

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

4,800

Open access books available

122,000

International authors and editors

135M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.

For more information visit www.intechopen.com



Infeasibility Handling in Constrained MPC

Rubens Junqueira Magalhães Afonso and
Roberto Kawakami Harrop Galvão
*Instituto Tecnológico de Aeronáutica
Brazil*

1. Introduction

1.1 Aim of the chapter

Predictive Control optimization problems may be rendered infeasible in the presence of constraints due to model-plant mismatches, external perturbations, noise or faults. This may cause the optimizer to issue a control sequence which is impossible to implement, leading to prediction errors, as well as loss of stability of the control loop. Such a problem motivates the development of techniques aimed at recovering feasibility without violating hard physical constraints imposed by the nature of the plant. Currently, setpoint management approaches and techniques dealing with changes in the constraints are two of the most effective solutions to recover feasibility with low computational demand. In this chapter a review of techniques that can be understood as one of the aforementioned is presented along with some illustrative simulation examples.

1.2 Concepts and literature review

One of the main advantages of Predictive Control is the ability to deal with constraints over the inputs and states of the plant in an explicit manner, which brings better performance and more safety to the operation of the plant (Maciejowski, 2002), (Rossiter, 2003). Constraints over the excursion of the control signals are particularly common in processes that operate near optimal conditions (Rodrigues & Odloak, 2005). However, if the optimization becomes infeasible, possibly due to model-plant mismatches, external perturbations, noise or faults, a control sequence which is impossible to implement may be issued, leading to prediction errors, as well as loss of stability of the control loop (Maciejowski, 2002). Such a problem motivates the development of techniques aimed at recovering feasibility without violating hard physical constraints imposed by the nature of the plant.

The MPC formulation itself allows for a simple solution, which consists of enlarging the horizons, as means to allow for more degrees of freedom in the optimization. On the other hand, an increase in the computational burden associated to the solution of the optimization problem results, since there are more decision variables as well as constraints. Moreover, enlarging the horizons cannot solve all sorts of infeasibilities.

Constraint relaxation is one alternative which involves less decision variables and is usually effective. Nevertheless, it is often not obvious which constraints to relax and the amount by which they should be relaxed in order to attain a feasible optimization problem. There are

different approaches for this purpose, some of which will be briefly discussed in this chapter. Initially, one must differentiate between two types of constraints (Alvarez & de Prada, 1997), (Vada et al., 2001):

Physical constraints: those limits that can never be surpassed and are determined by the physical functioning of the system. For instance, a valve cannot be opened more than 100% or less than 0%.

Operating constraints: those limits fixed by the plant operator. These limits, which are usually more restrictive than the physical constraints, define the band within which the variables are expected to be under normal operating conditions. For instance, it may be more profitable to operate a chemical reactor in a certain range of temperatures, in order to favor the kinetics of the desired reaction that forms products of economical interest. However, if maintaining such operating condition would compromise the safety of operation of the plant at some point, then the associated constraints could be relaxed.

The literature has many different approaches to constraint relaxation. Some infeasibility handling techniques are described in Rawlings & Muske (1993) and Sokaert & Rawlings (1999):

Minimal time approach: An algorithm identifies the smallest time, $\kappa(x)$, which depends on the current state x , beyond which the state constraint can be satisfied over an infinite horizon. Prior to time $\kappa(x)$, the state constraint is ignored, and the control law enforces the state constraint only after that time. An advantage of this method is that it leads to the earliest possible constraint satisfaction. Transient constraint violations, however, can be large.

Soft-constraint approach: Violations of the state constraints are allowed, but an additional term is introduced in the cost function to penalize the constraint violation.

In Zafiriou & Chiou (1993) the authors propose a method for calculating the smallest magnitude of the relaxation that renders the optimization feasible for a SISO system.

The paper by Sokaert (1994) presents many suggestions to circumvent the problem of infeasibility, among which, one that classifies the constraints in priority levels and tries to enforce the ones with higher priority through relaxation of the others.

Sokaert & Rawlings (1999) introduce an approach capable of minimizing the peak and duration of the constraint violation, with advantages concerning the transient response.

A relaxation procedure that can be applied either to the controls or to the system outputs is described by Alvarez & de Prada (1997). The control-related approach consists of relaxing the operating constraints on the control amplitude or rate of change according to a priority schedule. The output-related approach consists of relaxing the operating constraints on the output amplitude or modifying the time interval where such constraints are imposed within the prediction horizon.

In Vada et al. (2001) the proposed scheme involves the classification of the constraints in priority levels and the solution of a linear programming problem parallel to the MPC optimization. In Afonso & Galvão (2010a), different weights are employed for the relaxation of operating output constraints, up to the values of physical constraints, as means to overcome infeasibility caused by actuator faults.

Another alternative to recover feasibility are the so-called setpoint management procedures (Bemporad & Mosca, 1994), (Gilbert & Kolmanovsky, 1995), (Bemporad et al., 1997), which

artificially reduce the distance between the actual plant state and the constraint set. The reference governor proposed by Kapasouris et al. (1988) inspired many techniques to deal with problems involving actuator saturation through manipulation of the setpoint or the tracking error (Gilbert & Kolmanovsky, 1995). There are also papers aiming at imposing a reference model to the behavior of the plant that employ setpoint management in order to obtain feasibility when the control signals are bounded (Montandon et al., 2008).

Stability guarantees may be achieved with setpoint management by using a terminal constraint invariant set parameterized by the setpoint. Limon et al. (2008) employ this technique parameterizing the terminal set in terms of the control and state setpoints. The authors show that an optimal management of the setpoint may be achieved, guaranteeing the smallest distance between the desired setpoint and the one used by the MPC. This procedure increases the domain of attraction of the controller dramatically.

An application of the parameterization of the terminal set in terms of the steady-state value of the control can be found in Almeida & Leissling (2010). In that paper, the technique is employed to circumvent infeasibility caused by actuator faults which limit the range of values of control that the actuator can deploy. On the other hand, in Afonso & Galvão (2010b) the authors manage the setpoint of a state variable that does not affect the control setpoint, making parameterization of the terminal set unnecessary, as means to overcome infeasibility brought about by similar actuator faults.

In this chapter, the treatment of infeasibility in the optimization problem of constrained MPC will be discussed. Some illustrative simulations will provide a basic coverage of this topic, which is of great importance to practical implementations of MPC due to the capability of circumventing problems brought about by model-plant mismatch, faults, noise, disturbances or simply reducing the computational burden required to calculate an adequate control sequence.

2. Adopted MPC formulation

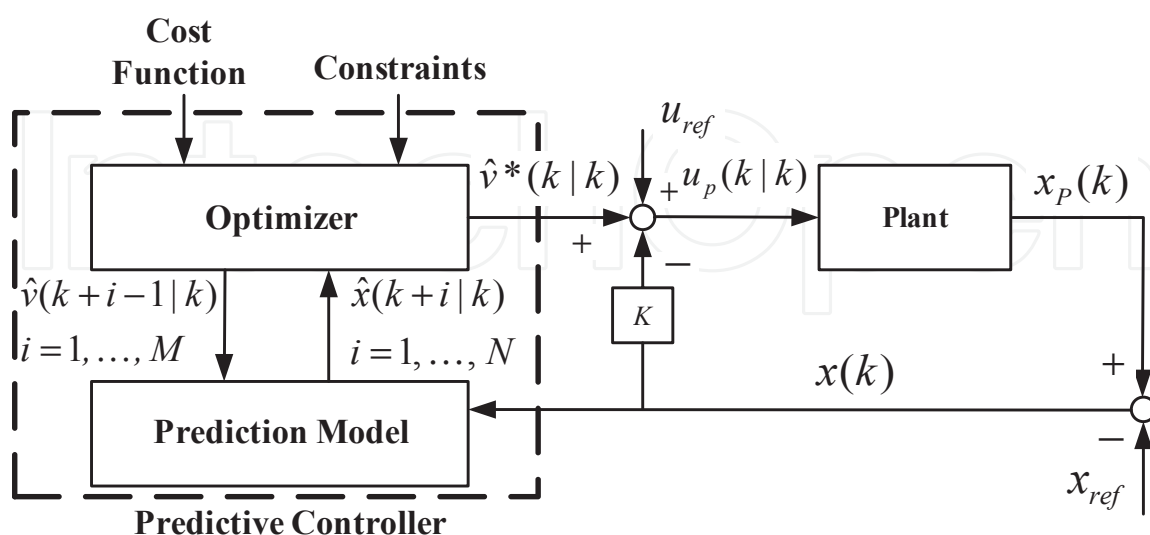


Fig. 1. MPC with inner feedback loop.

Fig. 1 presents the main elements of the MPC formulation adopted in this chapter. Since this is a regulator scheme, the desired equilibrium value x_{ref} for the state must be subtracted from the measured state of the plant x_P , in order to generate the state x read by the controller:

$$x = x_P - x_{ref} \quad (1)$$

In a similar manner, the corresponding equilibrium value of the control u_{ref} must be added to the output of the controller u to generate the control u_P to be applied to the plant, that is:

$$u = u_P - u_{ref} \quad (2)$$

A mathematical model of the plant is employed to calculate state predictions N steps ahead, over the so-called "Prediction Horizon". These predictions are determined on the basis of the current state ($x(k) \in \mathbb{R}^n$) and are also dependent on the future control sequence. $\hat{\bullet}(k+i|k)$ denotes the predicted value of variable \bullet at time $k+i$ ($i \geq 1$) based on the information available at time k . The optimization algorithm determines a control sequence, over a Control Horizon of M steps ($\hat{v}(k+i-1|k)$, $i = 1, \dots, M$), that minimizes the cost function specified for the problem, possibly subject to state and/or input constraints. It is assumed that the MPC control sequence is set to zero after the end of the Control Horizon, i.e. $\hat{v}(k+i-1|k) = 0$, $i > M$. The control is implemented in a receding horizon fashion, i.e., only the first element of the optimized control sequence is applied to the plant and the solution is recalculated at the next sampling period taking into account the new sensor readings. Therefore, the controller output at time k is given by $u(k) = \hat{u}^*(k|k) = \hat{v}^*(k|k) - Kx(k)$, where K is the gain of an internal loop.

It is assumed that the dynamics of the plant can be described by a discrete state-space equation of the form $x_P(k+1) = Ax_P(k) + Bu_P(k)$. Therefore, the relation between u and x is given by

$$x(k+1) = Ax(k) + Bu(k) \quad (3)$$

The MPC controller is designed to enforce constraints of the type

$$u_{P,min} \leq u_P \leq u_{P,max} \quad (4)$$

$$x_{P,min} \leq x_P \leq x_{P,max} \quad (5)$$

Considering Eqs. (1) and (2), the constraints in Eqs. (4) and (5) can be expressed as

$$u_{P,min} - u_{ref} \leq u \leq u_{P,max} - u_{ref} \quad (6)$$

$$x_{P,min} - x_{ref} \leq x \leq x_{P,max} - x_{ref} \quad (7)$$

The optimization problem to be solved at instant k consists of minimizing a cost function of the form

$$J_{mpc} = \sum_{i=0}^{M-1} \hat{v}^T(k+i|k) \Psi \hat{v}(k+i|k) \quad (8)$$

subject to the following constraints:

$$\hat{u}(k+i|k) = -K\hat{x}(k+i|k) + \hat{v}(k+i|k), \quad i \geq 0 \quad (9)$$

$$\hat{v}(k+i|k) = 0, \quad i \geq M \quad (10)$$

$$\hat{x}(k+i+1|k) = A\hat{x}(k+i|k) + B\hat{u}(k+i|k), \quad i \geq 0 \quad (11)$$

$$\hat{x}(k|k) = x(k) \quad (12)$$

$$\hat{y}(k+i|k) = C\hat{x}(k+i|k), \quad i \geq 0 \quad (13)$$

$$\hat{u}(k+i|k) \in \mathbf{U}, \quad i \geq 0 \quad (14)$$

$$\hat{x}(k+i|k) \in \mathbf{X}, \quad i > 0 \quad (15)$$

in which $\Psi = \Psi^T > 0$ is a weight matrix and \mathbf{U} and \mathbf{X} are the sets of admissible controls and states, respectively, according to Eqs. (6) and (7).

Following a receding horizon policy, the control at the k -th instant is given by $u(k) = \hat{v}^*(k|k) - Kx(k)$, where K is the gain of the internal loop represented in Fig. 1. At time $k + 1$, the optimization is repeated to obtain $v^*(k + 1|k + 1)$.

The inner-loop controller is designed as a Linear Quadratic Regulator (LQR) with the following cost function:

$$J_{lqr} = \sum_{i=0}^{\infty} \left[\hat{x}^T(k + i|k) Q_{lqr} \hat{x}(k + i|k) + \hat{u}^T(k + i|k) R_{lqr} \hat{u}(k + i|k) \right], \quad (16)$$

$$Q_{lqr} = Q_{lqr}^T \geq 0, \quad R_{lqr} = R_{lqr}^T > 0$$

with Q_{lqr} chosen so that the pair $(A, Q_{lqr}^{\frac{1}{2}})$ is detectable.

Let P be the only non-negative symmetric solution of the Algebraic Riccati Equation $P = A^T P A - A^T P B (R_{lqr} + B^T P B)^{-1} B^T P A + Q_{lqr}$. It can then be shown that, if the weight matrix Ψ is chosen as $\Psi = R_{lqr} + B^T P B$, then the minimization of the cost in Eq. (8) subject to the constraints of Eqs. (9) – (15) is equivalent to the minimization of the cost of Eq. (16) subject to the constraints of Eqs. (11) – (15) (Chisci et al., 2001). The outcome is that the cost function has an infinite horizon, which is useful for stability guarantees (Sokaert & Rawlings, 1998), (Kouvaritakis et al., 1998). It is worth noting that, due to the penalization of the control signal \hat{v} in the cost of Eq. (8), the MPC acts only when it is necessary to correct the inner-loop control in order to avoid violations of the constraints stated in Eqs. (14) and (15).

Defining vector \hat{V} and matrix $\bar{\Psi}$ as

$$\hat{V} = \begin{bmatrix} \hat{v}(k|k) \\ \vdots \\ \hat{v}(k + M - 1|k) \end{bmatrix}, \quad \bar{\Psi} = \begin{bmatrix} \Psi & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \Psi \end{bmatrix}, \quad (17)$$

the cost function can be rewritten as

$$J_{mpc} = \hat{V}^T \bar{\Psi} \hat{V} \quad (18)$$

which is quadratic in terms of \hat{V} .

Defining the vectors

$$\hat{X} = \begin{bmatrix} \hat{x}(k + 1|k) \\ \vdots \\ \hat{x}(k + N|k) \end{bmatrix}, \quad \hat{U} = \begin{bmatrix} \hat{u}(k|k) \\ \vdots \\ \hat{u}(k + N - 1|k) \end{bmatrix}, \quad (19)$$

the state and control prediction vectors may be related to \hat{V} as (Maciejowski, 2002):

$$\begin{aligned} \hat{X} &= H \hat{V} + \Phi x(k) \\ \hat{U} &= H_u \hat{V} + \Phi_u x(k) \end{aligned} \quad (20)$$

It is important to remark that the presence of an infinite number of constraints in Eqs. (14) and (15) does not allow the employment of computational methods for the solution of the

optimization problem. However, this issue can be circumvented by introducing a terminal constraint for the state in the form of a Maximal Output Admissible Set (MAS) (Gilbert & Tan, 1991). This problem will be tackled in section 4. For now, it is sufficient to state that there exists a finite horizon within which enforcement of the constraints leads to enforcement of the constraints over an infinite horizon, given some reasonable assumptions on the plant dynamics (Rawlings & Muske, 1993).

3. Constraint relaxation approaches

3.1 Minimal-time approach

Minimal-time approaches allow constraint violations for a certain period of time, which is to be minimized. There is no commitment to reduce the peaks of the violations during this period. These are, respectively, the strongest advantage and the weakest drawback of these methods. The constraint violations are usually allowed to take place in the beginning of the control task, which reduces the time taken to achieve feasibility at the cost of degrading the transient response of the control-loop. Sokaert & Rawlings (1999) introduce an approach of minimal-time solution that considers the peak violation of the constraints as a secondary objective, after the minimization of the time to enforce the constraints. This avoids unnecessarily large peak violations.

One possibility to avoid control constraint violations, which are usually physical ones, is to enforce them while relaxing operating constraints on the state. This way, the problem always becomes feasible. One algorithm that implements a solution of this type may be stated as:

Data: $x(k)$

Result: Optimized control sequence \hat{V}^*

Solve constrained MPC problem;

if *infeasible* **then**

 Remove constraints on the state;

 Solve MPC problem;

 Find $\kappa = \kappa_{unc}$, which is the instant at which the state constraints are all enforced;

else

 Employ obtained control sequence;

 Terminate.

end

while *feasible* **do**

$\kappa \leftarrow \kappa - 1$;

 Solve MPC problem with state constraints enforced from time κ until the end of the prediction horizon;

end

Employ last feasible control sequence;

Terminate.

Algorithm 1: Minimal-time algorithm

This algorithm determines the smallest time window over which the state constraints must be removed at the beginning of the prediction horizon in order to attain feasibility.

3.2 Soft-constraint approach

In this approach the cost function is modified to include a penalization on the violation of operating constraints. This way, a compromise is achieved between time and peak values of the violations, as well as performance of the control-loop. Scokaert & Rawlings (1999) propose the penalization of the sum of the square of the values of the violations instead of the peak as means to reduce their time length. This can be accomplished by simply adding slack variables to the state/output constraints of Eq. (7) in case of infeasibility and adding a term to the right-hand side of Eq. (8), as follows:

$$J_{Soft} = \sum_{i=0}^{N-1} \hat{\delta}^T(k+i|k) \Psi \hat{\delta}(k+i|k) + \epsilon_p^T W_{\epsilon_p} \epsilon_p + \epsilon_n^T W_{\epsilon_n} \epsilon_n \quad (21)$$

$$\begin{aligned} x_{P,min} - x_{ref} - \epsilon_n \leq x \leq x_{P,max} - x_{ref} + \epsilon_p, \\ \epsilon_p, \epsilon_n \geq 0 \end{aligned} \quad (22)$$

where W_{ϵ_n} and W_{ϵ_p} are positive-definite weight matrices. The additional restrictions $\epsilon_p, \epsilon_n \geq 0$ impose that the constraints are not made more restrictive than their original settings.

With the cost function of Eq. (21) subject to the constraints of Eq. (22), the amount by which each constraint is prioritized can be tuned by the choice of the weight matrices.

To this end, a rule of thumb known as ‘‘Bryson’s rule’’ (Franklin et al., 2005), (Bryson & Ho, 1969) can be used as a guideline. It states that one may use the limits of the variables as parameters to choose their weights in the cost function so that their contribution is normalized. Therefore, the weights must be chosen so that the product between the admissible range (maximum value - minimum value) and the weight is approximately the same for all variables. However, in the present case, it is desirable that deviations of the slack variables from zero are more penalized than control deviations in order to enforce the constraints when possible. Therefore, it is reasonable to choose the weights for these variables an order of magnitude greater than the values obtained via Bryson’s rule.

Scokaert & Rawlings (1999) discuss the inclusion of a linear term of penalization of the slack variables as means to obtain exact relaxations, i. e., the controller relaxes the constraints only when necessary. This can be achieved by tuning the weights of this term based on the Lagrange multipliers associated to the constrained minimization problem. However, an advantage of introducing terms that penalize the square of the slack variables is that the choice of a positive-definite weight matrix leads to a well-posed quadratic program, since the associated Hessian is positive definite.

3.3 Hard constraint relaxation with prioritization

There are methods which relax the operating constraints, possibly according to a priority list, in order to achieve feasibility of the optimization problem. There are various techniques employing such policies, some of which resort to optimization problems parallel to the MPC optimization in order to determine the minimum relaxation that is necessary to achieve feasibility. In this line, the priority list can be explored by solving many Linear Programming (LP) problems relaxing the constraints of lower priority until feasibility is achieved or by solving a single LP problem online as proposed by Vada et al. (2001). In their work, offline computations of the weights of the slack variables that relax the constraints are performed.

The calculated weights have the property of relaxing the constraints according to the defined priority in a single LP problem.

3.4 Simulation example

This example is based on a double integrator model, with sampling period of 1 time unit. Double integrators can be used to model a number of real-world systems, such as a vehicle moving in an environment where friction is negligible (space, for instance).

The discrete-time model matrices are:

$$A = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}, B = \begin{bmatrix} 0.5 \\ 1 \end{bmatrix} \quad (23)$$

and the LQR weight matrices are:

$$Q_{lqr} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, R_{lqr} = 1 \quad (24)$$

The control and prediction horizons were set to $M = 7$ and $N = 20$, respectively.

The constraints are: $-0.5 \leq x_1 \leq 0.5$ (position), $-0.1 \leq x_2 \leq 0.1$ (velocity) and $-0.01 \leq u \leq 0.01$ (acceleration).

A comparison between the results obtained with a minimal-time solution and a soft constraint approach is presented. Two choices of weight matrices were considered:

$$W_{\epsilon_n}^1 = W_{\epsilon_p}^1 = W^1 = \begin{bmatrix} 10 & 0 \\ 0 & 20 \end{bmatrix}, W_{\epsilon_n}^2 = W_{\epsilon_p}^2 = W^2 = \begin{bmatrix} 100 & 0 \\ 0 & 10000 \end{bmatrix} \quad (25)$$

The application of Bryson's rule to adjust the weight matrices would require the definition of an acceptable violation of the constraints, which could be established as the difference between physical and operating state constraints. However, since this example does not discriminate between these two types of constraints, the W^1 and W^2 matrices were chosen for the sole purpose of illustrating the effect of varying the weights.

The initial state of the system is $x_0 = [1.5 \ 0]^T$, which violates the constraints on x_1 .

The first comparison involves the two infeasibility handling techniques (minimal-time and soft constraint). For this purpose, the W^1 weight matrix was employed. Figures 2 and 3 show the resulting state trajectories. It can be seen that the minimal-time approach leads to a faster recovery of feasibility, as the soft constraint approach takes longer to enforce all the constraints. This result can also be associated to the control profile presented in Fig. 4. In fact, the control obtained with the minimal-time approach reverses its sign earlier, as compared to the soft constraint approach.

The second comparison involves three scenarios: no state constraints and soft constraint approach with weights W^1 and W^2 . Figures 5, 6 and 7 show the resulting state and control trajectories. As can be seen, a reduction in the weights tends to generate a solution closer to the unconstrained case. In fact, smaller weights on the slack variables result in a smaller penalization of the constraint violations. In the limit, if the weights are made equal to zero, the constraints can be relaxed as much as it is needed and therefore the unconstrained optimal solution is obtained.

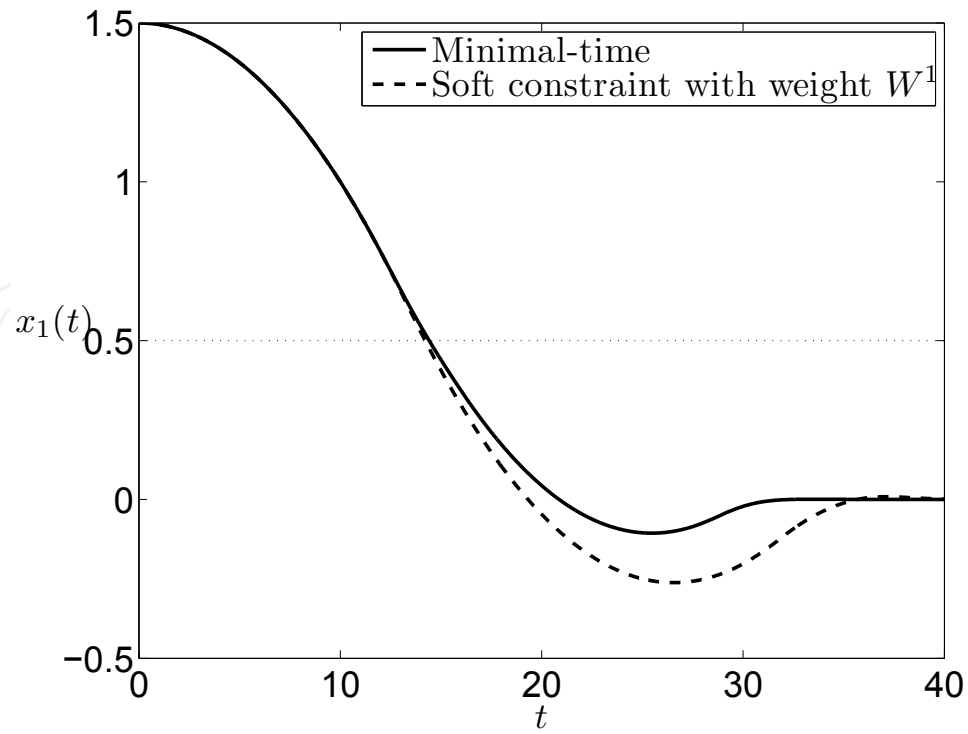


Fig. 2. Position (x_1) with constraint relaxation.

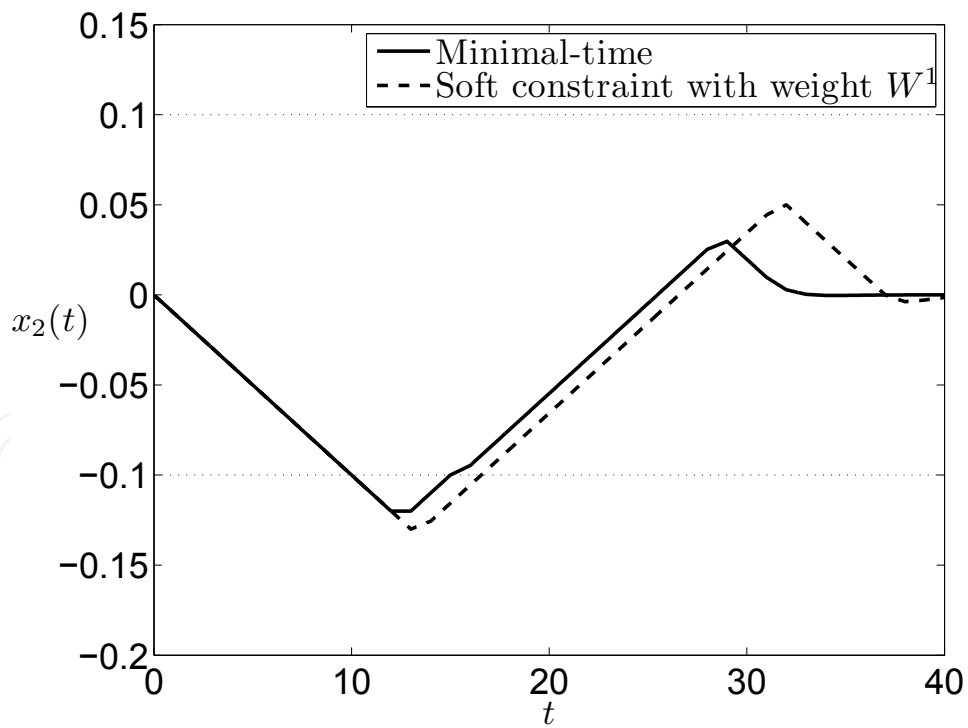


Fig. 3. Velocity (x_2) with constraint relaxation.

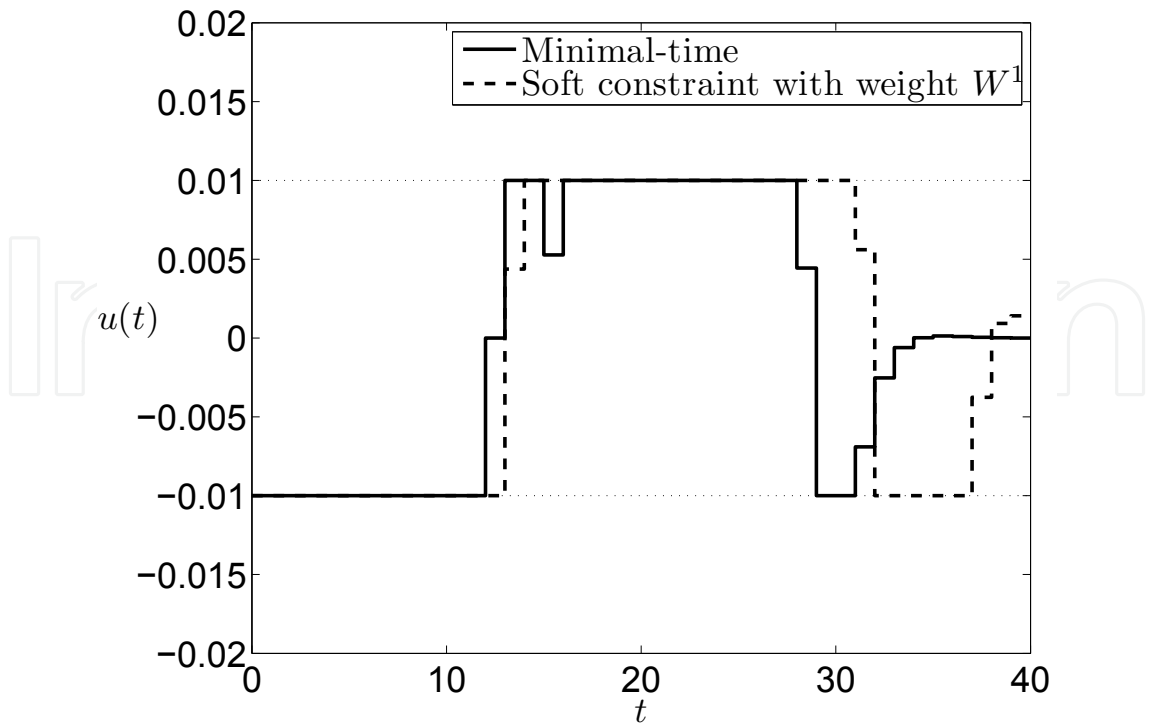


Fig. 4. Acceleration (u) with constraint relaxation.

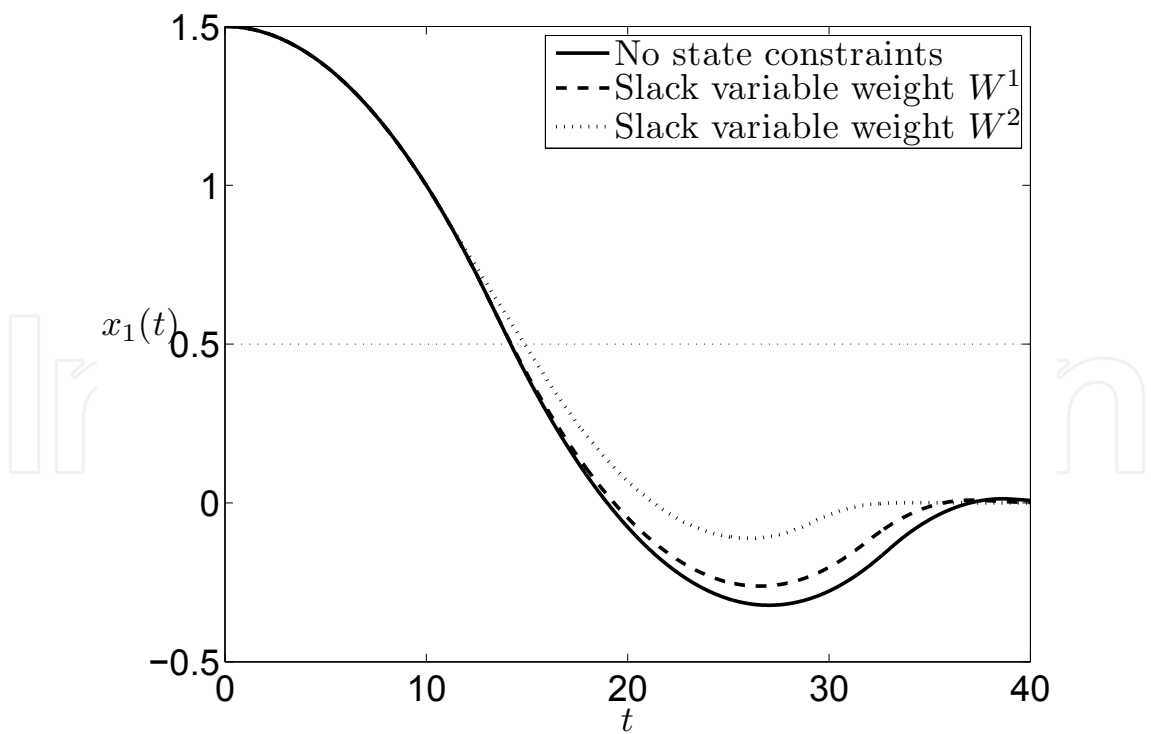


Fig. 5. Position (x_1) without state constraints and with soft constraint relaxation.

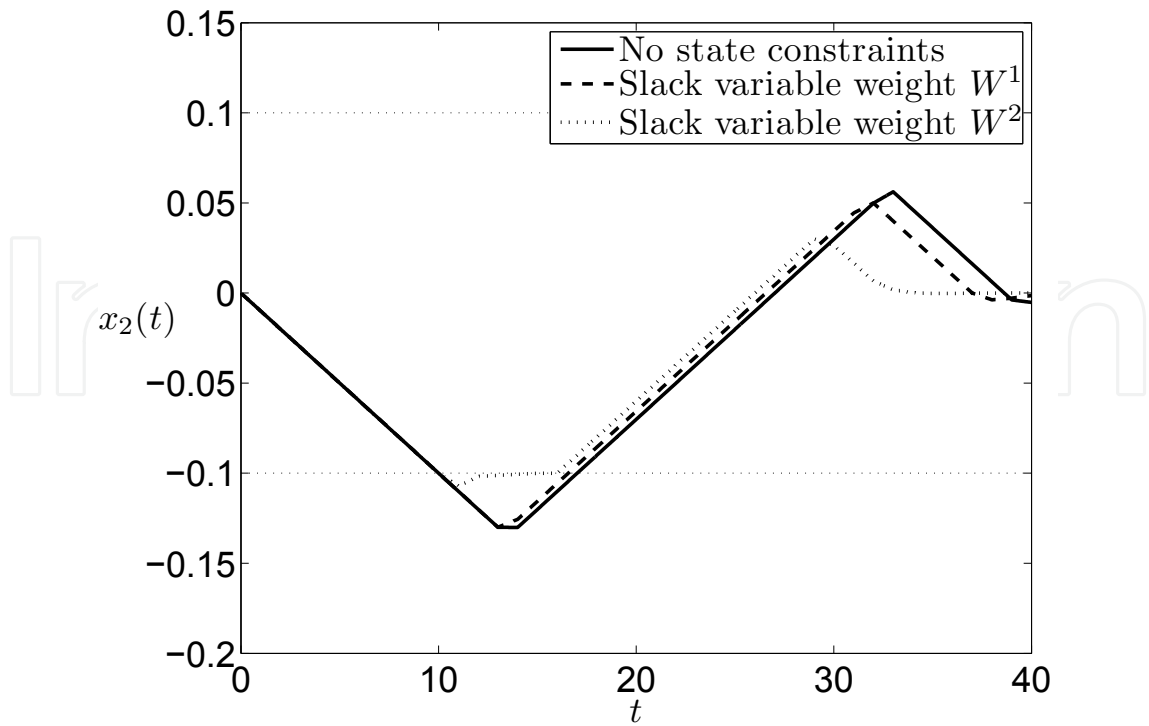


Fig. 6. Velocity (x_2) without state constraints and with soft constraint relaxation.

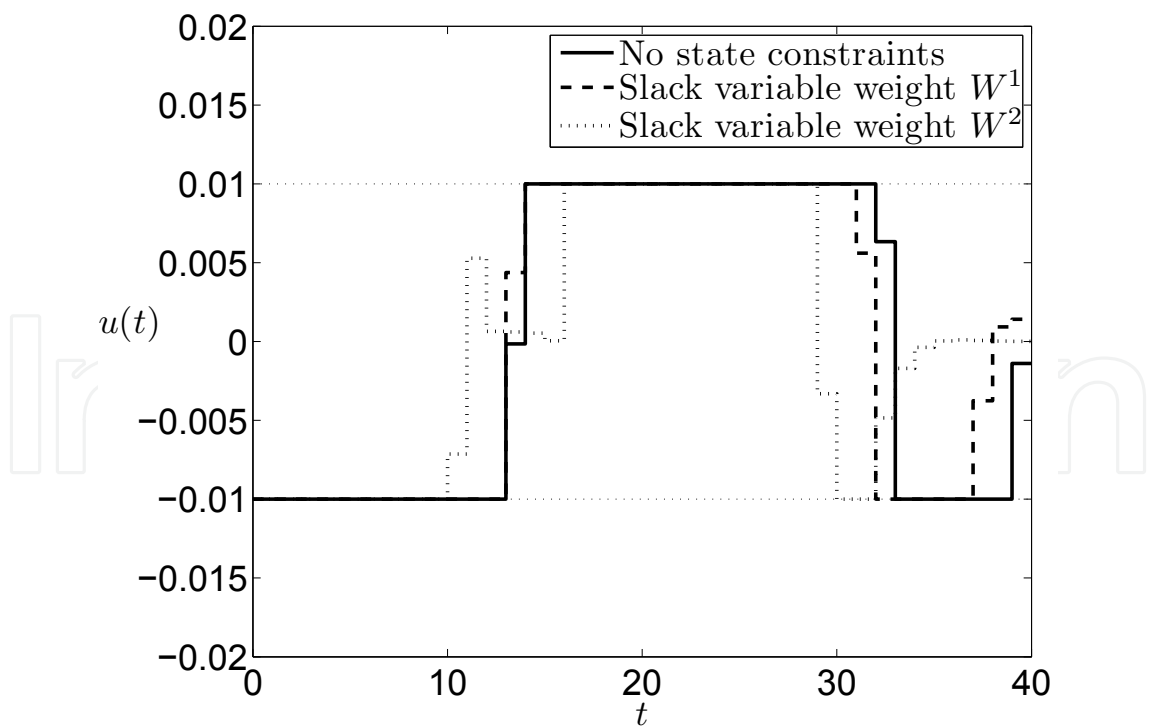


Fig. 7. Acceleration (u) without state constraints and with soft constraint relaxation.

4. Setpoint management approaches

The main idea behind setpoint management schemes is to find a new setpoint $x'_{ref}(k) = x_{ref}(k) - C\mu$ at each time k in order to make the problem feasible and to progressively steer the system state towards the original setpoint x_{ref} . $\mu \in \mathbb{R}^n$ is the setpoint management variable and $C \in \mathbb{R}^{q \times n}$ is a constant matrix. It is worth noting that, in the general case, changing the setpoint x_{ref} would also affect the corresponding setpoint u_{ref} for the control. As a result, the bounds on the control u would need to be changed, which would require the online recalculation of the terminal constraint set. Therefore, the class of systems considered in this study are restricted to those which require no adjustment in the control setpoint after a change in the state setpoint. This is a property of plants with integral behavior.

It is worth noting that these setpoint modifications impose a need of redetermination of the MAS every time the value of μ changes. The approach presented in the following subsection introduces a parameterization of the MAS in terms of the possible values of μ , avoiding the necessity to repeat the determination of the terminal set online.

4.1 Parameterization of the MAS

The parameterization of the MAS may be carried out through the employment of an augmented state vector \bar{x} defined as (Almeida & Leissing, 2010)

$$\bar{x} = \begin{bmatrix} x \\ \mu \end{bmatrix}, \quad (26)$$

which evolves inside the MAS according to

$$\bar{x}(k+1) = \bar{A}\bar{x}(k), \quad \bar{A} = \begin{bmatrix} A - BK & 0 \\ 0 & I_n \end{bmatrix}. \quad (27)$$

It is worth noting that the identity matrix $I_n \in \mathbb{R}^{n \times n}$ multiplies the additional components of the state because these are supposed to remain constant along the prediction horizon. Although \bar{A} has eigenvalues in the border of the unit circle (eigenvalues at +1 associated to the matrix I_n), it is still possible to determine the MAS in a finite number of steps because the dynamics given by Eq. (27) is stable in the Lyapunov sense (Gilbert & Tan, 1991).

The state constraints are altered by the management variable μ in the following fashion:

$$x_{P,min} - x_{ref} + C\mu \leq x \leq x_{P,max} - x_{ref} + C\mu \quad (28)$$

where C is a matrix that relates the vector $\mu \in \mathbb{R}^n$ of setpoint management variables to the corresponding component of the state vector $x \in \mathbb{R}^n$ whose setpoint is managed.

In order to incorporate the constraints to the parameterization, an auxiliary output variable \bar{z} may be defined as

$$\bar{z} = \begin{bmatrix} x - C\mu \\ -x + C\mu \end{bmatrix} \quad (29)$$

which is subject to the following constraints:

$$\bar{z} \leq \begin{bmatrix} x_{P,max} - x_{ref} \\ x_{ref} - x_{P,min} \end{bmatrix} \quad (30)$$

Since $u = -Kx$ inside the MAS, the output function for the determination of the MAS becomes $\bar{z} = \bar{C}\bar{x}$ with

$$\bar{C} = \begin{bmatrix} I_n & -C \\ -I_n & C \end{bmatrix} \quad (31)$$

Having determined the MAS (\bar{O}_∞) associated to the dynamics of Eq. (27) with the constraints of Eq. (30), it can be particularized online by fixing the value of μ . The set \bar{O}_∞ obtained is invariant regarding matrix \bar{A} . It is convenient to note that the terminal constraint $\hat{x}(k+N|k) \in \bar{O}_\infty$ for a particular choice of μ can replace the constraints from $i = N$ onwards in Eqs. (14) and (15). Imposing $\hat{x}(k+N|k) \in \bar{O}_\infty$ is equivalent to imposing the constraints $\hat{u}(k+i|k) \in \mathbb{U}$ and $\hat{x}(k+i|k) \in \mathbb{X}$ until $i = N + t^*$, with t^* obtained during the offline determination of the parameterized MAS. Therefore, the infinite set of constraints of Eqs. (14) and (15) is reduced to a finite one.

4.2 Optimization problem formulation

Considering the setpoint management, the optimization problem to be solved at time k now involves \hat{V} and μ as decision variables.

Thus, the optimization problem becomes

$$\min_{\hat{V}, \mu} \hat{V}^T \Psi \hat{V} + \mu^T W_\mu \mu \quad (32)$$

s.t.

$$\begin{bmatrix} H_U \\ -H_U \\ H \\ -H \end{bmatrix} \hat{V} \leq \begin{bmatrix} [u_{max} - u_{ref}]_{N+t^*+1} - \Phi_U(x_P(k) - x_{ref} + C\mu) \\ \Phi_U(x_P(k) - x_{ref} + C\mu) - [u_{min} - u_{ref}]_{N+t^*+1} \\ [x_{P,max} - x_{ref} + C\mu]_{N+t^*} - \Phi(x_P(k) - x_{ref} + C\mu) \\ \Phi(x_P(k) - x_{ref} + C\mu) - [x_{P,min} - x_{ref} + C\mu]_{N+t^*} \end{bmatrix}$$

where W_μ is a positive-definite weight matrix, the operator $[\bullet]_j$ stacks j copies of vector \bullet , and H , H_U , Φ and Φ_U are in accordance with Eq. 20.

The greater the weights in W_μ in comparison to Ψ , the closer the solution is to the one obtained without the need of setpoint management.

After the solution of the optimization problem of Eq. (32), the control signal to be applied to the plant is given by

$$u_P(k) = u_{ref} + \hat{v}^*(k|k) - K(x_P(k) - x_{ref} + C\mu^*) \quad (33)$$

4.3 Simulation example

The simulation scenario employed in this example is the same as that of subsection 3.4. Only the constraints over the position variable are different ($-1 \leq x_1 \leq 1$). The determination of the MAS leads to $t^* = 7$ and M remains equal to 7. Therefore, the constraint horizon in order to guarantee that the constraints are enforced over an infinite horizon is $N = M + t^* = 14$.

The initial state is $x_0 = [1 \ 0]^T$, which respects the constraints. However, the problem is infeasible, making the employment of a technique to recover feasibility mandatory. The procedure described in this section can be used to recover feasibility. The setpoint of the position is chosen for management, meaning that $\mu \in \mathbb{R}$ and

$$C = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad (34)$$

It is desirable to keep the setpoint management as close to zero as possible. To this end, the weight of the setpoint management variable is chosen as $W_\mu = 1000$.

Figure 8 shows the position variable, which starts at the edge of the constraint and is steered to the origin without violating the constraints.

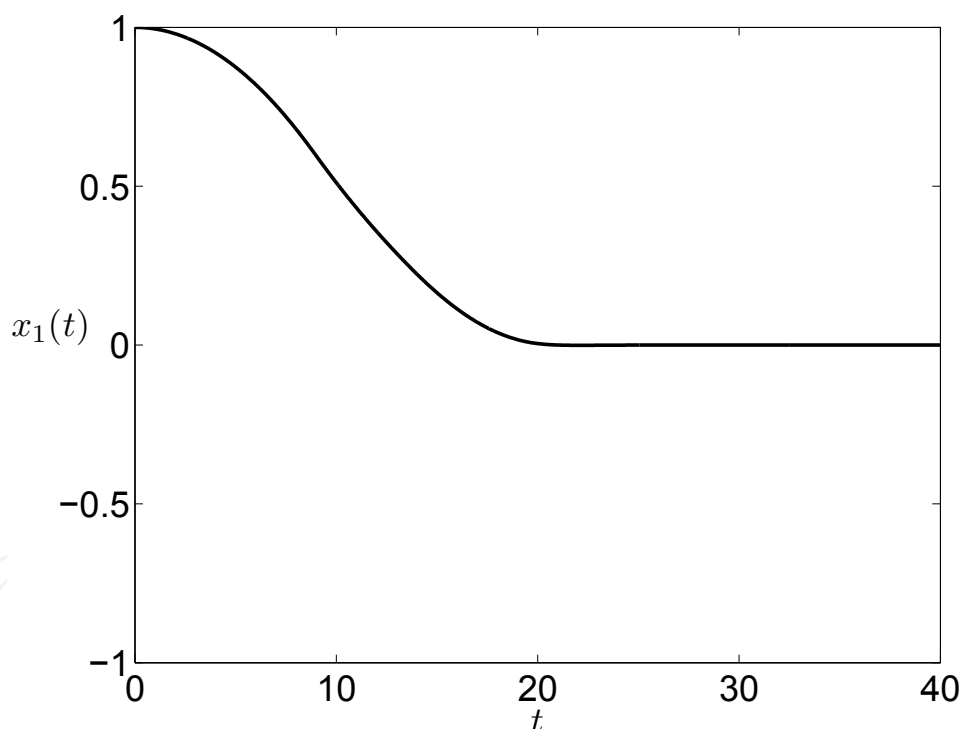


Fig. 8. Position (x_1) with setpoint management.

It can be seen in Fig. 9 that the velocity variable gets close to its lower bound (-0.1), but this constraint is also satisfied. Figure 10 shows that the constraints on the acceleration are active in the beginning of the maneuver, but are not violated.

The setpoint management variable μ is shown in Fig. 11. It can be seen that the management technique is applied up to time $t = 10$. This time coincides with the change in the acceleration from negative to positive.

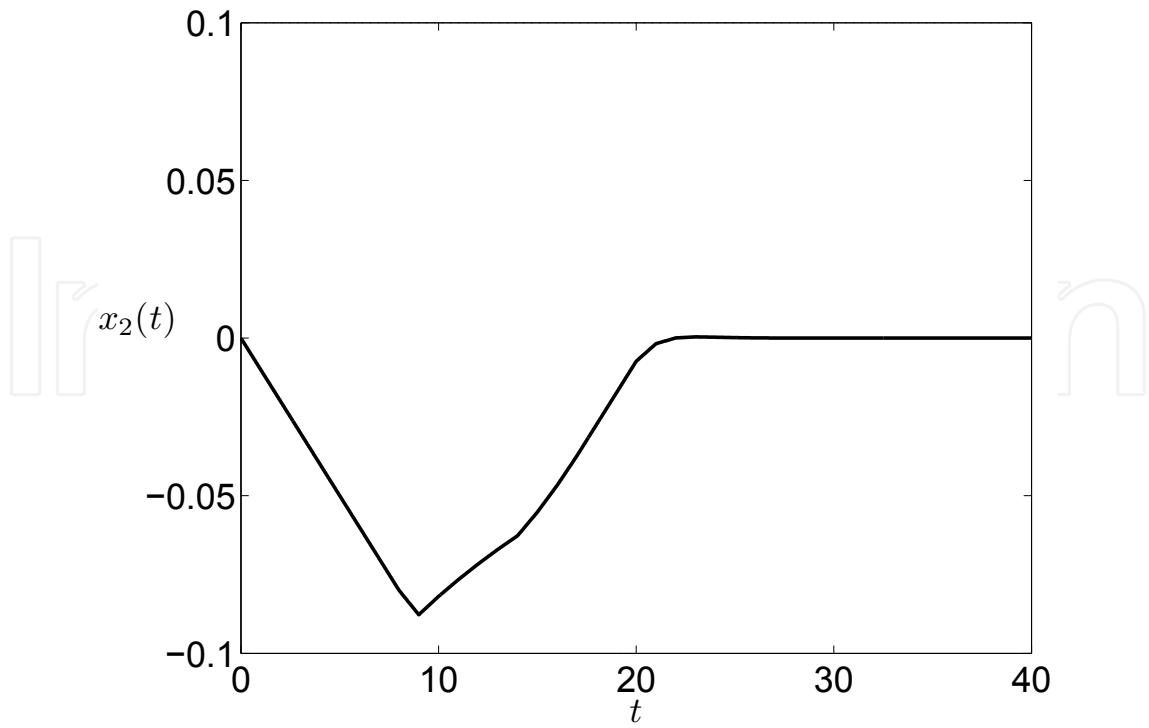


Fig. 9. Velocity (x_2) with setpoint management.

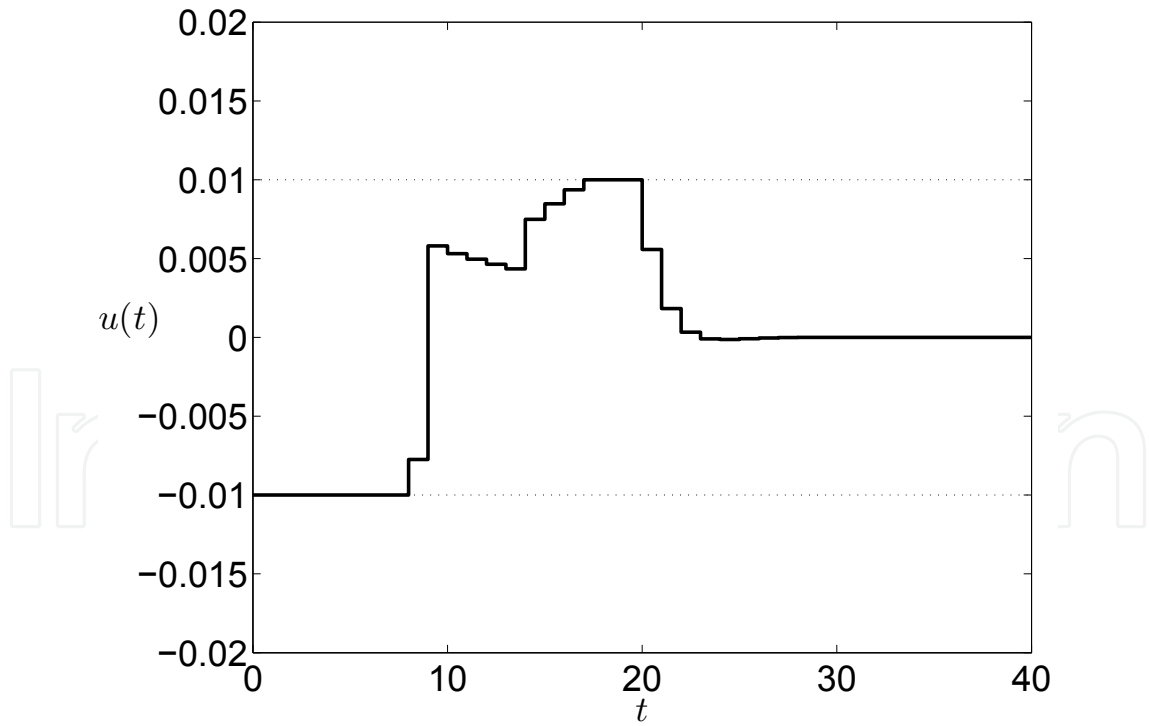


Fig. 10. Acceleration (u) with setpoint management.

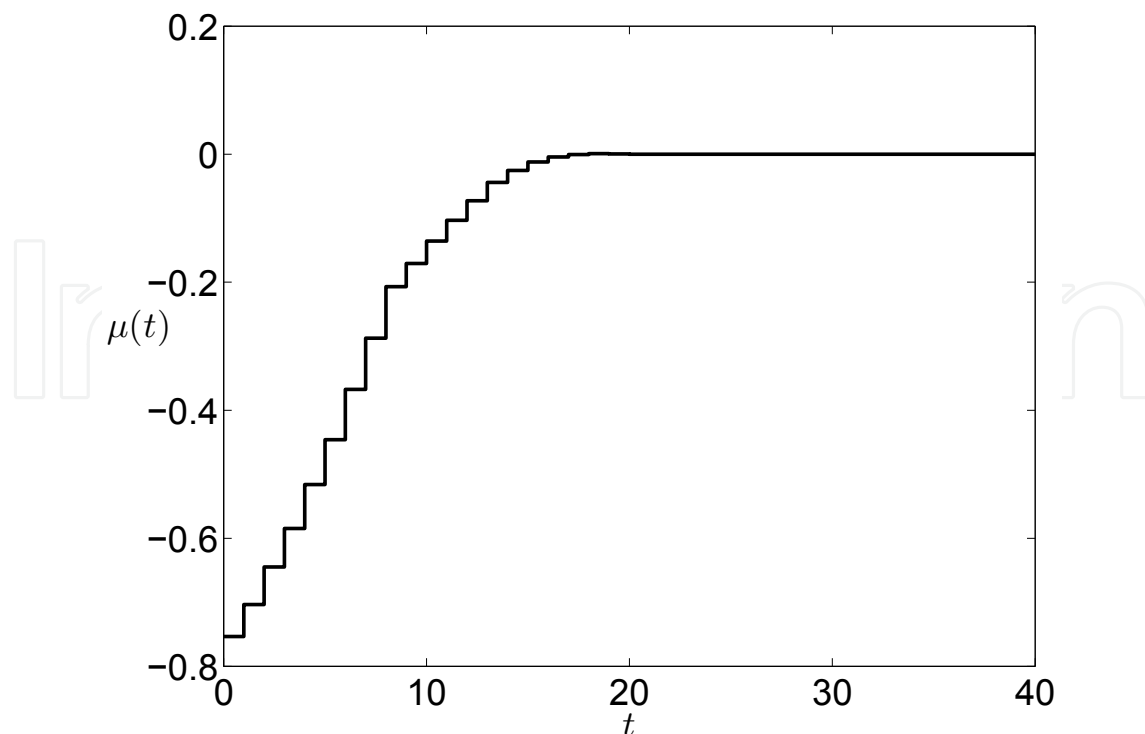


Fig. 11. Position setpoint management variable (μ).

5. Conclusions

In real applications of MPC controllers, noise, disturbances, model-plant mismatches and faults are commonly found. Therefore, infeasibility of the associated optimization problem can be a recurrent issue. This justifies the study of techniques capable of driving the system to a feasible region, since infeasibility may cause prediction errors, deployment of impracticable control sequences and instability of the control loop. Computational workload is also of great concern in real applications, thus the adopted techniques must be simple enough to be executed in a commercial off-the-shelf computer within the sample period and effective enough to make the problem feasible. In this chapter a review of the literature regarding feasibility issues was presented and two of the more widely adopted approaches (constraint relaxation and setpoint management) were described. Simulation examples of some illustrative techniques were presented in order to clarify the advantages, drawbacks and difficulties in implementation of some techniques.

6. Acknowledgements

The authors acknowledge the financial support of FAPESP (MSc scholarship 2009/12674-0) and CNPq (research fellowship).

7. References

Afonso, R. J. M. & Galvão, R. K. H. (2010a). Controle preditivo com garantia de estabilidade nominal aplicado a um helicóptero com três graus de liberdade empregando relaxamento de restrições de saída (Predictive control with nominal stability guarantee applied to a helicopter with three degrees of freedom employing

- output constraint relaxation - text in portuguese), *Proc. XVIII Congresso Brasileiro de Automática*, pp. 1797 – 1804.
- Afonso, R. J. M. & Galvão, R. K. H. (2010b). Predictive control of a helicopter model with tolerance to actuator faults, *Proc. Conf. Control and Fault-Tolerant Systems (SysTol)*, 2010, pp. 744 – 751.
- Almeida, F. A. & Leissling, D. (2010). Fault-tolerant model predictive control with flight-test results, *J. Guid. Control Dyn.* 33(2): 363 – 375.
- Alvarez, T. & de Prada, C. (1997). Handling infeasibilities in predictive control, *Computers & chemical engineering* 21: S577 – S582.
- Bemporad, A., Casavola, A. & Mosca, E. (1997). Nonlinear control of constrained linear systems via predictive reference management, *IEEE Trans. Automatic Control* 42(3): 340 – 349.
- Bemporad, A. & Mosca, E. (1994). Constraint fulfilment in feedback control via predictive reference management, *Proc. 3rd IEEE Conf. Control Applications*, Glasgow, UK, pp. 1909 – 1914.
- Bryson, A. E. & Ho, Y.-C. (1969). *Applied Optimal Control*, Blaisdell, Waltham, MA.
- Chisci, L., Rossiter, J. A. & Zappa, G. (2001). Systems with persistent disturbances: predictive control with restricted constraints, *Automatica* 37(7): 1019–1028.
- Franklin, G., Powell, J. & Emami-Naeini, A. (2005). *Feedback Control of Dynamic Systems*, 5th edn, Prentice Hall, Upper Saddle River, NJ.
- Gilbert, E. G. & Kolmanovsky, I. (1995). Discrete-time reference governors for systems with state and control constraints and disturbance inputs, *Proc. 34th IEEE Conference on Decision and Control*.
- Gilbert, E. G. & Tan, K. T. (1991). Linear systems with state and control constraints: the theory and application of maximal output admissible sets, *IEEE Trans. Automatic Control* 36(9): 1008–1020.
- Kapasouris, P., Athans, M. & Stein, G. (1988). Design of feedback control systems for stable plants with saturating actuators, *Proc. 27th IEEE Conference on Decision and Control*.
- Kouvaritakis, B., Rossiter, J. A. & Cannon, M. (1998). Linear quadratic feasible predictive control, *Automatica* 34(12): 1583–1592.
- Limon, D., Alvarado, I., Alamo, T. & Camacho, E. (2008). MPC for tracking piecewise constant references for constrained linear systems, *Automatica* 44(9): 2382–2387.
- Maciejowski, J. M. (2002). *Predictive Control with Constraints*, 1st edn, Prentice Hall, Harlow, England.
- Montandon, A. G., Borges, R. M. & Henrique, H. M. (2008). Experimental application of a neural constrained model predictive controller based on reference system, *Latin American applied research* 38: 51 – 62.
- Rawlings, J. & Muske, K. (1993). The stability of constrained receding horizon control, *IEEE Trans. Automatic Control* 38(10): 1512–1516.
- Rodrigues, M. A. & Odloak, D. (2005). Robust mpc for systems with output feedback and input saturation, *Journal of Process Control* 15: 837 – 846.
- Rossiter, J. A. (2003). *Model-based Predictive Control: a practical approach*, 1st edn, CRC Press, Boca Raton.
- Scokaert, P. (1994). *Constrained Predictive Control*, PhD thesis, Univ. Oxford, UK.
- Scokaert, P. & Rawlings, J. (1998). Constrained linear quadratic regulation, *IEEE Trans. Automatic Control* 43(8): 1163–1169.

- Scokaert, P. & Rawlings, J. (1999). Feasibility issues in linear model predictive control, *AIChE Journal* 45(8): 1649 – 1659.
- Vada, J., Slupphaug, O., Johansen, T. & Foss, B. (2001). Linear mpc with optimal prioritized infeasibility handling: application, computational issues and stability, *Automatica* 37(11): 1835 – 1843.
- Zafiriou, E. & Chiou, H. (1993). Output constraint softening for siso model predictive control, *American Control Conference*.

IntechOpen

IntechOpen



Frontiers of Model Predictive Control

Edited by Prof. Tao Zheng

ISBN 978-953-51-0119-2

Hard cover, 156 pages

Publisher InTech

Published online 24, February, 2012

Published in print edition February, 2012

Model Predictive Control (MPC) usually refers to a class of control algorithms in which a dynamic process model is used to predict and optimize process performance, but it is can also be seen as a term denoting a natural control strategy that matches the human thought form most closely. Half a century after its birth, it has been widely accepted in many engineering fields and has brought much benefit to us. The purpose of the book is to show the recent advancements of MPC to the readers, both in theory and in engineering. The idea was to offer guidance to researchers and engineers who are interested in the frontiers of MPC. The examples provided in the first part of this exciting collection will help you comprehend some typical boundaries in theoretical research of MPC. In the second part of the book, some excellent applications of MPC in modern engineering field are presented. With the rapid development of modeling and computational technology, we believe that MPC will remain as energetic in the future.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Rubens Junqueira Magalhães Afonso and Roberto Kawakami Harrop Galvão (2012). Infeasibility Handling in Constrained MPC, *Frontiers of Model Predictive Control*, Prof. Tao Zheng (Ed.), ISBN: 978-953-51-0119-2, InTech, Available from: <http://www.intechopen.com/books/frontiers-of-model-predictive-control/infeasibility-handling-in-constrained-mpc>

INTECH
open science | open minds

InTech Europe

University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China

Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2012 The Author(s). Licensee IntechOpen. This is an open access article distributed under the terms of the [Creative Commons Attribution 3.0 License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

IntechOpen

IntechOpen