t) \equiv \frac{\pi}{4}$, $T = 0.001$ s. The slight chatter on the β and torque trajectories is caused by the approximations involved in the optimization procedure described in Sect. 5.4. The results described above were obtained on a 486 DX2/66 personal computer, using Matlab 4.2 and Simulink 1.3 with embedded C code. The CPU time required by the RG to select a single $\beta(t)$ ranged between 7 and 18 ms.

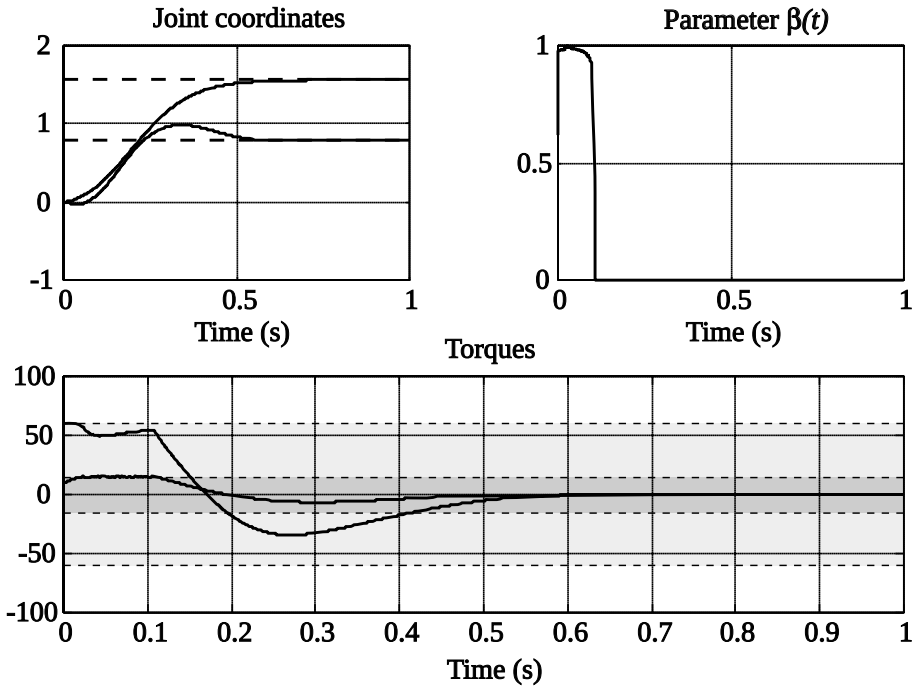


Figure 5.4: Response with RG ($T = 0.001$ s).

5.6 Concluding Remarks

For a broad class of nonlinear continuous-time systems and input/state hard constraints, this chapter has addressed the RG problem, viz. the one of filtering the desired reference trajectory in such a way that a nonlinear primal compensated control system can operate in a stable way with satisfactory tracking performance and no constraint violation. The resulting computational burden turns out to be moderate and the related operations executable with current computing hardware. Alternatively, in some applications, the trajectory generated by the RG can be computed off line and stored for subsequent task executions. Future developments of this research will be addressed towards numerical criteria for the determination of the constraint horizon, and to an independent parameterization of the components of the reference.

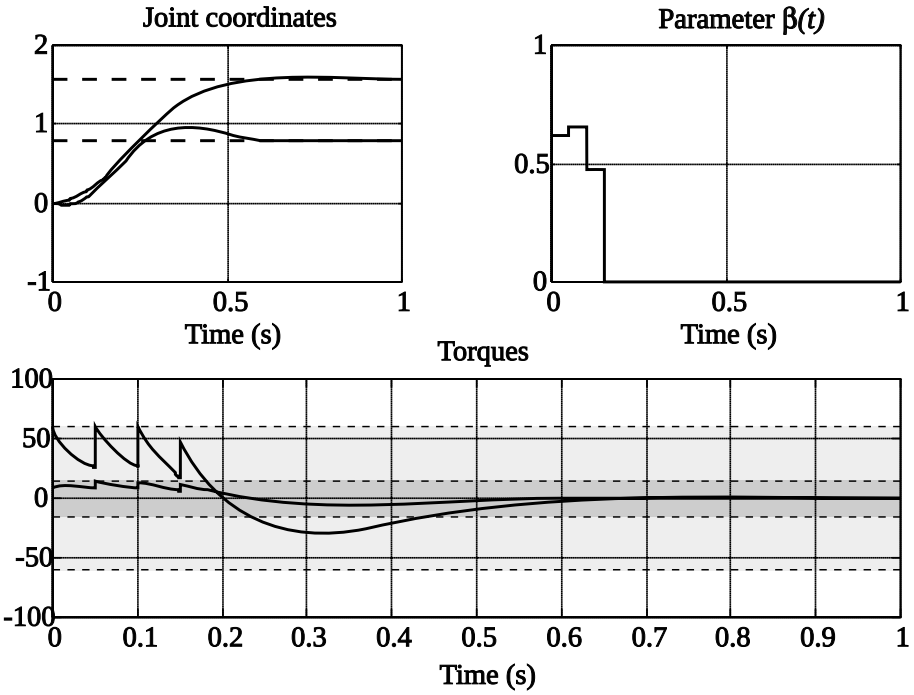


Figure 5.5: Response with RG ($T = 0.05$ s).

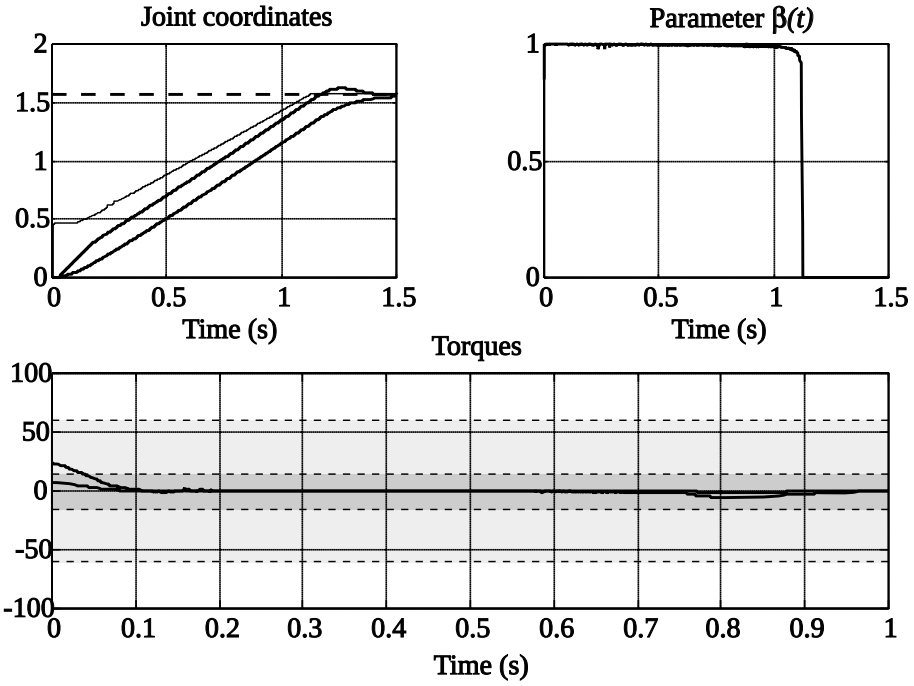


Figure 5.6: Response with RG, torque constraints, and the constraint $|\theta_1 - \theta_2| \leq 0.2$ rad. The generated reference input is depicted (thin line) together with the joint trajectories (thick lines)

Chapter 6

Robotic Systems

This chapter addresses the problem of satisfying input/state constraints for robotic systems tracking a given geometric path. According to a prediction of the evolution of the robot from the current state, a discrete-time device called *Path Governor* (PG) generates on line a suitable time-parameterization of the path to be tracked, by solving at fixed intervals a constrained scalar optimization problem. Higher level switching commands are also taken into account by simply associating a different optimization criterion to each mode of operation. Experimental results are reported for a 3-DOF PUMA 560 manipulator subject to absolute position error, Cartesian velocity, and motor voltage constraints.

6.1 Introduction

Tracking a given geometric path in the presence of physical constraints is a task which often occurs in robotic applications. Physical limits may involve: joint-motor voltages, currents, and consequently torque saturations; power consumption; joint positions, which are limited for reasons of mechanical construction; and specifications on manufacture tolerances, which impose the fulfilling of constraints on the tracking error. Usually, this leads to *off-line* constrained optimal path planning problems, for instance minimal time problems with torque constraints. Many of these methods [BDG85, SM85, DN90] translate the original physical limits on the torques, reference trajectory, and joint coordinates q into constraints on the only reference trajectory r, by substituting q with r in the equations of the robot. For example, torque

saturations are converted in constraints on the velocity and acceleration of the reference trajectory. However, this entails in assuming perfect tracking, namely $q = r$, and consequently the closed-loop dynamics induced by the adopted feedback controller is neglected. Although this approach leads to computationally efficient strategies, it turns out to be inadequate in several applications. For instance, problems with constraints on the tracking error $q-r$ cannot be handled, because this would result constantly null. In addition, in the case of saturating joint torques, these methods do not leave any room for the amount of torque required by the feedback law; therefore, even if nominally satisfied, during the execution of the task the robot could require a total torque exceeding the limits. More complicated constrained path-planning problems can be formulated taking into account the feedback torque controller, and therefore the overall closed-loop dynamics; however, in most cases the resulting computational burden turns out to be huge, and the presence of measurement noise and unmodeled dynamics frustrates the effort of such an accurate formulation.

Based on the *time-scaling* concept introduced by [Hol84] (and extended for multi-robot configurations by [MA91]), [DN90] and, later, [KKP96] suggested *on-line* trajectory time-scaling algorithms which take into account the overall closed-loop dynamics. Basically, given a desired path $r(s)$, the path acceleration $\ddot{s}(t)$ is selected on-line within a range interval directly derived by the given torque limits and measurements. However, these methods are limited to problems with *input* constraints, and require a previously computed nominal optimal time-parameterization $s_n(t)$ of the desired path.

For a given desired path $r(s)$ to be tracked by the joint coordinates of the robot, the approach described in this chapter copes with generic *input/state* constraints—e. g. tracking error, torque, and/or position constraints— and does not require any previous time-parameterization.

We assume that a feedback controller has been already designed in order to guarantee, in the absence of constraints, nice stability and tracking properties ($q \approx r$). However, fast reference signals may result in a violation of the constraints. In order to avoid this, we add to the pre-designed control system a new discrete-time device, denominated *Path Governor* (PG), which, on the basis of current position and velocity measurements, generates on line a suitable parameterization $s(t)$ of the desired reference $r(s)$, as depicted in

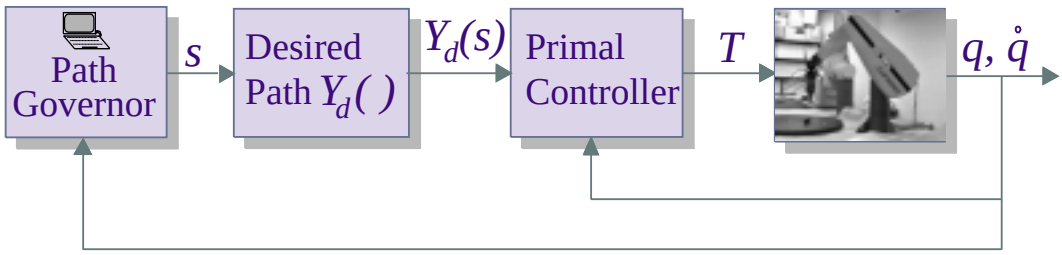


Figure 6.1: Path tracking with on-line path parameterization

Fig 6.1.

The PG attempts to reduce the computational complexity in two ways: first, only a portion of the desired path is considered at a time; second, the resulting sub-trajectory depends only on a scalar parameter—its end-point. As for predictive controllers [GPM89, CS91], these simpler planning processes evolve according to a *receding horizon* strategy: the planned parameterization is applied until new measurements are available. Then, a new parameterization is evaluated which replaces the previous one. This provides the desired robustness against both model and measurement disturbances. The selection of the *temporary* end-point is performed by considering two objectives: (i) minimize the traversal time, i.e. the time required to track the desired path, and (ii) guarantee that the constraints are and will be fulfilled—i.e., no "blind-alley" is entered. The idea of reducing the complexity of constrained tracking problems was exploited in the previous chapters.

The chapter is organized as follows. In Sect. 6.2 we describe the PG's path-parameterization strategy. In Sect. 6.3 we state the assumptions which are required to prove the main properties of the PG in Sect. 6.4. The constrained optimization problem related to the PG is briefly described in Sect. 6.5, and some extensions are discussed in Sect. 6.6 to cope with switching commands and partially known desired paths. Simulative examples on a two-link planar manipulator are described in Sect. 6.7. Finally, experimental results on a 3-DOF PUMA 560 manipulator subject to absolute position error, Cartesian velocity, and motor voltage constraints are presented in Sects. 6.8.

6.2 Path Governor Formulation

The robot closed-loop dynamics is expressed by

$$\begin{cases} \dot{x} &= \varphi(x, r, \dots, r^{(h)}) \\ c &= \ell(x, r, \dots, r^{(h)}) \\ r &= r(s) \\ x(0) &= x_0 \end{cases} \tag{6.1}$$

where $x = [q' \ \dot{q}' \ x_i' \ x_c']'$ collects the robot positions $q \in \mathbb{R}^m$, velocities $\dot{q}$, possible internal states x^i (e.g. electrical dynamics), and the state x_c of the controller, $x \in \mathbb{R}^n$, and the initial condition $x_0 \in \mathcal{X}_0$ for some compact set $\mathcal{X}_0 \subset \mathbb{R}^n$; $r(s) \in \mathbb{R}^m$ is the reference to be tracked by q, and is a given function of the scalar s, $s_0 \le s \le s_f$, determined by the specific task; $r^{(j)} \triangleq \frac{d^j r}{dt^j}$, $j = 0, \dots, h$, where h is the number of derivatives involved in the control law, usually $h = 0, 1$ or 2; $c \in \mathbb{R}^p$ is the vector we wish to satisfy the constraints

$$c(t) \in \mathcal{C}, \ \forall t \ge 0. \tag{6.2}$$

The aim of this chapter is to design a device, referred to as *Path Governor* (PG), which on-line selects the parameter $s(t)$ so as to fulfill (6.2) and minimize the traversal time. Since, as one can expect, this selection involves a non negligible amount of computations, this device will operate in discrete time, namely every T seconds. In order to avoid "blind-alleys", rather than selecting $s(t)$ for only $kT < t \le (k+1)T$, $k \in \mathbb{Z}_+ \triangleq \{0, 1, \dots\}$, the PG cautiously considers an entire *virtual parameterization* $s(\tau; kT, s_\infty)$, where $\tau \in [0, +\infty)$ represents the prediction time, kT the current time, and s_∞ is a free scalar, $s_\infty = \lim_{\tau \to \infty} s(\tau; t, s_\infty)$, denominated *temporary end-point*. Based on the available measurements $q(kT), \dot{q}(kT)$, the scalar s_∞^k is selected at time kT, by solving a constrained optimization problem. This is aimed at minimizing the time required to track the desired path and taking into account that the predicted evolution $c(\tau; kT, s_\infty)$— generated by applying $r(s(\tau; kT, s_\infty))$ from the initial state $x(kT)$— satisfies the given constraints. The algorithm used by the PG can be formulated as follows:

Algorithm 1

0 Let Δs_∞, α be fixed positive scalars, and let $s_\infty^{-1} \triangleq s_0$.

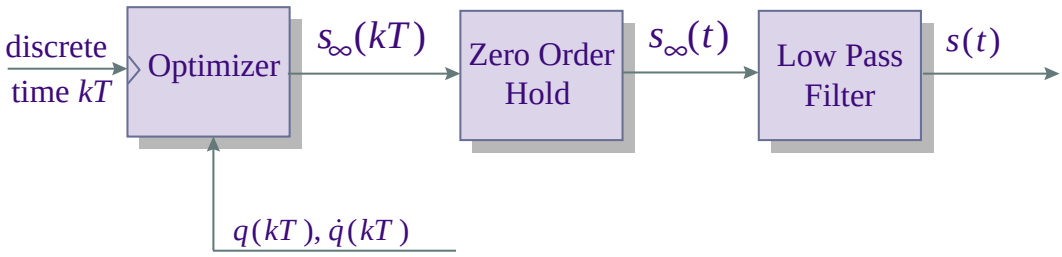


Figure 6.2: Path Governor (PG)

1 At time $t = kT$, find the temporary end-point $s_\infty^k \in [s_\infty^{k-1}, s_\infty^{k-1} + \Delta s_\infty]$ which maximizes

$$J(s_\infty) \triangleq s(T; kT, s_\infty) \tag{6.3}$$

with respect to s_∞ subject to the constraint that the virtual parameterization

$$s(\tau; kT, s_\infty) \triangleq s_\infty + [s(kT) - s_\infty]e^{-\alpha\tau}, \ \tau > 0, \tag{6.4}$$

where $s(kT) = s(T; (k-1)T, s_\infty^{k-1})$ has been determined at time $(k-1)T$, satisfies the constraints

$$c(\tau; kT, s_\infty) \in \mathcal{C}, \ \forall \tau > 0 \tag{6.5}$$

2 Apply $r(s(t)) = r(s(t - kT; kT, s_\infty^k))$ to the closed loop system (6.1) only for $t \in (kT, (k+1)T]$.

3 Repeat the procedure at time $(k+1)T$ until $s((k+1)T) \geq s_f$.

A schematic of the PG is depicted in Fig. 6.2.

We underline the notational difference which will be used hereafter between $s(\tau; kT, s_\infty)$, representing the *virtual* parameter at the prediction time τ, and $s(t)$ which is instead actually used to parameterize the desired path at time t.

Definition 6.1 *At time kT and given the current state $x(kT)$ a temporary end-point s_∞ is* admissible *if the corresponding virtual evolution $c(\tau; t, s_\infty) \in \mathcal{C}$, $\forall \tau > 0$*

Remark 1 Notice that (6.3) is aimed to minimize the time required for the tracking of the desired path. On the other hand, it also holds that

$$s_\infty^k = \begin{cases} \max\limits_{s_\infty^{k-1} \leq s_\infty \leq s_\infty^{k-1} + \Delta s_\infty} s_\infty \\ \text{subject to } c(\tau; kT, s_\infty) \in \mathcal{C}, \ \forall \tau > 0 \end{cases} \tag{6.6}$$

Remark 2 The generated path parameterization satisfies the property

$$\dot{s}(t) > 0, \ \forall t \geq 0$$

Remark 3 By setting $T \to 0$, the previous strategy can be regarded as a way to select at each time t the derivative $\dot{s}(t) = \dot{s}(0; t, s_\infty) = \alpha[s_\infty - s(t)]$. Therefore, maximization of s_∞ corresponds to maximization of $\dot{s}$ and hence to minimization of the traversal time

$$t_f = \int_0^{t_f} dt = \int_{s_0}^{s_f} \frac{1}{\dot{s}} ds$$

Remark 4 The upper-bound induced by a finite Δs_∞ prevents that the solution $s_\infty^k = +\infty$. In order to reach the maximum tracking speed, s_∞^k should be determined by the constraints on vector c rather than this artificial upper-bound, and therefore the scalar optimization problem (6.6) should generate $s_\infty^k < s_\infty^{k-1} + \Delta s_\infty$. Since Δs_∞ can be selected arbitrarily large (for instance, $\Delta s_\infty = 2(s_f - s_0)$), this in general can be achieved.

The formulation of Algorithm 1 does not take into account the time required for the computation of s_∞^k, which will be denoted by T_c. Henceforth, for real-time applications, Algorithm 1 should be modified as follows. Let $T > T_c$. At time $(k-1)T$, the current measurement of the state $x((k-1)T)$ is used to predict $x(T; (k-1)T, s_\infty^{k-1})$. This replaces $x(kT)$ in Algorithm 1. Then s_∞^k is computed during the time interval $[(k-1)T, (k-1)T + T_c]$, and is available at time kT for the generation of the desired path $r(s(t))$, $t \in (kT, (k+1)T]$. Since this modification only introduces a time delay equal to T, in that during the first time interval $[0, T]$ $s(t) \equiv s_0$ is applied, in the following sections we shall neglect this computational aspect, which instead will be discussed in Sect. 6.8.

6.3 Assumptions

In order to prove nice properties of the PG in Sect. 6.4, we consider the class of systems (6.1) and references r which fulfill the following assumptions. The notation $B(x_0, \epsilon)$ will be used to denote the ball $\{x : \ \|x - x_0\| \leq \epsilon\}$.

Assumption 6.1 *The reference path $r : [s_0, s_f] \to \mathbb{R}^m$ is continuous and piecewise differentiable h times, and*

$$\|\frac{d^j r(s)}{ds^j}\| \leq R_j, \forall j = 0, \ldots, h$$

for some positive R_j.

In particular, Assumption 1 implies that there exists a compact set $\mathcal{R} \subset \mathbb{R}^m$ such that $r(s) \in \mathcal{R}$, $\forall s \in [s_0, s_f]$.

Assumption 6.2 *At time* $t = 0$ *the temporary end-point* $s_\infty = s_0$ *is admissible from the initial state* x_0.

This allows to initialize Algorithm 1 by setting $s_\infty^{-1} \triangleq s_0$. As an example, Assumption 6.2 is satisfied for $q(0) = r(s_0)$, $\dot{q}(0) = 0$, i.e. when the robot is initially at rest on the initial point of the desired path.

Definition 6.2 *The reference path* r *is extended for* $s > s_f$ *by setting*

$$r(s) \triangleq r(s_f), \ \forall s \geq s_f$$

Notice that the properties in Assumption 6.1 still hold when r is redefined as in Definition 6.2.

Assumption 6.3 $\varphi : \mathbb{R}^n \times \mathcal{R} \times B(0, R_1) \times \ldots \times B(0, R_h) \to \mathbb{R}^n$ *is continuous in* $(x, r, \ldots, r^{(h)})$.

Assumption 6.4 $\ell : \mathbb{R}^n \times \mathcal{R} \times B(0, R_1) \times \ldots \times B(0, R_h) \to \mathbb{R}^n$ *is uniformly continuous in* $(x, r, \ldots, r^{(h)})$.

Assumption 6.5 *For all* $x_0 \in \mathcal{X}_0$ *and for all* $r \in \mathcal{R}$, *if* $r(s(t)) \equiv r$, $\forall t \geq \bar{t}$ *then as* $t \to \infty$

$$\begin{cases} q & \to & r \\ \dot{q} & \to & 0 \\ x^i & \to & x_r^i \\ x^c & \to & x_r^c \end{cases}$$

and x_r^i, x_r^c *depend continuously on and are uniquely determined by* r. *Moreover, by letting* $x_r \triangleq [r' \ 0' \ {x^i}_r' \ {x^c}_r']'$ *and* $c_r \triangleq \ell(x_r, r, 0, \ldots, 0)$, *stability is uniform w.r.t.* x_r, *in that*

$$\forall \epsilon > 0 \ \exists \rho(\epsilon) > 0 : \ \|x(t_1) - x_r\| \leq \rho(\epsilon) \ \Rightarrow \ \|x(t) - x_r\| \leq \epsilon, \ \forall t \geq t_1 \quad (6.7)$$

Assumption 6.6 $\forall r \in \mathcal{R}$, *if* $r(s(t)) \to r$, *and* $r^{(j)}(s(t)) \to 0$ *as* $t \to \infty$, $\forall j = 1, \ldots, h$, *then* $q(t) \to r$, $\dot{q} \to 0$, $x^c(t) \to x_r^c$.

Notice that often in practical applications, because of finite numerical precision, Assumption 6.5 also implies Assumption 6.6. It is clear that, since the PG introduces a further feedback loop (see Fig 6.1), stability and tracking properties of the overall system cannot be a priori inferred from Assumptions 6.5, 6.6. These properties will be investigated in Sect. 6.4.

Assumption 6.7 *The constraint set $\mathcal{C}$ has a nonempty interior.*

Assumption 6.7 entails in requiring that there is some "maneuver space" inside $\mathcal{C}$, and that no equality constraints can be handled. A simple instance of $\mathcal{C}$ is a hyper-rectangle having nonzero volume.

Assumption 6.8 *Let δ be a fixed (arbitrarily small) positive real. Then $\mathcal{R}$ is such that $B(c_r, \delta) \subseteq \mathcal{C}$, $\forall r \in \mathcal{R}$.*

Assumption 6.8 entails in requiring that the commanded reference positions, each taken as set-point, are restricted to the ones which, in steady-state, give a corresponding constrained vector c_r which "lies away" from the border of $\mathcal{C}$ of at least a distance $\delta > 0$. By Assumption 6.7 such a δ always exists.

6.4 Main Results

In this section, we will study some properties exploited by the Path Governor formulated in Sect. 6.2. Lemma 6.1 will first prove that an admissible temporary end-point s_∞^k can be found at each time kT. Lemma 6.2 will show that s_∞^k cannot jam on a value between s_0 and s_f, in that a better admissible temporary end-point is always found within a finite time. Lemma 6.3, on the other hand, will prove that if the generated $s(t)$ converges to a final value s_f, then this limit is reached after a finite time. Theorem 6.1 will make use of both lemmas to show that $s(t) = s_f$ after a finite time t_f. Theorem 6.2 will summarize the overall PG properties.

Lemma 6.1 *$\forall k \in \mathbb{Z}_+$ there exists a temporary end-point $s_\infty^k \geq s_\infty^{k-1}$ which is admissible from the current state $x(kT)$.*

Proof. The proof easily follows by induction. Assumption 6.2 states that an admissible s_∞^k can be found at least for $k = 0$. Assume that an admissible

temporary end-point s_∞^{k-1} has been found at time $(k-1)T$. Now notice that $s(kT) = s_\infty^{k-1} + [s((k-1)T) - s_\infty^{k-1}]e^{-\alpha T}$ and hence

$$\begin{aligned} s(\tau; kT, s_\infty^{k-1}) &= s_\infty^{k-1} + [s(kT) - s_\infty^{k-1}]e^{-\alpha\tau} \\ &= s_\infty^{k-1} + [s((k-1)T) - s_\infty^{k-1}]e^{-\alpha T}e^{-\alpha\tau} \\ &= s(\tau + T; (k-1)T, s_\infty^{k-1}). \end{aligned}$$

Furthermore, $c(\tau; (k-1)T, s_\infty^{k-1}) \in \mathcal{C}$, $\forall \tau \geq 0$, in particular $\forall \tau \geq T$. Since also after T seconds the state has moved exactly to $x(kT) = x(T; (k-1)T, s_\infty^{k-1})$, it follows that

$$c(\tau; kT, s_\infty^k) = c(T + \tau; (k-1)T, s_\infty^{k-1}) \in \mathcal{C}, \ \forall \tau \geq 0,$$

Therefore, at least $s_\infty^k \triangleq s_\infty^{k-1}$ is admissible at time $(k+1)T$ from $x((k+1)T)$. □

Lemma 6.1 has proved that the sequence $\{s_\infty^k\}_{k=0}^\infty$ is defined and nondecreasing. Next Lemma 6.2 shows that such a sequence cannot assume a constant value less or equal than s_f.

Lemma 6.2 *Let s_∞^k be admissible at time kT, $s_\infty^k \leq s_f$. Then, by applying $s(t) = s(t - kT; kT, s_\infty^k)$, there exists a time $\bar{t} \geq kT$ and $\bar{s}_\infty > s_\infty^k$ such that $\bar{s}_\infty$ is admissible at time $\bar{t}$.*

Proof. Let $r_\infty^k \triangleq r(s_\infty^k)$, $x_\infty^k \triangleq x_{r_\infty^k}$. Since $s(t) = s(t - kT; kT, s_\infty^k) \to s_\infty^k$ as $t \to \infty$, by Assumption 6.1 $r(s(t)) \to r_\infty^k$, and, by Assumption 6.6, $x(t) \to x_\infty^k$. By Assumption 6.4, $c(t) \to c_\infty^k \triangleq \ell(x_\infty^k, r_\infty^k, 0, \ldots, 0)$, and also $\exists \epsilon = \epsilon(\delta) > 0$ such that, $\forall x \in \mathbb{R}^n$, $\forall r, \ldots, r^{(h)}$ satisfying Assumption 6.1, $\forall \bar{s}_\infty \in [s_0, s_f]$, $\bar{r}_\infty \triangleq r(\bar{s}_\infty)$, $\bar{x}_\infty \triangleq x_{\bar{r}_\infty}$,

$$\|x - \bar{x}_\infty\| + \|r - \bar{r}_\infty\| + \sum_{j=1}^{h} \|r^{(h)} - 0\| \leq \epsilon \tag{6.8}$$

implies $\|c - \bar{c}_\infty\| \leq \delta$, where $c = \ell(x, r, \ldots, r^{(h)})$, and $\bar{c}_\infty = \ell(\bar{x}_\infty, \bar{r}_\infty, 0, \ldots, 0)$. We wish to find a time $\bar{t}$ and a parameter $\bar{s}_\infty > s_\infty^k$ such that (6.8) holds for $x = x(\tau; \bar{t}, \bar{s}_\infty)$, $r = r(s(\tau; \bar{t}, \bar{s}_\infty))$, and $\forall \tau > 0$, in order to claim that $\bar{s}_\infty$ is admissible from $x(\bar{t})$. In order to accomplish this task, let t_a such that

$$\|x(t) - x_\infty^k\| \leq \frac{1}{2}\rho\left(\frac{\epsilon}{h+2}\right)$$

for all $t \geq t_a$, where the function ρ is defined in (6.7). By virtue of Assumption 6.5 (continuity of x_r^c with respect to r) and Assumption 6.1 (continuity of $r(s)$), let $s_\infty^a > s_\infty^k$ such that the corresponding equilibrium state $x_\infty \triangleq x_{r(s_\infty)}$ satisfies

$$\|x_\infty - x_\infty^k\| \leq \frac{1}{2}\rho\left(\frac{\epsilon}{h+2}\right)$$

$\forall s_\infty \in (s_\infty^k, s_\infty^a]$. By continuity of $r(s)$, there exists a $\Delta s_\infty^b > 0$ such that

$$\|r(s_\infty) - r_\infty^k\| \leq \frac{\epsilon}{2(h+2)}$$

for all $s_\infty \in [s_\infty^k - \Delta s_\infty^b, s_\infty^k + \Delta s_\infty^b]$. Because $s(t) \to s_\infty^k$, take t_b such that $s(t) \geq s_\infty^k - \Delta s_\infty^b$ for all $t \geq t_b$. Since, for every temporary end-point s_∞ and time t, $s(\tau; t, s_\infty)$ monotonically increases from $s(t)$ to s_∞ as τ increases, the conditions $t \geq t_b$ and $s_\infty \in [s_\infty^k - \Delta s_\infty^b, s_\infty^k + \Delta s_\infty^b]$ imply

$$\|r(s(\tau; t, s_\infty)) - r_\infty^k\| \leq \frac{\epsilon}{2(h+2)}, \ \forall \tau \geq 0.$$

Consider now

$$\frac{dr}{dt}(s(\tau; t, s_\infty)) = \frac{dr}{ds}\dot{s}(\tau; t, s_\infty) = \frac{dr}{ds}\alpha[s_\infty - s_\infty^k + (s(kT) - s_\infty^k)e^{-\alpha(t-kT)}]e^{-\alpha\tau};$$

since $\|dr/ds\| \leq R_1$, one can find $s_\infty^c > s_\infty^k$ and t_c such that $\|\dot{r}(s(\tau; t, s_\infty))\| \leq \frac{\epsilon}{2+h}$, $\forall \tau \geq 0$, $\forall s_\infty \in (s_\infty^k, s_\infty^c]$, and $\forall t \geq t_c$. Similarly, since by Assumption 6.1 $\|\frac{d^j r}{ds^j}\| \leq R_j$, $\forall j = 1, \ldots, h$, one can select s_∞^c and t_c so that $\|r^{(j)}(s(\tau; t, s_\infty))\| \leq \frac{\epsilon}{2+h}$, $\forall \tau \geq 0$, $\forall j = 1, \ldots, h$, $\forall s_\infty \geq s_\infty^c$, $\forall t \geq t_c$. Then, by selecting

$$\bar{s}_\infty \triangleq \min\{s_\infty^a, s_\infty^k + \Delta s_\infty^b, s_\infty^c\}$$

and

$$\bar{t} \triangleq \max\{t_a, t_b, t_c\}$$

one has

$$\begin{aligned}
&\|x(\tau; \bar{t}, \bar{s}_\infty) - \bar{x}_\infty\| + \|r(s(\tau; \bar{t}, \bar{s}_\infty)) - \bar{r}_\infty\| + \sum_{j=1}^{h} \|r^{(h)}(s(\tau; \bar{t}, \bar{s}_\infty))\| \\
&\leq \|x(\tau; \bar{t}, \bar{s}_\infty) - x_\infty\| + \|x_\infty - \bar{x}_\infty\| + \|r(s(\tau; \bar{t}, \bar{s}_\infty)) - r_\infty\| + \|r_\infty - \bar{r}_\infty\| + \\
&\sum_{j=1}^{h} \|r^{(h)}(s(\tau; \bar{t}, \bar{s}_\infty))\| \leq \epsilon, \ \forall \tau \geq 0
\end{aligned}$$

and thus $\|c(\tau;\bar{t},\bar{s}_\infty)-\bar{c}_\infty\|\leq\delta$, $\forall\tau\geq 0$. By Assumption 6.8, $c(\tau;\bar{t},\bar{s}_\infty)\in\mathcal{C}$, $\forall\tau\geq 0$, follows. Then, $\bar{s}_\infty > s^k_\infty$ is admissible at time $\bar{t}$. □

Lemma 6.2 proved that, if $s^k_\infty < s_f$ at time kT, then after a finite time another admissible $s_\infty > s^k_\infty$ can be found. Next Lemma 6.3 shows that if $s(t)\to\bar{s}$, than this limit value is reached by s^k_∞ in a finite time.

Lemma 6.3 *Let $\lim_{t\to\infty} s(t)=\bar{s}_\infty\in[s_0,s_f]$. Then, there exists a finite time $\bar{t}\geq 0$ such that $\bar{s}_\infty$ is admissible at time $\bar{t}$.*

Proof. Since

$$s((k+1)T)=s^k_\infty+[s(kT)-s^k_\infty]e^{-\alpha T},$$

it follows that

$$\lim_{k\to\infty} s^k_\infty=\lim_{k\to\infty}\frac{s((k+1)T)-s(kT)e^{-\alpha T}}{1-e^{-\alpha T}}=\bar{s}_\infty$$

By setting $\bar{r}\triangleq r(\bar{s}_\infty)$, $\bar{x}\triangleq x_{\bar{r}}$, $\bar{c}\triangleq\ell(\bar{x},\bar{r},0,\ldots,0)$, and following arguments similar to those used in the proof of Lemma 6.2, we can find a time $\bar{t}$ such that

$$\|x(\tau;\bar{t},\bar{s}_\infty)-\bar{x}\|+\|r(\tau;\bar{t},\bar{s}_\infty)-\bar{r}\|+\sum_{j=1}^{h}\|r^{(h)}(\tau;\bar{t},\bar{s}_\infty)\|\leq\epsilon \tag{6.9}$$

for all $\tau\geq 0$, or equivalently $\|c(\tau;\bar{t},\bar{s}_\infty)-\bar{c}\|\leq\delta$. By Assumption 6.8, $c(\tau;\bar{t},\bar{s}_\infty)\in\mathcal{C}$, $\forall\tau\geq 0$ follows, or, equivalently, $\bar{s}_\infty$ is admissible at time $\bar{t}$. □

Next Theorem 6.1 proves that the Path Governor generates a desired-path parameterization $s(t)$ such that s_f is reached in a finite time t_f.

Theorem 6.1 *There exists a finite time t_f such that $s(t_f)=s_f$.*

Proof. Assume by contradiction that $s(t)<s_f$, $\forall t\geq t_0$. Since $\dot{s}(t)>0$, $s(t)$ is a real monotonically increasing and upper-bounded function of the time t, and hence there exists $\bar{s}_\infty\triangleq\lim_{t\to\infty}s(t)\leq s_f$. By Lemma 6.3 there exists a time $\bar{t}_1$ such that $s^k_\infty=\bar{s}_\infty$, $\forall kT\geq\bar{t}_1$. Then, by Lemma 6.2, there exists a time $\bar{t}_2$ such that for $kT\geq\bar{t}_2$ there exists $s^k_\infty>\bar{s}_\infty$ which is admissible. This contradicts the optimality of $\bar{s}_\infty$. □

Next Theorem 6.2 summarizes the properties of the proposed Path Governor

Theorem 6.2 *Let $s(t)$ be selected according to the Path Governor (Algorithm 1) formulated in Sect. 6.2. Then*

i. there exists a finite time t_f such that $s(t_f)=s_f$;

ii. *the constraints $c(t) \in \mathcal{C}$ are fulfilled for all $t \geq 0$ while the robot is driven along the path $r(s(t))$, $t \in [0, t_f]$;*

iii. $\lim_{t\to\infty} q(t) = r(s_f)$, $\lim_{t\to\infty} \dot{q}(t) = 0$.

Proof. Existence of such a t_f is guaranteed by Theorem 6.1. Constraint fulfillment follows by the selection criterion for the temporary end-points s_∞^k, in that $c(t, x(t), r(s(t)), \ldots, r^{(h)}(s(t))) = c(t, x(t - kT; kT, s_\infty^k), r(s(t - kT; kT, s_\infty^k)), \ldots, r^{(h)}(s(t - kT; kT, s_\infty^k)),\ \forall t \in (kT, (k+1)T]$. Convergence of q, $\dot{q}$ follows by Assumption 6.5. □

6.5 Optimization Algorithm

In order to implement the PG described in the previous sections, the optimization problem (6.6),(6.16) is solved by using a bisection algorithm over the interval $[s_\infty^{k-1}, s_\infty^{k-1} + \Delta s_\infty]$. Let N denote the number of parameters s_∞ which can be evaluated during the selected period T. Because $\mathcal{C}$ is generic and the model of the robot is nonlinear, no convexity properties of the set of admissible s_∞ can be invoked. Then, the adopted bisection algorithm only provides local minima. By following an approach similar to [Bem97b], it can be proved that this does not affect the convergence results proved in Sect. 6.4. However, it is clear that if global minimization procedures were adopted in selecting s_∞^k, better tracking properties might be achieved, at the expense of an increased computational effort. Testing the admissibility of a given s_∞ requires the evaluation of (6.5), and consequently the numerical integration of the closed-loop equations (6.1) from initial state $x(kT)$. When instead feedback linearization is adopted as primal control strategy, the numerical integration can be carried out on a discrete-time version of the resulting linear system, verifying the (nonlinear) constraints at sample steps. We describe an alternative form which is more suitable for algorithmic implementation for general structures of (6.1). At each time kT the PG must solve the optimization problem (6.6). Despite the simple structure of the cost function (6.3), the problem involves continuous state constraints (6.5) over an infinite horizon. We translate (6.6) in a general form which is more suitable for algorithmic implementation via Runge-Kutta methods. Let the set $\mathcal{C}$ be

defined as

$$\mathcal{C} = \{c \in \mathbb{R}^p : \ \varphi_i(c) \leq 0, \ i = 1, \ldots, q\}$$

Then, the constraints in (6.5) can be expressed in the form

$$g_i(x, s, \dot{s}, \ldots, s^{(h)}) \leq 0, \ i = 1, \ldots, q \tag{6.10}$$

where the functions g_i derive from the composition of φ_i, ℓ, the desired path $r(s)$, and its derivatives $\frac{d^j r}{ds^j}$, $j = 0, \ldots, h$. For the sake of simplicity, we assume that the constrained vector c does not depend on the derivatives of the reference r, $c = \ell(x, r)$, which allows us to drop the dependence on the derivatives of s in (6.10). At a fixed time $t = kT$, system (6.1),(6.4) can be rewritten as

$$\begin{cases} \dot{x} &= \varphi(x, r(s), \ldots, r^{(h)}(s)) \\ \dot{s} &= \alpha[s_\infty - s] \\ x(\tau = 0) &= x(kT) \\ s(\tau = 0) &= s(kT) \end{cases} \tag{6.11}$$

Then, the constraints in (6.5) become

$$g_i(x(\tau; kT, s_\infty), s(\tau; kT, s_\infty)) \leq 0, \ \forall \tau > 0, \ i = 1, \ldots, q \tag{6.12}$$

According to the procedure in [TJ89], the condition (6.12) is equivalent to the scalar constraint equality

$$G(s_\infty) \triangleq \sum_{i=1}^{q} \int_0^{+\infty} \max\{g_i(x(\tau; kT, s_\infty), s(\tau; kT, s_\infty)), 0\} d\tau = 0 \tag{6.13}$$

By defining for a small $\epsilon > 0$ the function $\sigma_\epsilon : \mathbb{R} \mapsto \mathbb{R}$

$$\sigma_\epsilon(g) \triangleq \begin{cases} g & \text{if} \quad g > \epsilon \\ \frac{(g+\epsilon)^2}{4\epsilon} & \text{if} \quad |g| \leq \epsilon \\ 0 & \text{if} \quad g < -\epsilon \end{cases}$$

the fulfillment of the constraint (6.13) is guaranteed by the condition

$$G_\epsilon(s_\infty) \triangleq \sum_{i=1}^{q} \int_0^{+\infty} \sigma_\epsilon(g_i(x(\tau; kT, s_\infty), s(\tau; kT, s_\infty)), 0) d\tau = 0 \tag{6.14}$$

which ensures better numerical conditioning and allow the derivative $\frac{dG_\epsilon}{ds_\infty}$ to be analytically computed, when this is required by gradient-based optimization algorithms. However, the evaluation of G_ϵ still requires the integration of (6.11)

over an infinite horizon. This can be avoided by integrating the differential equations

$$\begin{cases} \dfrac{dx}{ds} &= \dfrac{\varphi(x, r(s), \ldots, r^{(h)}(s))}{\alpha(s_\infty - s)} \\ x(s(kT)) &= x(kT) \end{cases} \tag{6.15}$$

over the finite interval $s \in [s(kT), s_\infty]$. Similarly, (6.14) is transformed in the finite integral

$$G_\epsilon(s_\infty) \triangleq \sum_{i=1}^{q} \int_{s(kT)}^{s_\infty} \sigma_\epsilon(g_i(x(s; kT, s_\infty), s), 0) \frac{1}{\alpha(s_\infty - s)} ds = 0 \tag{6.16}$$

Notice that, as a consequence of Assumption 6.8, convergence of the integral (6.13) or, equivalently, (6.16), can be guaranteed by choosing ϵ sufficiently small so that $B(c, \delta) \subset \mathcal{C} \Rightarrow \varphi_i(c) \leq \epsilon$, $i = 1, \ldots, q$. In this case, after a finite time $\tau^* \triangleq \frac{1}{\alpha} \log \frac{s(kT) - s_\infty}{s^* - s_\infty}$, where $s^* \in (s(kT), s_\infty)$, the function to be integrated is zero.

6.6 Switching Commands and Partially Known Paths

We will present some slight modifications Algorithm 1 which allow the application of the PG when higher level commands are added to the (autonomous) tracking task, and/or the whole desired path is not completely known in advance.

6.6.1 Switching Commands

We wish to take into account higher level commands, which consist in switching the autonomous operation among the following: (i) stop the motion along the trajectory, for example because an unexpected obstacle has been detected; (ii) slow down the motion; (iii) invert the motion; (iv) resume the normal (autonomous and as fast as possible) execution of the tracking task. This commands can be taken into account by different options in the optimization involved in Algorithm 1, as follows:

- **STOP**: minimize s_∞, $s_\infty \in [s(kT), s_\infty^{k-1}]$.

- **SLOW DOWN**: find the maximum admissible $s_\infty^* \in [s_\infty^{k-1}, s_\infty^{k-1} + \Delta s_\infty]$ and set $s_\infty^k = s_\infty^{k-1} + [s_\infty^* - s_\infty^{k-1}]\frac{1}{N}$, where $N > 1$ is proportional to how much the tracking must be slowed down.
- **GO BACK** : minimize s_∞, $s_\infty \in [s_\infty^{k-1} - \Delta s_\infty, s_\infty^{k-1}]$.
- **GO, RESUME**: maximize s_∞, $s_\infty \in [s_\infty^{k-1}, s_\infty^{k-1} + \Delta s_\infty]$.

Each option also includes (6.5), and therefore constraint fulfillment warranty is preserved.

6.6.2 Partially Unknown Desired Paths

Assume that the desired path is not completely known in advance, i.e. $r(s)$ is known at time t only for $s \leq s(t) + \sigma$, $\sigma \geq 0$. We distinguish two situations: $\sigma = 0$, which corresponds to a task where the end-effector has to track an object whose motion is not known in advance; $\sigma > 0$, for example if new pieces of trajectory are appended before the completion of the tracking task. These modes of operations can be both taken into account by dynamically redefining the desired path. Define recursively, for $s > s(kT)$,

$$r_k(s) \triangleq \begin{cases} r_{k-1}(s) & \text{if } s \leq s(kT) + \sigma \\ f_k(s) & \text{otherwise} \end{cases}$$

where $f_k(s)$ is a function which is constructed on-line on the basis of the data available at time kT, and satisfies the following properties

1. $f_k(s(kT) + \sigma) = r_{k-1}(s(kT) + \sigma)$ (continuity);
2. $f_k(s) = r_{k-1}(s)$, $\forall s \geq s(kT)$, if no admissible temporary end-point can be found otherwise.

For instance, assuming that the desired path is only known for $s \leq s(kT) + \sigma$, consider for $s > s(kT) + \sigma$

$$f_k(s) \triangleq \begin{cases} r(s(kT) + \sigma) & \forall s > s(kT) + \sigma, \text{ if an admissible } s_\infty^k \text{ can be found} \\ r_{k-1}(s) & \text{otherwise} \end{cases}$$

Constraint fulfillment is preserved for all $t \geq 0$, and, in the worst case, as $t \to \infty$, the robot joint coordinates $q(t)$ will jam on $r(s(k_0T) + \sigma)$ for some $k_0 \in \mathbb{Z}_+$.

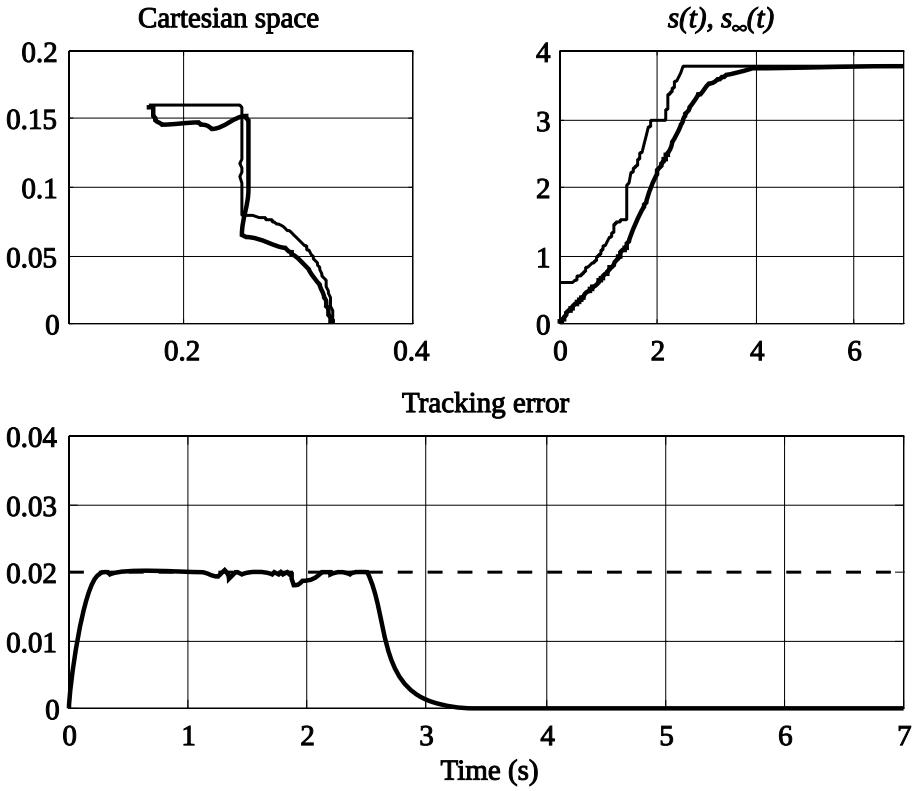


Figure 6.3: Tracking error constraints $e_{\max} = 2$ cm, $T = 0.05$ s.

6.7 Simulation Results

The performance of the PG presented in the previous sections has been tested by computer simulations on a two link robot moving on a horizontal plane. Each joint is equipped with a motor for providing input torque, encoders and tachometers are used for measuring the joint positions $q = [\theta_1,\ \theta_2]'$, and velocities $\dot{q} = [\dot{\theta}_1,\ \dot{\theta}_2]'$. By using Lagrangian equations, and by setting

$$\mathcal{T} = \begin{bmatrix} \mathcal{T}_1 \\ \mathcal{T}_2 \end{bmatrix} \quad r = \begin{bmatrix} \theta_{1d} \\ \theta_{2d} \end{bmatrix},$$

where θ_{1d}, θ_{2d} denote the desired values for joint positions and $\mathcal{T}_1$, $\mathcal{T}_2$ the motor torques, the dynamic model of the robot can be expressed in the standard form

$$H(q)\ddot{q} + C(q,\dot{q})\dot{q} = \mathcal{T} \tag{6.17}$$

where

$$H(q) = \begin{bmatrix} m_1 l_{c_1}^2 + I_1 + m_2[l_1^2 + l_{c_2}^2 + 2l_1 l_{c_2}\cos(\theta_2)] + I_2 & m_2 l_1 l_{c_2}\cos(\theta_2) + m_2 l_{c_2}^2 + I_2 \\ m_2 l_1 l_{c_2}\cos(\theta_2) + m_2 l_{c_2}^2 + I_2 & m_2 l_{c_2}^2 + I_2 \end{bmatrix}$$

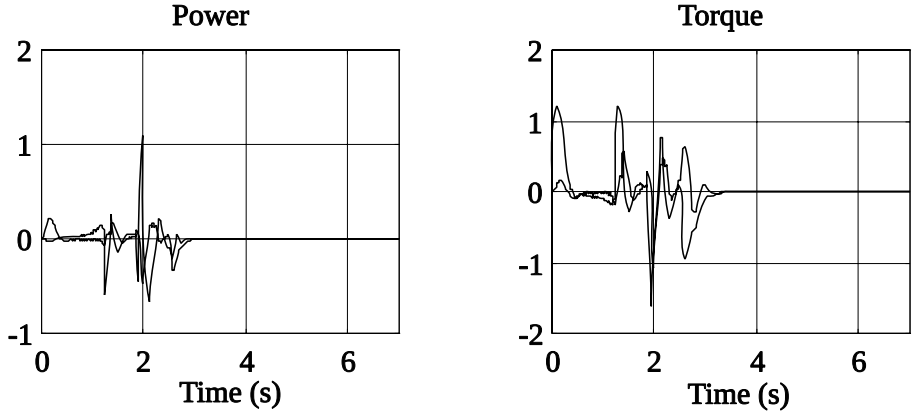


Figure 6.4: Power $\mathcal{T}_i\dot{\theta}_i$ and torque $\mathcal{T}_i$, $i = 1, 2$, for the trajectories of Fig. 6.3.

$$C(q,\dot{q}) = m_2 l_1 l_{c_2} \sin(\theta_2) \begin{bmatrix} -\dot{\theta}_2 & -\dot{\theta}_1 - \dot{\theta}_2 \\ \dot{\theta}_1 & 0 \end{bmatrix}$$

Individual joint PD controllers

$$\mathcal{T} = -\begin{bmatrix} k_{p1}(\theta_1 - \theta_{1d}) + k_{d1}\dot{\theta}_1 \\ k_{p2}(\theta_2 - \theta_{2d}) + k_{d2}\dot{\theta}_2 \end{bmatrix} \tag{6.18}$$

provide reference tracking.

Let (y, z) denote the position of the end-effector in the cartesian plane,

$$\begin{aligned} y &= l_1\cos(\theta_1) + l_2\cos(\theta_1+\theta_2) \\ z &= l_1\sin(\theta_1) + l_2\sin(\theta_1+\theta_2) \end{aligned}$$

We wish that (y, z) tracks the desired path $r_{yz}(s) = [r_y(s), r_z(s)]$,

$$r_{yz}(s) \triangleq \begin{cases} [\frac{l_2}{2}\cos(\frac{\pi}{2}s) + l_1, \frac{l_2}{2}\sin(\frac{\pi}{2}s)] & \text{if} \quad 0 \le s \le 1 \\ [l_1, \frac{l_2}{2}s] & \text{if} \quad 1 < s \le 2 \\ [l_1 - \frac{l_2}{2}(s-2), l_2] & \text{if} \quad 2 < s \le 3 \\ [l_1 - \frac{l_2}{2}, l_2] & \text{if} \quad s > 3 \end{cases} \tag{6.19}$$

for $s \in [0, 3]$, while satisfying the tracking error constraint

$$c(t) \triangleq (y(t) - r_y(s(t)))^2 + (z(t) - r_z(s(t)))^2 \le e_{\max}^2 \tag{6.20}$$

The desired path $r(s) = [\theta_{d1}(s), \theta_{d2}(s)]$ in the joint-space is obtained by using the well-known inverse transformation

$$\begin{cases} D &= \dfrac{r_y^2 + r_z^2 - l_1^2 - l_2^2}{2l_1 l_2} \\ \theta_{2d} &= \text{atan2}(-\sqrt{1-D^2}, D) \\ \theta_{1d} &= \text{atan2}(r_z, r_y) - \text{atan2}(l_2\sin\theta_{2d}, l_1 + l_2\cos\theta_{2d}) \end{cases}$$

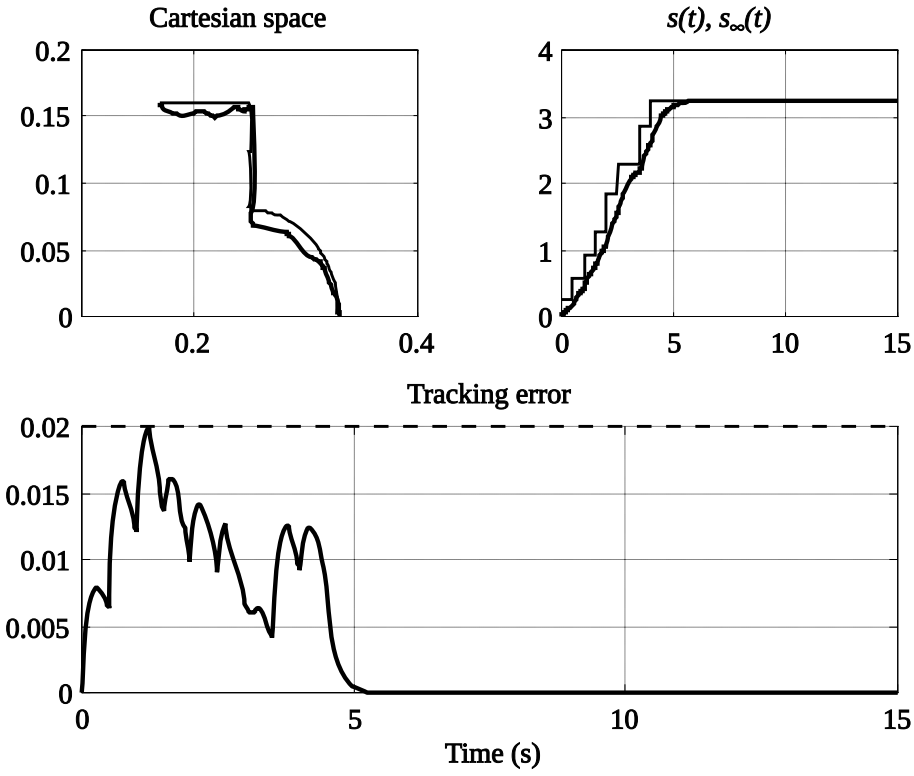


Figure 6.5: Tracking error constraints $e_{\max} = 2$ cm, $T = 0.5$ s.

Notice that $r(s)$ satisfies Assumption 6.1. Assumption 6.4 is fulfilled as well for the choices (6.19) and (6.20). In order to show that system (6.17)–(6.18) fulfills the Assumption (6.5), let $x \triangleq [q' \ \dot{q}']' = [x_1 \ x_2 \ x_3 \ x_4]'$ and consider the following function

$$\begin{aligned} V(x) &= \frac{1}{2}\begin{bmatrix} x_2 \\ x_4 \end{bmatrix}' H(x) \begin{bmatrix} x_2 \\ x_4 \end{bmatrix} + \frac{1}{2}\begin{bmatrix} \theta_{1d} - x_1 \\ \theta_{2d} - x_3 \end{bmatrix}' K_p \begin{bmatrix} \theta_{1d} - x_1 \\ \theta_{2d} - x_3 \end{bmatrix}, \\ K_p &= \begin{bmatrix} k_{p1} & 0 \\ 0 & k_{p2} \end{bmatrix} > 0 \end{aligned}$$

which is a Lyapunov function for (6.17)–(6.18) [AS86]. Since its derivative along the trajectories of the system

$$\begin{aligned} \dot{V}(x) &= -\begin{bmatrix} x_2 \\ x_4 \end{bmatrix}' K_d \begin{bmatrix} x_2 \\ x_4 \end{bmatrix} \leq 0, \\ K_d &= \begin{bmatrix} k_{d1} & 0 \\ 0 & k_{d2} \end{bmatrix} > 0, \end{aligned}$$

and $V(x) = 0$ iff $x = [\theta_{1d} \ 0 \ \theta_{2d} \ 0]'$, the first part of Assumption 6.5 is satisfied. Uniform stability is proved as follows. By contradiction, suppose

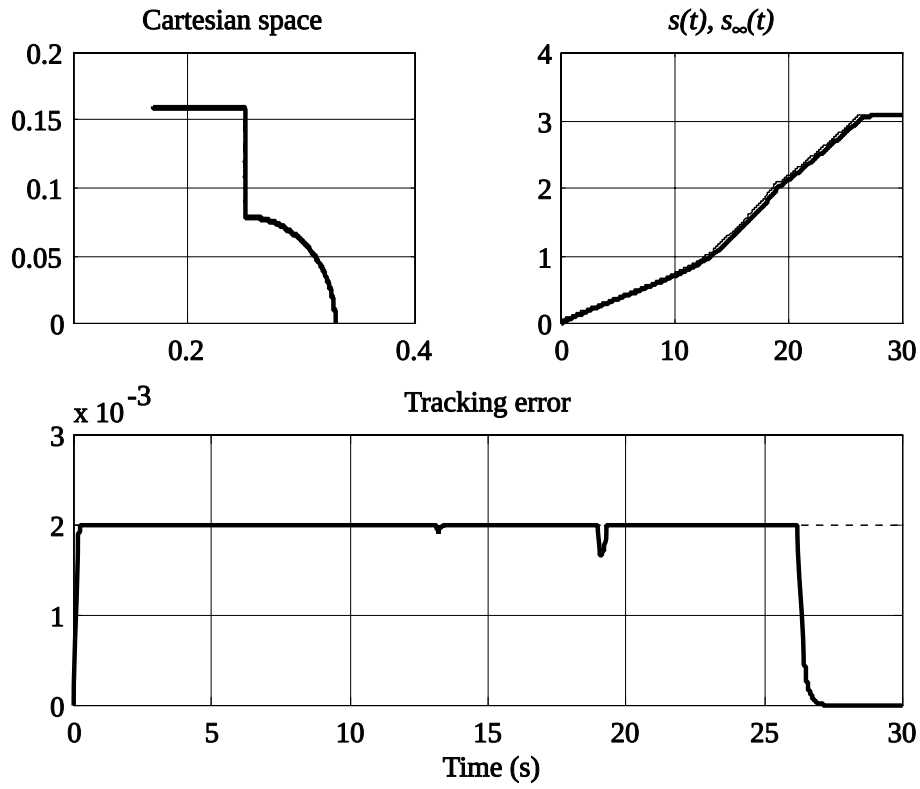


Figure 6.6: Tracking error constraints $e_{\max} = 2$ mm, $T = 0.01$ s.

that there exists an $\epsilon > 0$ such that, $\forall \rho > 0$, there exists x_r and t_1 with $\|x(0) - x_r\| \leq \rho$ and $\|x(t_1) - x_r\| > \epsilon$. Since $\gamma_1 I \leq H(x) \leq \gamma_2 I$ for some positive γ_1, γ_2, by denoting by $\lambda_m(K_p)$, $\lambda_M(K_p)$ respectively the minimum and maximum eigenvalue of K_p, and by setting $\gamma_3 = \min\{\lambda_m(K_p), \gamma_1\}$, $\gamma_4 = \max\{\lambda_M(K_p), \gamma_3\}$, it follows that $\|x(t_1) - x_r\| \leq \frac{2}{\gamma_3} V(x(t_1)) \leq \frac{2}{\gamma_3} V(x(0)) \leq \frac{\gamma_4}{\gamma_3}\rho$ for any arbitrary positive ρ, a contradiction. Moreover, in practice, the reference $r(s(t))$ is expressed by a finite numerical precision, and therefore, if $r(s(t))$ tends toward r, after a finite time $r(s(t)) \equiv r$, and Assumption 6.6 is verified. It is easy to check that Assumptions 6.3, 6.7, and 6.8 are satisfied as well.

Simulations have been carried out with the system parameters reported in [KS94]. The results described hereafter were obtained under Matlab 4.2 + Simulink 1.3. On-line optimization has been performed by using the bisection method mentioned in Sect. 6.5, and standard Runge-Kutta methods have been adopted for numerical integration of (6.15). The PG is applied in order to satisfy the tracking error constraint (6.20). Figs. 6.3-6.4 show the resulting

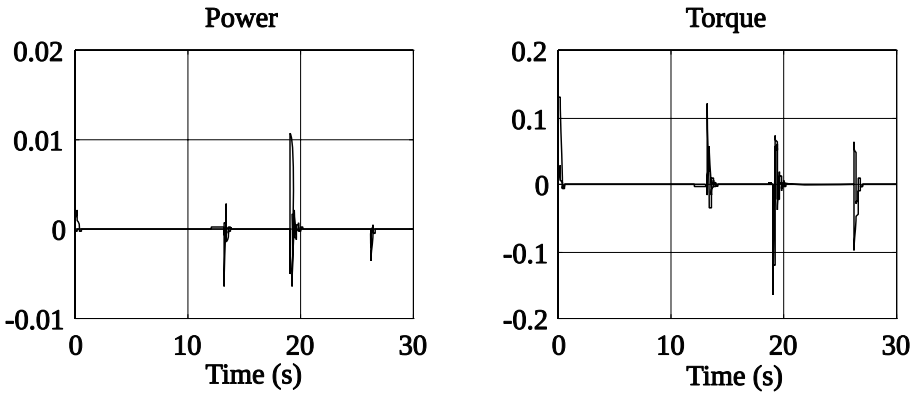


Figure 6.7: Power $\mathcal{T}_i\dot{\theta}_i$ and torque $\mathcal{T}_i$, $i = 1, 2$, for the trajectories of Fig. 6.6.

trajectories for $e_{\max} = 2$ cm (this large value has been chosen to better display the results). The initial condition $[\theta_1(0),\ \theta_2(0)]' = r(0)$, $\dot{\theta}_1(0) = \dot{\theta}_2(0) = 0$ and $s_\infty = 0$ satisfy Assumption 6.2. A PG period $T = 0.05$ s, $\alpha = 2$, $\Delta s_\infty = 0.2$, and $N = 12$ evaluations per period are selected as parameters of the PG.

Because of the adopted receding horizon strategy and particular structure (6.4), the resulting path parameterization $s(t)$ is only near-minimum time. The performance is also affected by the available computational power. In fact, the PG necessarily operates in discrete time, namely every T seconds, in order to perform the required computations. In Fig. 6.5 the PG period is increased to $T = 0.5$ s, and a consequent degradation of the performance obtained in Fig. 6.3 can be noted.

In Figs. 6.6-6.7, the more severe constraint $e_{\max} = 2$ mm is taken into account, and consequently the traversal time increases to 28 s. Notice that a shorter time could be achieved by defining $c(t)$ as the distance between the end-effector $[y(t),\ z(t)]$ and the geometric path $\{r(s)\}_{s\in[0,3]}$; however such a definition of $c(t)$ does not fit with the formulation adopted in the present chapter.

In Figs. 6.8-6.9 the torque constraints

$$|\mathcal{T}_i| \leq 0.5 \text{ Nm},\ i = 1, 2$$

and the power constraints

$$\mathcal{T}_i\dot{\theta}_i \leq 0.5 \text{ W},\ i = 1, 2$$

have been added to the error constraint $e_{\max} = 2$ cm.

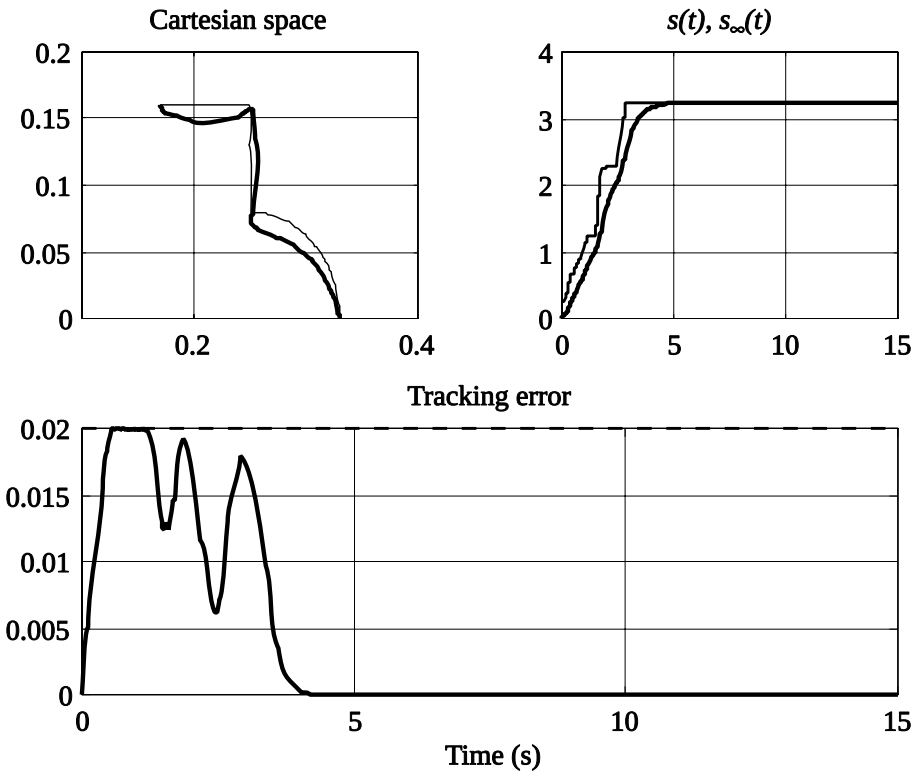


Figure 6.8: Tracking error, torque, and power constraints, $T = 0.05$ s.

Finally, the robustness of the PG against model mismatching has been tested. The mass, length, and inertia of the links in the robot model used by the PG are modified approximately by 30% of the real value. The resulting trajectories are shown in Fig. 6.10. Notice that every time interval $T = 0.05$ s the PG attempts to enforce the fulfillment of the constraint (6.20).

6.8 Experimental Results

The performance of the PG presented in the previous sections has been tested for constrained position control of a three-degree-of-freedom PUMA 560 manipulator. The PG has been implemented on the experimental system in the Center for Robotics and Automation at Washington University (Fig. 6.12). A multiprocessor SGI IRIX 4D/340VGX allows the parallel real-time computation of the parameter s_∞^k at the PG rate $1/T = 2.5$ Hz, and path generation and primal feedback control at the sampling rate $1/\tau_s = 1000$ Hz.

By incorporating the end-effector in the third link, the dynamics is given

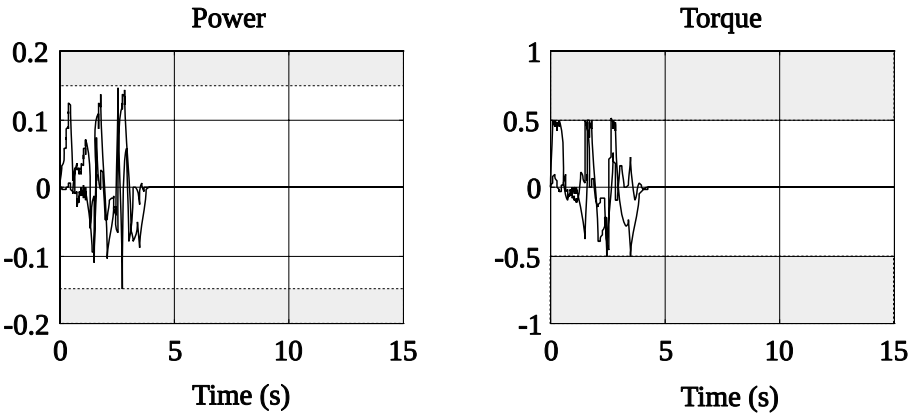


Figure 6.9: Power $\mathcal{T}_i\dot{\theta}_i$ and torque $\mathcal{T}_i$, $i = 1, 2$, for the trajectories of Fig. 6.8.

by the following equation

$$D(q)\ddot{q} + C(q, \dot{q}) + G(q) = \mathcal{T} \tag{6.21}$$

where $\mathcal{T}$ is three vector of joint torques, q is the three vector of joint displacements (with $\dot{q}$ and $\ddot{q}$ the first and second derivatives), $D(q)$ is the three-by-three inertia matrix, $C(q, \dot{q})$ is the three vector of centripetal and Coriolis terms, and $G(q)$ is the three vector of gravity terms [TBY85].

The Cartesian task space position Y of the end-effector is related to the joint coordinates q by the direct/inverse kinematic equations reported in [TBMR91]. The end-effector has to track the desired path

$$Y_d(s) = \begin{bmatrix} r_0 x_d(s) + p_x \\ r_0 y_d(s) + p_y \\ p_z \end{bmatrix} \tag{6.22}$$

which is depicted in Fig. 6.14, where $[x_d, y_d]$ is the quarter of circle defined as

$$\begin{cases} [\cos(\frac{\pi}{2}s), \sin(\frac{\pi}{2}s)] & \text{if} \quad 0 \leq s \leq 1 \\ [0, 2 - s] & \text{if} \quad 1 < s \leq 2 \\ [s - 2, 0] & \text{if} \quad 2 < s \leq 3 \\ [1, 0] & \text{if} \quad s > 3 \end{cases}$$

$r_0 = 0.3$, and $[p_x, p_y, p_z] = [0.3, 0, -0.25]$ (units are given in the MKS system where unspecified). The desired path $r(s) = [\theta_{d1}(s), \theta_{d2}(s), \theta_{d3}(s)]$ in the joint-space is obtained by inverse kinematics, and satisfies Assumption 6.1.

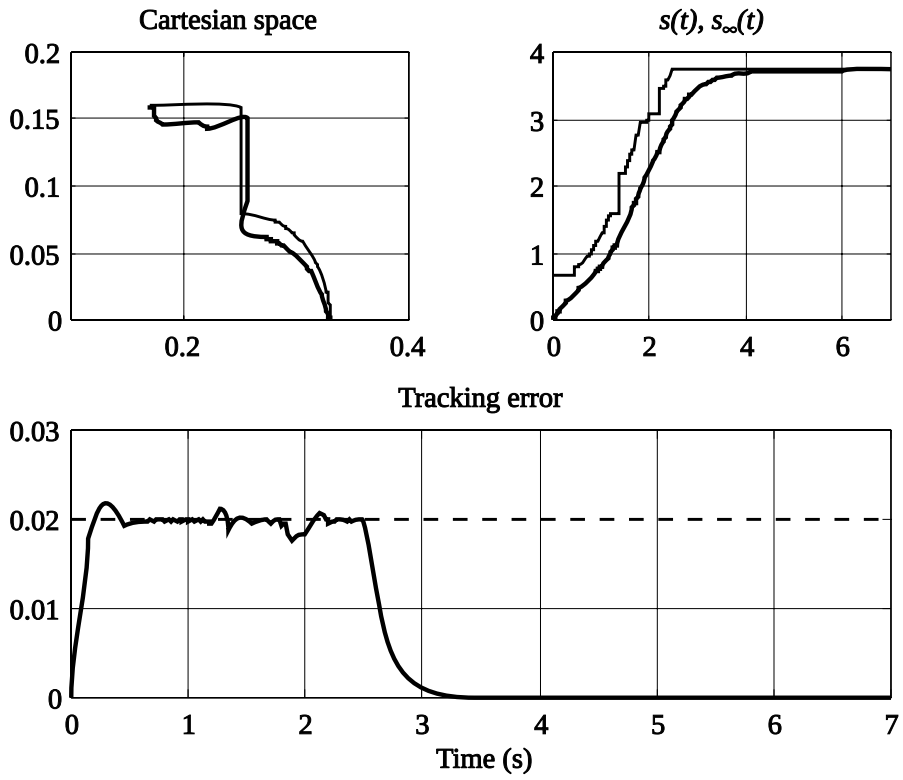


Figure 6.10: Tracking error constraints $e_{\max} = 2$ cm in the presence of model mismatching.

In order to allow the end-effector's Cartesian position Y to track the reference signal $Y_d(s(t))$ in (6.22), the nonlinear feedback task controller (NFTC) reported in [TBMR93]

$$\mathcal{T} = D(q)J(q)^{-1}\left[\ddot{Y}_d(s(t)) + K_p(Y_d - Y) + K_d(\dot{Y}_d - \dot{Y}) - \dot{J}(q,\dot{q})\dot{q}\right] + C(q,\dot{q}) + G(q) \tag{6.23}$$

is adopted as primal controller, with $K_p = 6000 \cdot I_3$, $K_d = 80 \cdot I_3$, As a general rule to design controllers to be used in connection with a PG, in order to maximize the tracking properties one should try to select a primal controller which provides a fast closed-loop response of system (6.1). Usually this corresponds to large violations of the constraints, which therefore can be enforced by inserting a PG. The closed-loop equations (6.1) resulting from (6.21) and (6.23)

$$\ddot{e} + K_d\dot{e} + K_p e = 0 \tag{6.24}$$

are linear, and therefore it is easy to show that Assumptions 6.5, 6.6, 6.3, 6.7, and 6.8 are satisfied. However, these are fulfilled for a wider class of closed-

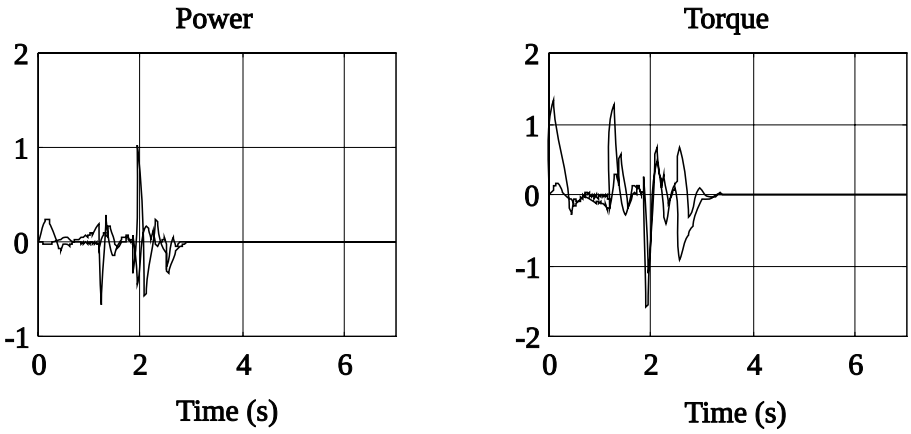


Figure 6.11: Power $\mathcal{T}_i\dot{\theta}_i$ and torque $\mathcal{T}_i$, $i = 1, 2$, for the trajectories of Fig. 6.10.

loop systems. Consider for instance simple individual joint PD controllers with gravity compensation

$$\mathcal{T} = -K_p(q - q_d) - K_d(\dot{q} - \dot{q}_d) + G(q), \tag{6.25}$$

let $x \triangleq [q' \ \dot{q}']'$, $q_d(t) \equiv q_d$, and consider the following function

$$V(x) \ = \ \frac{1}{2}\dot{q}'D(q)\dot{q} + \frac{1}{2}(q_d - q)'K_p(q_d - q)$$

which is a Lyapunov function for (6.21)–(6.25) [AS86]. Since its derivative along the trajectories of the system

$$\dot{V}(x) \ = \ -\dot{q}'K_d\dot{q} > 0,$$

and $V(x) = 0$ iff $x = [q_d' \ 0']$, the first part of Assumption 6.5 is satisfied. Uniform stability is proved as follows. By contradiction, suppose that there exists an $\epsilon > 0$ such that, $\forall\rho > 0$, there exists x_r and t_1 with $\|x(0) - x_r\| \leq \rho$ and $\|x(t_1) - x_r\| > \epsilon$. Since $\gamma_1 I \leq H(x) \leq \gamma_2 I$ for some positive γ_1, γ_2, by denoting by $\lambda_m(K_p)$, $\lambda_M(K_p)$ respectively the minimum and maximum eigenvalue of K_p, and by setting $\gamma_3 = \min\{\lambda_m(K_p), \gamma_1\}$, $\gamma_4 = \max\{\lambda_M(K_p), \gamma_3\}$, it follows that $\|x(t_1) - x_r\| \leq \frac{2}{\gamma_3}V(x(t_1)) \leq \frac{2}{\gamma_3}V(x(0)) \leq \frac{\gamma_4}{\gamma_3}\rho$ for any arbitrary positive ρ, a contradiction. Moreover, in practice, the reference $r(s(t))$ is expressed by a finite numerical precision, and therefore, if $r(s(t))$ tends toward r, after a finite time $r(s(t)) \equiv r$, and Assumption 6.6 is verified. It is easy to check that Assumptions 6.3, 6.7, and 6.8 are satisfied as well.

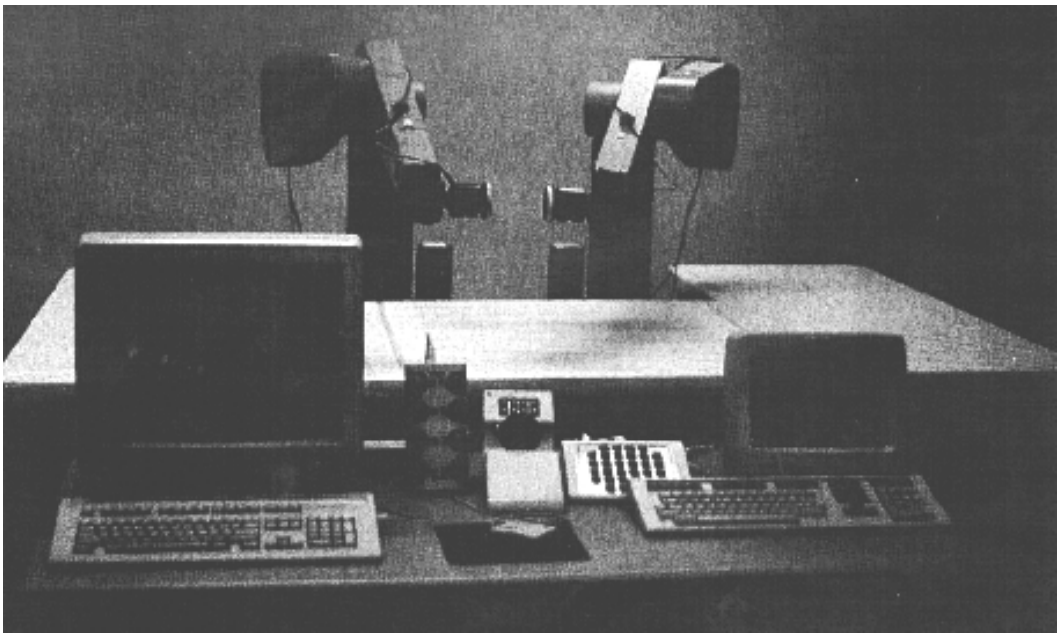


Figure 6.12: Experimental facilities at the Center for Robotics and Automation, Washington University, St.Louis, MO, USA.

We wish to impose the following constraint on the absolute position error

$$\|e\| = \sqrt{(x - x_d)^2 + (y - y_d)^2 + (z - z_d)^2} \leq 5 \text{ mm}, \tag{6.26}$$

on the Cartesian velocity

$$\|\dot{Y}\| \leq 0.6 \text{ m/s}, \tag{6.27}$$

and on the motor voltages

$$V_i = \left(\frac{R_i}{N_i K_i^t}\right) \mathcal{T}_i + (K_i^b N_i)\dot{q}_i \leq 10 \text{ V} \tag{6.28}$$

where the values of armature resistance R_i, gear ratio N_i, torque constant K_i^t, and back EMF constant K_i^b are reported in [TBMR93]. Notice that, because of the choices (6.22) and (6.26)–(6.28) , the constrained vector $c \triangleq [\|e\|^2, \|\dot{Y}\|^2, V_1, V_2, V_3]'$ fulfills Assumption 6.4.

On-line optimization has been performed by using the bisection method mentioned in Sect. 6.5. For numerical integration of (6.1), the linear system (6.24) has been discretized with sampling period $T_{pred} = 0.002$ s in order to predict $x(T; (k-1)T, s_\infty^{k-1})$, and $T_{constr} = 0.005$ s for constraint checking. The resulting program is executed on one CPU in 0.06 s to 0.28 s, allowing the selection of the PG period $T = 0.4$ s. The initial condition $q(0) \approx q_d(0)$, $\dot{q} = 0$

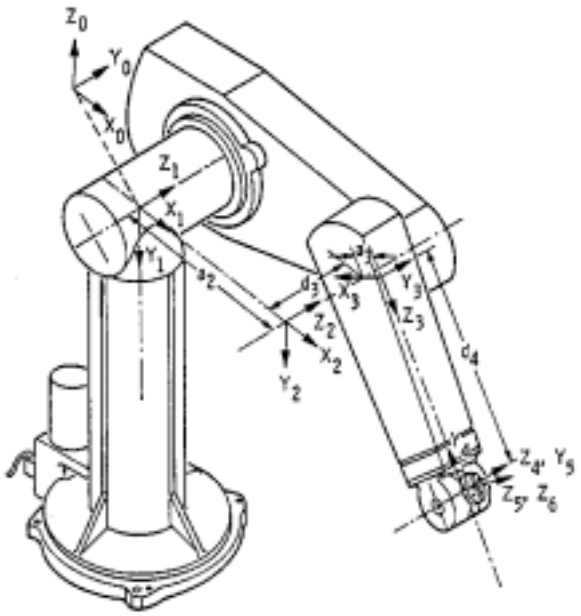


Figure 6.13: Link coordinate assignments for PUMA 560.

and $s_\infty = 0$ satisfies Assumption 6.2. The parameters $\alpha = 2$, $\Delta s_\infty = 0.03$, and $N = 5$ evaluations per period have been selected.

The trajectories recorded during the experiment are depicted in Figs. 6.15, 6.16. The voltage constraints are slightly violated, because of the mismatching between the predictive model and the real system. However, it is clear that one can easily assign more stringent limitations in order to still guarantee constraint fulfillment. Since the derivatives $\dot{s}(t), \ddot{s}(t)$ are only piecewise continuous, $s(t)$ has been smoothed out by a low-pass filter before being used to parameterize the reference $Y_d(s)$. The resulting filtered signal $\gamma(t)$ is also depicted in Fig. 6.15. Finally, we point out that, because of the adopted receding horizon strategy and the particular structure (6.4), the resulting path parameterization $s(t)$ is only near-minimum time. The performance is also affected by the available computational power, which sets a lower bound on T, and therefore on how often the PG can receive feedback and provide new temporary end-points.

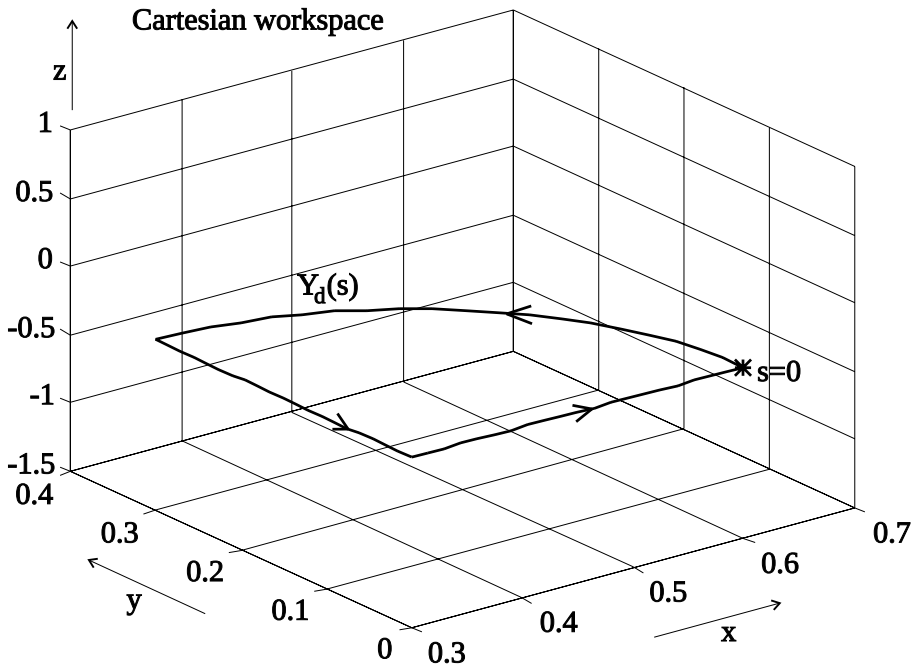


Figure 6.14: Desired geometric path $Y_d(s)$.

6.9 Concluding Remarks

For robotic systems, this chapter has addressed the problem of tracking a given geometric path while satisfying constraints on the variables of the system. A time-parameterization of the path is generated on-line by performing at fixed intervals a scalar constrained optimization based on the integration of the robot's dynamic equations, and the method has been shown to be implementable in real-time.

The proposed strategy introduces a new design approach. Tedious trial-and-error sessions to tune the primal controller parameters in order to satisfy physical constraints—e.g. on motor voltages—are no longer required: Once the primal controller's design knobs have been roughly selected, the constraints are automatically enforced by the PG. Because of the general class of robot models, primal controllers, and physical constraints considered in this chapter, we believe that the proposed approach is versatile enough to cope with many different practical robotic applications.

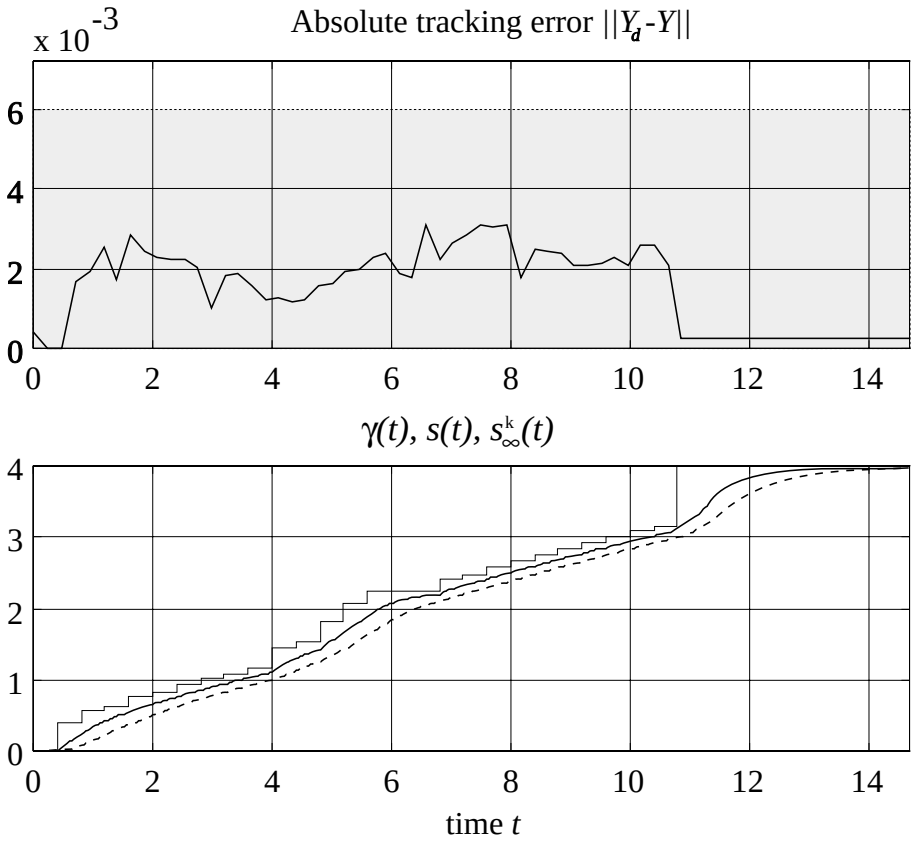


Figure 6.15: Absolute tracking error, generated $s(t)$ (thick line), s_∞^k (thin line), and $\gamma(t)$ (dashed line).

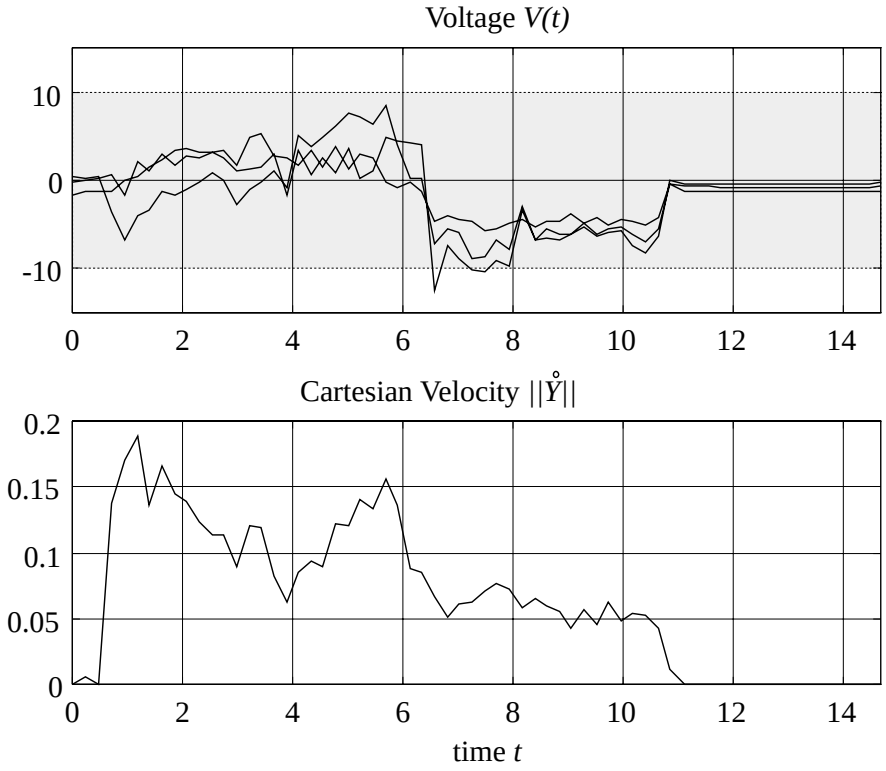


Figure 6.16: Motor voltage V_i, $i = 1, 2, 3$, and Cartesian velocity $\|\dot{Y}\|$ for the trajectories of Fig. 6.15.

Chapter 7

Teleoperated Systems

7.1 Introduction

The teleoperation problem consists of controlling a plant from a remote location through a communication channel, which is usually affected by signal transmission delays and failures. Delays can be imposed by: limits on the speed of light during radio transmission, e.g. 0.4 s for vehicles in low earth orbit [She93]; limits on the speed of sound during acoustic telemetry, for example 2 s on a 1700 m round-trip; network traffic, such as on a TCP/IP connection on the Internet. While the first two are exactly predictable or even constant (for example in space applications), the third kind of delay is a-priori unbounded. The presence of delays highly disturbs the human operator's intuition and can lead to instability of the overall teleoperation. A widely adopted solution consists of using computer-graphic predictors which, on the basis of the last available state measurement from the remote location and a reliable dynamic model of the plant, generate virtual representations for the human operator. Therefore, the predictor's action compensates the delay in the back channel. By even generating predictions which go further in the future, the delay on the forward channel can be canceled as well, as the operator generates commands according to a "future" situation, commands which can be executed in time as they arrive at the remote location. This scheme assumes that time-delays can be upperbounded. On the other hand, when time-delays can be arbitrarily large, a possible lack of commands can take place at the remote location. In such a circumstance, it is desirable to have (open-loop)

emergency sequences of commands which allow to avoid unwanted behaviours of the plant. Predictive controllers [RM93, MLZ90, CS91, BCM97] naturally provide this kind of sequences. In fact, a predictive controller selects a virtual command sequence at each time step t by solving a convex (constrained) optimization problem. This sequence maximizes the tracking performance while producing a predicted evolution of the system which fulfils some possible hard constraints on input/state variables. Then the first command is applied and the remaining ones are thrown away, and a new one is selected at time $t+1$, according to the so called *receding horizon* strategy. However, when the loop is broken by the forward time-delay and therefore the new sequence does not arrive, the subsequent terms of the last sequence can be applied to the plant. In this chapter we describe the use of predictive controllers, for teleoperation of systems subject to input/state constraints through transmission channels with unbounded delays. In particular, we focus out attention on special class of predictive controllers, the reference governors (RG) [BCM97, Bem97b, GKT95]. We assume that the plant to be controlled has some stability properties, possibly provided by a local feedback loop. On the other side of the channel, a RG filters the desired trajectories to be tracked, either generated by human operators or by computer programs, so that at the remote side fulfillment of constraints on input/state variables of the plant is guaranteed, and the tracking performance is maximized.

7.2 Problem Formulation

The reference governors developed in the previous sections can be effectively adopted for the teleoperation through a channel with unbounded time-delays, such as Internet TCP/IP connections. For the sake of simplicity of notation, we shall focus out attention on discrete-time primal systems of the form

$$\begin{cases} x(t+1) &= \phi(x(t), w(t)) \\ y(t) &= h(x(t), w(t)) \\ c(t) &= \ell(x(t), w(t)) \end{cases} \tag{7.1}$$

where the output y has to track a desired reference r while c satisfies the constraint $c(t) \in \mathcal{C}$, $\forall t \geq 0$. We assume that system (7.1), the reference r, and the constraint set $\mathcal{C}$ satisfy certain assumptions so that, by applying one of the

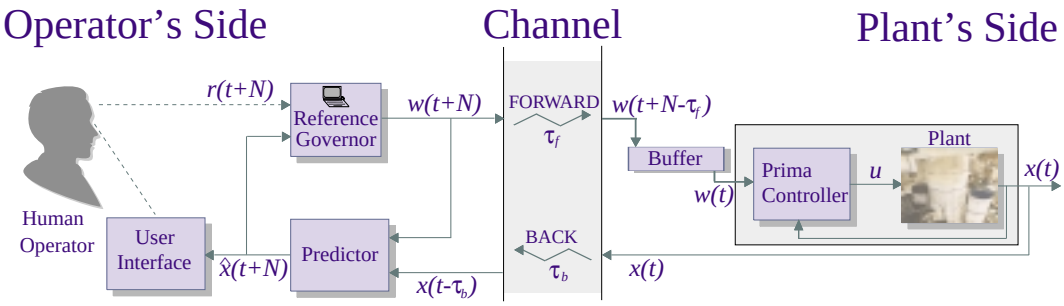


Figure 7.1: Reference Governor and teleoperation through a channel with delays

reference governors developed in the previous chapters, the desired convergence properties are obtained in the absence of time-delays.

The control configuration through a delayed channel is depicted in Fig. 7.1. Let τ_f and τ_b denote respectively the forward and backward communication delays. The plant is equipped by a local controller for stability, tracking, and robustness purposes. At time t, from the plant's side the measurement of the current state $x(t)$ is sent to the operator's side. Here, at the same time t, $x(t-\tau_b)$ is received. This value is used to predict an estimate $\hat{x}(t+N-1)$ of the future state, which is visualized through a user interface to the operator. Accordingly, the operator generates a new command $r(t+N-1)$, which is processed by the RG to obtain the filtered desired reference $w(t+N-1)$. Simultaneously, at the plant's side $w(t+N-1-\tau_f)$ is received and buffered. This scheme is synchronized and able to provide $w(t)$ at time t if $0 \leq \tau_f \leq N-1$.

In principle, any controller could be used in the above scheme instead of the RG. However, when time-delays cannot be a-priori upperbounded, for example on an Internet connection, the above scheme is not able to mantain synchronization between the operator's and the plant's side. As remarked above, an interesting feature of predictive controllers like the RG is that seminfinite virtual command sequences are available at each time-step and can be "safely " applied to the plant—i.e. produces an evelution which satisfies the given constraints— no matter how large is the forward time-delay. Henceforth, predictive controllers can be used to still ensure constraint fulfillment and avoid loss of synchronization between the teleoperator and the plant by introducing some additional logic. This is depicted in the flow-charts in Figs. 7.2-7.3. Consider the plant's side algorithm (Fig. 7.3). As soon as the buffer becomes

empty because of a forward delay greater than $N-1$, say this happens at time t_r, an alarm is sent to the RG, and the plant enters a *recovery-mode*, where it is supplied by the last available virtual command sequence until acknowledged by the RG. The alarm is labeled with the key $t^p_{old} \triangleq t_r - 1$. The recovery-mode is terminated as soon as new commands arrive from the RG, in time to be executed, which are labeled as t^p_{old}.

On the RG's side (Fig. 7.2), the predictor begins to provide wrong estimates as soon as the plant enters a recovery-mode, since the RG commands issued for times $t > t_r$ have no longer been executed by the plant. As a new key t^p_{old} is received at time t^r_{rec}, the predicted state estimate is corrected and the issued commands are labeled with the new key t^p_{old}. This allows the plant to disregard the commands received through the channel until those ones labeled as t^p_{old} are received. The RG assumes that the plant will end the recovery state later on at time $t_c \triangleq t^r_{rec} + N - 1$. As soon as the RG receives from the plant data referring to the time $t - \tau_b = t_c$, the RG actually verifies that the recovery mode was terminated properly, i.e. at time t_c the plant was not sending alarms ($t^r = -1$). If this is not the case, again the predicted state estimate is corrected and t^r_{rec} is set to the current time t.

Assume that the plant is at an equilibrium state x_0 corresponding to the command $\theta_0 = [\mu_0' \; w_0']'$. The execution of the algorithm at the RG's side is started as soon as x_0 is received through the channel at time $t = 0$. The two algorithms are intialized as follows: `status`=0, `recovery`=0, $t^p_{old} = t^r_{old} = -1$, $\theta^p_{old} = \theta_0$, $t^r_{rec} = -1$, $t_c = -1$.

7.3 Simulation Results

The teleoperation scheme described in the previous sections is applied to the nonminimum-phase SISO system (same as Example 1, Chap. 2)

$$\begin{cases} x(t+1) &= \begin{bmatrix} 1.5402 & -0.6703 \\ 1 & 0 \end{bmatrix} x(t) + \begin{bmatrix} 1 \\ 0 \end{bmatrix} w(t) \\ y(t) &= \begin{bmatrix} -0.8935 & 1.0237 \end{bmatrix} x(t) \\ c(t) &= y(t) \end{cases} \tag{7.2}$$

The unit step response of (7.2) is depicted in Fig. 7.4a (thin line). As a predictive controller, the RG developed in Chap. 2 has been selected along

with the parameters $\delta = 0.05$, $\gamma = 0.99$, $\Psi_\mu = 1$, $\Psi_w = 10$, $\Psi_y = 0$. The task of the CG is to bound the output between -0.5 and 5. Accordingly, $c(t) = y(t)$ and $\mathcal{C} = [-0.5, 5]$. The related constrained unit step response is shown in Fig. 7.4a (thick line). Fig. 7.4b depicts the generated command trajectory $g(t)$ (thick line), the reference trajectory $r(t)$ (thick dashed line), and minimizing parameters $\mu(t)$ (thin line) and $w(t)$ (thin dashed line).

Teleoperation is performed through a delayed channel, where delays $\tau_f(t)$, $\tau_b(t)$ are not known in advance and unbounded. These are randomly generated between 1 and 19, with the constraint $\tau_f(t) - \tau_f(t-1) \leq 1$, $\tau_b(t) - \tau_b(t-1) \leq 1$. The commands are generated $N = 10$ steps in advance. The resulting trajectories are depicted in Fig. 7.5

In order to consider the effects of noise, a similar experiment is repeated by adding a measurement noise

$$x(t - \tau_b) = x_{real}(t - \tau_b) + \xi$$

where $|\xi_1|$, $|\xi_2| \leq 0.3$. Fig. 7.6 exhibits the related output response (thick dashed line). The prescribed lower bound is slightly violated.

7.4 Conclusions

This chapter has presented a teleoperation scheme through communication channels with unbounded delays for constrained systems. This is achieved by using predictive controllers, because of their feature to supply safe commands despite arbitrarily large time delays. The reason to place the predictive controller at the operator's side is twofold. First, the controller requires dedicated hardware (the more powerful the more accurate is the model of the plant) which sometimes can be not worth to place remotely, for example in space applications or in highly noisy environments. Second, a single remote server can run several predictive controllers for as many plants positioned in different locations and wired through a newtwork. For these reasons, we wish that the proposed approach will be experimented for teleoperation through the Internet.

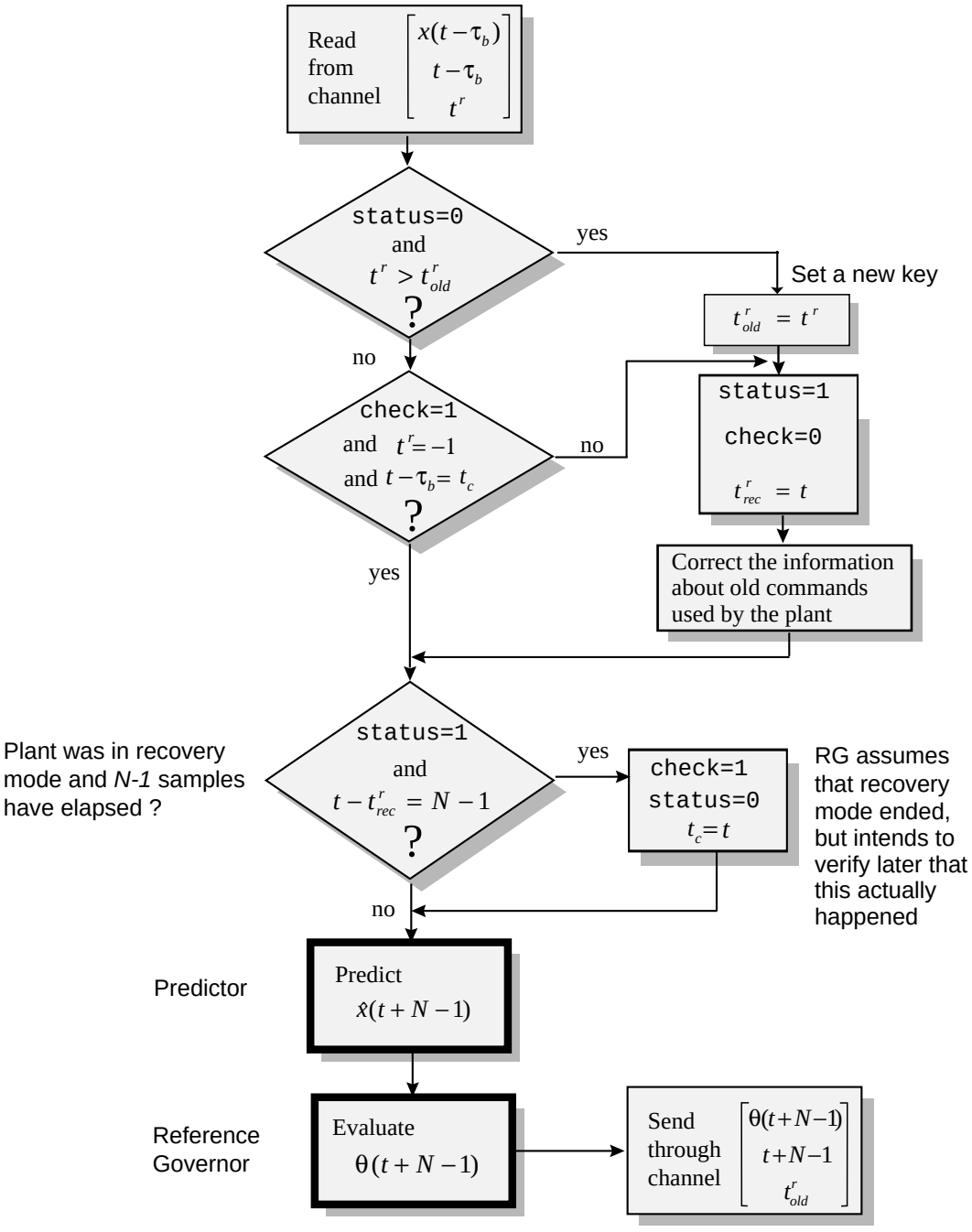


Figure 7.2: RG's side algorithm

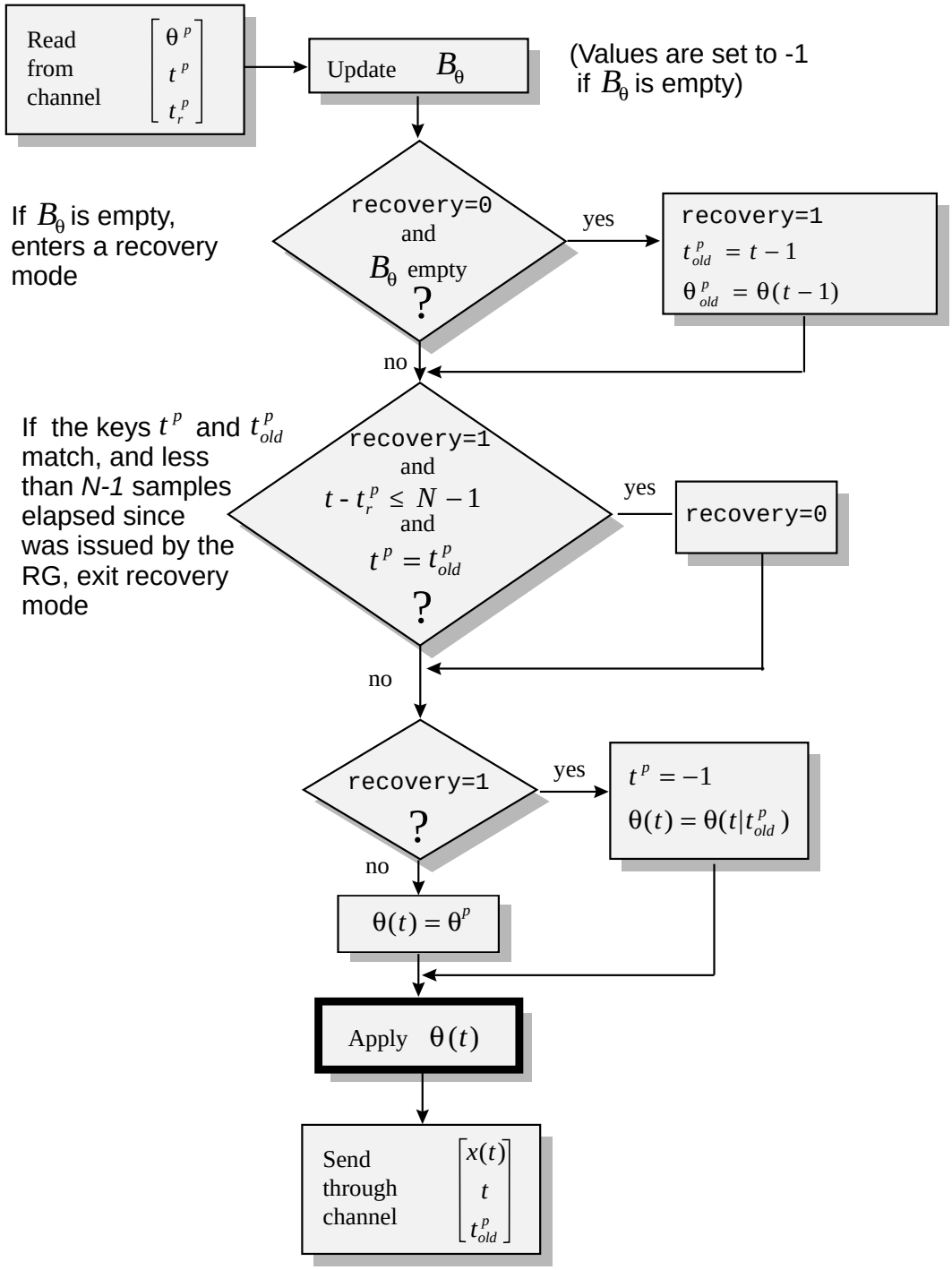


Figure 7.3: Plant's side algorithm

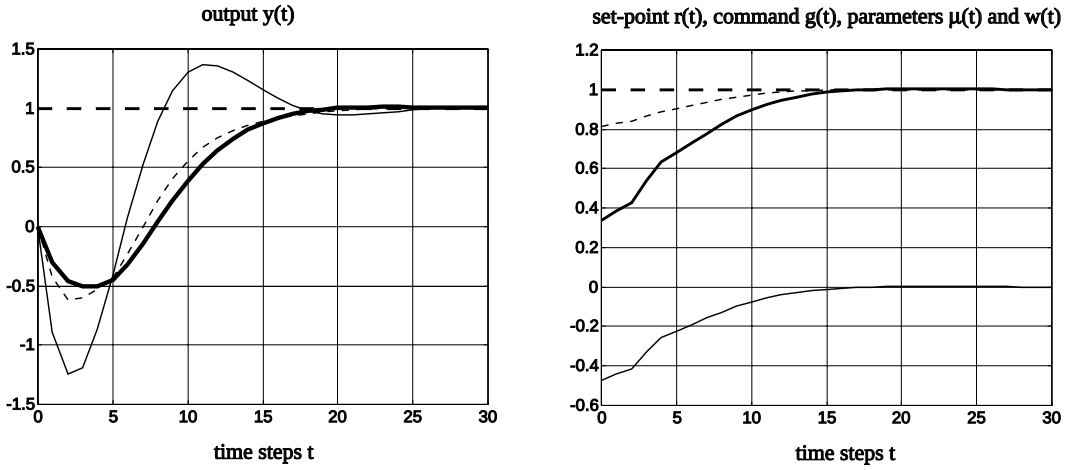


Figure 7.4: Example 1: (a) [left] Unit step response with no CG (thin line) and with CG ($\delta = 0.05$; $\gamma = 0.99$; $\Psi_\mu = 1$; $\Psi_w = 10$; $\Psi_y = 0$; thick line) for the nominal plant (7.2); Response with CG for the perturbed plant (2.63) (dashed line). (b) [right] Reference trajectory $r(t)$ (thick dashed line); Generated command trajectory $g(t)$ (thick line); minimizing parameters $\mu(t)$ (thin solid line) and $w(t)$ (thin dashed line).

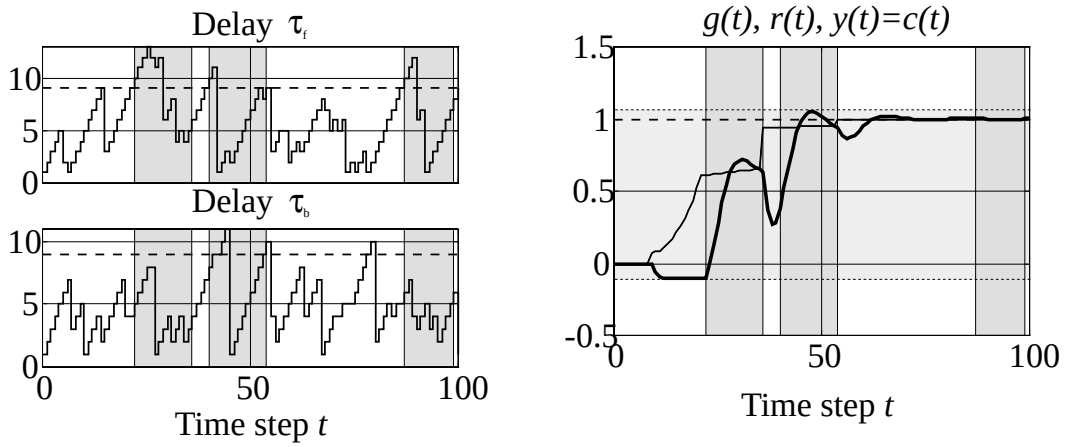


Figure 7.5: Teleoperation through a delayed channel. The intervals of recovery-state are depicted as gray areas

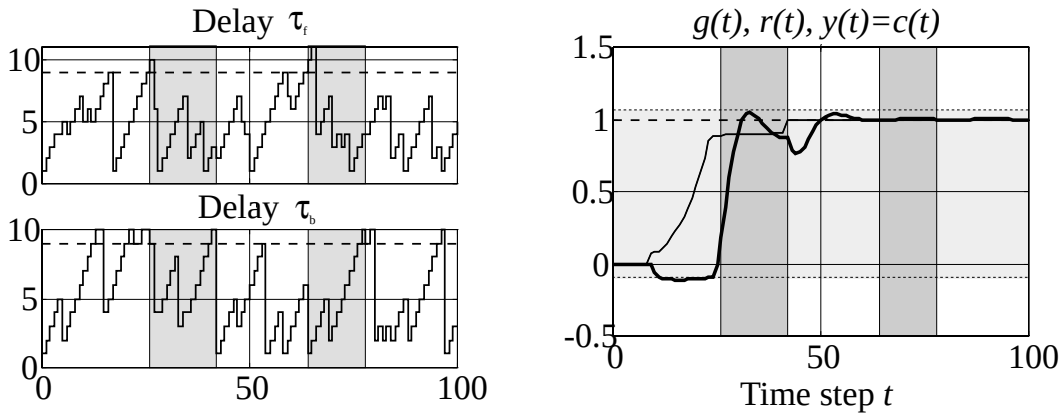


Figure 7.6: Teleoperation through a delayed channel, with state-measurement noise. The intervals of recovery-state are depicted as gray areas

Conclusions and Directions for Future Resarch

This dissertation has introduced and investigated the *reference governor* (RG) approach to fulfillment of input/state hard constraints. This consists of a higher-level module which prefilters the desired reference signal in such a way that a primal compensated control system can operate in a stable way with satisfactory tracking performance and no constraint violation. In contrast with other predictive control approaches which attempt to solve stabilization/tracking and constraint fulfillment at the same time, we have adopted a new hierarchical philosophy, based on a separation of these two requirements: First, stabilize the system, by using one of the many available control techniques, for the unconstrained case; then, impose constraint fulfillment by adding the RG. The RG has been shown to require a computational burden which is much more moderate than traditional predictive control schemes, whose application has been limited up to now to slow processes.

Each chapter has developed and specialized theoretical, computational, and simulative/experimental results for a broad class of systems: Linear, linear uncertain, linear with disturbances and partial state information, nonlinear, robotic, teleoperated through channels with uncertain delays (e.g. the Internet).

Future research might be directed towards the investigation of reference governors which are able to integrate decision (logic) variables in the optimization problem, for which efficient integer programming techniques are now available. In particular, the ability to provide fault tolerance seems to be a key requirement in future intelligent control systems. Besides further theoretical developments, it remains to explore a wide area of possible

applications, in order to understand where RG schemes might provide feasible control solutions.

As a final step, for a complete professional and personal satisfaction, we wish that, after a successful acceptance by the scientific community [BCM97, BM97, Bem97b], the RG will find industrial application. Not *in place* of existing conventional controllers, but *in addition* to conventional controllers, as a supervising higher-level module for constraint fulfilment and improved performance.

Bibliography

[ABKL94] J. Anderson, T. Bachx, M. King, and J. Van Loon. Getting the most from advanced process control. *Chemical and Engineering*, pages 78–89, March 1994.

[AP92] J.C. Allwright and G.C. Papavasiliou. On linear programming and robust model-predictive control using impulse-responses. *Systems & Control Letters*, 18:159–164, 1992.

[APR96] C. Basso Amolat, S. Polimeni, and A. Rocchetti. Controllo di constraints – la via più economica all'ottimizzazione. *Automazione e Strumentazione*, pages 129–137, January 1996. In Italian.

[AS86] H. Asada and J.-J. E. Slotine. *Robot Analysis and Control*. John Wiley and Sons, 1986.

[Aub91] J.P. Aubin. *Viability Theory*. Birkhäuser, Boston, 1991.

[BB88] A. Benzaouia and C. Burgat. The regulator problem for a class of linear systems with constrained control. *Systems & Control Letters*, 10:357–363, 1988.

[BB91] S.P. Boyd and C.H. Barrat. *Linear Controller Design: Limits of Performance*. Prentice-Hall, Englewood Cliffs, N. J., 1991.

[BCM94] A. Bemporad, L. Chisci, and E. Mosca. On the stabilizing property of the zero terminal state receding horizon regulation. *Automatica*, 30(12):2013–2015, 1994.

[BCM97] A. Bemporad, A. Casavola, and E. Mosca. Nonlinear control of constrained linear systems via predictive reference management. *IEEE Trans. Automat. Control*, AC-42(3):340–349, 1997.

[BDG85] J. Bobrow, S. Dubowsky, and J. Gibson. Time-optimal control of robotic manipulators along specified paths. *Int. J. Robotics Research*, 4(3):3–17, 1985.

[Bem97a] A. Bemporad. Control of constrained nonlinear systems via reference management. In *American Control Conference*, Albuquerque, U.S.A., 1997.

[Bem97b] A. Bemporad. Reference governor for constrained nonlinear systems. *IEEE Trans. Autom. Control*, 1997. to appear.

[BG96] G. Bitsoris and E. Gravalou. Design techniques for the control discrete-time systems subject to state and control constraints. Technical Report Tech. Rep. UP CSL 96006, University of Patras, Control Systems Laboratory, June 1996.

[BG97] A. Bemporad and A. Garulli. Predictive control via set-membership state estimation for constrained linear systems with disturbances. In *European Control Conference*, Bruxelles, Belgium, 1997.

[BGed] A. Bemporad and A. Garulli. Output-feedback predictive control of constrained linear systems with disturbances via set-membership state estimation. *Int. Journ. Robust and Nonlinear Control*, submitted.

[BGV90] R.R. Bitmead, M. Gevers, and Wertz V. *Adaptive Optimal Control—The Thinking Man's GPC*. Prentice Hall of Australia Pty Ltd., 1990.

[Bit88a] G. Bitsoris. On the positive invariance of polyhedral sets for discrete-time systems. *Systems & Control Letters*, 11:243–248, 1988.

[Bit88b] G. Bitsoris. Positive invariant sets of discrete-time linear systems. *Int. J. Contr.*, 47(6):1713–1726, 1988.

[Bla90a] F. Blanchini. Control synthesis for discrete time systems with control and state bounds in the presence of disturbances. *J. of Optimization Theory and Applications*, 65(1):29–40, 1990.

[Bla90b] F. Blanchini. Feedback control for linear time-invariant systems with state and control bounds in the presence of disturbances. *IEEE Trans. Autom. Control*, AC-35(11):1231–1234, 1990.

[BM94a] A. Bemporad and E. Mosca. Constraint fulfilment in control systems via predictive reference management. In *Proc. 33rd IEEE Conf. on Decision and Control*, pages 3016–3022, Lake Buena Vista, FL., U.S.A., 1994.

[BM94b] A. Bemporad and E. Mosca. Constraint fulfilment in feedback control via predictive reference management. In *Proc. 3rd IEEE Conf. on Control Applications*, pages 1909–1914, Glasgow, U.K., 1994.

[BM95a] A. Bemporad and E. Mosca. Nonlinear predictive reference filtering for constrained tracking. In *Proc. 3rd European Control Conf.*, pages 1720–1725, Roma, Italy, 1995.

[BM95b] A. Bemporad and E. Mosca. Nonlinear predictive reference governor for constrained control systems. In *Proc. 34th IEEE Conf. on Decision and Control*, pages 1205–1210, New Orleans, Lousiana, U.S.A., 1995.

[BM95c] D.S. Bernstein and A.N. Michel. A chronological bibliography on saturating actuators. *Int. J. Robust and Nonlinear Control*, 5(5):375–380, 1995. Special issue on saturating actuators.

[BM96] A. Bemporad and E. Mosca. Robust nonlinear reference filtering for constrained linear systems with uncertain impulse/step responses. In *35th IEEE Conference on Decision and Control*, pages 3527–3528, Kobe, Japan, 1996.

[BM97] A. Bemporad and E. Mosca. Fulfilling hard constraints in uncertain linear systems by reference managing. *Automatica*, 1997. to appear.

[BR71] D.P. Bertsekas and I.B. Rhodes. Recursive state estimation for a set-membership description of uncertainty. *IEEE Trans. Aut, Control*, 16:117–128, 1971.

[BS79] M.S. Bazaraa and C.M. Shetty. *Nonlinear Programming — Theory and Algorithms*. John Wiley & Sons, 1979.

[BS90] V. Broman and M.J. Shensa. A compact algorithm for the intersection and approximation of n-dimensional polytopes. *Mathematics and Computers in Simulation*, 32:469–480, 1990.

[BT97] A. Bemporad and T.J. Tarn. On-line path parameterization for manipulators with input/state constraints. In *Proc. IEEE/ASME Intern. Conf. on Adv. Intell. Mechatr. (AIM97)*, Tokio, Japan, 1997.

[BTX97] A. Bemporad, T.J. Tarn, and N. Xi. Predictive path parameterization for constrained robot control. Technical Report Lab Report SSM-RL-97-22, Dept. Systems Science and Mathematics, Washington University, St. Louis, MO, 1997. submitted IEEE Trans. Control Systems Technology.

[BV93] G. Bitsoris and M. Vassilaki. Design techniques of linear constrained discrete-time control systems. *Control and Dynamic Systems*, 56:1–49, 1993.

[Cam90] P.J. Campo. *Studies in Robust Control of Systems Subject to Constraints*. Ph.D. dissertation, California Institute of Technology, Pasadena, CA, U.S.A., 1990.

[CGZ19] L. Chisci, A. Garulli, and G. Zappa. Recursive state bounding by parallelotopes. *Automatica*, 32(7):1049–1056, 19.

[Che80] F.L. Chernousko. Optimal guaranteed estimates of indeterminacies with the aid of ellipsoids. parts i-iii. *Engineering Cybernetics*, 18, 1980.

[Chi97] S.L. Chiu. Developing commercial applications of intelligent control. *IEEE Control Systems Magazine*, pages 94–97, April 1997.

[CL55] E. A. Coddington and N. Levinson. *Theory of Ordinary Differential Equations*. McGraw Hill, 1955.

[Cla94] D.W. Clarke. Advances in model-based predictive control. In *Advances in Model-Based Predictive Control*, pages 3–21. Oxford University Press Inc., New York, 1994.

[CM87] P.J. Campo and M. Morari. Robust model predictive control. In *Proc. American Cont. Conf.*, pages 1021–1026, 1987.

[CM89] D.W. Clarke and C. Mohtadi. Properties of generalized predictive control. *Automatica*, 25(6):859–875, 1989.

[CMT87a] D.W. Clarke, C. Mohtadi, and P.S. Tuffs. Generalized predictive control – ii. extensions and interpretations. *Automatica*, 23:149–160, 1987.

[CMT87b] D.W. Clarke, C. Mothadi, and P.S. Tuffs. Generalized predictive control. *Proc. IEE*, 140:247–354, 1987.

[Cor94] P.I. Corke. *The Unimation Puma servo system*, volume `ftp://janus.cat.csiro.au /pub/pic/pumaservo.ps.Z`. MTM-226, 1994.

[CR80] C.R. Cuttler and B.C. Ramaker. Dynamic matrix control—a computer control algorithm. In *American Control Conference*, volume WP5-B, San Francisco, USA, 1980.

[CS91] D.W. Clarke and R. Scattolini. Constrained receding-horizon predictive control. *Proc. IEE*, 140:247–354, 1991.

[DN90] O. Dahl and L. Nielsen. Torque-limited path following by on-line trajectory scaling. *IEEE Trans. Robotics Automat.*, RA-6(5):554–561, 1990.

[DSE87] J.C. Doyle, R.S. Smith, and D.F. Enns. Control of plants with input saturation nonlinearities. In *Proc. American Cont. Conf.*, pages 1034–1039, Minneapolis, MN, 1987.

[FH82] E. Fogel and F. Huang. On the value of information in system identification - bounded noise case. *Automatica*, 18:229–238, 1982.

[FKSV94] T.F. Filippova, A.B. Kurzhanski, K. Sugimoto, and I. Valyi. Ellipsoidal state estimation for uncertain dynamical systems. In *Bounding Approaches to System Identification*. Plinum Poers, 1994.

[GC86] P.O. Gutman and M. Cwikel. Admissible sets and feedback control for discrete-time linear dynamical systems with bounded control and states. *IEEE Trans. Automat. Control*, AC-31(4):373–376, 1986.

[GC87] P.O. Gutman and M. Cwikel. An algorithm to find maximal state constraint for discrete-time linear dynamical systems with bounded control and states. *IEEE Trans. Automat. Control*, AC-32(3):251–254, 1987.

[GK92] T.J. Graettinger and B.H. Krogh. On the computation of reference signal constraints for guaranteed tracking performance. *Automatica*, 28:1125–1141, 1992.

[GK95a] E.G. Gilbert and I. Kolmanovsky. Discrete-time reference governors for systems with state and control constraints and disturbance inputs. In *Proc. 34th IEEE Conf. on Decision and Control*, pages 1189–1194, 1995.

[GK95b] E.G. Gilbert and I. Kolmanovsky. Maximal output admissible sets for discrete-time systems with disturbance inputs. In *Proc. American Control Conf.*, pages 2000–2005, 1995.

[GKT95] E.G. Gilbert, I. Kolmanovsky, and K. Tin Tan. Discrete-time reference governors and the nonlinear control of systems with state and control constraints. *Int. Journal of Robust and Nonlinear Control*, pages 487–504, 1995.

[GPM89] C.E. García, D.M. Prett, and M. Morari. Model predictive control: theory and practice – a survey. *Automatica*, 25(3):335–348, 1989.

[GS83] A. H. Glattfelder and W. Schaufelberger. Stability analysis of single-loop control systems with saturation and antireset-windup circuits. *IEEE Trans. Aut. Control*, 28(12):1074–1081, December 1983.

[GT91] E.G. Gilbert and K. Tin Tan. Linear systems with state and control constraints: the theory and applications of maximal output admissible sets. *IEEE Trans. Automat. Control*, 36:1008–1020, 1991.

[Hol84] J.M. Hollerbach. Dynamics scaling of manipulator trajectories. *ASME J. Dyn. Syst., Meas. , Cont.*, 105:102–106, 1984.

[KA85] P. Kapasouris and M. Athans. Multivariable control systems with saturating actuators antireset windup stategies. In *Proc. American Cont. Conf.*, pages 1579–1584, 1985.

[KA90] P. Kapasouris and M. Athans. Control systems with rate and magnitude saturation for neutrally stable open loop systems. In *Proc. 29th Conf. Decision and Control*, pages 3404–3409, Honolulu, HI, December 1990.

[KAS88] P. Kapasouris, M. Athans, and G. Stein. Design of feedback control systems for stable plants with saturating actuators. In *Proc. 27th IEEE Conf. on Decision and Control*, pages 469–479, Austin, Texas, U.S.A., 1988.

[KAS90] P. Kapasouris, M. Athans, and G. Stein. Design of feedback control systems for unstable plants with saturating actuators. In *Proc. IFAC Symp. on Nonlinear Control System Design*, pages 302–307, Pergamon Press, 1990.

[KBM96] M.V. Kothare, V. Balakrishnan, and M. Morari. Robust constrained model predictive control using linear matrix inequalities. *Automatica*, 32(10):1361–1379, 1996.

[KG85] S.S. Keerthi and E.G. Gilbert. An existence theorem for discrete-time infinite-horizon optimal control problems. *IEEE Trans. Autom. Control*, AC-30(9):907–909, 1985.

[KG87] S.S. Keerthi and E.G. Gilbert. Computation of minimum time feedback control laws for discrete-time systems with state-control constraints. *IEEE Trans. Autom. Control*, AC-32(5):432–435, 1987.

[KG88] S.S. Keerthi and E.G. Gilbert. Optimal infinite-horizon feedback control laws for a general class of constrained discrete-time systems: stability and moving-horizon approximations. *J. Opt. Theory and Applications*, 57:265–293, 1988.

[KKP96] A. Kumagai, D. Kohli, and R. Perez. Near-minimum time feedback controller for manipulators using on-line time scaling of trajectories. *ASME J. Dyn. Syst., Meas. , Cont.*, 118:300–308, 1996.

[KNM95] M.V. Kothare, V. Nevistić, and M. Morari. Robust constrained model predictive control for nonlinear systems: A comparative study. In *Proc. 34th Conf. Decision and Control*, pages 2884–2885, New Orleans, LO, December 1995.

[Kos83] R.L. Kosut. Design of linear systems with saturating linear control and bounded states. *IEEE Trans. Autom. Control*, AC-28(1):121–124, 1983.

[Kot97] M.V. Kothare. *Control of Systems Subject to Constraints*. Ph.D. dissertation, California Institute of Technology, Pasadena, CA, U.S.A., March 1997.

[KS94] R. Kelly and R. Salgado. Pd control with computed feedforward of robot manipulators: a design procedure. *IEEE Trans. Robotics Automat.*, 10:566–571, 1994.

[Lau94] G. K. Lausterer. Power plants. economic benefits of advanced control. *Automazione e Strumentazione*, pages 69–78, September 1994.

[MA91] S.B. Moon and S. Ahmad. Time scaling of cooperative multi-robot trajectories. *IEEE Trans. Syst., Man, Cyber.*, 21(4):900–908, 1991.

[MLZ90] E. Mosca, J. M. Lemos, and J. Zhang. Stabilizing i/o receding-horizon control. In *Proc. 29th IEEE CDC*, pages 2518–2523, 1990.

[MM90] D.Q. Mayne and H. Michalska. Receding horizon control of nonlinear systems. *IEEE Trans. Automat. Control*, 35:814–824, 1990.

[Mos95] E. Mosca. *Optimal, Predictive, and Adaptive Control*. Prentice Hall, Englewood Cliffs, New York, 1995.

[MP93] D.Q. Mayne and E. Polak. Optimization based design and control. In *Preprints 12th IFAC World Congress*, volume 3, pages 129–138, Sydney, 1993.

[MZ89] M. Morari and E. Zafiriou. *Robust Process Control*. Prentice Hall, Inc., Englewood Cliffs, NJ, 1989.

[NMS96] G. De Nicolao, L. Magni, and R. Scattolini. Robust predictive control of systems with uncertain impulse response. *Automatica*, 32:1475–1479, 1996.

[Oli96] S.L. De Oliveira. *Model Predictive Control (MPC) for Constrained Nonlinear Systems*. Ph.D. dissertation, California Institute of Technology, Pasadena, CA, U.S.A., March 1996.

[OM94] S.L. De Oliveira and M. Morari. Robust model predictive control for nonlinear systems. In *Proc. 33rd Conf. Decision and Control*, pages 3561–3567, Orlando, FL, 1994.

[PY93a] E. Polak and T.H. Yang. Moving horizon control of linear systems with input saturation and plant uncertainty–part 1. robustness. *Int. Journal of Control*, 58(3):613–638, 1993.

[PY93b] E. Polak and T.H. Yang. Moving horizon control of linear systems with input saturation and plant uncertainty–part 2. disturbance rejection and tracking. *Int. Journal of Control*, 58(3):639–663, 1993.

[Ric93] J. Richalet. Industrial applications of model based predictive control. *Automatica*, 29:1251–1274, 1993.

[RM93] J.B. Rawlings and K.R. Muske. The stability of constrained receding-horizon control. *IEEE Trans. Automat. Control*, 38:1512–1516, 1993.

[RRTP78] J. Richalet, A. Rault, J. L. Testud, and J. Papon. Model predictive heuristic control-application to industrial processes. *Automatica*, 14:413–428, 1978.

[San76a] J. M. Martín Sanchez. Adaptive predictive control system. *U.S. Patent No. 4, 196, 576*, 1976.

[San76b] J. M. Martín Sanchez. A new solution to adaptive control. *Proc. IEEE*, 64:1209–1218, 1976.

[SC94] P.O.M. Scokaert and D. W. Clarke. Stability and feasibility in constrained predictive control. In *Advances in Model-Based Predictive Control*, pages 217–229. Oxford Press Inc.,N. Y., New York, 1994.

[Sch68] F.C. Schweppe. Recursive state estimation: unknown but bounded errors and system inputs. *IEEE Trans. Aut. Control*, 13:22–28, 1968.

[She93] T.B. Sheridan. Space teleoperation through time delay: Review and prognosis. *IEEE Trans. Robotics Automat.*, 9(5):592–606, 1993.

[SM85] K. Shin and N.D. McKay. Minimum-time control of robotic manipulators with geometric path constraints. *IEEE Trans. Robotics Automat.*, 30(6):531–541, 1985.

[Soe92] R. Soeterboek. *Predictive Control. A Unified Approach.* Prentice-Hall, Englwood Hill, N. J., 1992.

[Son95] E. Sontag. On the input-to-state stability property. *European Journal of Control*, 1(2):24–36, 1995.

[SR96] P.O.M. Scokaert and J.B. Rawlings. Infinite horizon linear quadratic control with constraints. In *Proc. IFAC*, volume 7a-04 1, pages 109–114, San Francisco, USA, 1996.

[SRM97] P.O.M. Scokaert, J.B. Rawlings, and E.S. Meadows. Discrete-time stability with perturbations: Application to model predictive control. *Automatica*, 33(3):463–470, 1997.

[SS90] E.D. Sontag and H.J. Sussman. Nonlinear output feedback design for linear systems with saturating controls. In *Proc. 29th Conf. Decision and Control*, pages 3414–3416, 1990.

[SSF84] J.M. Martín Sánchez, S.L. Shah, and D.G. Fisher. A stable adaptive predictive control system. *Int. J. Contr.*, 39:215–234, 1984.

[SSY94] H.J. Sussmann, E.D. Sontag, and Y. Yang. A general result on the stabilization of linear systems using bounded controls. *IEEE Trans. Automat. Control*, 39:2411–2424, 1994.

[Ste89] G. Stein. Respect the unstable – bode lecture for being recipient of the 1989 bode prize of the ieee control systems society. In *Proc. 28th IEEE Conf. on Decision and Control*, Tampa, FL, 1989.

[TBMR91] T.J. Tarn, A.K. Bejczy, G.T. Marth, and A.K. Ramadorai. Kinematic characterization of the puma 560 manipulator. Technical Report Robotics Laboratory Rep. SSM-RL-91-15, Dept. of Systems Science and Mathematics, Washington Univ., St. Louis, MO, 1991.

[TBMR93] T.J. Tarn, A.K. Bejczy, G.T. Marth, and A.K. Ramadorai. Performance comparison of four manipulator servo schemes. *IEEE Control Systems Magazine*, February:22–29, 1993.

[TBY85] T.J. Tarn, A.K. Bejczy, and Xiaoping Yun. Dynamic equations for puma 560 robot arm. Technical Report Robotics Laboratory Rep. SSM-RL-85-02, Dept. of Systems Science and Mathematics, Washington Univ., St. Louis, MO, 1985.

[TC88] T.T.C. Tsang and D.W. Clarke. Generalised predictive control with input constraints. *Proc. IEE - part D*, 135(6):451–460, 1988.

[TJ89] K. L. Teo and L. S. Jennings. Nonlinear optimal control problems with continuous state inequality constraints. *Journal of Optimization Theory and Applications*, 63(1), 1989.

[TXB96] T.J. Tarn, N. Xi, and A.K. Bejczy. Path-based approach to integrated planning and control for robotic systems. *Automatica*, 32(12):1675, December 1996.

[TXGB95] T.J. Tarn, N. Xi, C. Guo, and A.K. Bejczy. Human/machine sharing control for telerobotic systems. *Intelligent Robots and Systems*, pages 373–380, 1995.

[VZ96] A. Vicino and G. Zappa. Sequential approximation of feasible parameter sets for identification with set membership uncertainty. *IEEE Trans. Autom. Control*, 41:774–785, 1996.

[YP93] T.H. Yang and E. Polak. Moving horizon control of nonlinear systems with input saturation, disturbances and plant uncertainty. *Int. Journal of Control*, 58:875–903, 1993.

[ZBM95] A. Zheng, V. Balakrishnan, and M. Morari. Constrained stabilization of discrete-time systems. *Int. J. Robust and Nonlinear Control*, 5(5):461–485, 1995.

[ZFAW93] M. Zohdy, M.S. Fadali, and A. Abdel-Wahab. Robust control system design with state constraints. *Int. J. Systems Sci.*, 24(1):193–202, 1993.

[Zhe95] Z. Q. Zheng. *Robust Control of Systems Subject to Constraints*. Ph.D. dissertation, California Institute of Technology, Pasadena, CA, U.S.A., 1995.

[ZM93] A. Zheng and M. Morari. Robust stability of constrained model predictive control. In *Proc. American Control Conf.*, volume 1, pages 379–383, San Francisco, CA, 1993.

[ZM95] A. Zheng and M. Morari. Stability of model predictive control with mixed constraints. *IEEE Trans. Automat. Control*, 40:1818–1823, 1995.