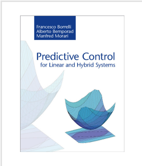


Predictive Control: for Linear and Hybrid Systems
ISBN 9781139061759
IV Constrained Optimal Control of Linear Systems
12 Receding Horizon Control

12
Receding Horizon Control

In this chapter we review the basics of Receding Horizon Control (RHC). In the first part we discuss the stability and the feasibility of RHC and we provide guidelines for choosing the terminal weight so that closed-loop stability is achieved.

The second part of the chapter focuses on the RHC implementation. Since RHC requires at each sampling time to solve an open-loop constrained finite time optimal control problem as a function of the current state, the results of the previous chapters imply two possible approaches for RHC implementation.

In the first approach a mathematical program is solved at each time step for the current initial state. In the second approach the explicit piecewise affine feedback policy (that provides the optimal control for all states) is precomputed off-line. This reduces the on-line computation of the RHC law to a function evaluation, thus avoiding the on-line solution of a quadratic or linear program. This technique is attractive for a wide range of practical problems where the computational complexity of on-line optimization is prohibitive. It also provides insight into the structure underlying optimization-based controllers, describing the behavior of the RHC controller in different regions of the state space. Moreover, for applications where safety is crucial, the correctness of a piecewise affine control law is easier to verify than that of a mathematical program solver.

12.1 RHC Idea

In the previous chapter we discussed the solution of constrained finite time and infinite time optimal control problems for linear systems. An infinite horizon suboptimal controller can be designed by repeatedly solving finite time optimal control problems in a receding horizon fashion as described next. At each sampling time, starting at the current state, an open-loop optimal control problem is solved over a finite horizon (top diagram in Figure 12.1). The computed optimal manipulated input signal is applied to the process only during the following sampling interval $[t, t+1]$. At the next time step $t+1$ a new optimal control problem based on new measurements of the state is solved over a shifted horizon (bottom diagram in Figure 12.1). The resulting controller is referred to as a Receding Horizon Controller (RHC). A receding horizon controller where the finite time optimal control law is computed by solving an optimization problem on-line is usually referred to as *Model Predictive Control* (MPC).

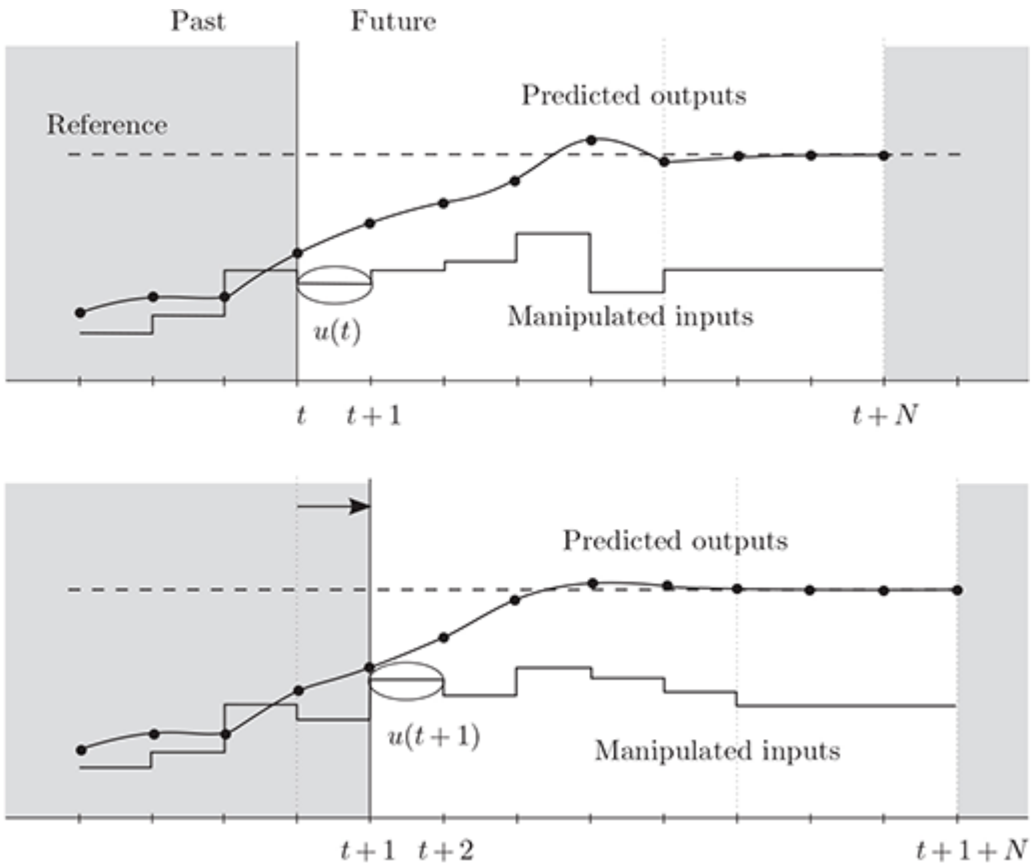


Figure 12.1 Receding Horizon Idea.

12.2 RHC Implementation

Consider the problem of regulating to the origin the discrete-time linear time-invariant system

$$x(t+1) = Ax(t) + Bu(t), \tag{12.1}$$

where $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^m$ are the state and input vectors, respectively, subject to the constraints

$$x(t) \in \mathcal{X},\ u(t) \in \mathcal{U},\ \forall t \geq 0, \tag{12.2}$$

where the sets $\mathcal{X} \subseteq \mathbb{R}^n$ and $\mathcal{U} \subseteq \mathbb{R}^m$ are polyhedra. Receding Horizon Control (RHC) approaches such a constrained regulation problem in the following way. Assume that a full measurement or estimate of the state $x(t)$ is available at the current time t. Then the finite time optimal control problem

$$
\begin{aligned}
J_t^*(x(t)) = \min_{U_{t\to t+N|t}} \quad & J_t(x(t), U_{t\to t+N|t}) = p(x_{t+N|t}) + \sum_{k=0}^{N-1} q\left(x_{t+k|t}, u_{t+k|t}\right) \\
\text{subj. to} \quad & x_{t+k+1|t} = Ax_{t+k|t} + Bu_{t+k|t},\ k = 0, \dots, N-1 \\
& x_{t+k|t} \in \mathcal{X},\ u_{t+k|t} \in \mathcal{U},\ k = 0, \dots, N-1 \\
& x_{t+N|t} \in \mathcal{X}_f \\
& x_{t|t} = x(t)
\end{aligned}
\tag{12.3}
$$

is solved at time t, where $U_{t\to t+N/t} = \{u_{t/t}, \dots, u_{t+N-1/t}\}$ and where $x_{t+k/t}$ denotes the state vector at time $t + k$ predicted at time t obtained by starting from the current state $x_{t/t} = x(t)$ and applying to the system model

$$x_{t+k+1|t} = Ax_{t+k|t} + Bu_{t+k|t}$$

the input sequence $u_{t/t}, \dots, u_{t+k-1/t}$. Often the symbol $x_{t+k/t}$ is read as "the state x at time $t+k$ predicted at time t." Similarly, $u_{t+k/t}$ is read as "the input u at time $t+k$ computed at time t." For instance, $x_{3/1}$ represents the predicted state at time 3 when the prediction is done at time $t = 1$ starting from the current state $x(1)$. It is different, in general, from $x_{3/2}$ which is the predicted state at time 3 when the prediction is done at time $t = 2$ starting from the current state $x(2)$.

Let $U^*_{t\to t+N|t} = \{u^*_{t|t}, \dots, u^*_{t+N-1|t}\}$ be the optimal solution of (12.3) at time t and $J_t^*(x(t))$ the corresponding value function. Then, the first element of $U^*_{t\to t+N|t}$ is applied to system (12.1)

$$u(t) = u^*_{t|t}(x(t)). \tag{12.4}$$

The optimization problem (12.3) is repeated at time $t + 1$, based on the new state $x_{t+1/t+1} = x(t + 1)$, yielding a *moving* or *receding horizon* control strategy.

Let $f_t : \mathbb{R}^n \to \mathbb{R}^m$ denote the *receding horizon* control law that associates the optimal input $u^*_{t|t}$ to the current state $x(t)$, $f_t(x(t)) = u^*_{t|t}(x(t))$. Then the closed-loop system obtained by controlling (12.1) with the RHC (12.3)–(12.4) is

$$x(k+1) = Ax(k) + Bf_k(x(k)) = f_{cl}(x(k), k),\ k \geq 0. \tag{12.5}$$

Note that the notation used in this chapter is slightly different from the one used in Chapter 11. Because of the receding horizon strategy, there is the need to distinguish between the input $u^*(t + k)$ applied to the plant at time $t + k$, and optimizer $u^*_{t+k|t}$ of the problem (12.3) at time $t + k$ obtained by solving (12.3) at time t with $x_{t/t} = x(t)$.

Consider problem (12.3). As the system, the constraints and the cost function are time-invariant, the solution to problem (12.3) is a time-invariant function of the initial state $x(t)$. Therefore, in order to simplify the notation, we can set $t = 0$ in (12.3) and remove the term "|0" since it is now redundant and rewrite (12.3) as

$$
\begin{aligned}
J_0^*(x(t)) = \min_{U_0} \quad & J_0(x(t), U_0) = p(x_N) + \sum_{k=0}^{N-1} q(x_k, u_k) \\
\text{subj. to} \quad & x_{k+1} = Ax_k + Bu_k, \ k = 0, \dots, N-1 \\
& x_k \in \mathcal{X}, \ u_k \in \mathcal{U}, \ k = 0, \dots, N-1 \\
& x_N \in \mathcal{X}_f \\
& x_0 = x(t),
\end{aligned}
\tag{12.6}
$$

where $U_0 = \{u_0, \dots, u_{N-1}\}$ and the notation in Remark 7.1 applies. Similarly as in previous chapters, we will focus on two classes of cost functions. If the 1-norm or ∞-norm is used in (12.6), then we set $p(x_N) = ||Px_N||_p$ and $q(x_k, u_k) = ||Qx_k||_p + ||Ru_k||_p$ with $p = 1$ or $p = \infty$ and P, Q, R full column rank matrices. The cost function is rewritten as

$$
J_0(x(0), U_0) = \|Px_N\|_p + \sum_{k=0}^{N-1} \|Qx_k\|_p + \|Ru_k\|_p. \tag{12.7}
$$

If the squared Euclidian norm is used in (12.6), then we set $p(x_N) = x_N' P x_N$ and $q(x_k, u_k) = x_k' Q x_k + u_k' R u_k$ with $P \succeq 0, \ Q \succeq 0$ and $R \succ 0$. The cost function is rewritten as

$$
J_0(x(0), U_0) = x_N' P x_N + \sum_{k=0}^{N-1} x_k' Q x_k + u_k' R u_k. \tag{12.8}
$$

The control law (12.4)

$$
u(t) = f_0(x(t)) = u_0^*(x(t)) \tag{12.9}
$$

and closed-loop system (12.5)

$$
x(k+1) = Ax(k) + Bf_0(x(k)) = f_{cl}(x(k)), \ k \geq 0 \tag{12.10}
$$

are time-invariant as well.

Note that the notation in (12.6) does not allow us to distinguish at which time step a certain state prediction or optimizer is computed and is valid for time-invariant problems only. Nevertheless, we will prefer the RHC notation in (12.6) to the one in (12.3) in order to simplify the exposition.

Compare problem (12.6) and the CFTOC (11.9). The *only* difference is that problem (12.6) is solved for $x_0 = x(t)$, $t \geq 0$ rather than for $x_0 = x(0)$. For this reason we can make use of all the results of the previous chapter. In particular, $\mathcal{X}_0$ denotes the set of feasible states $x(t)$ for problem (12.6) as defined and studied in Section 11.2. Recall from Section 11.2 that $\mathcal{X}_0$ is a polyhedron.

From the above explanations it is clear that a fixed prediction horizon is shifted or *recedes* over time, hence its name, receding horizon control. The procedure of this *on-line* optimal control technique is summarized in the following algorithm.

Algorithm 12.1 *On-line receding horizon control*

Input: State $x(t)$ at time instant t
Output: Receding horizon control input $u(x(t))$
  Obtain $U_0^*(x(t))$ by solving the optimization problem (12.6)
  If 'problem infeasible' **Then** stop
  Return the first element u_0^* of U_0^*

Example 12.1 Consider the double integrator system (11.23) rewritten below:

$$x(t+1) = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t) \tag{12.11}$$

The aim is to compute the receding horizon controller that solves the optimization problem (12.6) with $p(x_N) = x_N' P x_N,\ q(x_k, u_k) = x_k' Q x_k + u_k' R u_k\ \ N = 3, P = Q = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},\ R = 10,\ \mathcal{X}_f = \mathbb{R}^2$ subject to the input constraints

$$-0.5 \le u(k) \le 0.5,\ k = 0, \dots, 3 \tag{12.12}$$

and the state constraints

$$\begin{bmatrix} -5 \\ -5 \end{bmatrix} \le x(k) \le \begin{bmatrix} 5 \\ 5 \end{bmatrix},\ k = 0, \dots, 3. \tag{12.13}$$

The QP problem associated with the RHC has the form (11.31) with

$$H = \begin{bmatrix} 13.50 & -10.00 & -0.50 \\ -10.00 & 22.00 & -10.00 \\ -0.50 & -10.00 & 31.50 \end{bmatrix},\ F = \begin{bmatrix} -10.50 & 10.00 & -0.50 \\ -20.50 & 10.00 & 9.50 \end{bmatrix},\ Y = \begin{bmatrix} 14.50 & 23.50 \\ 23.50 & 54.50 \end{bmatrix} \tag{12.14}$$

and

$$G_0 = \begin{bmatrix} 0.50 & -1.00 & 0.50 \\ -0.50 & 1.00 & -0.50 \\ -0.50 & 0.00 & 0.50 \\ -0.50 & 0.00 & -0.50 \\ 0.50 & 0.00 & -0.50 \\ 0.50 & 0.00 & 0.50 \\ -1.00 & 0.00 & 0.00 \\ 0.00 & -1.00 & 0.00 \\ 1.00 & 0.00 & 0.00 \\ 0.00 & 1.00 & 0.00 \\ 0.00 & 0.00 & -1.00 \\ 0.00 & 0.00 & 1.00 \\ 0.00 & 0.00 & 0.00 \\ -0.50 & 0.00 & 0.50 \\ 0.00 & 0.00 & 0.00 \\ 0.50 & 0.00 & -0.50 \\ -0.50 & 0.00 & 0.50 \\ 0.50 & 0.00 & -0.50 \\ 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 \end{bmatrix},\ E_0 = \begin{bmatrix} 0.50 & 0.50 \\ -0.50 & -0.50 \\ 0.50 & 0.50 \\ -0.50 & -0.50 \\ -0.50 & -0.50 \\ 0.50 & 0.50 \\ 0.00 & 0.00 \\ 0.00 & 0.00 \\ 0.00 & 0.00 \\ 0.00 & 0.00 \\ 0.00 & 0.00 \\ 0.00 & 0.00 \\ 1.00 & 1.00 \\ -0.50 & -0.50 \\ -1.00 & -1.00 \\ 0.50 & 0.50 \\ -0.50 & -1.50 \\ 0.50 & 1.50 \\ 1.00 & 0.00 \\ 0.00 & 1.00 \\ -1.00 & 0.00 \\ 0.00 & -1.00 \end{bmatrix},\ w_0 = \begin{bmatrix} 0.50 \\ 0.50 \\ 5.00 \\ 5.00 \\ 5.00 \\ 5.00 \\ 5.00 \\ 5.00 \\ 5.00 \\ 5.00 \\ 0.50 \\ 0.50 \\ 5.00 \\ 5.00 \\ 5.00 \\ 5.00 \\ 0.50 \\ 0.50 \\ 5.00 \\ 5.00 \\ 5.00 \\ 5.00 \end{bmatrix} \tag{12.15}$$

The RHC (12.6)–(12.9) algorithm becomes Algorithm 12.2. We refer to the solution U_0^* of the QP (11.31) as $[U_0^*, \mathrm{Flag}] = \mathrm{QP}(H, 2F'x(0), G_0, w_0 + E_0 x(0))$ where "Flag" indicates if the QP was found to be feasible or not.

Algorithm 12.2 *QP-based on-line receding horizon control*

Input: State $x(t)$ at time instant t
Output: Receding horizon control input $u(x(t))$
Compute $\tilde{F} = 2F'x(t)$ and $\tilde{w}_0 = w_0 + E_0 x(t)$
Obtain $U_0^*(x(t))$ by solving the optimization problem $[U_0^*, \text{Flag}] = \text{QP}(H, \tilde{F}, G_0, \tilde{w}_0)$
If Flag = infeasible **Then** stop
Return the first element u_0^* of U_0^*

Figure 12.2 shows two closed-loop trajectories starting at state $x(0) = [-4.5, 2]$ and $x(0) = [-4.5, 3]$. The trajectory starting from $x(0) = [-4.5, 2]$ converges to the origin and satisfies input and state constraints. The trajectory starting from $x(0) = [-4.5, 3]$ stops at $x(2) = [1, 2]$ because of infeasibility. At each time step, the open-loop predictions are depicted with dashed lines. This shows that the closed-loop trajectories are different from the open-loop predicted trajectories because of the receding horizon nature of the controller.

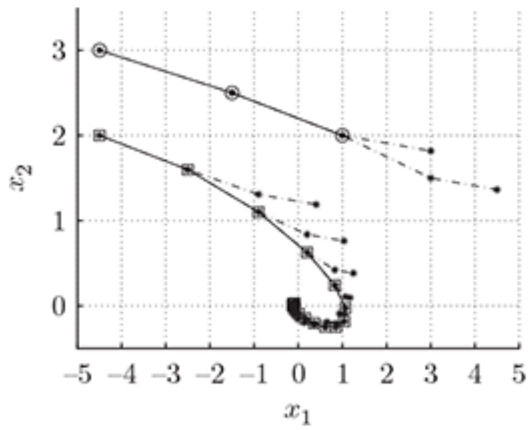
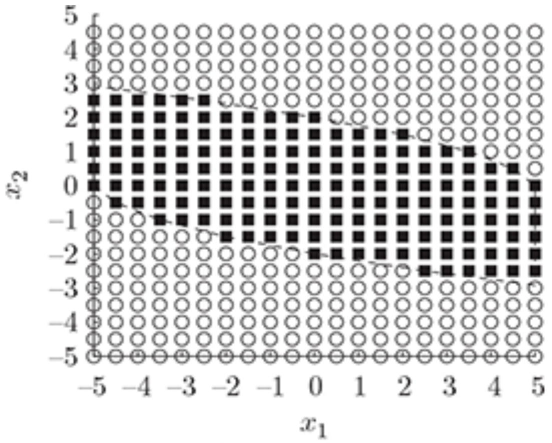
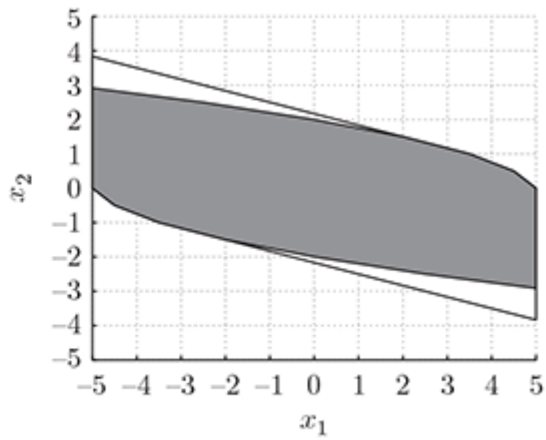


Figure 12.2 Example 12.1. Closed-loop trajectories realized (solid) and predicted (dashed) for two initial states x(0)=[–4.5,2] (*boxes*) and x(0)=[–4.5,3] (*circles*).

In Figure 12.3(a) the feasible state space was gridded and each point of the grid was marked with a square if the RHC law (12.6)–(12.9) generates feasible closed-loop trajectories and with a circle if it does not. The set of all initial conditions generating feasible closed-loop trajectories is the maximal positive invariant set $\mathcal{O}_\infty$ of the autonomous system (12.10). We remark that this set is different from the set $\mathcal{X}_0$ of feasible initial conditions for the QP problem (11.31) with matrices (12.15). Both sets $\mathcal{O}_\infty$ and $\mathcal{X}_0$ are depicted in Figure 12.3(b). The computation of f_0 is discussed later in this chapter. Because of the nonlinear nature of f_0, the computation of $\mathcal{O}_\infty$ for the system (12.10) is not an easy task. Therefore, we will show how to choose a terminal invariant set $\mathcal{X}_f$ such that $\mathcal{O}_\infty = \mathcal{X}_0$ is guaranteed automatically.



(a) *Boxes* (*Circles*) are initial points leading (not leading) to feasible closed-loop trajectories.



(b) Maximal positive invariant set $\mathcal{O}_\infty$ (gray) and set of initial feasible states $\mathcal{X}_0$ (white and gray).

Figure 12.3 Example 12.1. Double integrator with RHC.

Note that a feasible closed-loop trajectory does not necessarily converge to the origin. Feasibility, convergence and stability of RHC are discussed in detail in the next sections. Before that we want to illustrate these issues through another example.

Example 12.2 Consider the unstable system

$$x(t+1) = \begin{bmatrix} 2 & 1 \\ 0 & 0.5 \end{bmatrix} x(t) + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u(t) \tag{12.16}$$

with the input constraints

$$-1 \le u(k) \le 1, \; k = 0, \ldots, N-1 \tag{12.17}$$

and the state constraints

$$\begin{bmatrix} -10 \\ -10 \end{bmatrix} \le x(k) \le \begin{bmatrix} 10 \\ 10 \end{bmatrix}, \; k = 0, \ldots, N-1. \tag{12.18}$$

In the following, we study the receding horizon control problem (12.6) with $p(x_N) = x_N' P x_N$, $q(x_k, u_k) = x_k' Q x_k + u_k' R u_k$ for different horizons N and weights R. We set $Q = I$ and omit both the terminal set constraint and the terminal weight, i.e., $\mathcal{X}_f = \mathbb{R}^2$, $P = 0$.

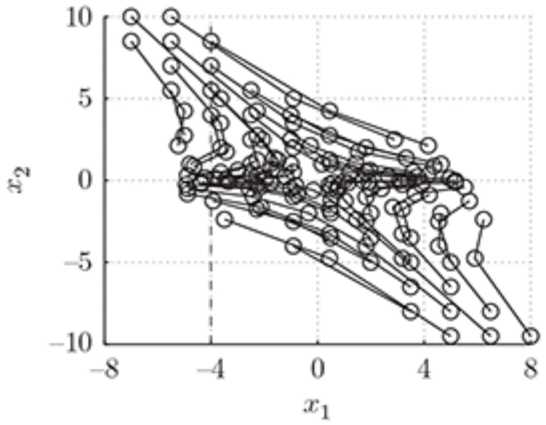
Figure 12.4 shows closed-loop trajectories for receding horizon control loops that were obtained with the following parameter settings

Setting 1: $N = 2$, $R = 10$

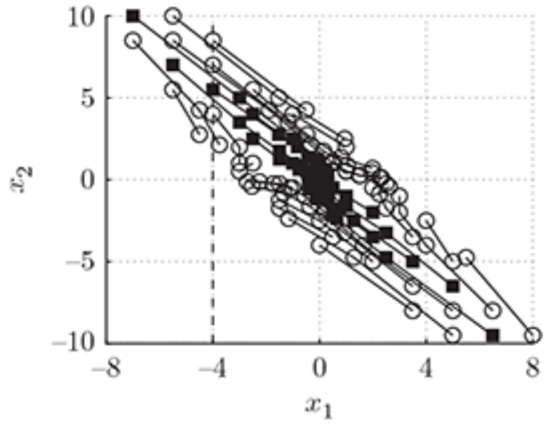
Setting 2: $N = 3$, $R = 2$

Setting 3: $N = 4$, $R = 1$

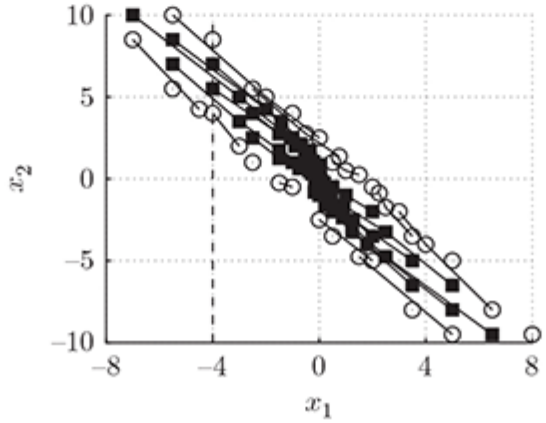
For *Setting 1* (Figure 12.4(a)) there is evidently no initial state that can be steered to the origin. Indeed, it turns out, that *all* nonzero initial states $x(0) \in \mathbb{R}^2$ diverge from the origin and eventually become infeasible. In contrast, *Setting 2* leads to a receding horizon controller, that manages to get some of the initial states converge to the origin, as seen in Figure 12.4(b). Finally, Figure 12.4(c) shows that *Setting 3* can expand the set of those initial states that can be brought to the origin.



(a) *Setting 1*: $N = 2, R = 10$.



(b) *Setting 2*: $N = 3, R = 2$.



(c) *Setting 3*: $N = 4, R = 1$.

Figure 12.4 Example 12.2. Closed-loop trajectories for different settings of horizon *N* and weight *R*. *Boxes* (*Circles*) are initial points leading (not leading) to feasible closed-loop trajectories.

Note the behavior of particular initial states:

1. Closed-loop trajectories starting at state $x(0) = [–4, 7]$ behave differently depending on the chosen setting. Both *Setting 1* and *Setting 2* cannot bring this state to the origin, but the controller with *Setting 3* succeeds.
2. There are initial states, e.g., $x(0) = [–4, 8.5]$, that always lead to infeasible trajectories independent of the chosen settings. It turns out, that *no* setting can be found that brings those states to the origin.

These results illustrate that the choice of parameters for receding horizon control influences the behavior of the resulting closed-loop trajectories in a complex manner. A better understanding of the effect of parameter changes can be gained from an inspection of maximal positive invariant sets $\mathcal{O}_\infty$ for the different settings, and the maximal control invariant set $\mathcal{C}_\infty$ as depicted in Figure 12.5.

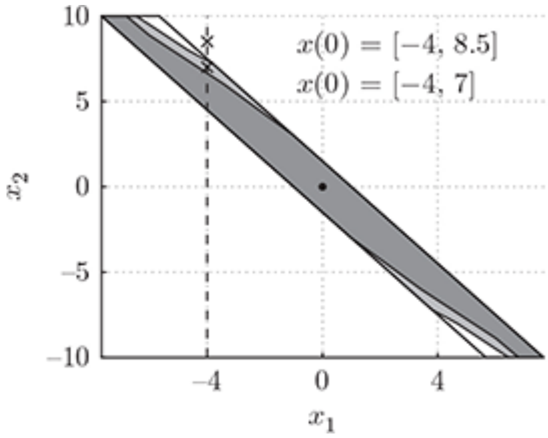


Figure 12.5 Example 12.2. Maximal positive invariant sets $\mathcal{O}_\infty$ for different parameter settings: *Setting 1* (origin), *Setting 2* (dark-gray) and *Setting 3* (gray and dark-gray). Also depicted is the maximal control invariant set $\mathcal{C}_\infty$ (white and gray and dark-gray).

The maximal positive invariant set stemming from *Setting 1* only contains the origin ($\mathcal{O}_\infty = \{0\}$) which explains why all nonzero initial states diverge from the origin. For *Setting 2* the maximal positive invariant set has grown considerably, but does not contain the initial state $x(0) = [-4, 7]$, thus leading to infeasibility eventually. *Setting 3* leads to a maximal positive invariant set that contains this state and thus keeps the closed-loop trajectory inside this set for all future time steps.

From Figure 12.5 we also see that a trajectory starting at $x(0) = [-4, 8.5]$ cannot be kept inside any bounded set by *any* setting (indeed, by any controller) since it is outside the maximal control invariant set $\mathcal{C}_\infty$.

12.3 RHC Main Issues

If we solve the receding horizon problem for the special case of an infinite horizon (setting $N = \infty$ in (12.6) as we did for LQR in Section 8.5 and CLQR in Section 11.3.4) then it is almost immediate that the closed-loop system with this controller has some nice properties. Most importantly, the differences between the open-loop predicted and the actual closed-loop trajectories observed in Example 12.1 disappear. As a consequence, if the optimization problem is feasible, then the closed-loop trajectories will be feasible for all times. If the optimization problem has a finite solution, then in closed-loop the states and inputs will converge to the origin asymptotically.

In RHC, when we solve the optimization problem over a finite horizon repeatedly at each time step, we hope that the controller resulting from this “short-sighted” strategy will lead to a closed-loop behavior that mimics that of the infinite horizon controller. The examples in the last section indicated that at least two problems may occur. First of all, the controller may lead us into a situation where after a few steps the finite horizon optimal control problem that we need to solve at each time step is infeasible, i.e., that there does not exist a sequence of control inputs for which the constraints are obeyed. Second, even if the feasibility problem does not occur, the generated control inputs may not lead to trajectories that converge to the origin, i.e., that the closed-loop system is asymptotically stable.

In general, stability and feasibility are not ensured by the RHC law (12.6)– (12.9). In principle, we could analyze the RHC law for feasibility, stability and convergence but this is difficult as the examples in the last section illustrated. Therefore, conditions will be derived on how the terminal weight P and the terminal constraint set $\mathcal{X}_f$ should be chosen such that closed-loop stability and feasibility are ensured.

12.3.1 Feasibility of RHC

The examples in the last section illustrate that feasibility at the initial time $x(0) \in \mathcal{X}_0$ does not necessarily imply feasibility for all future times. It is desirable to design an RHC such that feasibility for all future times is guaranteed, a property we refer to as *persistent feasibility*.

We would like to gain some insight when persistent feasibility occurs and how it is affected by the formulation of the control problem and the choice of the controller parameters. Let us first recall the various sets introduced in Section 10.2 and Section 11.2 and how they are influenced.

- $\mathcal{C}_\infty$: The maximal control invariant set $\mathcal{C}_\infty$ is only affected by the sets $\mathcal{X}$ and $\mathcal{U}$, the constraints on states and inputs. It is the largest set over which we can expect *any* controller to work.
- $\mathcal{X}_0$: A control input U_0 can only be found, i.e., the control problem is feasible, if $x(0) \in \mathcal{X}_0$. The set $\mathcal{X}_0$ depends on $\mathcal{X}$ and $\mathcal{U}$, on the controller horizon N and on the controller terminal set $\mathcal{X}_f$. It does not depend on the objective function and it has generally no relation with $\mathcal{C}_\infty$ (it can be larger, smaller, etc.).
- $\mathcal{O}_\infty$: The maximal positive invariant set for the closed-loop system depends on the controller and as such on all parameters affecting the controller, i.e., $\mathcal{X}$, $\mathcal{U}$, N, $\mathcal{X}_f$ and the objective function with its parameters P, Q and R. Clearly $\mathcal{O}_\infty \subseteq \mathcal{X}_0$ because if it were not there would be points in $\mathcal{O}_\infty$ for which the control problem is not feasible. Because of invariance, the closed-loop is persistently feasible for all states $x(0) \in \mathcal{O}_\infty$. Clearly, $\mathcal{O}_\infty \subseteq \mathcal{C}_\infty$.

We can now state necessary and sufficient conditions guaranteeing persistent feasibility by means of invariant set theory.

Lemma 12.1 *Let $\mathcal{O}_\infty$ be the maximal positive invariant set for the closed-loop system $x(k+1) = f_{cl}(x(k))$ in (12.10) with constraints (12.2). The RHC problem is persistently feasible if and only if $\mathcal{X}_0 = \mathcal{O}_\infty$.*

Proof: For the RHC problem to be persistently feasible $\mathcal{X}_0$ must be positive invariant for the closed-loop system. We argued above that $\mathcal{O}_\infty \subseteq \mathcal{X}_0$. As the positive invariant set $\mathcal{X}_0$ cannot be larger than the maximal positive invariant set $\mathcal{O}_\infty$, it follows that $\mathcal{X}_0 = \mathcal{O}_\infty$. ■

As $\mathcal{X}_0$ does not depend on the controller parameters P, Q and R but $\mathcal{O}_\infty$ does, the requirement $\mathcal{X}_0 = \mathcal{O}_\infty$ for persistent feasibility shows that, in general, only some P, Q and R are allowed. The parameters P, Q and R affect the performance. The complex effect they have on persistent feasibility makes their choice extremely difficult for the design engineer. In the following we will remedy this undesirable situation. We will make use of the following important *sufficient* condition for persistent feasibility.

Lemma 12.2 *Consider the RHC law (12.6)–(12.9) with $N \geq 1$. If $\mathcal{X}_1$ is a control invariant set for system (12.1)–(12.2) then the RHC is persistently feasible. Also, $\mathcal{O}_\infty = \mathcal{X}_0$ is independent of P, Q and R.*

Proof: If $\mathcal{X}_1$ is control invariant then, by definition, $\mathcal{X}_1 \subseteq \text{Pre}(\mathcal{X}_1)$. Also recall that $\text{Pre}(\mathcal{X}_1) = \mathcal{X}_0$ from the properties of the feasible sets in equation (11.20) (note that $\text{Pre}(\mathcal{X}_1) \cap \mathcal{X} = \text{Pre}(\mathcal{X}_1)$ from control invariance). Pick some $x \in \mathcal{X}_0$ and some feasible control u for that x and define $x^+ = Ax + Bu \in \mathcal{X}_1$. Then $x^+ \in \mathcal{X}_1 \subseteq \text{Pre}(\mathcal{X}_1) = \mathcal{X}_0$. As u was arbitrary (as long as it is feasible) $x^+ \in \mathcal{X}_0$ for all feasible u. As $\mathcal{X}_0$ is positive invariant, $\mathcal{X}_0 = \mathcal{O}_\infty$ from Lemma 12.1. As $\mathcal{X}_0$ is positive invariant for all feasible u, $\mathcal{O}_\infty$ does not depend on P, Q and R. ■

Note that in the proof of Lemma 12.2, persistent feasibility does not depend on the input u as long as it is feasible. For this reason, sometimes in the literature this property is referred to "persistently feasible for all feasible u."

We can use Lemma 12.2 in the following manner. For $N = 1$, $\mathcal{X}_1 = \mathcal{X}_f$. If we choose the terminal set to be control invariant then $\mathcal{X}_0 = \mathcal{O}_\infty$ and RHC will be persistently feasible independent of chosen control objectives and parameters. Thus the designer can choose the parameters to affect performance without affecting persistent feasibility. A control horizon of $N = 1$ is often too restrictive, but we can easily extend Lemma 12.2.

Theorem 12.1 *Consider the RHC law (12.6)–(12.9) with $N \geq 1$. If $\mathcal{X}_f$ is a control invariant set for system (12.1)–(12.2) then the RHC is persistently feasible.*

Proof: If $\mathcal{X}_f$ is control invariant, then $\mathcal{X}_{N-1}, \mathcal{X}_{N-2}, \ldots, \mathcal{X}_1$ are control invariant and Lemma 12.2 establishes persistent feasibility for all feasible u. ■

Corollary 12.1 *Consider the RHC law (12.6)–(12.9) with $N \geq 1$. If there exists $i \in [1, N]$ such that $\mathcal{X}_i$ is a control invariant set for system (12.1)–(12.2), then the RHC is persistently feasible for all cost functions.*

Proof: Follows directly from the proof of Theorem 12.1. ■

Recall that Theorem 11.2 together with Remark 11.4 define the properties of the set $\mathcal{X}_0$ as N varies. Therefore, Theorem 12.1 and Corollary 12.1 provide also guidelines on the choice of the horizon N for guaranteeing persistent feasibility for all feasible u. For instance, if the RHC problem (12.6) for $N = \overline{N}$ yields a control invariant set $\mathcal{X}_0$, then from Theorem 11.2 the RHC law (12.6)–(12.9) with $N = \overline{N} + 1$ is persistently feasible for all feasible u. Moreover, from Corollary 12.1 the RHC law (12.6)–(12.9) with $N \geq \overline{N} + 1$ is persistently feasible for all feasible u.

Corollary 12.2 *Consider the RHC problem (12.6)–(12.9). If N is greater than the determinedness index $\overline{N}$ of $\mathcal{K}_\infty(\mathcal{X}_f)$ for system (12.1)–(12.2), then the RHC is persistently feasible.*

Proof: The feasible set $\mathcal{X}_i$ for $i = 1, \ldots, N-1$ is equal to the $(N-i)$-step controllable set $\mathcal{X}_i = \mathcal{K}_{N-i}(\mathcal{X}_f)$. If the maximal controllable set is finitely determined then $\mathcal{X}_i = \mathcal{K}_\infty(\mathcal{X}_f)$ for $i \leq N - \overline{N}$. Note that $\mathcal{K}_\infty(\mathcal{X}_f)$ is control invariant. Then persistent feasibility for all feasible u follows from Corollary 12.1. ■

Persistent feasibility does not guarantee that the closed-loop trajectories converge towards the desired equilibrium point. From Theorem 12.1 it is clear that one can only guarantee that $x(k) \in \mathcal{X}_1$ for all $k > 0$ if $x(0) \in \mathcal{X}_0$.

One of the most popular approaches to guarantee persistent feasibility and stability of the RHC law (12.6)–(12.9) makes use of a control invariant terminal set $\mathcal{X}_f$ and a terminal cost P which drives the closed-loop optimal trajectories towards the origin. A detailed discussion follows in the next section.

12.3.2 Stability of RHC

In this section we will derive the main stability result for RHC. Our objective is to find a Lyapunov function for the closed-loop system. We will show next that if the terminal cost and constraint are appropriately chosen, then the value function $J_0^*(\cdot)$ is a Lyapunov function.

Theorem 12.2 *Consider system (12.1)–(12.2), the RHC law (12.6)–(12.9) and the closed-loop system (12.10). Assume that*

- *(A0) The stage cost $q(x, u)$ and terminal cost $p(x)$ are continuous and positive definite functions.*
- *(A1) The sets $\mathcal{X}$, $\mathcal{X}_f$ and $\mathcal{U}$ contain the origin in their interior and are closed.*
- *(A2) $\mathcal{X}_f$ is control invariant, $\mathcal{X}_f \subseteq \mathcal{X}$.*
- *(A3)* $\min_{v \in \mathcal{U},\ Ax+Bv \in \mathcal{X}_f} (-p(x) + q(x, v) + p(Ax + Bv)) \leq 0,\ \forall x \in \mathcal{X}_f.$

Then, the origin of the closed-loop system (12.10) is asymptotically stable with domain of attraction $\mathcal{X}_0$.

Proof: From hypothesis (A2), Theorem 12.1 and Lemma 12.1, we conclude that $\mathcal{X}_0 = \mathcal{O}_\infty$ is a positive invariant set for the closed-loop system (12.10) for any choice of the cost function. Thus persistent feasibility for any feasible input is guaranteed in $\mathcal{X}_0$.

Next, we prove convergence and stability. We establish that the function $J_0^*(\cdot)$ in (12.6) is a Lyapunov function for the closed-loop system. Because the cost J_0, the system and the constraints are time-invariant we can study the properties of J_0^* between step $k = 0$ and step $k + 1 = 1$.

Consider problem (12.6) at time $t = 0$. Let $x(0) \in \mathcal{X}_0$ and let $U_0^* = \{u_0^*, \ldots, u_{N-1}^*\}$ be the optimizer of problem (12.6) and $\mathbf{x}_0 = \{x(0), x_1, \ldots, x_N\}$ be the corresponding optimal state trajectory. After the implementation of u_0^* we obtain $x(1) = x_1 = Ax(0) + Bu_0^*$. Consider now problem (12.6) for $t = 1$. We will construct an upper bound on $J_0^*(x(1))$. Consider the sequence $\tilde{U}_1 = \{u_1^*, \ldots, u_{N-1}^*, v\}$ and the corresponding state trajectory resulting from the initial state $x(1)$, $\tilde{\mathbf{x}}_1 = \{x_1, \ldots, x_N, Ax_N + Bv\}$. Because $x_N \in \mathcal{X}_f$ and (A2) there exists a feasible v such that $x_{N+1} = Ax_N + Bv \in \mathcal{X}_f$ and with this v the sequence $\tilde{U}_1 = \{u_1^*, \dots, u_{N-1}^*, v\}$ is feasible. Because $\tilde{U}_1$ is not optimal $J_0(x(1), \tilde{U}_1)$ is an upper bound on $J_0^*(x(1))$.

Since the trajectories generated by U_0^* and $\tilde{U}_1$ overlap, except for the first and last sampling intervals, it is immediate to show that

$$J_0^*(x(1)) \le J_0(x(1), \tilde{U}_1) = J_0^*(x(0)) - q(x_0, u_0^*) - p(x_N) + (q(x_N, v) + p(Ax_N + Bv)). \tag{12.19}$$

Let $x = x_0 = x(0)$ and $u = u_0^*$. Under assumption (A3) equation (12.19) becomes

$$J_0^*(Ax + Bu) - J_0^*(x) \le -q(x, u),\ \forall x \in \mathcal{X}_0. \tag{12.20}$$

Equation (12.20) and the hypothesis (A0) on the stage cost $q(\cdot)$ ensure that $J_0^*(x)$ strictly decreases along the state trajectories of the closed-loop system (12.10) for any $x \in \mathcal{X}_0$, $x \neq 0$. In addition to the fact that $J_0^*(x)$ decreases, $J_0^*(x)$ is lower-bounded by zero and since the state trajectories generated by the closed-loop system (12.10) starting from any $x(0) \in \mathcal{X}_0$ lie in $\mathcal{X}_0$ for all $k \ge 0$, equation (12.20) is sufficient to ensure that the state of the closed-loop system converges to zero as $k \to 0$ if the initial state lies in $\mathcal{X}_0$. We have proven (i).

In order to prove stability via Theorem 7.2 we have to establish that $J_0^*(x)$ is a Lyapunov function. Positivity holds by the hypothesis (A0), decrease follows from (12.20). For continuity at the origin we will show that $J_0^*(x) \le p(x),\ \forall x \in \mathcal{X}_f$ and as $p(x)$ is continuous at the origin (by hypothesis (A0)) $J_0^*(x)$ must be continuous as well. From assumption (A2), $\mathcal{X}_f$ is control invariant and thus for any $x \in \mathcal{X}_f$ there exists a feasible input sequence $\{u_0, \dots, u_{N-1}\}$ for problem (12.6) starting from the initial state $x_0 = x$ whose corresponding state trajectory is $\{x_0, x_1, \dots, x_N\}$ stays in $\mathcal{X}_f$, i.e., $x_i \in \mathcal{X}_f\ \forall\ i = 0, \dots, N$. Among all the aforementioned input sequences $\{u_0, \dots, u_{N-1}\}$ we focus on the one where u_i satisfies assumption (A3) for all $i = 0, \dots, N-1$. Such a sequence provides an upper bound on the function J_0^*:

$$J_0^*(x_0) \le \left(\sum_{i=0}^{N-1} q(x_i, u_i)\right) + p(x_N),\ x_i \in \mathcal{X}_f,\ i = 0, \dots, N, \tag{12.21}$$

which can be rewritten as

$$\begin{aligned} J_0^*(x_0) &\le \left(\sum_{i=0}^{N-1} q(x_i, u_i)\right) + p(x_N) \\ &= p(x_0) + \left(\sum_{i=0}^{N-1} q(x_i, u_i) + p(x_{i+1}) - p(x_i)\right)\ x_i \in \mathcal{X}_f,\ i = 0, \dots, N, \end{aligned} \tag{12.22}$$

which from assumption (A3) yields

$$J_0^*(x) \le p(x), \ \forall x \in \mathcal{X}_f. \tag{12.23}$$

In conclusion, there exist a finite time in which any $x \in \mathcal{X}_0$ is steered to a level set of $J_0^*(x)$ contained in $\mathcal{X}_f$ after which convergence to and stability of the origin follows. ■

Remark 12.1 The assumption on the positive definiteness of the stage cost $q(\cdot)$ in Theorem 12.2 can be relaxed as in standard optimal control. For instance, for the 2-norm based cost function (12.8), one can allow $Q \succeq 0$ with $(Q^{\frac{1}{2}}, A)$ observable.

Remark 12.2 The procedure outlined in Theorem 12.2 is, in general, conservative because it requires the introduction of an artificial terminal set $\mathcal{X}_f$ to guarantee persistent feasibility and a terminal cost to guarantee stability. Requiring $x_N \in \mathcal{X}_f$ usually decreases the size of the region of attraction $\mathcal{X}_0 = \mathcal{O}_\infty$. Also the performance may be negatively affected.

Remark 12.3 A function $p(x)$ satisfying assumption (A3) of Theorem 12.2 is often called control Lyapunov function.

The hypothesis (A2) of Theorem 12.2 is required for guaranteeing persistent feasibility as discussed in Section 12.3.1. In some part of the literature the constraint $\mathcal{X}_f$ is not used. However, in this literature the terminal region constraint $\mathcal{X}_f$ is implicit. In fact, it is typically required that the horizon N is sufficiently large to ensure feasibility of the RHC (12.6)–(12.9) at all time instants t. Technically this means that N has to be greater than the determinedness index $\overline{N}$ of system (12.1)–(12.2) which by Corollary 12.2 guarantees persistent feasibility for all inputs. We refer the reader to Section 12.3.1 for more details on feasibility.

Next we will show a few simple choices for P and $\mathcal{X}_f$ satisfying the hypothesis (A2) and (A3) of Theorem 12.2.

Stability, 2-Norm Case

Consider system (12.1)–(12.2), the RHC law (12.6)–(12.9), the cost function (12.8) and the closed-loop system (12.10). A simple choice for $\mathcal{X}_f$ is the maximal positive invariant set (see Section 10.1) for the closed-loop system $x(k+1) = (A+BF_\infty)x(k)$ where F_∞ is the associated unconstrained infinite time optimal controller (8.33). With this choice the assumption (A3) in Theorem 12.2 becomes

$$x'(A'(P - PB(B'PB + R)^{-1}BP)A + Q - P)x \le 0, \ \forall x \in \mathcal{X}_f, \tag{12.24}$$

which is satisfied as an equality if P is chosen as the solution P_∞ of the Algebraic Riccati Equation (8.32) for system (12.1).

In general, instead of F_∞ we can choose any controller F which stabilizes $A + BF$. With $v = Fx$ the assumption (A3) in Theorem 12.2 becomes

$$-P + (Q + F'RF) + (A + BF)'P(A + BF) \leq 0. \tag{12.25}$$

It is satisfied as an equality if we choose P as a solution of the corresponding Lyapunov equation.

We learned in Section 7.5.2 that P satisfying (12.24) or (12.25) as equalities expresses the infinite horizon cost

$$J^*_\infty(x_0) = x_0'Px_0 = \sum_{k=0}^{\infty} x_k'Qx_k + u_k'Ru_k. \tag{12.26}$$

In summary, we conclude that the closed-loop system (12.10) is stable if there exist a controller F which stabilizes the unconstrained system inside the controlled invariant terminal region $\mathcal{X}_f$ and if the infinite horizon cost $x'Px$ incurred with this controller is used in the cost function (12.8). Thus, the two terms in the objective (12.8) reflect the infinite horizon cost, one the initial finite horizon cost when the controller is constrained and the second one the cost for the infinite tail of the trajectory incurred after the system enters $\mathcal{X}_f$ and the controller is unconstrained.

If the open loop system (12.1) is asymptotically stable, then we may even select $F = 0$. Note that depending on the choice of the controller the controlled invariant terminal region $\mathcal{X}_f$ changes.

For any of the discussed choices for F and $\mathcal{X}_f$ stability implies exponential stability. The argument is simple. As the system is closed-loop stable it enters the terminal region in finite time. If $\mathcal{X}_f$ is chosen as suggested, the closed-loop system is unconstrained after entering $\mathcal{X}_f$. For an unconstrained linear system the convergence to the origin is exponential.

Stability, 1-Norm and ∞-Norm Case

Consider system (12.1)–(12.2), the RHC law (12.6)–(12.9), the cost function (12.7) and the closed-loop system (12.10). Let $p = 1$ or $p = \infty$. If system (12.1) is asymptotically stable, then $\mathcal{X}_f$ can be chosen as the positively invariant set of the autonomous system $x(k + 1) = Ax(k)$ subject to the state constraints $x \in \mathcal{X}$. Therefore in $\mathcal{X}_f$ the input $\mathbf{0}$ is feasible and the assumption (A3) in Theorem 12.2 becomes

$$-\|Px\|_p + \|PAx\|_p + \|Qx\|_p \leq 0, \forall x \in \mathcal{X}_f, \tag{12.27}$$

which is the corresponding Lyapunov inequality for the 1-norm and ∞-norm case (7.55) whose solution has been discussed in Section 7.5.3.

In general, if the unconstrained optimal controller (9.31) exists it is PPWA. In this case the computation of the maximal invariant set $\mathcal{X}_f$ for the closed-loop PWA system

$$x(k+1) = (A + F^i)x(k) \quad \text{if} \quad H^i x \leq 0, \ i = 1, \ldots, N^r \tag{12.28}$$

is more involved. However if such $\mathcal{X}_f$ can be computed it can be used as terminal constraint in

Theorem 12.2. With this choice the assumption (A3) in Theorem 12.2 is satisfied by the infinite time unconstrained optimal cost matrix P_∞ in (9.32).

12.4 State Feedback Solution of RHC, 2-Norm Case

The state feedback receding horizon controller (12.9) with cost (12.8) for system (12.1) is

$$u(t) = f_0^*(x(t)), \tag{12.29}$$

where $f_0^*(x_0) : \mathbb{R}^n \to \mathbb{R}^m$ is the piecewise affine solution to the CFTOC (12.6) and is obtained as explained in Section 11.3.

We remark that the implicit form (12.6) and the explicit form (12.29) describe the same function, and therefore the stability, feasibility, and performance properties mentioned in the previous sections are automatically inherited by the piecewise affine control law (12.29). Clearly, the explicit form (12.29) has the advantage of being easier to implement, and provides insight into the type of controller action in different regions CR_i of the state space.

Example 12.3 Consider the double integrator system (12.11) subject to the input constraints

$$-1 \le u(k) \le 1 \tag{12.30}$$

and the state constraints

$$-10 \le x(k) \le 10. \tag{12.31}$$

We want to regulate the system to the origin by using the RHC problem (12.6)–(12.9) with cost (12.8), $Q = \left[\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix}\right]$, $R = 0.01$, and $P = P_\infty$ where P_∞ solves the algebraic Riccati equation (8.32). We consider three cases:

Case 1. $N = 2$, $\mathcal{X}_f = 0$,

Case 2. $N = 2$, $\mathcal{X}_f$ is the positively invariant set of the closed-loop system $x(k+1) = (A + BF_\infty)$ where F_∞ is the infinite time unconstrained optimal controller (8.33).

Case 3. No terminal state constraints: $\mathcal{X}_f = \mathbb{R}^2$ and $N = 6$ = determinedness index+1.

From the results presented in this chapter, all three cases guarantee persistent feasibility for all cost functions and asymptotic stability of the origin with region of attraction $\mathcal{X}_0$ (with $\mathcal{X}_0$ different for each case). Next, we will detail the matrices of the quadratic program for the on-line solution as well as the explicit solution for the three cases.

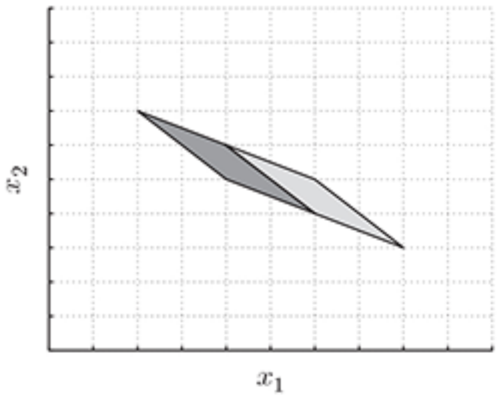
Case 1. $\mathcal{X}_f = 0$. The mp-QP problem associated with the RHC has the form (11.31) with

$$H = \left[\begin{smallmatrix} 19.08 & 8.55 \\ 8.55 & 5.31 \end{smallmatrix}\right],\ F = \left[\begin{smallmatrix} -10.57 & -5.29 \\ -10.59 & -5.29 \end{smallmatrix}\right],\ Y = \left[\begin{smallmatrix} 10.31 & 9.33 \\ 9.33 & 10.37 \end{smallmatrix}\right] \tag{12.32}$$

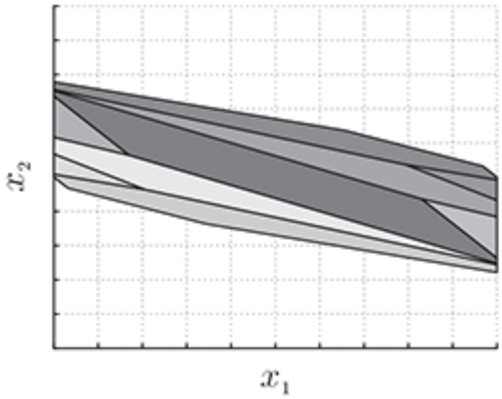
and

$$
G_0 = \begin{bmatrix} 0.00 & -1.00 \\ 0.00 & 1.00 \\ 0.00 & 0.00 \\ -1.00 & 0.00 \\ 0.00 & 0.00 \\ 1.00 & 0.00 \\ -1.00 & 0.00 \\ -1.00 & -1.00 \\ 1.00 & 0.00 \\ 1.00 & 1.00 \\ 0.00 & 0.00 \\ -1.00 & 0.00 \\ 0.00 & 0.00 \\ 1.00 & 0.00 \\ -1.00 & 0.00 \\ 1.00 & 0.00 \\ 0.00 & 0.00 \\ 0.00 & 0.00 \\ 0.00 & 0.00 \\ 0.00 & 0.00 \\ 0.00 & 0.00 \\ 0.00 & 0.00 \\ 0.00 & 0.00 \\ 0.00 & 0.00 \\ 1.00 & 0.00 \\ 1.00 & 1.00 \\ -1.00 & 0.00 \\ -1.00 & -1.00 \end{bmatrix}, \quad E_0 = \begin{bmatrix} 0.00 & 0.00 \\ 0.00 & 0.00 \\ 1.00 & 1.00 \\ -1.00 & -1.00 \\ -1.00 & -1.00 \\ 1.00 & 1.00 \\ 0.00 & 0.00 \\ -1.00 & -1.00 \\ 0.00 & 0.00 \\ 1.00 & 1.00 \\ 1.00 & 1.00 \\ -1.00 & -1.00 \\ -1.00 & -1.00 \\ 1.00 & 1.00 \\ -1.00 & -2.00 \\ 1.00 & 2.00 \\ 1.00 & 0.00 \\ 0.00 & 1.00 \\ -1.00 & 0.00 \\ 0.00 & -1.00 \\ 1.00 & 0.00 \\ 0.00 & 1.00 \\ -1.00 & 0.00 \\ 0.00 & -1.00 \\ 0.00 & 0.00 \\ 1.00 & 1.00 \\ 0.00 & 0.00 \\ -1.00 & -1.00 \end{bmatrix}, \quad w_0 = \begin{bmatrix} 1.00 \\ 1.00 \\ 10.00 \\ 10.00 \\ 10.00 \\ 10.00 \\ 10.00 \\ 10.00 \\ 10.00 \\ 10.00 \\ 5.00 \\ 5.00 \\ 5.00 \\ 5.00 \\ 1.00 \\ 1.00 \\ 10.00 \\ 10.00 \\ 10.00 \\ 10.00 \\ 5.00 \\ 5.00 \\ 5.00 \\ 5.00 \\ 0.00 \\ 0.00 \\ 0.00 \\ 0.00 \end{bmatrix} \tag{12.33}
$$

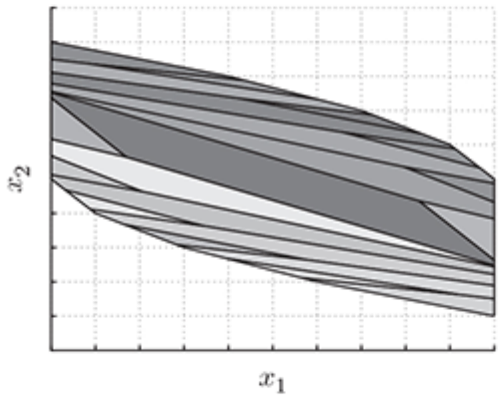
The corresponding polyhedral partition of the state space is depicted in Figure 12.6(a).



(a) Polyhedral partition of $\mathcal{X}_0$, Case 1. $N = 2$, $\mathcal{X}_f = 0$, $N_0^r = 7$.



(b) Polyhedral partition of $\mathcal{X}_0$, Case 2. $N = 2$, $\mathcal{X}_f = \mathcal{O}_\infty^{\mathrm{LQR}}$, $N_0^r = 9$.



(c) Polyhedral partition of $\mathcal{X}_0$, Case 3. $N = 6$, $\mathcal{X}_f = \mathbb{R}^2$, $N_0^r = 13$.

Figure 12.6 Example 12.3. Double integrator. RHC with 2-norm. Region of attraction $\mathcal{X}_0$ for different horizons N and terminal regions $\mathcal{X}_f$.

The RHC law is:

$$
u = \begin{cases}
[-0.61 \ -1.61]\,x & \text{if } \begin{bmatrix} 0.70 & 0.71 \\ -0.70 & -0.71 \\ -0.70 & -0.71 \\ 0.70 & 0.71 \end{bmatrix} x \le \begin{bmatrix} 0.00 \\ 0.00 \\ 0.00 \\ 0.00 \end{bmatrix} & \text{(Region \#1)} \\
[-1.00 \ -2.00]\,x & \text{if } \begin{bmatrix} -0.71 & -0.71 \\ -0.70 & -0.71 \\ -0.45 & -0.89 \\ 0.45 & 0.89 \\ 0.71 & 0.71 \\ -0.70 & -0.71 \end{bmatrix} x \le \begin{bmatrix} 0.00 \\ -0.00 \\ 0.45 \\ 0.45 \\ 0.71 \\ -0.00 \end{bmatrix} & \text{(Region \#2)} \\
[-1.00 \ -2.00]\,x & \text{if } \begin{bmatrix} 0.45 & 0.89 \\ -0.70 & -0.71 \\ 0.71 & 0.71 \end{bmatrix} x \le \begin{bmatrix} 0.45 \\ -0.00 \\ -0.00 \end{bmatrix} & \text{(Region \#3)} \\
[-0.72 \ -1.72]\,x & \text{if } \begin{bmatrix} 0.39 & 0.92 \\ 0.70 & 0.71 \\ -0.70 & -0.71 \\ 0.70 & 0.71 \end{bmatrix} x \le \begin{bmatrix} 0.54 \\ 0.00 \\ 0.00 \\ -0.00 \end{bmatrix} & \text{(Region \#4)} \\
[-1.00 \ -2.00]\,x & \text{if } \begin{bmatrix} 0.45 & 0.89 \\ -0.71 & -0.71 \\ 0.70 & 0.71 \\ -0.45 & -0.89 \\ 0.71 & 0.71 \\ 0.70 & 0.71 \end{bmatrix} x \le \begin{bmatrix} 0.45 \\ 0.71 \\ -0.00 \\ 0.45 \\ 0.00 \\ -0.00 \end{bmatrix} & \text{(Region \#5)} \\
[-1.00 \ -2.00]\,x & \text{if } \begin{bmatrix} -0.45 & -0.89 \\ -0.71 & -0.71 \\ 0.70 & 0.71 \end{bmatrix} x \le \begin{bmatrix} 0.45 \\ -0.00 \\ -0.00 \end{bmatrix} & \text{(Region \#6)} \\
[-0.72 \ -1.72]\,x & \text{if } \begin{bmatrix} -0.39 & -0.92 \\ 0.70 & 0.71 \\ -0.70 & -0.71 \\ -0.70 & -0.71 \end{bmatrix} x \le \begin{bmatrix} 0.54 \\ 0.00 \\ 0.00 \\ -0.00 \end{bmatrix} & \text{(Region \#7)}
\end{cases}
$$

The union of the regions depicted in Figure 12.6(a) is $\mathcal{X}_0$. From Theorem 12.2, $\mathcal{X}_0$ is also the domain of attraction of the RHC law.

Case 2. $\mathcal{X}_f$ positively invariant set. The set $\mathcal{X}_f$ is

$$
\mathcal{X}_f = \left\{ x \in \mathbb{R}^2 : \begin{bmatrix} -0.35617 & -0.93442 \\ 0.35617 & 0.93442 \\ 0.71286 & 0.70131 \\ -0.71286 & -0.70131 \end{bmatrix} x \le \begin{bmatrix} 0.58043 \\ 0.58043 \\ 1.9049 \\ 1.9049 \end{bmatrix} \right\}. \tag{12.34}
$$

The corresponding polyhedral partition of the state space is depicted in Figure 12.6(b). The union of the regions depicted in Figure 12.6(b) is $\mathcal{X}_0$. Note that from Theorem 12.2 the set $\mathcal{X}_0$ is also the domain of attraction of the RHC law.

Case 3. $\mathcal{X}_f = \mathbb{R}^n$, $N = 6$. The corresponding polyhedral partition of the state space is depicted in Figure 12.6(c).

Comparing the feasibility regions $\mathcal{X}_0$ in Figure 12.6 we notice that in Case 2 we obtain a larger region than in Case 1 and that in Case 3 we obtain a feasibility region larger than Case 1 and Case 2. This can be easily explained from the theory presented in this and the previous chapter. In particular we have seen that if a control invariant set is chosen as terminal constraint $\mathcal{X}_f$, the size of the feasibility region increases with the number of control moves (increase from Case 2 to Case 3) (Remark 11.3). Actually in Case 3, $\mathcal{X}_0 = \mathcal{K}_\infty(\mathcal{X}_f)$ with $\mathcal{X}_f = \mathbb{R}^2$, the maximal controllable set. Also, the size of the feasibility region increases with the size of the target set (increase from Case 1 to Case 2).

12.5 State Feedback Solution of RHC, 1-Norm, ∞-Norm Case

The state feedback receding horizon controller (12.6)–(12.9) with cost (12.7) for system (12.1) is

$$u(t) = f_0^*(x(t)), \tag{12.35}$$

where $f_0^*(x_0) : \mathbb{R}^n \to \mathbb{R}^m$ is the piecewise affine solution to the CFTOC (12.6) and is computed as explained in Section 11.4. As in the 2-norm case the explicit form (12.35) has the advantage of being easier to implement, and provides insight into the type of control action in different regions CR_i of the state space.

Example 12.4 Consider the double integrator system (12.11) subject to the input constraints

$$-1 \leq u(k) \leq 1 \tag{12.36}$$

and the state constraints

$$-5 \leq x(k) \leq 5. \tag{12.37}$$

We want to regulate the system to the origin by using the RHC controller (12.6)– (12.9) with cost (12.7), $p = \infty$, $Q = \left[\begin{smallmatrix}1 & 0\\ 0 & 1\end{smallmatrix}\right]$, $R = 20$. We consider two cases:

Case 1. $\mathcal{X}_f = \mathbb{R}^n$, $N = 6$ (determinedness index+1) and $P = Q$

Case 2. $\mathcal{X}_f = \mathbb{R}^n$, $N = 6$ and $P = P_\infty$ given in (9.34) measuring the infinite time unconstrained optimal cost in (9.32).

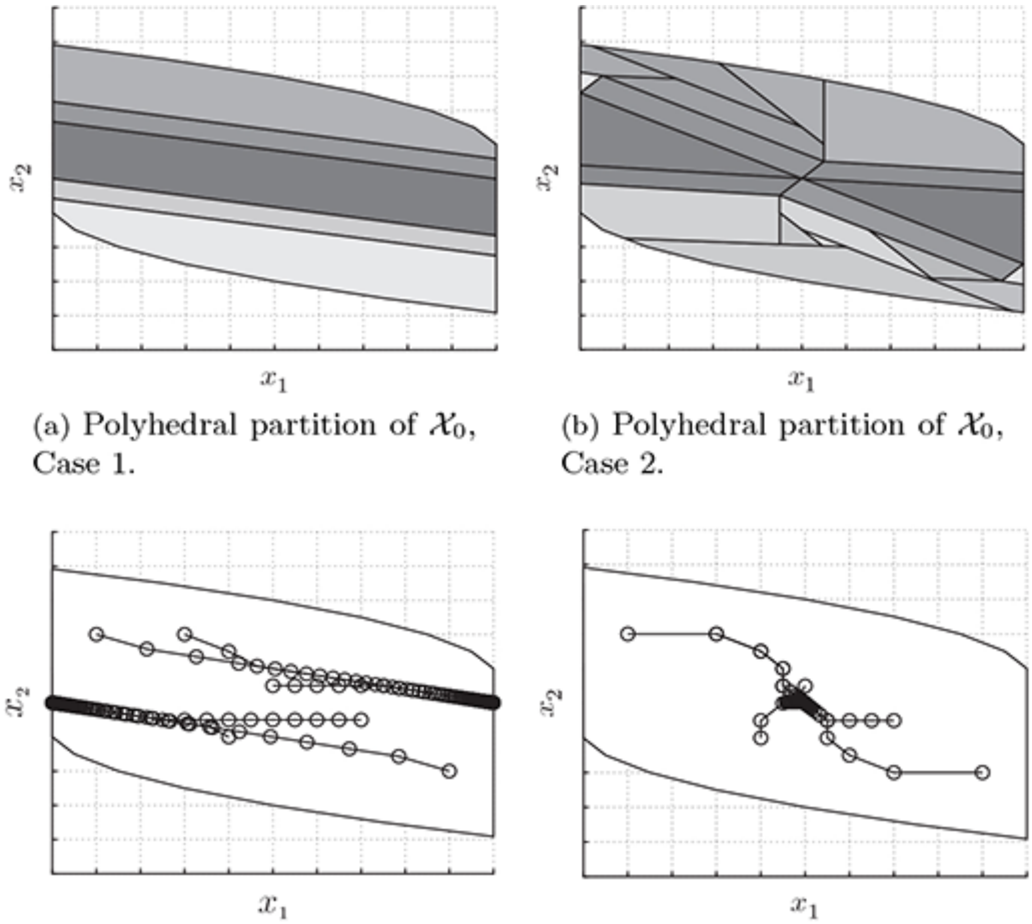
From Corollary 12.2 in both cases persistent feasibility is guaranteed for all cost functions and $\mathcal{X}_0 = \mathcal{C}_\infty$. However, in Case 1 the terminal cost P does not satisfy (12.27) which is assumption (A3) in Theorem 12.2 and therefore the convergence to and the stability of the origin cannot be guaranteed. In order to satisfy assumption (A3) in Theorem 12.2, in Case 2 we select the terminal cost to be equal to the infinite time unconstrained optimal cost computed in Example 9.1.

Next we will detail the explicit solutions for the two cases.

Case 1. The LP problem associated with the RHC has the form (11.56) with $\bar{G}_0 \in \mathbb{R}^{124\times18}$, $\bar{S}_0 \in \mathbb{R}^{124\times2}$ and $c' = [\mathbf{0}_6\ \mathbf{1}_{12}]$. The corresponding polyhedral partition of the state space is depicted in Figure 12.7(a). The RHC law is:

$$u = \begin{cases} 0 & \text{if } \begin{bmatrix} 0.16 & 0.99 \\ -0.16 & -0.99 \\ -1.00 & 0.00 \\ 1.00 & 0.00 \end{bmatrix} x \leq \begin{bmatrix} 0.82 \\ 0.82 \\ 5.00 \\ 5.00 \end{bmatrix} & \text{(Region \#1)} \\ \begin{bmatrix} -0.29 & -1.71 \end{bmatrix} x + 1.43 & \text{if } \begin{bmatrix} 1.00 & 0.00 \\ -0.16 & -0.99 \\ -1.00 & 0.00 \\ 0.16 & 0.99 \end{bmatrix} x \leq \begin{bmatrix} 5.00 \\ -0.82 \\ 5.00 \\ 1.40 \end{bmatrix} & \text{(Region \#2)} \\ -1.00 & \text{if } \begin{bmatrix} -0.16 & -0.99 \\ 1.00 & 0.00 \\ 0.71 & 0.71 \\ -1.00 & 0.00 \\ 0.20 & 0.98 \\ 0.16 & 0.99 \\ 0.24 & 0.97 \\ 0.45 & 0.89 \\ 0.32 & 0.95 \end{bmatrix} x \leq \begin{bmatrix} -1.40 \\ 5.00 \\ 4.24 \\ 5.00 \\ 2.94 \\ 3.04 \\ 2.91 \\ 3.35 \\ 3.00 \end{bmatrix} & \text{(Region \#3)} \\ \begin{bmatrix} -0.29 & -1.71 \end{bmatrix} x - 1.43 & \text{if } \begin{bmatrix} -1.00 & 0.00 \\ 0.16 & 0.99 \\ 1.00 & 0.00 \\ -0.16 & -0.99 \end{bmatrix} x \leq \begin{bmatrix} 5.00 \\ -0.82 \\ 5.00 \\ 1.40 \end{bmatrix} & \text{(Region \#4)} \\ 1.00 & \text{if } \begin{bmatrix} -0.32 & -0.95 \\ -0.24 & -0.97 \\ -0.20 & -0.98 \\ -0.16 & -0.99 \\ -1.00 & 0.00 \\ 0.16 & 0.99 \\ -0.71 & -0.71 \\ -0.45 & -0.89 \\ 1.00 & 0.00 \end{bmatrix} x \leq \begin{bmatrix} 3.00 \\ 2.91 \\ 2.94 \\ 3.04 \\ 5.00 \\ -1.40 \\ 4.24 \\ 3.35 \\ 5.00 \end{bmatrix} & \text{(Region \#5)} \end{cases}$$

The union of the regions depicted in Figure 12.7(a) is $\mathcal{X}_0$ and is shown in white in Figure 12.7(c). Since N is equal to the determinedness index plus one, $\mathcal{X}_0$ is a positive invariant set for the closed-loop system and thus persistent feasibility is guaranteed for all $x(0) \in \mathcal{X}_0$. However, it can be noticed from Figure 12.7(c) that convergence to the origin is not guaranteed. Starting from the initial conditions [–4,2], [–2, 2], [0, 0.5], [4,–2], [–1,–1] and [2,–0.5], the closed-loop system converges to either [–5, 0] or [5, 0].



(a) Polyhedral partition of $\mathcal{X}_0$, Case 1.

(b) Polyhedral partition of $\mathcal{X}_0$, Case 2.

(c) Closed-loop Trajectories, Case 1. Convergence to origin to guaranteed.

(d) Closed-loop Trajectories, Case 2.

Figure 12.7 Example 12.4. Double integrator. RHC with ∞-norm cost function, behavior for different terminal weights.

Case 2. The LP problem associated with the RHC has the form (11.56) with $\bar{G}_0 \in \mathbb{R}^{174\times 18}$, $\bar{S}_0 \in \mathbb{R}^{174\times 2}$ and $c' = [\mathbf{0}_6\ \mathbf{1}_{12}]$. The RHC law is defined over 21 regions and the corresponding polyhedral partition of the state space is depicted in Figure 12.7(b).

The union of the regions depicted in Figure 12.7(b) is $\mathcal{X}_0$ and is shown in white in Figure 12.7(d). Since N is equal to the determinedness index plus one, $\mathcal{X}_0$ is a positive invariant set for the closed-loop system and thus persistent feasibility is guaranteed for all $x(0) \in \mathcal{X}_0$. Convergence to the origin is also guaranteed by the choice of P as shown by the closed-loop trajectories in Figure 12.7(d) starting from the same initial conditions as in Case 1.

12.6 Tuning and Practical Use

Receding Horizon Control in the different variants and with the many parameter options as introduced in this chapter is a very powerful control technique to address complex control problems in practice. At present there is no other technique to design controllers for general large linear multivariable systems with input and output constraints with a stability guarantee.

The fact that there is no effective tool to *analyze* the stability of large constrained multivariable systems makes this stability guarantee *by design* so important. The application of RHC in practice, however, requires the designer to make many choices. In this section we will offer some guidance.

Objective Function

The squared 2-norm is employed more often as an indicator of control quality in the objective function than the 1- or ∞-norm. The former has many advantages.

The 1- or ∞-norm formulation leads to an LP (11.56) from which the optimal control action is determined. The solution lies always at the intersection of constraints and changes *discontinuously* as the tuning parameters are varied. This makes the formulation of the control problem (what constraints need to be added for good performance?) and the choice of the weights in the objective function difficult.

The optimizer may be nonunique in the case of dual degeneracy. This may cause the control action to vary randomly as the solver picks up different optimizers in succeeding time steps.

The construction of the control invariant target set $\mathcal{X}_f$ and of a suitable terminal cost $p(x_N)$ needed in Theorem 12.2 is more difficult for the 1- or ∞-norm formulation because the unconstrained controller is PPWA (Section 12.3.2).

If the optimal control problem is stated in an *ad hoc* manner by trial and error rather than on the basis of Theorem 12.2 undesirable response characteristics may result. The controller may cease to take any action and the system may get stuck. This happens if any control action leads to a short-term increase in the objective function, which can occur, for example, for a system with inverse response characteristics.

Finally, the 1- or ∞-norm formulation involves many more constraints than the 2-norm formulation. In general, this will lead to a larger number of regions of the explicit control law. These regions do not need to be stored, however, but only the value function and the controller, and the search for the region containing the present state can be executed very efficiently (Chapter 14).

Despite the discussed deficiencies it may be advantageous to use the 1- or ∞-norm in cases when the control objective is not just an indirect tool to achieve a control specification but reflects the economics of the process. For example, the costs of electrical power depend sometimes on peak power that can be captured with the ∞-norm. However, even in these cases, quadratic terms are often added to the objective for “regularization.”

Design via Theorem 12.2

First, we need to choose the horizon length N and the control invariant target set $\mathcal{X}_f$. Then we can vary the parameters Q and R freely to affect the control performance in the same spirit as we do for designing an LQR. Stability is assured as long as we adjust P according to the outlined procedures when changing Q and R.

The longer the horizon N, the larger the maximal controllable set $\mathcal{K}_N(\mathcal{X}_f)$ over which the closed-loop system is guaranteed to be able to operate (this is true as long as the horizon is smaller than the determinedness index of $\mathcal{K}_\infty(\mathcal{X}_f)$). On the other hand, with the control horizon increases the on-line computational effort or the complexity of the explicit controller determined off-line.

We need to keep in mind that the terminal set $\mathcal{X}_f$ is introduced artificially for the sole purpose of leading to a *sufficient* condition for persistent feasibility. We want it to be large so that it does not compromise closed-loop performance. The larger $\mathcal{X}_f$, the larger $\mathcal{K}_N(\mathcal{X}_f)$. Though it is simplest to choose $\mathcal{X}_f = 0$, it is undesirable unless N is chosen large. Ideally $\mathcal{X}_f$ should be the maximal control invariant set achieved with the unconstrained controller.

Specifically, we first design the unconstrained optimal controller as suggested at the end of Section 12.3.2. From this construction we also obtain the terminal cost satisfying condition (A3) of Theorem 12.2 to use in the RHC design. Then we determine the maximal positive invariant set for the closed-loop system with the unconstrained controller and use this set as $\mathcal{X}_f$. This set is usually difficult to compute for systems of large dimension with the algorithms we introduced in Section (10.2).

Note that for stable systems without state constraints $\mathcal{K}_\infty(\mathcal{X}_f) = \mathbb{R}^n$ always, i.e., the choice of $\mathcal{X}_f$ is less critical. For unstable systems $\mathcal{K}_\infty(\mathcal{X}_f)$ is the region over which the system can operate stably in the presence of input constraints and is of eminent practical importance.

State/Output Constraints

State constraints arise from practical restrictions on the allowed operating range of the system. Thus, contrary to input constraints, they are rarely "hard." They can lead to complications in the controller implementation, however. As it can never be excluded that the state of the real system moves outside the constraint range chosen for the controller design, special provisions must be made (patches in the algorithm) to move the state back into the range. This is difficult and these types of patches are exactly what one wanted to avoid by choosing MPC in the first place.

Thus, typically, state constraints are "softened" in the MPC formulation. For example,

$$x \le x_{\max}$$

is approximated by

$$x \le x_{\max} + \epsilon, \ \ \epsilon \ge 0,$$

and a term $l(\epsilon)$ is added to the objective function to penalize violations of the constraint. This formulation may, however, lead to a violation of the constraint even when a feasible input exists that avoids it but is not optimal for the new objective function with the penalty term added. Let us analyze how to choose $l(\cdot)$ such that this does not occur by using the exact penalty method [173].

As an example, take

$$\begin{array}{rll} J^* = \min_z & f(z) \\ \text{subj. to} & g(z) \le 0, \end{array} \tag{12.38}$$

where $g(z)$ is assumed to be scalar for simplicity and $f(z)$ to be strictly convex. Let us soften the constraints and add the penalty as suggested

$$\begin{array}{rll} p(\epsilon) = \min_z & f(z) + l(\epsilon) \\ \text{subj. to} & g(z) \le \epsilon, \end{array} \tag{12.39}$$

where $\epsilon \ge 0$. We want to choose the penalty $l(\epsilon)$ so that by minimizing the augmented objective we recover the solution to the original problem if it exists:

$$p(0) = J^*$$
and
$$\arg\min_{\epsilon \geq 0} p(\epsilon) = 0$$

which are equivalent to

$$p(0) = J^* \quad \text{and} \tag{12.40}$$

$$p(\epsilon) > p(0), \ \forall \epsilon > 0. \tag{12.41}$$

For (12.40) we require $l(0) = 0$. To construct $l(\epsilon)$ to satisfy (12.41) we assume that strong duality holds and u^* exists so that

$$J^* = \min_z (f(z) + u^* g(z)), \tag{12.42}$$

where u^* is an optimal dual variable. As the optimizer z^* of (12.38) satisfies $g(z^*) \leq 0$ we can add a redundant constraint without affecting the solution

$$J^* = \begin{array}{ll} \min_z & (f(z) + u^* g(z)) \\ \text{subj. to} & g(z) \leq \epsilon \end{array} \quad \forall \epsilon \geq 0. \tag{12.43}$$

Then we can state the bounds

$$J^* \leq \begin{array}{ll} \min_z & (f(z) + u^* \epsilon) \\ \text{subj. to} & g(z) \leq \epsilon \end{array} < \begin{array}{ll} p(\epsilon) = \min_z & (f(z) + u\epsilon) \\ \text{subj. to} & g(z) \leq \epsilon \end{array} \quad \forall \epsilon > 0, \ u > u^*. \tag{12.44}$$

Thus $l(\epsilon) = u\epsilon$ with $u > u^* \geq 0$ is a possible penalty term satisfying the requirements (12.40)–(12.41).

Because of smoothness

$$l(\epsilon) = u\epsilon + v\epsilon^2, \ u > u^*, v > 0 \tag{12.45}$$

is preferable. On the other hand, note that $l(\epsilon) = v\epsilon^2$ does not satisfy an inequality of the form (12.44). Therefore, it should not be used as it can lead to optimizers z^* in (12.39) which violate $g(z) \leq 0$ even if a feasible optimizer to the original problem exists.

These ideas can be extended to multiple constraints $g_j(z) \leq 0$, $j = 1, \ldots, r$ via the penalty term

$$l(\epsilon) = u \sum_{j=1}^{r} \epsilon_j + v \sum_{j=1}^{r} \epsilon_j^2, \tag{12.46}$$

where

$$u > \max_{j \in \{1,\ldots,r\}} u_j^*, \; v \geq 0. \tag{12.47}$$

Formulations also exist where the necessary constraint violations are following a prescribed order so that less important constraints are violated first [174].

Time-varying references, constraints, disturbances and system parameters

The standard RHC formulation (12.6) can be easily extended to include these features. Known disturbances are simply included in the prediction model. If the system states are not to return to the origin, but some output y is to follow some trajectory r, then appropriate penalties of the error $e = y - r$ are included in the control objective. How to do this and to achieve offset-free tracking is described in Section 12.7. Theorem 12.2 can be used to design the controller in these cases.

If the constraints are time-varying then $\mathcal{X}$ and $\mathcal{U}$ become time-varying. For example, the constraints may be shaped like a funnel tightening towards the end of the horizon.

If the underlying system is nonlinear, one often uses a locally linearized model for the prediction and updates it at each time step. Note that the results of Theorem 12.2 do not apply when the system model and/or the constraints are time-varying.

In all these cases, the optimization problem to be solved on line, (11.31) or (11.56), does not change in structure but some of the defining matrices will now change at each time step, which will increase the necessary on-line computational effort somewhat.

If the controller is to be computed explicitly off-line, then all the varying parameters (disturbances, constraints, references), which we will denote by θ, become *parameters* in the multiparametric optimization problem and the resulting controller becomes an explicit function of them: $u(t) = F(x(t), \theta)$. As we have learned, the complexity of the solution of mp-QP and mp-LP problems depends primarily on the number of constraints. If the number of parameters affects the number of constraints then this may only be possible for a relatively small number of parameters. Thus, the possibilities to take into account time-varying control problem features in explicit MPC are rather limited. Time-varying models cannot be handled at all.

Multiple Horizons and Move-Blocking

Unfortunately, as we started to discuss in the previous paragraphs, in challenging applications, the design procedure implied by Theorem 12.2 may not be strictly applicable. Then the closed-loop behavior with the RHC has to be analyzed by other means, in the worst case through extensive simulation studies. In principle, this offers us also more freedom in the problem formulation and the choice of the parameters, for example, in order to reduce the computational complexity. The basic RHC formulation (12.6) may be modified as follows:

$$\begin{array}{ll} \min\limits_{U_0} & p(x_{N_y}) + \sum\limits_{k=0}^{N_y-1} q(x_k, u_k) \\ \text{subj. to} & x_{k+1} = Ax_k + Bu_k, \; k = 0, \ldots, N_y - 1 \\ & x_k \in \mathcal{X}, k = 0, \ldots, N_c \\ & u_k \in \mathcal{U}, \; k = 0, \ldots, N_u \\ & u_k = Kx_k, \; N_u < k < N_y \end{array} \tag{12.48}$$

where K is some feedback gain, N_y, N_u, N_c are the prediction, input, and state constraint horizons, respectively, with $N_u \le N_y$ and $N_c \le N_y$. This formulation reduces the number of constraints and as a consequence makes the long horizon prediction used in the optimization less accurate as it is not forced to obey all the constraints. As this approximation affects only the states far in the future, it is hoped that it will not influence significantly the present control action.

Generalized Predictive Control (GPC) in its most common formulation [89] has multiple horizons as an inherent feature but does not include constraints. Experience and theoretical analysis [56] have shown that it is very difficult to choose all these horizons that affect not only performance but even stability in a nonintuitive fashion. Thus, for problems where constraints are not important and the adaptive features of GPC are not needed, it is much preferable to resort to the well established LQR and LQG controllers for which a wealth of stability, performance and robustness results have been established.

Another more effective way to reduce the computational effort is *move-blocking* where the manipulated variables are assumed to be fixed over time intervals in the future thus reducing the degrees of freedom in the optimization problem. By choosing the blocking strategies carefully, the available RHC stability results remain applicable [72].

12.7 Offset-Free Reference Tracking

This section describes how the RHC problem has to be formulated to track constant references without offset under model mismatch. We distinguish between the number p of measured outputs, the number r of outputs which one desires to track (called “tracked outputs”), and the number n_d of disturbances. First, we summarize the conditions that need to be satisfied to obtain offset-free RHC by using the arguments of the internal model principle. Then, we provide a simple proof of zero steady-state offset when $r \le p = n_d$. Extensive treatment of reference tracking for RHC can be found in in [13, 218, 225, 226, 198]. Consider the discrete-time time-invariant system

$$\begin{cases} x_m(t+1) = f(x_m(t), u(t)) \\ y_m(t) = g(x_m(t)) \\ z(t) = H y_m(t). \end{cases} \tag{12.49}$$

In (12.49), $x_m(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^m$ and $y_m(t) \in \mathbb{R}^p$ are the state, input, measured output vector, respectively. The controlled variables $z(t) \in \mathbb{R}^r$ are a linear combination of the measured variables for which offset-free behavior is sought. Without any loss of generality we assume H to have full row rank.

The objective is to design an RHC based on the linear system model (12.1) of (12.49) in order to have $z(t)$ track $r(t)$, where $r(t) \in \mathbb{R}^r$ is the reference signal, which we assume to converge to a constant, i.e., $r(t) \to r_\infty$ as $t \to \infty$. We require zero steady-state tracking error, i.e., $(z(t) - r(t)) \to 0$ for $t \to \infty$.

The Observer Design

The plant model (12.1) is augmented with a disturbance model in order to capture the mismatch between (12.49) and (12.1) in steady state. Several disturbance models have been presented in the literature [13, 205, 193, 226, 225, 294]. Here we follow [226] and use the form:

$$\begin{cases} x(t+1) = Ax(t) + Bu(t) + B_d d(t) \\ d(t+1) = d(t) \\ \quad y(t) = Cx(t) + C_d d(t) \end{cases} \tag{12.50}$$

with $d(t) \in \mathbb{R}^{n_d}$. With abuse of notation we have used the same symbols for state and outputs of system (12.1) and system (12.50). Later we will focus on specific versions of the model (12.50).

The observer estimates both states and disturbances based on this augmented model. Conditions for the observability of (12.50) are given in the following theorem.

Theorem 12.3 *[215, 216, 226, 13] The augmented system (12.50) is observable if and only if* (C, A) *is observable and*

$$\begin{bmatrix} A-I & B_d \\ C & C_d \end{bmatrix} \tag{12.51}$$

has full column rank.

Proof: From the Hautus observability condition system (12.50) is observable iff

$$\begin{bmatrix} A'-\lambda I & 0 & C' \\ B_d' & I-\lambda I & C_d' \end{bmatrix} \text{ has full row rank } \forall\lambda. \tag{12.52}$$

Again from the Hautus condition, the first set of rows is linearly independent iff (C, A) is observable. The second set of rows is linearly independent from the first n rows except possibly for $\lambda = 1$. Thus, for the augmented system the Hautus condition needs to be checked for $\lambda = 1$ only, where it becomes (12.51). ■

Remark 12.4 Note that for condition (12.51) to be satisfied the number of disturbances in d needs to be smaller or equal to the number of available measurements in y, $n_d \leqslant p$. Condition (12.51) can be nicely interpreted. It requires that the model of the disturbance effect on the output $d \to y$ must not have a zero at (1, 0). Alternatively we can look at the steady state of system (12.50)

$$\begin{bmatrix} A-I & B_d \\ C & C_d \end{bmatrix} \begin{bmatrix} x_\infty \\ d_\infty \end{bmatrix} = \begin{bmatrix} 0 \\ y_\infty \end{bmatrix}, \tag{12.53}$$

where we have denoted the steady state values with a subscript ∞ and have omitted the forcing term u for simplicity. We note that from the observability condition (12.51) for system (12.50) Equation (12.53) is required to have a unique solution, which means, that we must be able to deduce a unique value for the disturbance d_∞ from a measurement of y_∞ in steady state.

The following corollary follows directly from Theorem 12.3.

Corollary 12.3 *The augmented system (12.50) with $n_d = p$ and $C_d = I$ is observable if and only if* (C, A) *is observable and*

$$det\begin{bmatrix} A-I & B_d \\ C & I \end{bmatrix} = det(A-I-B_dC) \neq 0. \tag{12.54}$$

Remark 12.5 We note here how the observability requirement restricts the choice of the disturbance model. If the plant has no integrators, then det $(A - I) \neq 0$ and we can choose $B_d = 0$. If the plant has integrators then B_d has to be chosen specifically to make det $(A - I - B_dC) \neq 0$.

The state and disturbance estimator is designed based on the augmented model as follows:

$$\begin{bmatrix} \hat{x}(t+1) \\ \hat{d}(t+1) \end{bmatrix} = \begin{bmatrix} A & B_d \\ 0 & I \end{bmatrix}\begin{bmatrix} \hat{x}(t) \\ \hat{d}(t) \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u(t) + \begin{bmatrix} L_x \\ L_d \end{bmatrix}(-y_m(t)+C\hat{x}(t)+C_d\hat{d}(t)), \tag{12.55}$$

where L_x and L_d are chosen so that the estimator is stable. We remark that the results below are independent of the choice of the method for computing L_x and L_d. We then have the following property.

Lemma 12.3 *Suppose the observer (12.55) is stable. Then,* rank$(L_d) = n_d$.

Proof: From (12.55) it follows

$$\begin{bmatrix} \hat{x}(t+1) \\ \hat{d}(t+1) \end{bmatrix} = \begin{bmatrix} A + L_xC & B_d + L_xC_d \\ L_dC & I + L_dC_d \end{bmatrix}\begin{bmatrix} \hat{x}(t) \\ \hat{d}(t) \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u(t) - \begin{bmatrix} L_x \\ L_d \end{bmatrix} y_m(t). \tag{12.56}$$

By stability, the observer has no poles at (1, 0) and therefore

$$\det\left(\begin{bmatrix} A - I + L_xC & B_d + L_xC_d \\ L_dC & L_dC_d \end{bmatrix}\right) \neq 0. \tag{12.57}$$

For (12.57) to hold, the last n_d rows of the matrix have to be of full row rank. A necessary condition is that L_d has full row rank. ■

In the rest of this section we will focus on the case $n_d = p$.

Lemma 12.4 *Suppose the observer (12.55) is stable. Choose $n_d = p$. The steady state of the observer (12.55) satisfies:*

$$\begin{bmatrix} A-I & B \\ C & 0 \end{bmatrix}\begin{bmatrix} \hat{x}_\infty \\ u_\infty \end{bmatrix} = \begin{bmatrix} -B_d\hat{d}_\infty \\ y_{m,\infty} - C_d\hat{d}_\infty \end{bmatrix}, \tag{12.58}$$

where $y_{m,\infty}$ and u_∞ are the steady state measured output and input of the system (12.49), $\hat{x}_\infty$ and $\hat{d}_\infty$ are state and disturbance estimates from the observer (12.55) at steady state, respectively.

Proof: From (12.55) we note that the disturbance estimate $\hat{d}$ converges only if $L_d(-y_{m,\infty} + C\hat{x}_\infty + C_d\hat{d}_\infty) = 0$. As L_d is square by assumption and nonsingular by Lemma 12.3 this implies that at steady state, the observer estimates (12.55) satisfy

$$-y_{m,\infty} + C\hat{x}_\infty + C_d\hat{d}_\infty = 0. \tag{12.59}$$

Equation (12.58) follows directly from (12.59) and (12.55). ■

The MPC Design

Denote by $z_\infty = Hy_{m,\infty}$ and r_∞ the tracked measured outputs and their references at steady state, respectively. For offset-free tracking at steady state we want $z_\infty = r_\infty$. The observer condition (12.58) suggests that at steady state the MPC should satisfy

$$\begin{bmatrix} A-I & B \\ HC & 0 \end{bmatrix} \begin{bmatrix} x_\infty \\ u_\infty \end{bmatrix} = \begin{bmatrix} -B_d\hat{d}_\infty \\ r_\infty - HC_d\hat{d}_\infty \end{bmatrix}, \tag{12.60}$$

where x_∞ is the MPC state at steady state. For x_∞ and u_∞ to exist for any $\hat{d}_\infty$ and r_∞ the matrix $\begin{bmatrix} A-I & B \\ HC & 0 \end{bmatrix}$ must be of full row rank which implies $m \geq r$.

The MPC is designed as follows

$$\begin{array}{ll} \min_{U_0} & (x_N - \bar{x}_t)'P(x_N - \bar{x}_t) + \sum_{k=0}^{N-1} (x_k - \bar{x}_t)'Q(x_k - \bar{x}_t) + (u_k - \bar{u}_t)'R(u_k - \bar{u}_t) \\ \text{subj. to} & x_{k+1} = Ax_k + Bu_k + B_d d_k,\ k = 0, \ldots, N \\ & x_k \in \mathcal{X},\ u_k \in \mathcal{U},\ k = 0, \ldots, N-1 \\ & x_N \in \mathcal{X}_f \\ & d_{k+1} = d_k,\ k = 0, \ldots, N-1 \\ & x_0 = \hat{x}(t) \\ & d_0 = \hat{d}(t), \end{array} \tag{12.61}$$

with the targets $\bar{u}_t$ and $\bar{x}_t$ given by

$$\begin{bmatrix} A-I & B \\ HC & 0 \end{bmatrix} \begin{bmatrix} \bar{x}_t \\ \bar{u}_t \end{bmatrix} = \begin{bmatrix} -B_d\hat{d}(t) \\ r(t) - HC_d\hat{d}(t) \end{bmatrix} \tag{12.62}$$

and where $Q \succeq 0$, $R \succ 0$, and $P \succ 0$.

Let $U^*(t) = \{u_0^*, \ldots, u_{N-1}^*\}$ be the optimal solution of (12.61)–(12.62) at time t. Then, the first sample of $U^*(t)$ is applied to system (12.49)

$$u(t) = u_0^*. \tag{12.63}$$

Denote by $c_0(\hat{x}(t), \hat{d}(t), r(t)) = u_0^*(\hat{x}(t), \hat{d}(t), r(t))$ the control law when the estimated state and disturbance are $\hat{x}(t)$ and $\hat{d}(t)$, respectively. Then the closed-loop system obtained by controlling (12.49) with the MPC (12.61)–(12.62)–(12.63) and the observer (12.55) is:

$$
\begin{aligned}
x(t+1) &= f(x(t), c_0(\hat{x}(t), \hat{d}(t), r(t))) \\
\hat{x}(t+1) &= (A + L_x C)\hat{x}(t) + (B_d + L_x C_d)\hat{d}(t) + Bc_0(\hat{x}(t), \hat{d}(t), r(t)) - L_x y_m(t) \\
\hat{d}(t+1) &= L_d C\hat{x}(t) + (I + L_d C_d)\hat{d}(t) - L_d y_m(t).
\end{aligned} \tag{12.64}
$$

Often in practice, one desires to track all measured outputs with zero offset. Choosing $n_d = p = r$ is thus a natural choice. Such a zero-offset property continues to hold if only a subset of the measured outputs are to be tracked, i.e., $n_d = p > r$. Next we provide a very simple proof for offset-free control when $n_d = p$.

Theorem 12.4 *Consider the case $n_d = p$. Assume that for $r(t) \to r_\infty$ as $t \to \infty$, the MPC problem (12.61)–(12.62) is feasible for all $t \in \mathbb{N}_+$, unconstrained for $t \geq j$ with $j \in \mathbb{N}_+$ and the closed-loop system (12.64) converges to $\hat{x}_\infty$, $\hat{d}_\infty$, $y_{m,\infty}$, i.e., $\hat{x}(t) \to \hat{x}_\infty$, $\hat{d}(t) \to \hat{d}_\infty$, $y_m(t) \to y_{m,\infty}$ as $t \to \infty$. Then $z(t) = Hy_m(t) \to r_\infty$ as $t \to \infty$.*

Proof: Consider the MPC problem (12.61)–(12.62). At steady state $u(t) \to u_\infty = c_0(\hat{x}_\infty, \hat{d}_\infty, r_\infty)$, $\bar{x}_t \to \bar{x}_\infty$ and $\bar{u}_t \to \bar{u}_\infty$. Note that the steady state controller input u_∞ (computed and implemented) might be different from the steady state target input $\bar{u}_\infty$.

The asymptotic values $\hat{x}_\infty$, $\bar{x}_\infty$, u_∞ and $\bar{u}_\infty$ satisfy the observer conditions (12.58)

$$
\begin{bmatrix} A - I & B \\ C & 0 \end{bmatrix} \begin{bmatrix} \hat{x}_\infty \\ u_\infty \end{bmatrix} = \begin{bmatrix} -B_d \hat{d}_\infty \\ y_{m,\infty} - C_d \hat{d}_\infty \end{bmatrix} \tag{12.65}
$$

and the controller requirement (12.62)

$$
\begin{bmatrix} A - I & B \\ HC & 0 \end{bmatrix} \begin{bmatrix} \bar{x}_\infty \\ \bar{u}_\infty \end{bmatrix} = \begin{bmatrix} -B_d \hat{d}_\infty \\ r_\infty - HC_d \hat{d}_\infty \end{bmatrix}. \tag{12.66}
$$

Define $\delta x = \hat{x}_\infty - \bar{x}_\infty$, $\delta u = u_\infty - \bar{u}_\infty$ and the offset $\varepsilon = z_\infty - r_\infty$. Notice that the steady state target values $\bar{x}_\infty$ and $\bar{u}_\infty$ are both functions of r_∞ and $\hat{d}_\infty$ as given by (12.66). Left multiplying the second row of (12.65) by H and subtracting (12.66) from the result, we obtain

$$
\begin{aligned}
&(A - I)\delta x + B\delta u = 0 \\
&HC\delta x = \varepsilon.
\end{aligned} \tag{12.67}
$$

Next we prove that $\delta x = 0$ and thus $\varepsilon = 0$.

Consider the MPC problem (12.61)–(12.62) and the following change of variables $\delta x_k = x_k - \bar{x}_t$, $\delta u_k = u_k - \bar{u}_t$. Notice that $Hy_k - r(t) = HCx_k + HC_d d_k - r(t) = HC\delta x_k + HC\bar{x}_t + HC_d d_k - r(t) = HC\delta x_k$ from condition (12.62) with $\hat{d}(t) = d_k$. Similarly, one can show that $\delta x_{k+1} = A\delta x_k + B\delta u_k$. Then, around the origin where all constraints are inactive the MPC problem (12.61) becomes:

$$\begin{array}{ll} \min_{\delta u_0,\ldots,\delta u_{N-1}} & \delta x_N' P \delta x_N + \sum_{k=0}^{N-1} \delta x_k' Q \delta x_k + \delta u_k' R \delta u_k \\ \text{subj. to} & \delta x_{k+1} = A\delta x_k + B\delta u_k, \quad 0 \le k \le N \\ & \delta x_0 = \delta x(t), \\ & \delta x(t) = \hat{x}(t) - \bar{x}_t. \end{array} \tag{12.68}$$

Denote by K_{MPC} the unconstrained MPC controller (12.68), i.e., $\delta u_0^* = K_{MPC}\delta x(t)$. At steady state $\delta u_0^* \to u_\infty - \bar{u}_\infty = \delta u$ and $\delta x(t) \to \hat{x}_\infty - \bar{x}_\infty = \delta x$. Therefore, at steady state, $\delta u = K_{MPC}\delta x$. From (12.67)

$$(A - I + BK_{MPC})\delta x = 0. \tag{12.69}$$

By assumption the unconstrained system with the MPC controller converges. Thus K_{MPC} is a stabilizing control law, which implies that $(A - I + BK_{MPC})$ is nonsingular and hence $\delta x = 0$. ∎

Remark 12.6 Theorem 12.4 was proven in [226] by using a different approach.

Remark 12.7 Theorem 12.4 can be extended to prove local Lyapunov stability of the closed-loop system (12.64) under standard regularity assumptions on the state update function f in (12.64) [204].

Remark 12.8 The proof of Theorem 12.4 assumes only that the models used for the control design (12.1) and the observer design (12.50) are identical in steady state in the sense that they give rise to the same relation $z = z(u, d, r)$. It does not make any assumptions about the behavior of the real plant (12.49), i.e., the model-plant mismatch, with the exception that the closed-loop system (12.64) must converge to a fixed point. The models used in the controller and the observer could even be different as long as they satisfy the same steady state relation.

Remark 12.9 If condition (12.62) does not specify $\bar{x}_t$ and $\bar{u}_t$ uniquely, it is customary to determine $\bar{x}_t$ and $\bar{u}_t$ through an optimization problem, for example, minimizing the magnitude of $\bar{u}_t$ subject to the constraint (12.62) [226].

Remark 12.10 Note that in order to achieve no offset we augmented the model of the plant with as many disturbances (and integrators) as we have measurements ($n_d = p$) (cf. Equation (12.56)). Our design procedure requires the addition of p integrators even if we wish to control only a subset of $r < p$ measured variables. This is actually not necessary as we suspect from basic system theory. The design procedure for the case $n_d = r < p$ is, however, more involved [198].

If the squared 2-norm in the objective function of (12.61) is replaced with a 1- or ∞-norm $(\|P(x_N - \bar{x}_t)\|_p + \sum_{k=0}^{N-1} \|Q(x_k - \bar{x}_t)\|_p + \|R(u_k - \bar{u}_t)\|_p$, where $p = 1$ or $p = \infty$), then our results continue to hold. In particular, Theorem 12.4 continues to hold. The unconstrained MPC controlled K_{MPC} in (12.68) will be piecewise linear around the origin [44]. In particular, around the origin, $\delta u^*(t) = \delta u_0^* = K_{MPC}(\delta x(t))$ is a continuous piecewise linear function of the state variation δx:

$$K_{MPC}(\delta x) = F^j \delta x \quad \text{if} \quad H^j \delta x \leq K^j,\ j = 1, \ldots, N^r, \tag{12.70}$$

where H^j and K^j in equation (12.70) are the matrices describing the j-th polyhedron $CR^j = \{\delta x \in \mathbb{R}^n : H^j\delta x \leq K^j\}$ inside which the feedback optimal control law $\delta u^*(t)$ has the linear form $F^j\delta x(k)$. The polyhedra CR^j, $j = 1, \ldots, N^r$ are a partition of the set of feasible states of problem (12.61) and they all contain the origin.

Explicit Controller

Examining (12.61), (12.62) we note that the control law depends on $\hat{x}(t)$,$\hat{d}(t)$ and $r(t)$. Thus in order to achieve offset free tracking of r outputs out of p measurements we had to add the $p + r$ "parameters"$\hat{d}(t)$ and $r(t)$ to the usual parameters $\hat{x}(t)$.

There are more involved RHC design techniques to obtain offset-free control for models with $n_d < p$ and in particular, with minimum order disturbance models $n_d = r$. The total size of the parameter vector can thus be reduced to $n + 2r$. This is significant only if a small subset of the plant outputs are to be controlled. A greater reduction of parameters can be achieved by the following method. By Corollary 12.3, we are allowed to choose $B_d = 0$ in the disturbance model if the plant has no integrators. Recall the target conditions 12.62 with $B_d = 0$

$$\begin{bmatrix} A - I & B \\ HC & 0 \end{bmatrix} \begin{bmatrix} \bar{x}_t \\ \bar{u}_t \end{bmatrix} = \begin{bmatrix} 0 \\ r(t) - HC_d\hat{d}(t) \end{bmatrix}. \tag{12.71}$$

Clearly, any solution to (12.71) can be parameterized by $r(t){-}HC_d\hat{d}(t)$. The explicit control law is written as $u(t) = c_0(\hat{x}(t),\ r(t){-}HC_d\hat{d}(t))$, with only $n{+}r$ parameters. Since the observer is unconstrained, complexity is much less of an issue. Hence, a full disturbance model with $n_d = p$ can be chosen to yield offset-free control.

Remark 12.11 The choice of $B_d = 0$ might be limiting in practice. In [14], the authors have shown that for a wide range of systems, if $B_d = 0$ and a Kalman filter is chosen as observer, then the closed-loop system might suffer a dramatic performance deterioration.

Delta Input (δu) Formulation

In the δu formulation, the MPC scheme uses the following linear time-invariant system model of (12.49):

$$\begin{cases} x(t+1) = Ax(t) + Bu(t) \\ u(t) = u(t-1) + \delta u(t) \\ y(t) = Cx(t). \end{cases} \tag{12.72}$$

System (12.72) is controllable if (A, B) is controllable. The δu formulation often arises naturally in practice when the actuator is subject to uncertainty, e.g., the exact gain is unknown or is subject to drift. In these cases, it can be advantageous to consider changes in the control value as input to the plant. The absolute control value is estimated by the observer, which is expressed as follows

$$\begin{bmatrix} \hat{x}(t+1) \\ \hat{u}(t+1) \end{bmatrix} = \begin{bmatrix} A & B \\ 0 & I \end{bmatrix} \begin{bmatrix} \hat{x}(t) \\ \hat{u}(t) \end{bmatrix} + \begin{bmatrix} B \\ I \end{bmatrix} \delta u(t) + \begin{bmatrix} L_x \\ L_u \end{bmatrix} (-y_m(t) + C\hat{x}(t)). \tag{12.73}$$

The MPC problem is readily modified

$$\begin{array}{ll} \min_{\delta u_0,\ldots,\delta u_{N-1}} & \|y_k - r_k\|_Q^2 + \|\delta u_k\|_R^2 \\ \text{subj. to} & x_{k+1} = Ax_k + Bu_k,\ k \geq 0 \\ & y_k = Cx_k\ k \geq 0 \\ & x_k \in \mathcal{X},\ u_k \in \mathcal{U},\ k = 0,\ldots,N-1 \\ & x_N \in \mathcal{X}_f \\ & u_k = u_{k-1} + \delta u_k,\ k \geq 0 \\ & u_{-1} = \hat{u}(t) \\ & x_0 = \hat{x}(t). \end{array} \tag{12.74}$$

The control input applied to the system is

$$u(t) = \delta u_0^* + u(t-1). \tag{12.75}$$

The input estimate $\hat{u}(t)$ is not necessarily equal to the actual input $u(t)$. This scheme inherently achieves offset-free control, there is no need to add a disturbance model. To see this, we first note that $\delta u_0^* = 0$ in steady-state. Hence our analysis applies as the δu formulation is equivalent to a disturbance model in steady-state. This is due to the fact that any plant/model mismatch is lumped into $\hat{u}(t)$. Indeed this approach is equivalent to an input disturbance model ($B_d = B$, $C_d = 0$). If in (12.74) the measured $u(t)$ were substituted for its estimate, i.e., $u_{-1} = u(t - 1)$, then the algorithm would show offset.

In this formulation the computation of a target input $\bar{u}_t$ and state $\bar{x}_t$ is not required. A disadvantage of the formulation is that it is not applicable when there is an excess of manipulated variables u compared to measured variables y, since detectability of the augmented system (12.72) is then lost.

Minimum-Time Controller

In minimum-time control, the cost function minimizes the predicted number of steps necessary to reach a target region, usually the invariant set associated with the unconstrained LQR controller [167]. This scheme can reduce the on-line computation time significantly, especially for explicit controllers (Section 11.5). While minimum-time MPC is computed and implemented differently from standard MPC controllers, there is no difference between the two control schemes at steady-state. In particular, one can choose the target region to be the unconstrained region of (12.61)–(12.62). When the state and disturbance estimates and reference are within this region, the control law is switched to (12.61)–(12.62). The analysis and methods presented in this section therefore apply directly.

12.8 Literature Review

Although the basic idea of receding horizon control can be found in the theoretical work of Propoi [239] in 1963 it did not gain much attention until the mid-1970s, when Richalet and coauthors [248, 249] introduced their “Model Predictive Heuristic Control (MPHC)”.

Independently, in 1969 Charles Cutler proposed the concept to his PhD advisor Dr. Huang at the University of Houston. In 1973 Cutler implemented it successfully in the Shell Refinery in New Orleans, Louisiana [90].

Several years later Cutler and Ramaker [91] described this predictive control algorithm named Dynamic Matrix Control (DMC) in the literature. It has been hugely successful in the petrochemical industry. A vast variety of different methodologies with different names followed, such as Quadratic Dynamic Matrix Control (QDMC), Adaptive Predictive Control (APC), Generalized Predictive Control (GPC), Sequential Open Loop Optimization (SOLO), and others.

While the mentioned algorithms are seemingly different, they all share the same structural features: a model of the plant, the receding horizon idea, and an optimization procedure to obtain the control action by optimizing the system's predicted evolution.

Some of the first industrial MPC algorithms like IDCOM [249] and DMC [91] were developed for constrained MPC with quadratic performance indices. However, in those algorithms input and output constraints were treated in an indirect adhoc fashion. Only later, algorithms like QDMC [118] overcame this limitation by employing quadratic programming to solve constrained MPC problems with quadratic performance indices. During the same period, the use of linear programming was studied by Gutman and coauthors [137, 138, 140].

An extensive theoretical effort was devoted to analyze receding horizon control schemes, provide conditions for guaranteeing feasibility and closed-loop stability, and highlight the relations between MPC and the linear quadratic regulator [204, 203]. Theorem 12.2 in this book is the main result on feasibility and stability of MPC and was adopted from these publications.

The idea behind Theorem 12.2 dates back to Keerthi and Gilbert [168], the first researchers to propose specific choices for the terminal cost P and the terminal constraint $\mathcal{X}_f$, namely $\mathcal{X}_f = 0$ and $P = 0$. Under these assumptions Keerthi and Gilbert prove the stability for general nonlinear performance functions and nonlinear models. Their work has been followed by many other stability conditions for RHC including those in [168, 34, 35, 154, 84]. If properly analyzed all these results are based on the same concepts as Theorem 12.2.

In the past fifty years, the richness of theoretical and computational issues surrounding MPC has generated a large number of research studies that appeared in conference proceedings, archival journals and research monographs. The preceding paragraphs are not intended to summarize this vast literature. We refer the interested reader to some books on Model Predictive Control [76, 197, 247] that appeared in the last decade.

For complex constrained multivariable control problems, model predictive control has long been the accepted standard in the process industries [240, 241]. The results reported in this book have greatly reduced the on-line computational effort and have opened up this methodology to other application areas where hardware speed and costs are dominant — unlike in process control.