SEVENTH FRAMEWORK PROGRAMME
THEME – ICT
[Information and Communication Technologies]

SEVENTH FRAMEWORK PROGRAMME

Contract Number:	223854
Project Title:	Hierarchical and Distributed Model Predictive Control of Large-Scale Systems
Project Acronym:	HD-MPC

HD−MPC

Deliverable Number:	D3.2.1
Deliverable Type:	Report
Contractual Date of Delivery:	01/09/2009
Actual Date of Delivery:	**28/08/2009**
Title of Deliverable:	**Report on literature survey and analysis of (optimization) methods for robust distributed MPC**
Dissemination level:	Public
Workpackage contributing to the Deliverable:	WP3 - WP4
WP Leader:	Wolfgang Marquardt - Moritz Diehl
Partners:	TUD, POLIMI, RWTH, USE, UNC, UWM, SUPELEC, KUL
Author(s):	Carlo Savorgnan, Moritz Diehl

Table of contents

Project co-ordinator

Name: Bart De Schutter
Address: Delft Center for Systems and Control
Delft University of Technology
Mekelweg 2, 2628 Delft, The Netherlands
Phone Number: +31-15-2785113
Fax Number: +31-15-2786679
E-mail: b.deschutter@dcsc.tudelft.nl

Project web site: `http://www.ict-hd-mpc.eu`

Executive Summary

In this report we give an overview of the methods that can be used in robust distributed MPC. The literature on this subject is limited but we show how some methods from robust MPC and distributed optimization can be combined to obtain new distributed robust MPC schemes.

Chapter 1

Introduction

In this chapter we review the basics of standard and robust MPC.

1.1 Model predictive control

Consider a system described by the difference equation

$$x_{t+1} = \phi(x_t, u_t), \quad t = 0, 1, 2, \dots \tag{1.1}$$

where t, $x_t \in \mathbb{R}^n$ and $u_t \in \mathbb{R}^m$ represents the time, state and input, respectively. For a given initial condition $\bar{x}_0$, we consider the optimal control problem (OCP) of finding a control sequence $\{u_t\}_{t=0}^{\infty}$ that minimizes the cost

$$\sum_{t=0}^{\infty} \ell_t(x_t, u_t), \tag{1.2}$$

subject to the constraints

$$x_t \in \mathscr{X} \quad \text{and} \quad u_t \in \mathscr{U}, \quad \forall t = 0, 1, 2, \dots. \tag{1.3}$$

This problem is in general hard to solve due to the fact that the control horizon is infinite. MPC solves this problem by solving iteratively a finite horizon approximation of this OCP. The method can be summarized as follows

1. Choose a prediction horizon N;
2. Measure the current value of the state $\bar{x}$;
3. Solve the optimization problem

$$\begin{array}{rl} \min_{\substack{x_0,\dots,x_N \\ u_0,\dots,u_{N-1}}} & \sum_{t=0}^{N} \ell_t(x_t, u_t) \\ \text{s.t.} & x_0 = \bar{x} \\ & x_{t+1} = \phi(x_t, u_t) \;\; \forall t = 0, \dots, N-1 \\ & x_t \in \mathscr{X} \;\; \forall t = 0, \dots, N \\ & u_t \in \mathscr{U} \;\; \forall t = 0, \dots, N-1 \end{array} \tag{1.4}$$

4. Apply the computed value of the input u_0;
5. Go to step (2).

Before proceeding with the illustration of robust MPC we should underline two important facts.

- Applying MPC gives exactly the same result that we would obtain using the control derived with dynamic programming applied to the finite horizon approximation of the original OCP. The advantage of MPC lies in the fact it requires the solution of the finite dimensional optimization problem (1.4), while dynamic programming requires the solution of a problem which is usually infinite dimensional.

- MPC easily deals with state and input constraints in a seamless way. This constitutes a big advantage with respect to many other control techniques.

Remark 1 *The OCP considered and the MPC algorithm presented are basic. However, by modifying problem* (1.4)*, we can obtain better MPC formulations which can guarantee constraint satisfaction and stability (see, e.g., the survey [16]).*

1.2 Robust model predictive control

In practical control applications the system dynamics is not always known exactly. To model this fact we can add to the system model the uncertainty $w \in \mathbb{R}^p$:

$$x_{t+1} = \phi(x_t, u_t, w_t), \quad t = 0, 1, 2, \ldots \tag{1.5}$$

The uncertainty is assumed to take values in a compact set $\mathscr{W} \subset \mathbb{R}^p$.

In the optimization context it has been shown that little uncertainties in the problem data can cause a significant violation of the constraints (see [1] and the references therein). Therefore, including the information about the uncertainty w_t in the design of the input sequence $\{u_t\}_{t=0,\ldots,N-1}$ is of paramount importance for the MPC framework.

There are several papers dealing with robust versions of MPC. An account of the methods in the literature can be found in the books [14, 6, 19]. The easiest approach, which is commonly referred to as *open-loop MPC*, can be seen as a minor modification to the algorithm presented for certain systems in Section 1.1. In open-loop MPC, one identifies a nominal system dynamics

$$\tilde{x}_{t+1} = \tilde{\phi}(\tilde{x}_t, u_t) = \phi(\tilde{x}_t, u_t, \tilde{w}), \quad t = 0, 1, 2, \ldots. \tag{1.6}$$

obtained for specific value of the uncertainty $\tilde{w} \in \mathscr{W}$. The cost function is then minimized for the nominal trajectory $\{\tilde{x}_t\}_{t=0,\ldots,N-1}$, while the state and input constraints are satisfied for every possible uncertainty sequence $\{w_t\}_{t=0,\ldots,N-1} \in \mathscr{W}$. Some of the early results on robustness of nominal open-loop MPC can be found in [7, 22, 15]. A good survey about this subject is [16].

A significant drawback of open-loop MPC is that the choice of the control sequence is suboptimal and results in poor performance. This is due to the fact that this method doesn't take into account the fact that the real state is measured for every value of t and this allows to counteract the effect of the uncertainty. This is also the reason why open-loop MPC and dynamic programming are not equivalent when the system is uncertain.

To overcome the limitations of open-loop MPC, we can modify the optimization problem solved online such that we do not determine a control sequence but a sequence of control policies. The methodology obtained can be referred to as *feedback MPC* [10, 13, 21]. It is important noticing that there is a fundamental difference between calculating a classical feedback control law and using feedback MPC. While in the former we look for a control law which assigns an input value to *every* admissible state, in the latter we are interested in a control law which is defined only for the states on the trajectories that originated from the initial condition and correspond to an uncertainty sequence $\{w_t\}_{t=0,\dots,N-1} \in \mathscr{W}$. In other words, assuming that we can keep the real trajectory of the system close to the nominal trajectory, we want to find a control law which assigns an input value only for the states in a neighborhood of the nominal trajectory. This is the reason which makes feedback MPC more interesting from a computational point of view.

Although feedback MPC can be seen as a more viable alternative to classical feedback control, the optimization problem which should be solved iteratively inside the MPC loop is still computationally prohibitive. In fact, optimizing over (local) control policies is in general an infinite dimensional optimization problem. In practice, a common approach is to use a parametrization of the control policy around the nominal trajectory. Although this may be restrictive, this has shown to perform well in practice.

In the next subsection we illustrate the basics of feedback MPC for linear systems with additive uncertainty.

1.2.1 Feedback MPC for linear systems with additive uncertainty

Consider the uncertain system

$$x_{t+1} = Ax_t + Bu_t + w_t \tag{1.7}$$

where the state and uncertainty vectors are assumed to have the same dimension, $A \in \mathbb{R}^{n\times n}$ and $B \in \mathbb{R}^{n\times m}$ are the state and input matrices. We assume that the uncertainty set $\mathscr{W}$ is compact and contains the origin.

Since $0 \in \mathscr{W}$, we can define the nominal trajectory as

$$\tilde{x}_{t+1} = A\tilde{x}_t + B\tilde{u}_t \tag{1.8}$$

where $\tilde{u}_t$ is the nominal input. Defining the state error $e_t = x_t - \tilde{x}$ and $\hat{u}_t = u_t - \tilde{u}_t$ we obtain the following equation

$$x_{t+1} = \tilde{x}_{t+1} + e_{t+1} = A(\tilde{x}_t + e_t) + B(\tilde{u}_t + \hat{u}_t) + w_t \tag{1.9}$$

and therefore

$$e_{t+1} = Ae_t + B\hat{u}_t + w_t. \tag{1.10}$$

Thanks to the system linearity and the additivity of the uncertainty the evolution of $\tilde{x}_t$ and e_t can be seen as independent and we can calculate the input components $\tilde{u}_t$ and $\hat{u}_t$ so that we obtain a good nominal trajectory with the former and we minimize the effect of the uncertainty with the latter. Since the value of w_t is unknown a priory, in order to minimize the effect of w_t, we cannot use an open-loop input sequence $\{\hat{u}_t\}_{t=0,\dots,N-1}$ but we need to compute a sequence of feedback controls $\hat{u}_t = \gamma_t(e_t)$. As we have already mentioned above, optimizing over generic feedback laws is not computationally tractable and since the system is linear, a good compromise is the choice of linear feedback $\hat{u}_t = K_t e_t$. Unfortunately, also after this simplification the optimization problem to be solved can be computationally too demanding (the number of variables corresponding to the matrices K_t are $n \times m \times N$) and in practice it is easier to precompute only one gain matrix K (see [8] for another interesting approach). In this case, the dynamics governing the error e_t is

$$e_{t+1} = (A + BK)e + w_t. \tag{1.11}$$

If K is chosen such that the spectral radius of $A+BK$ is smaller than 1, there exists a bounded set $\mathscr{E}$ such that $e_t \in \mathscr{E}$ $\forall t$ (this is due to the fact that the uncertainty set $\mathscr{W}$ is bounded). The set $\mathscr{E}$ is said to robustly invariant for the system (1.11) (for the details on the computation of this set see the book [5] and the references therein). Since $0 \in \mathscr{W}$, it follows that $0 \in \mathscr{E}$.

Define the following set operations ($\mathscr{Z} \subseteq \mathbb{R}^n$ and $\mathscr{Y} \subseteq \mathbb{R}^n$):

- set addition
$$\mathscr{Y} \oplus \mathscr{Z} := \{y+z |\ y \in \mathscr{Y} \text{ and } z \in \mathscr{Z}\}; \tag{1.12}$$

- set subtraction
$$\mathscr{Y} \ominus \mathscr{Z} := \{x \in \mathbb{R}^n |\ x \oplus \mathscr{Z} \subseteq \mathscr{Y}\}. \tag{1.13}$$

Since $e_t \in \mathscr{E}\ \forall t$ and the dynamics of the system (1.7), it easy to conclude that the trajectory $\{x_t\}_{t=0,\dots,N}$ is such that $x_t \in \tilde{x} \oplus \mathscr{E}$ for all $t = 0,\dots,N$. In other words, the real trajectory is contained in a tube centered around the nominal trajectory (the concept of tubes in control was introduced in [2, 3]). We can therefore state that if the nominal trajectory is such that

$$\tilde{x}_t \in \mathscr{X} \ominus \mathscr{E} \quad \forall t = 0,\dots,N \tag{1.14}$$

then the real trajectory satisfies the state constraint $x_t \in \mathscr{X}\ \forall t = 0,\dots,N$. Also the input constraint $u_t \in \mathscr{U}$ can be rephrased in terms of the nominal input $\tilde{u}$:

$$\tilde{u}_t \in \mathscr{U} \ominus K\mathscr{E} \quad \forall t = 0,\dots,N \tag{1.15}$$

where $K\mathscr{E} = \{Ke |\ e \in \mathscr{E}\}$. Therefore, to obtain a robust MPC method we can solve online the same optimization problem (1.4) where we substitute the real trajectory and input sequence with the nominal one. The constraint sets $\mathscr{X}$ ans $\mathscr{U}$ should be substituted with the sets $\mathscr{X} \ominus \mathscr{E}$ and $\mathscr{U} \ominus K\mathscr{E}$ which can be computed offline.

1.2.2 Feedback MPC for general uncertain systems

In the previous subsection we have seen how a feedback MPC algorithm can be used to deal with the robust control of linear systems with additive uncertainty. It is important noticing that robustness with respect to constraint satisfaction has been obtained without requiring the online solution of an optimization problem which is more difficult than the one that should be solved for the same system if there was no uncertainty. This technique base on tubes can be extended also to other kinds of systems. The interested reader is referred to [19, Chapter 3], for a detailed illustration of feedback MPC with tubes for nonlinear systems and linear systems with parametric uncertainty (the methods in the book are based on [11, 17]).

Chapter 2

Robust distributed MPC

In the previous chapter we have illustrated some basic concepts of standard and robust MPC. In this chapter we consider networked systems and we discuss how robust MPC can be decentralized to cope with these systems.

2.1 Networked systems

Consider an uncertain networked system composed by M interconnected subsystems each one described by a difference equation of the form

$$x_{t+1}^i = \phi^i(x_t^j, u_t^j, w_t^i; j \in \mathscr{N}_i), \quad t = 0, 1, 2, \dots \tag{2.1}$$

where $x_t^i \in \mathbb{R}^{n_i}$, $u_t^i \in \mathbb{R}^{m_i}$ and $w_t^i \in \mathbb{R}^{p_i}$ represent the state, the input and the uncertainty of the subsystem i at time t. We assume that w_t^i takes values in the compact set $\mathscr{W}^i \subset \mathbb{R}^{p_i}$. The index set $\mathscr{N}^i$ contains the index i and all the indices of the subsystems which interact with the subsystem i. Consider for example the networked system in Figure 2.1 where the arrows indicate interaction between the subsystems

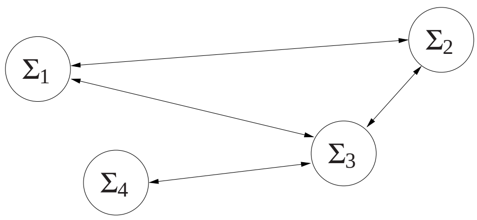

Figure 2.1: An example of networked systems.

$\Sigma_1, \Sigma_2, \Sigma_3, \Sigma_4$. If we consider Σ_4 we have $\mathscr{N}^4 = \{3, 4\}$ and therefore

$$x_{t+1}^4 = \phi^4(x_t^3, x_t^4, u_t^3, u_t^4, w_t^4). \tag{2.2}$$

Denote by x_t and u_t the state and the input of the centralized system:

$$x_t = \left[x_t^{1\,T} \dots\ x_t^{M\,T} \right]^T, \qquad u_t = \left[u_t^{1\,T} \dots\ u_t^{M\,T} \right]^T. \tag{2.3}$$

Denote by $\sum_{t=0}^{\infty} \ell_t(x_t, u_t)$ the cost which should be minimized in the given optimal control problem and by $x_t \in \mathscr{X}$ and $u_t \in \mathscr{U}$ the constraints on the state and input vectors. Depending on the problem considered we can have different notions of separability:

- the cost is separable if there exist a family of functions $\{\ell_t^i(\cdot,\cdot)\}_{i=1,\ldots,M}$ such that

$$\ell_t(x_t,u_t) = \sum_{i=1}^{M} \ell_t^i(x_t^i,u_t^i); \tag{2.4}$$

- the constraint set $\mathscr{X}$ is separable if there exist family of sets $\{\mathscr{X}^i\}_{i=1,\ldots,M}$ such that

$$(x_t \in \mathscr{X}) \iff (x_t^i \in \mathscr{X}^i \quad \forall i = 1,\ldots,M) \tag{2.5}$$

  where $\mathscr{X}^i \subseteq \mathbb{R}^{n_i}$;

- the constraint set $\mathscr{U}$ is separable if there exist family of sets $\{\mathscr{U}^i\}_{i=1,\ldots,M}$ such that

$$(u_t \in \mathscr{U}) \iff (u_t^i \in \mathscr{U}^i \quad \forall i = 1,\ldots,M) \tag{2.6}$$

  where $\mathscr{U}^i \subseteq \mathbb{R}^{n_i}$.

It is clear that when the subsystem dynamics are decoupled ($\mathscr{N}^i = \{i\}$) and the cost and the constraint sets are separable, the centralized MPC problem can trivially decomposed and solved separately. In the next section we shall consider problems where there is some coupling between the subsystems.

2.2 Robust distributed MPC for networked systems

The scientific literature on robust distributes MPC is quite limited. This can be explained by the fact that the interest in distributed MPC is quite recent and robust distributed MPC represents a second step in this direction. Another possible reason is that robust optimization is generally quite demanding from a computational point of view and, therefore, the time constraints for the online implementation of the algorithms can be too restrictive in practice. However, first attempts to cope with robust distributed MPC can be found in the literature. In [9] a distributes scheme is obtained using min-max optimization. [20] exploits a constraint tightening procedure to take into account of the coupling between the subsystems. In [24] robust MPC based on tubes is extended to networked systems. In [18], subsystem couplings are considered as additional uncertainties and input to state stability is investigated.

Although the literature on robust distributed MPC is limited, it should be noticed that some methods derived for robust centralized MPC can be seamlessly combined with distributed optimizations techniques and used for the control of networked systems. In the following subsection we will consider two classes of OCP where we can extend the feedback MPC framework to obtain a distributed control algorithm.

2.2.1 Linear subsystems with additive uncertainty sharing a common resource

Consider a system composed of two subsystems with decoupled dynamics:

$$x_{t+1}^1 = A^1 x_t^1 + B^1 u_t^1 + w_t^1 \qquad x_{t+1}^2 = A^2 x_t^2 + B^2 u_t^2 + w_t^2 \tag{2.7}$$

Assume the cost and the input constraint set $\mathscr{U}$ are separable and that the state constraint set $\mathscr{X}$ can be written as

$$\mathscr{X} = (\mathscr{X}^1 \times \mathscr{X}^2) \cap \{(x^1,x^2) | G^1x^1 + G^2x^2 \le h\} \tag{2.8}$$

This situation can be easily found in practical situations when two or more subsystems share a common resource[1]. By introducing a slack variable γ we can separate the set $\mathscr{X}$ with respect to the state vectors of the two subsystems:

$$\mathscr{X} = \left(\mathscr{X}^1 \times \cap \{x^1 | G^1 x^1 \leq h - \gamma\}\right) \times \left(\mathscr{X}^2 \cap \{x^2 | G^2 x^2 \leq \gamma\}\right) \tag{2.9}$$

For a fixed value of γ we can observe that the centralized OCP can be separated in two completely independent problems which can be solved using tube MPC. Using the same notation used in Subsection 1.2.1, in order to satisfy the state constraints, the nominal trajectories must satisfy the constraints

$$\tilde{x}_t^1 \in \left(\mathscr{X}^1 \times \cap \{x^1 | G^1 x^1 \leq h - \gamma\}\right) \ominus \mathscr{E}^1 \tag{2.10}$$

and

$$\tilde{x}_t^2 \in \left(\mathscr{X}^2 \cap \{x^2 | G^2 x^2 \leq \gamma\}\right) \ominus \mathscr{E}^2 \tag{2.11}$$

where $\mathscr{E}^1$ and $\mathscr{E}^2$ correspond to the set $\mathscr{E}$ for the two subsystems.

Denote by $\phi^1(\gamma)$ and $\phi^2(\gamma)$ the optimal cost associated to the optimization problem to be solved online when implementing tube MPC for the two subproblems for a fixed value of γ.

To obtain a distributed tube MPC scheme, we can solve at every MPC iteration the following master problem

$$\min_{\gamma} \quad \phi^1(\gamma) + \phi^2(\gamma) \ . \tag{2.12}$$

A solution of problem (2.12) can be obtained using a subgradient method

$$\gamma_{k+1} = \gamma_k - \alpha_k g_k \tag{2.13}$$

where α_k is the step size and g_k is the the subgradient of $\phi^1(\gamma) + \phi^2(\gamma)$. A value of g_k can be calculated solving in parallel the optimization subproblems and using the sensitivities w.r.t. γ obtained from the Lagrange multipliers.

The method we obtained can be seen as an application of the primal decomposition method (also known as resource allocation) [23] to tube MPC.

2.2.2 Cascaded linear subsystems with additive uncertainty

Consider two subsystems interconnected like in Figure 2.2.

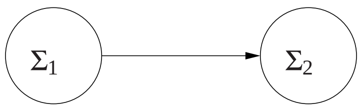

Figure 2.2: An example of cascaded subsystems.

Assume the subsystem dynamics are

$$x_{t+1}^1 = A^1 x_t^1 + B^1 u_t^1 + w_t^1 \qquad \text{and} \qquad x_{t+1}^2 = A^2 x_t^2 + B^2 u_t^2 + C x_t^1 + w_t^2. \tag{2.14}$$

where w_t^1 and w_t^2 belong the the compact sets $\mathscr{W}^1$ and $\mathscr{W}^2$ ($0 \in \mathscr{W}^1$ and $0 \in \mathscr{W}^2$). If we apply the same procedure of Subsection 1.2.1 to the first subsystem, we can write the state x_t^1 as $x_t^1 = \tilde{x}_t^1 + e_t^1$

[1] We can use the same method illustrated in the sequel to consider also the case where $\mathscr{U}$ is not separable but can be written in a form similar to (2.8).

($\tilde{x}_t^1$ is the nominal state of the first subsystem and e_t^1 represents the error). Splitting the input vector u_t^1 into two components $\tilde{u}_t^1$ and $\hat{u}_t^1$ and using the second to minimize e_t^1, we can guarantee that e_t^1 is contained inside the bounded set $\mathscr{E}^1$.

The dynamics of the second subsystem can by then written as

$$\begin{aligned} x_{t+1}^2 &= A^2x_t^2 + B^2u_t^2 + C\tilde{x}_t^1 + Ce_t^1 + w_t^2 \\ &= A^2x_t^2 + B^2u_t^2 + C\tilde{x}_t^1 + \overline{w}_t^2 \end{aligned} \quad . \tag{2.15}$$

We can now write $x_t^2 = \tilde{x}_t^2 + e_t^2$ and $u_t^2 = \tilde{u}_t^2 + \overline{u}_t^2$ where the nominal state $\tilde{x}_t^2$ is governed by the difference equation

$$\tilde{x}_{t+1}^2 = A^2\tilde{x}_t^2 + B^2\tilde{u}_t^2 + C\tilde{x}_t^1 \tag{2.16}$$

and the error e_t^2 is governed by the difference equation

$$e_{t+1}^2 = A^2e_t^2 + B^2\overline{u}_t^2 + \overline{w}_t^2. \tag{2.17}$$

Since $\overline{w}_t^2$ belongs to the bounded set $\mathscr{W}^2 \oplus C\mathscr{E}^1$, if the matrix pair (A^2, B^2) is controllable, we can design a feedback control $\overline{u}_t^2 = K^2e_t^2$ such that $e_t^2 \in \mathscr{E}^2$, where $\mathscr{E}^2$ is a bounded set.

If the cost and the constraint sets are separable, the optimization problem to be solved in the tube MPC iterations can be easily distributed among the subsystems using dual decomposition (see [12, 4] for a detailed description of this optimization method).

Remark 2 *From the example considered in this subsection we can notice that if the coupling between the subsystems is due to an interaction between the dynamics (i.e. for some subsystem we have that $\mathscr{N}^i$ does not contain only i), we need to take into account how the uncertainty propagates between the systems. This can be avoided when the coupling is due to an inequality constraint between variables belonging to different subsystems.*

2.3 Conclusions

As we have pointed out in the previous section, the literature about robust distributed MPC is limited. However, we have shown how we can combine techniques from robust MPC and distributed optimization to obtain new control schemes when the OCPs considered have a special structure.

Bibliography

[1] A. Ben-Tal, L. El Ghaoui, and A. Nemirovski. *Robust optimization*. Princeton University Press, 2009.

[2] D.P. Bertsekas and I.B. Rhodes. On the minimax reachability of target sets and target tubes. *Automatica*, 7:233–247, 1971.

[3] D.P. Bertsekas and I.B. Rhodes. Recursive state estimation for a set-membership description of uncertainty. *IEEE Trans. Automat. Contr.*, 16:117–128, 1971.

[4] D.P. Bertsekas and J. N. Tsitsiklis. *Parallel and distributed computation*. Prentice Hall, 1989.

[5] F. Blanchini and S. Miani. *Set-Theoretic Methods in Control*. Birkhäuser, 2008.

[6] E.F. Camacho and C. Bordons. *Model Predictive Control*. Springer, London, 2004.

[7] G. De Nicolao, L. Magni, and R. Scattolini. Robust Predictive Control of Systems with Uncertain Impulse Response. *Automatica*, 32:1475–1479, 1996.

[8] P. J. Goulart, E.C. Kerrigan, and J.M. Maciejowski. Optimization over state feedback policies for robust control with constraints. *Automatica*, 42:523–533, 2006.

[9] D. Jia and B.H. Krogh. Min-max Feedback Model Predictive control for Distributed Control with Communication. In *Proceedings of the American Control Conference*, 2002.

[10] M. Kothare, V. Balakrishnan, and M. Morari. Robust constrained model predictive control using linear matrix inequalities. *Automatica*, 32(10):1361–1379, November 1996.

[11] W. Langson, S.V. Rakovic I. Chryssochoos, and D. Q. Mayne. Robust model predictive control using tubes. *Automatica*, 40(1):125–133, 2004.

[12] Leon S. Lasdon. *Optimization theory for Large Systems*. Dover, 1970.

[13] J. H. Lee and Z. Yu. Worst-case Formulations of Model Predictive Control for Systems with Bounded Parameters. *Automatica*, 33(5):763–781, 1997.

[14] J.M. Maciejowski. *Predictive Control with Constraints*. Prentice Hall, 2002.

[15] L. Magni and R. Sepulchre. Stability margins of nonlinear receding-horizon control via inverse optimality. *Systems & Control Letters*, 32:241–245, 1997.

[16] D. Q. Mayne. Nonlinear model predictive control: Challenges and opportunities. In F. Allgöwer and A. Zheng, editors, *Nonlinear Predictive Control*, volume 26 of *Progress in Systems Theory*, pages 23–44, Basel Boston Berlin, 2000. Birkhäuser.

[17] D.Q. Mayne, M.M. Seron, and S.V. Rakovic. Robust model predictive control of constrained linear systems with bounded disturbances. *Automatica*, 41:219–224, 2005.

[18] D.M. Raimondo, L. Magni, and R. Scattolini. Decentralized MPC of nonlinear systems: An input-to-state stability approach. *Int. J. Robust Nonlinear Control*, 17:1651–1667, 2007.

[19] J.B. Rawlings and D.Q. Mayne. *Model Predictive Control: Theory and Design*. Nob Hill, 2009.

[20] A.G. Richards and J.P. How. A decentralized algorithm for robust constrained model predictive control. In *Proceedings of the 2004 American Control Conference*, 2004.

[21] P. O. M. Scokaert and D. Q. Mayne. Min-max feedback model predictive control for constrained linear systems. *IEEE Transactions on Automatic Control*, 43:1136–1142, 1998.

[22] P.O.M. Scokaert, J.B. Rawlings, and E.S. Meadows. Discrete-time Stability with Perturbations: Application to Model Predictive Control. *Automatica*, 33(3):463–470, 1997.

[23] G.J. Silverman. Primal decomposition of mathematical programs by resource allocation. I. Basic theory and a direction-finding procedure. *Operations Research*, 20:58–74, 1972.

[24] P. Trodden and A. Richards. Robust Distributed Model Predictive Control using Tubes. In *Proceedings of the 2006 American Control Conference*, 2006.