Robust and Stochastic MPC of Uncertain-parameter Systems

James Fleming

Worcester College

A thesis submitted in partial fulfilment of

the requirements for the degree of Doctor of Philosophy

Control group

Department of Engineering Science

University of Oxford

October 2016

Abstract

Constraint handling is difficult in model predictive control (MPC) of linear differential inclusions (LDIs) and linear parameter varying (LPV) systems. The designer is faced with a choice of using conservative bounds that may give poor performance, or accurate ones that require heavy online computation. This thesis presents a framework to achieve a more flexible trade-off between these two extremes by using a state tube, a sequence of parametrised polyhedra that is guaranteed to contain the future state.

To define controllers using a tube, one must ensure that the polyhedra are a subset of the region defined by constraints. Necessary and sufficient conditions for these subset relations follow from duality theory, and it is possible to apply these conditions to constrain predicted system states and inputs with only a little conservatism. This leads to a general method of MPC design for uncertain-parameter systems. The resulting controllers have strong theoretical properties, can be implemented using standard algorithms and outperform existing techniques.

Crucially, the online optimisation used in the controller is a convex problem with a number of constraints and variables that increases only linearly with the length of the prediction horizon. This holds true for both LDI and LPV systems. For the latter it is possible to optimise over a class of gain-scheduled control policies to improve performance, with a similar linear increase in problem size.

The framework extends to stochastic LDIs with chance constraints, for which there are efficient suboptimal methods using online sampling. Sample approximations of chance constraint-admissible sets are generally not positively invariant, which motivates the novel concept of 'sample-admissible' sets with this property to ensure recursive feasibility when using sampling methods. The thesis concludes by introducing a simple, convex alternative to chance-constrained MPC that applies a robust bound to the time average of constraint violations in closed-loop.

Acknowledgements

Firstly I would like to express my gratitude to the many friends and family members who have supported me over the course of my DPhil, especially during the long task of writing up. I am certain that you must all be tired of hearing about this thesis by now, but thanks for your patience, it helped greatly.

Further thanks go out to past and present members of the Control Group, for creating a great environment in which to carry out my work, as well as providing many interesting and entertaining discussions.

I am also grateful for the financial support of the Engineering and Physical Sciences Research Council, without which it would have been impossible to spend the last three and a half years in research.

Finally, but most importantly, I must thank my supervisors Professor Mark Cannon and Professor Basil Kouvaritakis for their guidance over the course of my studies. Thank you for sharing your wisdom. I could fill many theses with the knowledge I acquired from you both over the years; I have only written one, but I hope it is to your liking.

James Fleming

June 2016

Contents

Notation

$x_t,\ u_t$	:	state, input at time t
$x^k,\ u^k$	:	predicted state, input at time k
$x_t^k,\ u_t^k$	:	predicted state, input at time k as predicted at time t
$\sigma_E(\cdot),\ \gamma_E(\cdot)$	:	support function, gauge function (of E)
$\mathbb{P},\ \mathbb{E}$	:	probability measure, expectation
$H \geq 0$	:	matrix H has nonnegative entries
$P \succeq 0$	:	matrix P is positive semidefinite
$\mathcal{P}[F, b]$	:	the polyhedron $\{x \in \mathbb{R}^n \|\ Fx \leq b\}$
$\mathcal{R}_\Theta[F, b]$	:	the set $\{x \in \mathbb{R}^n \|\ F(\theta)x \leq b(\theta),\ \forall\theta \in \Theta\}$
$\mathcal{C}[F(\omega), b(\omega), \varepsilon]$	:	the set $\{x \in \mathbb{R}^n \|\ \mathbb{P}\{F(\omega)x \leq b(\omega)\} \geq 1 - \varepsilon\}$
$g^k(\cdot)$	:	tube parameterisation function at time k
$\alpha^k,\ \alpha_t^k$	:	tube parameters at time k, predicted at time t
$\Phi(\cdot),\ \Phi^{-1}(\cdot)$	:	image, preimage of set-valued map
$\lim\limits_{k\to\infty} \mathcal{S}_k$	:	set-theoretic limit
$P(E)$	:	power set of E
$\cup,\ \cap$	:	set union, intersection
$\vee,\ \wedge$	:	lattice join, meet

1 Introduction

1.1 Parametric uncertainty in MPC

This thesis is about the model predictive control of discrete-time systems that, for any fixed value of an uncertain time-varying parameter θ, have dynamics that are linear and time-invariant. Additive disturbances may act on the state, so that the dynamics take the general form:

$$x_{t+1} = A(\theta_t)x_t + B(\theta_t)u_t + w(\theta_t) \tag{1.1a}$$

$$y_t = C(\theta_t)x_t + D(\theta_t)u_t \tag{1.1b}$$

$$z_t = F(\theta_t)x_t + G(\theta_t)u_t \tag{1.1c}$$

These systems are to be controlled so that the first output, y_t, is made small in some sense (such as in l_2 norm) while the second output, z_t, robustly satisfies either the deterministic linear constraint

$$z_t \leq 1 \tag{1.2}$$

or the stochastic chance constraint

$$\mathbb{P}\{z_t \leq 1\} \geq 1 - \epsilon \tag{1.3}$$

for any possible realisation of the uncertainty parameter θ_t.

Importantly, we will allow this parameter to vary over time, as long as it is bounded within a compact set Θ at all times. In many cases of practical interest, such as when using the system dynamics (1.1) as a conservative approximation to an uncertain nonlinear system, θ_t may be an unknown function of the current system state so that the resulting dynamics are neither linear nor time-invariant and have many possible state trajectories. Nonetheless, the bound of the uncertain parameter in the compact set Θ will allow us to develop effective algorithms for control of such systems.

The motivation for studying constrained control of systems with this kind of uncertainty comes from two straightforward observations:

- The regulation or tracking control of all real-world processes is subject to either actuator limits or restriction to some safe region of operation.
- Almost all real-world processes are nonlinear, time-varying, or have system parameters that are not known exactly in practice.

The appeal of the formulation (1.1) is that it is general enough to conservatively represent many nonlinear or uncertain processes, while retaining enough mathematical structure to enable the design of effective constrained controllers. This latter condition often does not hold for more general nonlinear models, such as input-affine systems, as it difficult to find convex online optimisations in such cases. Model predictive control is our method of choice for this design because it provides a general framework for constrained control using mathematical optimisation techniques, and allows for guarantees of constraint satisfaction and closed-loop system stability.

1.1.1 A note on terminology

If the parameter θ_t is known and available to the controller at time t, then (1.1) is usually known as a linear parameter varying (LPV) system. Otherwise this system is a linear difference inclusion (LDI). In this thesis we shall see that if we restrict our analysis to LPV-A systems where B, D and G are not functions of the parameter θ_t, it is possible to handle the constrained MPC of both types of system in a similar framework. For that reason, we refer to them both as "linear parameter-uncertain" systems when there is no need to distinguish between the two.

1.2 Background

The general problem of optimal control is quite well understood, having been studied since the work of Pontryagin and Bellman in the 1950s [PBG62, Bel57]. A well-known early result due to Kalman is that, for quadratic cost functions, optimal controllers for unconstrained linear systems are given by linear state feedback controllers when the state is measurable, and the combination of a linear state feedback controller and an optimal linear estimator if it is not [Kal60], [K$^+$60]. These controllers can be found by solving certain quadratic matrix equations known as Riccati equations. More generally, it is possible to give a convex parameterisation (the 'Youla parameterisation') of all stabilising linear controllers for a linear plant [YJBJ$^+$76]. This allows for the design of linear controllers to minimise or bound the norms of transfer functions, such as the $\mathcal{H}_2$ or $\mathcal{H}_\infty$ norms. Methods are available for solving such problems with multiple inputs and multiple outputs using state-space formulations, leading again to the solution of a Riccati-type equation [D$^+$84].

Advances made in the theory of optimisation in the 1990s demonstrated that a large variety of convex optimisation problems are efficiently solvable by interior-point methods [NNY94]. Amongst many other applications, this enables efficient solution of problems involving linear matrix inequalities (LMIs), which are often encountered in control problems. In particular, conditions for the existence of a quadratic Lyapunov function can be expressed as LMI constraints. Using a standard change of variables that renders the problem convex, this allows linear state feedback controllers to be designed for large classes of uncertain systems [BEGFB94]. These controllers can be designed to minimise some norm of the system output such as the $\mathcal{L}_2$ norm, or bound the $\mathcal{L}_2$ gain from some disturbance input to the controlled output, and the results will be optimal in the sense that they give the best bounds provable by quadratic functions. Similar LMI methods are available for linear parameter-varying systems [WYPB95]. There also exists an analogous convexifying change of variables for the output-feedback control problem [SGC97], enabling quadratically-optimal feedback controllers to be designed in the more difficult case that the current state is unavailable for measurement.

The limitation of all these design techniques for unconstrained systems is that, when they are applied to systems with constraints, the resulting controller could be highly suboptimal and, worse, may not even lead to a stable closed-loop system once the effects of actuator saturation are taken into account. In principle, optimal controllers for constrained systems can be found by dynamic programming [Bel57], but in practice the method is far too computationally demanding to give solutions for systems with more than a few state variables. This has led researchers to consider various suboptimal methods for solving the constrained control problem, sacrificing optimality to search for simpler controllers that give guarantees of constraint satisfaction and stability.

One such method is model predictive control (MPC), in which a finite horizon optimal control problem is solved at each point in time to compute a control input to the system. The resulting input may well be suboptimal for the corresponding infinite-horizon problem, but the effect of this is mitigated somewhat by repeating the optimisation periodically and applying the newly-computed control to the system. For linear systems without uncertainty, this procedure can be optimal for the infinite horizon control problem if the prediction horizon is made long enough and a terminal cost introduced [SR98]. If the state at the end of the prediction horizon is constrained within a control invariant set [GT91, Ker01] when a suitable terminal cost is used, the resulting controller is guaranteed to be stable and satisfy constraints for all time.

When uncertainty or nonlinearity is present in the system dynamics, the MPC problem becomes considerably more difficult. If the uncertainty is additive, there are practical approaches to optimise over feedback policies that give guarantees of stability and constraint satisfaction with generally small levels of conservatism [MR09, Gou07]. But for constraint handling under parametric uncertainty, methods rely either on very conservative ellipsoidal bounding of the state [KBM96, CGM02], on solution algorithms that take time exponential in the length of the prediction horizon [PRSDM05c], or on other computationally intensive approaches such as offline dynamic programming [Bes10]. Hence there is a need for computationally tractable MPC controllers for parameter-uncertain systems that do not introduce large amounts of conservatism in the way that they handle constraints. This is the objective of the present thesis.

1.3 Thesis outline

This thesis is divided into three parts. Part I consists of background material on convex optimisation (Chapter 2) and model predictive control (Chapter 3). The aim here is to give a sense of the current state of the art in MPC, as well as providing some prerequisite knowledge for the remainder of the work.

Part II considers robust control, first developing the mathematical tools required to handle constraints for parameter-uncertain systems. The method that is advocated is online optimisation over perturbations on a feedback controller that is designed offline, and to bound the uncertain future state of the system in a 'prediction tube', which is a sequence of sets ('tube cross-sections') of future states parametrised by free variables in the MPC optimisation. This requires some study of how subset relations between parametrised sets can be applied by using convex constraints on the parameters (Chapter 4). The result of this analysis is a general framework for MPC of uncertain systems in which many parameterisations of the tube cross-sections are possible and constraint satisfaction and closed-loop stability can be guaranteed. This stability can take either the form of exponential convergence to a predetermined set or a bound on the l_2 gain from a system input (Chapter 5). These results are also extended to the case of LPV-A systems where measurements of the current system parameter can be exploited, and the optimisation can be carried out over a class of gain-scheduled control policies, improving performance (Chapter 6). In both cases, the resulting controllers are efficiently implementable using standard convex programming solvers and are demonstrated to greatly outperform constrained MPC methods that use ellipsoidal bounding for tractability.

Part III extends the theory to stochastic control. Motivated by the desire to ensure recursive feasibility in MPC problems with chance-constraints on state variables, a new class of admissible sets for parameter-uncertain systems under chance constraints is presented (Chapter 7). A chance-constrained formulation of the MPC is given, considering an expected-value cost function. As chance-constrained optimisation is often difficult in practice, approximate schemes that use sampling are considered, and an efficient heuristic method for sample removal is introduced. Recursive feasibility and stability are both retained as long as the uncertainty is bounded. In the special case that sampling is used only one step ahead, coloured noise can be handled and a probabilistic confidence guarantee given on constraint satisfaction (Chapter 8). Finally, we suggest an alternative to chance-constrained MPC by considering a limit on the number of violations of a constraint in a given period of time. This allows for construction of controllers that satisfy a robust bound on the number of violations of a constraint, without requiring chance-constrained programming (Chapter 9).

1.4 Related publications

Many of the ideas presented in this thesis have appeared in papers written during my time as a graduate student with the control group in Oxford. The theory of recursive feasibility when using polytopic tubes for robust MPC, which is presented in Part II, is found within the papers:

- James Fleming, Basil Kouvaritakis, and Mark Cannon. Regions of attraction and recursive feasibility in robust MPC. In *Control & Automation (MED), 2013 21st Mediterranean Conference on*, pages 801–806. IEEE, 2013
- James Fleming, Basil Kouvaritakis, and Mark Cannon. Robust tube MPC for linear systems with multiplicative uncertainty. *Automatic Control, IEEE Transactions on*, 60(4):1087–1092, 2015

However, the treatment in Part II is more complete, allowing for different tube parameterisations, many types of parametric uncertainty (including the extension to the LPV case in chapter 6) and using simplified terminal conditions. The approximation of Chapter 7 to the maximal admissible set under chance-constraints can be found in

- James Fleming and Mark Cannon. Admissible sets for chance-constrained difference inclusions. In *2016 American Control Conference, Boston, USA*, 2016

in a simplified form. Chapter 7 extends the theory to include additive disturbances and simplifies many proofs by considering set-valued mappings and closure properties of the set algebras. The remainder of the material of Part III on chance-constrained tube MPC formulations and sampling appear in the papers:

- James Fleming, Mark Cannon, and Basil Kouvaritakis. Stochastic tube MPC for LPV systems with probabilistic set inclusion conditions. In *53rd IEEE Conference on Decision and Control, Los Angeles, USA*, pages 4783–4788. IEEE, 2014
- James Fleming, Mark Cannon, and Basil Kouvaritakis. Stochastic MPC using probabilistic tubes and sample approximations of chance constraints. *To appear*, 2016

Again, Part III is a more thorough account, with various small improvements as well as the introduction of robust time-average constraints in Chapter 9.

Part I

Background material

2 Convex optimisation

2.1 Introduction

This introductory chapter considers the solution of convex optimisation problems, where it is desired to find the value of a vector x that minimises the value of a convex function subject to the condition that x must lie in a convex set. The motivation for studying convex problems in particular is that efficient solution algorithms are usually available when the problem is convex, often using freely available software packages. Hence, if we can show that some problem of interest in engineering or control is equivalent to a convex optimisation problem in some standard form, we can consider to have 'solved' the problem in the sense that we can quickly find a solution to any particular instance of it.

Most of the theory in this chapter can be found in the excellent book [BV04] on convex optimisation. The theory of support and gauge functions is covered in [Roc70], and a good reference for the material on robust optimisation is the review paper [BBC11]. The results on sampling methods for chance-constrained problems appears in [Cal10] and [CG11].

2.1.1 Convex sets

We first consider some important classes of convex sets, considered as subsets of the Euclidean space $\mathbb{R}^n$, equipped with the usual metric $d(x, y)$ giving the distance between points x and y. Recalling some basic terminology from the study of Euclidean spaces, we refer to subsets $E \subseteq \mathbb{R}^n$ as ***bounded*** if there is some point $y \in E$ and some $R \in \mathbb{R}_+$ such that $d(x, y) < R$ for all $x \in E$. The set E is ***closed*** if it contains the limits $\lim_{k\to\infty} x_k$ of all infinite sequences of points $\{x_k\}_{k=0,\ldots,\infty}$ with $x_k \in E$ for all k. A set is ***compact*** if it is closed and bounded.

The set E is ***convex*** if the convex combination $\mu x + (1-\mu)y \in E$ for any $x, y \in E$ and any $\mu \in [0, 1]$. Geometrically this states that the line segment between any two points in the set remains inside the set. It is a straightforward but vitally important fact that intersections of convex, closed, bounded and compact sets are convex, closed, bounded and compact respectively. If E is convex, affine mappings of the form

$$\{x \in \mathbb{R}^n | \;\; Ax + w, \;\; x \in E\}$$

for $A \in \mathbb{R}^{n \times n}$ and $w \in \mathbb{R}^n$ are also convex themselves. Conveniently, this allows us to describe a great variety of convex sets by considering finite intersections of a few basic types under affine transformations.

A ***polyhedron*** is the closed set defined by a finite number of nonstrict linear inequalities in the form

$$\{x \in \mathbb{R}^n | \;\; Fx \leq b\} \tag{2.1}$$

where $F \in \mathbb{R}^{q \times n}$ is a matrix and $g \in \mathbb{R}^q$ a vector. In the special case of a single constraint, so that F is a row vector and b a scalar, the resulting set is known as a ***half-space***, so that any polyhedron can be expressed equivalently as the intersection of a finite number of such half-spaces. The constraints $x \geq 0$ give the specific polyhedron usually called the ***positive orthant*** which is the prototypical example of a ***convex cone***, which is any convex set E containing μx for any $x \in E$ and any nonnegative scalar $\mu \geq 0$.

A ***polytope*** is a set defined by all convex combinations of a finite set of points, written as $\mathrm{Co}\{w_1, w_2, \ldots, w_m\}$ where 'Co' denotes the ***convex hull*** operation of finding the smallest convex set containing the given points. It is compact and convex and is expressible in the form

$$\{x \in \mathbb{R}^n | \;\; x = Wy, \;\; y \geq 0, \;\; \underline{1}^T y = 1\} \tag{2.2}$$

where W is a matrix with columns w_i, we have a vector $y \in \mathbb{R}^m$ and $\underline{1}$ denotes the vector of ones. It is a well-known result that a set is a polytope if and only if it is a bounded polyhedron. Hence there is an equivalence between the representations (2.2) and (2.1) whenever the latter describes a bounded set. But converting between the two representations can be a fairly difficult computation, especially in high-dimensional spaces where the number of vertices w_i may be large compared to the number of inequalities in (2.1). For this reason we will tend to prefer the representation (2.1) when using such sets in control.

An ***ellipsoid*** is a set of the form

$$\left\{x \in \mathbb{R}^n \middle| \ (x - x_0)^T P (x - x_0) \leq 1\right\}$$

where the matrix $P \in \mathbb{R}^{n\times n}$ is symmetric and positive definite so that the quantity on the left-hand-side of the equality is positive whenever $x \neq x_0$. We will not have much reason to consider ellipsoids in the remainder of the thesis, except to occasionally describe uncertainty in a system.

For any norm $\|\cdot\|$ the associated unit ***norm ball*** is the set

$$\{x \in \mathbb{R}^n | \ \|x\| \leq 1\} \tag{2.3}$$

which is convex and compact from the usual properties of norms. If we consider the p-norms, this is a polytope for $p = 1$ and $p = \infty$. Introducing another coordinate $t \in \mathbb{R}$, we can also consider the associated ***norm cone***

$$\{(x, t) \in \mathbb{R}^{n+1} | \ \|x\| \leq t\} \tag{2.4}$$

which is a convex cone in $\mathbb{R}^{n+1}$. In the case of a p-norm, this is a polyhedron for $p = 1$ or $p = \infty$. When $p = 2$, the resulting set is known as the ***second-order cone***. If we consider affine mappings of this cone, we are led to sets of the form:

$$\{x \in \mathbb{R}^n | \ \|Ax + b\|_2 \leq c^T x + d\} \tag{2.5}$$

Finally, recalling that a symmetric matrix $P \in \mathbb{S}^n$ is positive semidefinite (written $P \succeq 0$) if we have $x^T P x \geq 0$ for all $x \in \mathbb{R}^n$, we can define the ***semidefinite cone***

$$\mathbb{S}^+ = \{P \in \mathbb{S}^n | \ P \succeq 0\} \tag{2.6}$$

which is a convex cone. This allows us to consider inequalities between matrices like $P \succeq Q$, which means that $P - Q \in \mathbb{S}^+$. When dealing with these matrix inequalities we will often make use of the well-known result that the nonlinear matrix inequalities

$$C - BA^{-1}B^T \succeq 0, \quad A \succeq 0$$

are equivalent to the linear matrix inequality (LMI):

$$\begin{bmatrix} A & B \\ B^T & C \end{bmatrix} \succeq 0$$

This is usually known as the ***Schur complement lemma***.

2.1.2 Convex optimisation and duality

Our interest in these convex sets is that they can be used to define tractable convex optimisation problems. The convex optimisation problems encountered in this thesis can all be expressed in the standard form

$$\begin{aligned} \underset{x}{\text{minimise}} \quad & f(x) \\ \text{subject to} \quad & Ax = b \\ & g(x) \leq 0 \end{aligned} \tag{2.7}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ and $g : \mathbb{R}^n \to \mathbb{R}^q$ are convex functions. That is, they satisfy

$$f(\mu x + (1 - \mu)y) \leq \mu f(x) + (1 - \mu)f(y)$$

for any $x, y \in \mathbb{R}^n$ and any scalar $0 \leq \mu \leq 1$. We will refer to this as the ***primal*** optimisation problem. The constraint $g(x) \leq 0$ will almost always define one of the convex sets that we have described in the previous section. That is, a polyhedron, second-order cone, semidefinite cone or some intersection thereof. If the function f is linear in x, then the resulting optimisation problems are known as linear programs (LP), second-order cone programs (SOCP), and semidefinite programs (SDP) in these cases, respectively. If the function f is quadratic in x and $g(x) \leq 0$ defines a polyhedron then the optimisation is known as a quadratic program (QP). There is a chain of inclusions between these classes of problem

$$\text{LP} \subset \text{QP} \subset \text{SOCP} \subset \text{SDP}$$

in the sense that any LP can be expressed as a QP, SOCP or SDP, a QP can be expressed as an SOCP or SDP, and any SOCP expressed as an SDP.

For any optimisation problem in the standard form, we define the associated ***Lagrangian***

$$L(x, \lambda, \nu) = f(x) + \lambda^T g(x) + \nu^T (Ax - b) \tag{2.8}$$

and also the ***Lagrange dual function***

$$h(\lambda, \nu) = \inf_x L(x, \lambda, \nu) = \inf_x \{f(x) + \lambda^T g(x) + \nu^T (Ax - b)\} \tag{2.9}$$

and we define the ***dual*** of the standard form optimisation to be the maximisation of $h(\lambda, \nu)$ subject to nonnegativity constraints on the variable λ.

$$\begin{aligned} \underset{\lambda,\nu}{\text{maximise}} \quad & h(\lambda, \nu) \\ \text{subject to} \quad & \lambda \geq 0 \end{aligned} \tag{2.10}$$

The solution of the dual problem provides a lower bound on the optimal value of the primal problem. To see this, denote the optimum of the primal problem by p^* and the optimum of the dual by d^*, and by the definition of the dual problem we have:

$$d^* = \sup_{\lambda \geq 0, \nu} \inf_x L(x, \lambda, \nu)$$

A similar representation exists for the primal optimum. Note that $\sup_{\lambda \geq 0, \nu} \lambda^T g(x) < +\infty$ only if $g(x) \leq 0$. Similarly, $\sup_{\lambda \geq 0, \nu} \nu^T (Ax - b) < +\infty$ only if $Ax = b$. When both of these constraints are satisfied, both of these suprema are zero so that we have $\sup_{\lambda \geq 0, \nu} L(x, \lambda, \nu) = f(x)$. Hence, the primal optimal is equal to

$$p^* = \inf_x \sup_{\lambda \geq 0, \nu} L(x, \lambda, \nu)$$

and we have $d^* \leq p^*$ from the relationship

$$\sup_{\lambda \geq 0, \nu} \inf_x L(x, \lambda, \nu) \leq \inf_x \sup_{\lambda \geq 0, \nu} L(x, \lambda, \nu)$$

which holds for any function L. The condition $d^* \leq p^*$ is known as ***weak duality***.

Under some technical conditions, such as Slater's constraint qualification condition that there exists an x such that $g(x) < 0$ and $Ax = b$, it is possible to prove that the optimal value of the objective function in the primal and dual problems are equal if the primal is convex [BV04]. In this case, we say that ***strong duality*** holds. All convex problems considered in this thesis will be of this type and so we shall not have much reason to consider weak duality, but it is worth noting that weak duality also holds for nonconvex problems.

Later in the thesis, we will make use of these results for linear programming problems in order to develop convex conditions for subset relations, so we will briefly consider duality for this special case. For a linear program in the form

$$\begin{aligned} \underset{x}{\text{minimise}} \quad & c^T x \\ \text{subject to} \quad & Fx \leq g \end{aligned}$$

we have the Lagrange dual function:

$$h(\lambda) = \inf_x \{c^T x + \lambda^T (Fx - g)\}$$

This function will be unbounded below unless $c + F^T \lambda = 0$, in which case it satisfies $h(\lambda) = g^T \lambda$, and so the dual optimisation problem is equivalent to:

$$\begin{aligned} \underset{\lambda}{\text{maximise}} \quad & g^T \lambda \\ \text{subject to} \quad & c + F^T \lambda = 0 \\ & \lambda \geq 0 \end{aligned}$$

For a linear program and its dual, we always have strong duality.

2.2 Robust optimisation

An interesting extension to the standard form convex optimisation is to consider problems that have the form of (2.7) for any fixed value of an uncertain parameter. The usefulness of explicitly including uncertainty in an optimisation is clear, given that almost all real-world problems include uncertain data.

$$\begin{aligned} \underset{x}{\text{minimise}} \quad & f(x) \\ \text{subject to} \quad & Ax = b \\ & g(x, \theta) \leq 0, \quad \forall \theta \in \Theta \end{aligned} \tag{2.11}$$

The uncertain parameter θ is bounded in some compact set Θ, and constraints must be satisfied for all possible realisations. Note that we do not allow the uncertainty to be present in the objective function, but that there is no loss of generality in this. Indeed, if there was uncertainty in the objective, we could reformulate the problem into the form (2.11) by introducing an extra variable t and a constraint $t \geq f(x, \theta)$, and minimising t. We will also assume that the function $g(x, \theta)$ is convex in x, and affine in θ. This latter condition can be relaxed, but is typical of the problems we shall consider.

If the set Θ is finite with members $\theta^{(i)}$ for $i = 1, \ldots, N$, then (2.11) is equivalent to the convex problem

$$\begin{aligned} \underset{x}{\text{minimise}} \quad & f(x) \\ \text{subject to} \quad & Ax = b \\ & g(x, \theta^{(i)}) \leq 0, \quad i = 1, \ldots, N \end{aligned} \tag{2.12}$$

which can usually be solved by standard methods. Given an affine dependence of $g(x, \theta)$ on θ, this also solves the problem when Θ is polytopic, so that $\Theta = \mathrm{Co}\{\theta^{(i)}, i = 1, \ldots, N\}$. We will return to formulations like (2.12) later in the chapter when considering stochastic problems, in which case Θ will be a set of samples from some probability distribution.

When the set Θ contains an infinite number of members, this implies that the optimisation (2.11) has an infinite number of constraints (such cases are often called 'semi-infinite' optimisation problems). Nonetheless, it is often still possible to solve the problem if Θ is compact. Whether or not this leads to a convex problem depends on the form of the functions $g(x, \theta)$ and the uncertainty set Θ, so we now focus on the situation that $g(x, \theta)$ is affine in x, which will be the most important case in this thesis.

2.2.1 Robustifying linear constraints

Before considering the robust optimisation problem for linear inequality constraints, we introduce a few more definitions relating to convex sets that will be useful in solving robust optimisation problems. We assume throughout the discussion that E is a convex set.

The ***polar dual*** or polar set, of E is the set

$$\{x \in \mathbb{R}^n | \ y^T x \leq 1, \ \ \forall y \in E\}$$

which is convex. The ***support function*** of the set E, written $\sigma_E(\cdot)$, is the function:

$$\sigma_E(x) = \sup_y \{y^T x | \ y \in E\}$$

The name 'support function' reflects the fact that for any vector v, we have $v^T x \leq \sigma_E(v)$ for any $x \in E$, so that the function specifies all supporting hyperplanes of the convex set E. The ***gauge function*** (sometimes called 'Minkowski functional') of E is the function

$$\gamma_E(x) = \inf_{\mu \geq 0} \{\mu | \ x \in \mu E\}$$

which, intuitively, gives the minimum amount that we can 'scale' a set for it to contain a given point.

The support and gauge functions are duals of each other in a certain sense that will turn out to be useful when deriving convex equivalents of linear constraints. Suppose we know a convex set E and its polar dual E^*, then we have the two relationships [Roc70]:

$$\sigma_E(x) = \gamma_{E^*}(x), \quad \gamma_E(x) = \sigma_{E^*}(x) \tag{2.13}$$

In particular, the first of these two equations is very useful in dealing with robust linear constraints in cases where the polar set of the uncertainty is easily found. We consider the robust optimisation problem:

$$\begin{aligned} \underset{x}{\text{minimise}} \quad & f(x) \\ \text{subject to} \quad & Ax = b \\ & \bar{f}^T x + \theta^T F x \leq g, \quad \forall \theta \in \Theta \end{aligned} \tag{2.14}$$

Including the term with $\bar{f}^T$ in the constraint is quite useful, as it means that we can assume without any loss of generality that the uncertainty set Θ contains the origin. We can interpret $\theta = 0$ as corresponding to some 'nominal' value of the constraint function.

Now, the robust constraint can be written equivalently as

$$\bar{f}^T x + \sup_{\theta}\{\theta^T F x | \ \theta \in \Theta\} \leq g$$

which from the definition of the support function is

$$\bar{f}^T x + \sigma_{\Theta}(Fx) \leq g$$

and applying (2.13), we get:

$$\bar{f}^T x + \gamma_{\Theta^*}(Fx) \leq g$$

From the definition of the gauge function, this last inequality is satisfied if and only if there exists a μ such that

$$\bar{f}^T x + \mu \leq g, \quad Fx \in \mu\Theta^*$$

and since we know that the polar set Θ^* is convex, we have recovered convex constraints that are equivalent to the robust constraint in (2.14). Hence that robust optimisation can be expressed as:

$$
\begin{aligned}
\underset{x}{\text{minimise}} \quad & f(x) \\
\text{subject to} \quad & Ax = b \\
& \bar{f}^T x + \mu \leq g \\
& Fx \in \mu\Theta^*
\end{aligned}
\tag{2.15}
$$

To solve this latter problem, we of course need to know the form of the polar set Θ^*. We will consider first the case that Θ is a polyhedron. Under the mild assumption that the origin is contained within the interior of the set, we can express Θ as

$$
\begin{aligned}
\Theta &= \{\theta \in \mathbb{R}^q | \ C\theta \leq 1\} \\
&= \{\theta \in \mathbb{R}^q | \ c_i^T\theta \leq 1, \ \ i = 1, \ldots, p\}
\end{aligned}
$$

and from the definition of the polar dual, we have

$$
\begin{aligned}
\Theta^* &= \{a \in \mathbb{R}^q | \ \theta^T a \leq 1, \ \ \forall\theta : \ \ c_i^T\theta \leq 1, \ \ i = 1, \ldots, p\} \\
&= \mathrm{Co}\{c_i, \ \ i = 1, \ldots, p\} \\
&= \{a \in \mathbb{R}^n | \ a = C^T y, \ \ y \geq 0, \ \ \underline{1}^T y = 1\}
\end{aligned}
$$

so that in this instance the robust optimisation is equivalent to

$$
\begin{aligned}
\underset{x,y}{\text{minimise}} \quad & f(x) \\
\text{subject to} \quad & Ax = b \\
& \bar{f}^T x + \underline{1}^T y \leq g \\
& Fx = C^T y \\
& y \geq 0
\end{aligned}
\tag{2.16}
$$

which is just an LP. It is of course no coincidence that the constraints now implicitly include the Lagrange dual of the problem $\max_\theta\{x^T F^T \theta | \ C\theta \leq 1\}$, and we could have derived the robust counterpart directly from duality theory if desired.

The polar set has a similarly convenient form for ellipsoids. Supposing again that the origin is in the interior of the set, we can express a general ellipsoid as

$$\Theta = \left\{\theta \in \mathbb{R}^q \middle|\ \theta^T P \theta \leq 1\right\}$$

which has a polar set:

$$\Theta^* = \left\{a \in \mathbb{R}^q \middle|\ a^T\theta \leq 1\ \ \forall\theta :\ \ \theta^T P \theta \leq 1\right\}$$

Any particular a will be a member of this set if we have

$$\sup_\theta \left\{a^T\theta \middle|\ \theta^T P\theta \leq 1\right\} \leq 1$$

and the optimisation on the left-hand-side is simple enough to solve analytically. Supposing that the constraint is active at the solution, we can augment the constraint into the cost using the method of Lagrange multipliers to get:

$$J = a^T\theta + \lambda\left(\theta^T P\theta - 1\right) \tag{2.17}$$

We know that at the solution the gradient of this function will be zero, and hence

$$\frac{dJ}{d\theta} = a^T + 2\lambda\theta^T P = 0$$

so that:

$$\theta^* = -\frac{P^{-1}}{2\lambda}a \tag{2.18}$$

Plugging this into the equality constraint gives

$$\lambda^* = \pm\frac{1}{2}\sqrt{a^T P^{-1} a} \tag{2.19}$$

and substituting from (2.18) and (2.19) into (2.17) then gives the extrema:

$$J^* = \pm\sqrt{a^T P^{-1} a}$$

Hence the polar set is

$$\Theta^* = \{a \in \mathbb{R}^q |\ a^T P^{-1} a \leq 1\}$$

and the corresponding robust optimisation becomes

$$\begin{aligned}\underset{x}{\text{minimise}} \quad & f(x) \\ \text{subject to} \quad & Ax = b \\ & \bar{f}^T x + \|P^{-\frac{1}{2}} F x\|_2 \le g\end{aligned} \tag{2.20}$$

which is an SOCP. Here we use $P^{-1} = P^{-\frac{T}{2}} P^{-\frac{1}{2}}$, a Cholesky decomposition.

2.3 Chance-constrained optimisation

A different approach to uncertainty in optimisation is to consider the constraint function as a random variable and then apply the inequality constraint with at least a certain probability. Supposing that we have a probability space $(\Theta, \Sigma, \mathbb{P})$ for which $g(x, \cdot)$ is measurable for any x, we consider optimisations of the form:

$$\begin{aligned}\underset{x}{\text{minimise}} \quad & f(x) \\ \text{subject to} \quad & Ax = b \\ & \mathbb{P}\{g(x,\theta) \le 0\} \ge 1 - \varepsilon\end{aligned} \tag{2.21}$$

In general, this problem is difficult and may not be convex even when the function g is convex in x. If the set

$$S = \{(x, \theta) | \ g(x, \theta) \le 0\}$$

is convex, and θ has a logarithmically concave density function $p(\theta)$ so that

$$\mu \ln p(\theta_1) + (1 - \mu) \ln p(\theta_2) \le \ln p(\mu\theta_1 + (1 - \mu)\theta_2)$$

then since we have

$$\mathbb{P}\{g(x, \theta) \le 0\} = \int_\Omega 1_S(x, \theta) p(\theta) d\theta$$

the constraint function is also log-concave. Hence we can replace the chance constraint with an equivalent convex constraint.

$$\ln \mathbb{P}\{g(x, \theta) \le 0\} \ge \ln(1 - \varepsilon) \tag{2.22}$$

Many common probability distributions are log-concave, so we might imagine that we can solve many problems like (2.21). But unfortunately, in most cases there is no easy closed-form expression for the constraint function or its logarithm, so that solving this optimisation can be difficult even when it is convex.

2.3.1 Normal distributions

One special problem instance that can be solved exactly is when the constraint function $g(x, \theta)$ is affine and the distribution of θ a multivariate normal. In that case we can express the problem as

$$\begin{aligned} \underset{x}{\text{minimise}} \quad & f(x) \\ \text{subject to} \quad & Ax = b \\ & \mathbb{P}\{\theta^T x \leq g\} \geq 1 - \varepsilon \end{aligned} \tag{2.23}$$

where $\theta \sim \mathcal{N}(\bar{\theta}, \Sigma)$. As linear transformations of multivariate normals are also normal, the constraint function has distribution

$$\theta^T x - g \sim \mathcal{N}(\bar{\theta}^T x - g, x^T \Sigma x)$$

which is a univariate normal. Hence we can rewrite the chance constraint using the cumulative distribution function of a standard normal:

$$\Phi\left(\frac{g - \bar{\theta}^T x}{\sqrt{x^T \Sigma x}}\right) \geq 1 - \varepsilon$$

Applying the inverse Φ^{-1} of the CDF to both sides and rearranging gives

$$g - \bar{\theta}^T x \geq \Phi^{-1}(1 - \varepsilon) \|\Sigma^{\frac{1}{2}} x\|_2$$

which is a second-order cone constraint if $\Phi^{-1}(1-\varepsilon) > 0$. Hence, for $\varepsilon < 0.5$, we have the equivalent convex optimisation problem:

$$\begin{aligned} \underset{x}{\text{minimise}} \quad & f(x) \\ \text{subject to} \quad & Ax = b \\ & \|\Phi^{-1}(1-\varepsilon)\Sigma^{\frac{1}{2}} x\|_2 \leq g - \bar{\theta}^T x \end{aligned} \tag{2.24}$$

We should point out that this method is quite limited, and leads to a nonconvex optimisation problem if $\epsilon > 0.5$. It works only for normally distributed uncertainties and for individual chance constraints. In particular it does not generalise to joint constraints such as

$$\mathbb{P}\{\theta^T G x \leq h\} \geq 1 - \varepsilon$$

where G is a matrix and h a vector, and indeed the constraint function is quite hard to evaluate in this case.

2.3.2 Convex approhximations

For more general chance-constrained problems, various approximations exist to replace the chance constraint with some convex, tractable approximation. One useful proposal is given in [NS06], where it is noted that for any function $\psi : \mathbb{R} \to \mathbb{R}$ that is nondecreasing and satisfies $\psi(0) = 1$, we have the bound

$$\mathbb{E}\psi\left(\frac{g(x,\theta)}{\alpha}\right) \geq \mathbb{P}\left\{g(x,\theta) > 0\right\} \tag{2.25}$$

for any scalar $\alpha > 1$, so that the chance constraint is conservatively approximated by

$$\mathbb{E}\alpha\psi\left(\frac{g(x,\theta)}{\alpha}\right) \geq \alpha\epsilon$$

which is a convex constraint in α and x. Hence, it is possible to optimise over both α and x, and the chance-constrained optimisation becomes

$$
\begin{aligned}
\underset{x,\alpha>0}{\text{minimise}} \quad & f(x) \\
\text{subject to} \quad & Ax = b \\
& \mathbb{E}\alpha\psi\left(\frac{g(x,\theta)}{\alpha}\right) \geq \alpha\epsilon
\end{aligned}
\tag{2.26}
$$

which is a convex problem. The form of the constraint function (and how straightforward it is to evaluate) depends on the choice of function ψ. Typical choices are $\psi(u) = 1 + u$, $\psi(u) = (1 + u)^2$ or $\psi(u) = e^u$, and interestingly there are various equivalences between the bound (2.25) and classical concentration inequalities such as Markov, Chebyshev, and Hoeffding bounds in these cases [NS06]. Such bounds could also be used to derive conservative approximations of chance constraints directly.

2.3.3 Sample approximations

Another approach to solving the chance-constrained optimisation problem is to consider drawing N samples of the random variable θ and solve the optimisation

$$
\begin{aligned}
\underset{x}{\text{minimise}} \quad & f(x) \\
\text{subject to} \quad & Ax = b \\
& g(x,\theta^{(i)}) \leq 0, \quad i = 1,\dots,N
\end{aligned}
\tag{2.27}
$$

which will be convex if the function g is convex in x. This method is attractive because it often recovers a convex problem in many cases that the chance-constrained problem is nonconvex. 'Scenario' problems of this type have been present in the literature for quite some time, and have been given a sound theoretical basis in the form of probabilistic bounds that the solution of (2.27) is feasible for the chance-constrained problem (2.21). Specifically, if we assume that the samples $\theta^{(i)}$ are independently drawn, and we define $V(x) = \mathbb{P}\{g(x,\theta) > 0\}$, and if $\dim(x) = n$, then the bound

$$
\mathbb{P}^N\{V(x^*) > \varepsilon\} \leq \sum_{i=1}^{n-1} \binom{N}{i} \varepsilon^i (1-\varepsilon)^{N-i}
\tag{2.28}
$$

holds for the optimum x^* of (2.27), where $\mathbb{P}^N$ denotes the N-fold product probability measure [CG08]. The right-hand-side of this expression can be made as small as desired by increasing N, implying that we can solve the chance-constrained problem to any desired confidence level by choosing N sufficiently large. The expression on the right-hand-side is, in fact, the cumulative distribution of a binomial random variable. Functions to compute it are built in to many common mathematical software packages.

In [Cal10] this result was extended to the case that some of the sampled constraints are purposely violated. This allows the optimal value of the cost to be improved by removing some of the sampled constraints from the optimisation. This procedure is further considered in [CG11] where a simple 'greedy' method is advocated for removing sampled constraints one at a time. In particular, the optimisation becomes

$$\begin{aligned} \underset{x}{\text{minimise}} \quad & f(x) \\ \text{subject to} \quad & Ax = b \\ & g(x, \theta^{(i)}) \leq 0, \quad \forall i \in \mathcal{A}(\{\theta^{(i)}\}_{i=1,\ldots N}) \end{aligned} \tag{2.29}$$

where the set-valued function $\mathcal{A} : \Omega^N \to P(\{1, \ldots, N\})$ represents some procedure used to select $N - r$ of the sampled constraints to be 'active' in the optimisation. In this case, it is possible to prove the bound:

$$\mathbb{P}^N\{V(x^*) > \varepsilon\} \leq \binom{n+r-1}{r} \sum_{i=1}^{n+r-1} \binom{N}{i} \varepsilon^i (1-\varepsilon)^{N-i} \tag{2.30}$$

A key point here is that this bound holds for arbitrary removal procedures, so that any algorithm or rule can be used to determine which samples to remove. Later in the thesis, we exploit this in MPC by constructing a simple but effective heuristic for constraint removal.

2.4 Software

There are several toolboxes designed to aid in the specification of convex optimisation problems, and process them into forms that can be handled by standard solvers. Perhaps the most well-known examples are CVX [GBY08] and Yalmip [Lö04]. Both of these are free toolboxes designed to work with MATLAB, although a version of CVX that works with the general purpose programming language Python is also available [DCB14]. Yalmip was used in the preparation of this thesis, and in particular the robust optimisation module designed to automatically find convex equivalents of robust optimisation problems was very useful [Lö12].

There are too many optimisation solvers available to list them all here (indeed, the online documentation for Yalmip alone lists around 50 at time of writing), so we will restrict ourselves to mentioning the two used in preparing the present thesis. SeDuMi [Stu99] is a useful, and free, general purpose solver for medium-sized LP, SOCP and SDP problems that uses a primal-dual interior point method. Gurobi [O$^+$12] is a commercial solver for LP, QP, SOCP and mixed-integer problems that is freely available for users in academia. It contains a variety of different solution methods including both active-set and interior-point methods and tends to be faster than SeDuMi for large or sparse problems. In addition, for LP and QP problems it can be 'warm-started' from a known feasible solution, which can sometimes reduce computation time considerably in MPC applications.

For manipulations with polyhedra, the free MATLAB toolbox MPT is often useful [KGBM04]. MPT is designed as a toolbox for MPC design, mostly for linear and hybrid systems, and also has the capability to find explicit solutions of LPs and QPs as a function of parameters present in the problem.

3 Model predictive control

3.1 Introduction

This chapter is a brief introduction to the field of model predictive control, which is a methodology for solving constrained optimal control problems. Much of the reason for the success of MPC lies in the fact that it is capable of giving a good approximate solution to the optimal control problem, even in cases where the optimal control is difficult to calculate. We first review some results from the theory of MPC applied to linear time-invariant systems and give a stabilising controller for this case. This is done in some detail for the specific case of linear constraints and quadratic cost functions, as the resulting controller contains many of the elements that will be present in the MPC controllers developed later in the thesis. We then give some more general conditions for stability in MPC, before introducing the more difficult problems of robust and stochastic MPC involving additive and parametric uncertainty.

3.2 Linear quadratic control

We begin the overview of MPC by considering the discrete-time LTI system

$$x_{t+1} = Ax_t + Bu_t \tag{3.1a}$$

$$y_t = Cx_t + Du_t \tag{3.1b}$$

where the control objective at each time t is to minimise the l_2 norm of the output considered from t to ∞, that is we introduce a cost function:

$$V_t = \sum_{k=t}^{\infty} y_k^T y_k \tag{3.2}$$

If there are no constraints on the state or the input of this system, the solution is the well-known linear quadratic regulator. For simplicity, assume that $C^T D = 0$ so that the cost function contains no cross-product terms. Then we have the following optimal controller, which is the discrete-time analogue of a result due to Kalman [K+60].

Proposition 3.1. *The optimal unconstrained controller for (3.1) that minimises (3.2) is given by the linear feedback $u_t = Kx_t$ where*

$$K = -(B^T PB + D^T D)^{-1} B^T PA \tag{3.3}$$

and P solves the algebraic Riccati equation:

$$P = A^T PA + C^T C - A^T PB(B^T PB + D^T D)^{-1} B^T PA \tag{3.4}$$

The solution P of (3.4) is a unique positive definite matrix if $D^T D$ is positive definite and (C, A) are an observable pair. In addition, the optimal value of the cost function is $V_t^* = x_t^T Px_t$ for all times t, and this is a Lyapunov function for the closed-loop system so that we have asymptotic stability of the origin.

3.3 Linear MPC

When linear constraints are included in the problem, the situation becomes more complex as the optimal controller is no longer a simple linear feedback controller. Instead, we consider representing the optimal control inputs as the solution of a constrained optimisation problem. Suppose the system has a second output z_t, given by

$$z_t = Fx_t + Gu_t \tag{3.5}$$

and we require $z_t \leq 1$ for all times t. Now suppose that we search for the optimal input expressible in the form $u^k = c^k$ for $k < N$ and $u^k = Kx^k$ for $t \geq N$. The intuition behind this choice of input is that after some finite time, the constraints should not affect the solution and the optimal control will be identical to that of the unconstrained case. This leads to the semi-infinite optimisation problem:

$$\begin{aligned}
\underset{c^k}{\text{minimise}} \quad & \sum_{k=t}^{t+N-1} ((x^k)^T C^T C x^k + (c^k)^T D^T D c^k) + (x^N)^T P x^N && \text{(3.6a)}\\
\text{subject to} \quad & x^t = x_t && \text{(3.6b)}\\
& x^{k+1} = Ax^k + Bc^k, \;\; k = t, \ldots, t+N-1 && \text{(3.6c)}\\
& Fx^k + Gc^k \leq 1, \;\; k = t, \ldots, t+N-1 && \text{(3.6d)}\\
& (F+GK)(A+BK)^i x^{t+N} \leq 1, \;\; i = 0, \ldots, \infty && \text{(3.6e)}
\end{aligned}$$

This optimisation cannot be solved in its current form due to the infinite number of constraints (3.6d). However, as long as $A + BK$ is a stable matrix (which is certain if K is chosen as in Proposition 3.1), and the pair $(A+BK, F+GK)$ is observable, it is shown in [GT91] that this infinite set of constraints is equivalent to the constraints

$$(F+GK)(A+BK)^i x^{t+N} \leq 1, \;\; i = 0, \ldots, M \tag{3.7}$$

for some finite, computable M. This set of constraints is said to define the maximal admissible set (MAS) of the system (3.1) under the feedback control $u_t = Kx_t$. Hence, for a given x_0, we have a finite quadratic programming problem to compute our control sequence:

$$\begin{aligned}
\underset{c^k}{\text{minimise}} \quad & \sum_{k=t}^{t+N-1} ((x^k)^T C^T C x^k + (c^k)^T D^T D c^k) + (x^N)^T P x^N && \text{(3.8a)}\\
\text{subject to} \quad & x^t = x_t && \text{(3.8b)}\\
& x^{k+1} = Ax^k + Bc^k, \;\; k = t, \ldots, t+N-1 && \text{(3.8c)}\\
& Fx^k + Gc^k \leq 1, \;\; k = t, \ldots, t+N-1 && \text{(3.8d)}\\
& (F+GK)(A+BK)^i x^{t+N} \leq 1, \;\; i = 0, \ldots, M && \text{(3.8e)}
\end{aligned}$$

Under the standard assumptions that $D^T D$ is positive definite and (C, A) observable, this is a QP with a positive definite cost function, so the solution can be found in polynomial time using an interior point algorithm. Moreover, if K and P are chosen as in Proposition 3.1, it is shown in [SR98] that the solution to this QP is the solution to the infinite horizon constrained optimal control problem when N is sufficiently large.

Proposition 3.2. *If K and P satisfy (3.3) and (3.4), then for some sufficiently large but finite N, the optimal constrained control for (3.1) that minimises (3.2) subject to the constraints $Fx_t + Gu_t \leq 1$ is given by $u_t = (c^t)^*$ for $k < N$ and $u_t = Kx^t$ for $k \geq N$, where $(c^t)^*$ is given by the solution of (3.8).*

Remark 3.3. It is common to simplify (3.8) by eliminating the variables x^k for $k = 1, \ldots, N$ using the equality constraints (3.8b), leading to a 'condensed' formulation of the MPC problem. If $F + GK$ has more than one row, then it is also possible that some of the constraints given by (3.8d) may be redundant and could be removed.

It would not be desirable simply to solve for the optimal control once and then implement the entire computed sequence u^k, as there could be no feedback from future measurements of the state and hence, the controller could not compensate for any deviations of the actual state from the predicted state. The ad-hoc solution to this in MPC is to solve (3.8) at each time t with the initial condition $x^t = x_t$, and then apply only the first element of the resulting control sequence.

Algorithm 3.1 (LQ MPC for LTI systems)*.* For each t, solve (3.8) and set $u_t = (c^t)^*$.

If N is chosen large enough that the computed control sequence is equal to the infinite horizon optimal control, then (3.8a) will be equal to the infinite horizon cost and must decrease over time - this implies that it is a Lyapunov function, and the system is stable. In practice, we may want to choose N smaller than the value for which (3.8) gives the infinite horizon optimal control, perhaps because we desire a smaller optimisation problem that can be solved in less time. The standard method to prove stability in MPC in such cases is to show that the optimal value of the cost function used in the optimisation must not increase as t increases [MR09, Section 2.4.3].

In this instance we can readily show that the cost is a Lyapunov function. From (3.6d) the optimal solution to (3.8) at time t has a corresponding feasible point of the optimisation at time $t + 1$ given by choosing $c^k_{t+1} = c^k_t$ for $k = t, \ldots, N - 1$ and $c^{t+N}_{t+1} = Kx^{t+N}_{t+1}$. The optimal cost at time $t + 1$ must be no greater than the cost given by choosing $c_{k+1} = c^*_k$ and $c_N = Kx_N$ by the positive definiteness of D^TD and observability of (C, A), which must be less than the optimal cost at time t. Hence, by induction, the optimal value of the cost function is strictly decreasing with t. As it is positive definite, this forms a Lyapunov function for the state x_t. This yields the following result:

Proposition 3.4. *For any N, and sufficiently large M, Algorithm 3.1 asymptotically stabilises (3.1) so that $x_t \to 0$ as $t \to \infty$ and satisfies $Fx_t + Gu_t \leq 1$ for all $t = 0, \dots, \infty$.*

The procedure that we used to prove stability and constraint satisfaction is worth summarising, as it is repeated often in later parts of the thesis. Firstly, the use of the MAS (3.7) as a terminal constraint implies that any the optimisation is recursively feasible, and ensures satisfaction of constraints for all time. Secondly, the cost function is positive definite and must decrease for the solution at the next time, proving stability.

3.3.1 General linear MPC

The linear quadratic MPC of the previous section can readily be generalised to different cost functions and constraints, while retaining the results on stability and constraint satisfaction under some suitable assumptions. Suppose a general cost function is defined using a stage cost $l(x, u)$, a terminal cost $V_N(x)$, and there are constraints on the state and input given by $(x^k, c^k) \in \mathbb{F}$ and a terminal set $x^N \in \mathbb{T} \supseteq 0$, then a generic MPC optimisation is:

$$\begin{aligned} \underset{c^k}{\text{minimise}} \quad & \sum_{k=0}^{N-1} l(x^k, c^k) + V_N(x^N) && (3.9a) \\ \text{subject to} \quad & x^t = x_t && (3.9b) \\ & x^{k+1} = Ax^k + Bc^k, \;\; k = t, \dots, t+N-1 && (3.9c) \\ & (x^k, c^k) \in \mathbb{F}, \;\; k = t, \dots, t+N-1 && (3.9d) \\ & x^N \in \mathbb{T} && (3.9e) \end{aligned}$$

This finite horizon optimal control problem leads to an MPC algorithm given by:

Algorithm 3.2 (MPC for LTI systems)*.* For each t, solve (3.9) and set $u_t = (c^t)^*$.

Then there is the following result, given for instance in [MRRS00]. The proof proceeds along the usual lines of showing first recursive feasibility and then that the cost must decrease, so we omit it.

Proposition 3.5. *If there exists a nominal controller $u = \kappa_f(x)$ such that the following assumptions hold:*

1. $Ax + B\kappa_f(x) \in \mathbb{T}, \quad \forall x \in \mathbb{T}$
2. $(x, \kappa_f(x)) \in \mathbb{F}, \quad \forall x \in \mathbb{T}$
3. $V_N(Ax + B\kappa_f(x)) - V_N(x) \leq -l(x, \kappa_f(x)), \quad \forall x \in \mathbb{T}$
4. $V_N(0) = 0, \;\; V_N(x) \geq 0, \quad \forall x \in \mathbb{T}$

Then Algorithm 3.2 asymptotically stabilises (3.1) so that $x_t \to 0$ as $t \to \infty$ and satisfies $(x_t, c_t) \in \mathbb{F}$ for all $t = 0, \dots, \infty$.

Importantly, if the functions $l(x, c)$ and $V_N(x)$ and the sets $\mathbb{F}$ and $\mathbb{T}$ are convex then (3.9) is a convex optimisation problem for which efficient solution methods exist. It is worth noting that this would not be the case if nonlinear dynamics were used in place of (3.9b), as optimisation problems with nonlinear equality constraints are not convex.

3.4 Robust MPC for additive disturbances

Now consider the linear system of (3.1) with an additive disturbance $w_t \in \mathcal{W}$. We will suppose that we wish to apply constraints such as $Fx_t + Gu_t \leq 1$ such that they hold robustly, that is for all possible $w_t \in \mathcal{W}$.

$$x_{t+1} = Ax_t + Bu_t + w_t \tag{3.10a}$$

$$y_t = Cx_t + Du_t \tag{3.10b}$$

It is well known that in the presence of additive uncertainty, feedback must be applied when forming predictions in MPC to effectively apply constraints. A good discussion of this point is available in [MR09, Section 3.1.2].

Following on from a convexity result in [Löf03a], it is demonstrated in [GKM06] that it is possible to optimise over feedback policies of the form

$$u^k = \sum_{k=t}^{t+N-1} L^k w^k + c^k \tag{3.11}$$

using a convex optimisation with a number of variables that grows quadratically in N, and that this can outperform predicted inputs based on perturbations of linear feedback laws. This input parameterisation is quite versatile, and can be used for minimisation of expected-value costs as well as costs that guarantee an l_2 norm bound [Gou07]. It is also applicable to the case of output feedback, giving a very effective general framework for constrained control under additive uncertainty.

In turn [RKC$^+$12] gives a more complex formulation for linear systems with additive disturbances based on Tube MPC. This also uses a number of free variables that also grows quadratically in N and will in general outperform controllers using (3.11), at the cost of including a larger number of variables and constraints in the online optimisation. Finally, it has recently been shown that it is possible to solve the constrained dynamic programming problem corresponding to the dynamics (3.10) online using active set methods [BCK14, BCK16] which is less conservative in general than all previously mentioned implicit techniques when it is applicable - indeed, this gives the optimal controller.

3.4.1 Tube-based MPC

We now consider in some detail the 'Tube MPC' approach to constrained control of (3.10) which considers the predicted input to be given by an expression of the form

$$u^k = c^k + K(x^k - s^k) \tag{3.12}$$

where s^k is the predicted state of a nominal system, hence introducing feedback into the predictions of the future state. This leads to a method of robust MPC that, in this thesis, we shall generalise to parameter-uncertain systems.

To begin, we again consider the linear system with additive uncertainty (3.10) and suppose that we wish to minimise the predicted l_2-norm of the output y_t subject to the robust constraint:

$$Fx_t + Gu_t \leq 1 \tag{3.13}$$

The predicted dynamics (3.10) can be decomposed into a nominal and uncertain part by defining $x^k = s^k + e^k$ with s^k satisfying:

$$s^{k+1} = As^k + Bu^k \tag{3.14}$$

If a predicted input given by

$$u^k = c^k + K(x^k - s^k) \tag{3.15}$$

is applied to the system, then the "error" e^k evolves as

$$e^{k+1} = (A + BK)e^k + w^k \tag{3.16}$$

$$
\begin{aligned}
\underset{s^k, c^k}{\text{minimise}} \quad & \sum_{k=t}^{t+N-1} ((s^k)^T C^T C s^k + (c^k)^T D^T D c^k) + (x^N)^T P x^N && \text{(3.17a)}\\
\text{subject to} \quad & s^t \in \{x_0\} \oplus \mathcal{S} && \text{(3.17b)}\\
& s^{k+1} = As^k + Bc^k, \;\; k = t, \ldots, t+N-1 && \text{(3.17c)}\\
& Fs^k + Gc^k + \bar{f} \leq 1, \;\; k = t, \ldots, t+N-1 && \text{(3.17d)}\\
& Vs^{t+N} + \bar{v} \leq 1 && \text{(3.17e)}
\end{aligned}
$$

Once again, this is a convex QP. It gives the MPC algorithm:

Algorithm 3.3 (LTI Tube MPC)*.* At each t, solve (3.17) and set $u_t = (c^t)^* + K(x^t - (s^t)^*)$.

The use of the terminal disturbance invariant set along with the variable nature of s_0 ensures recursive feasibility, and hence constraint satisfaction for all time. As a result, the cost function also decreases in closed loop and we can prove a stability result. Unlike the case of MPC without uncertainty, the cost function is zero not just at the origin but on the whole set $\mathcal{S}$, so that we may only conclude that the state converges to $\mathcal{S}$ as $t \to \infty$. This is, however, completely natural given that we have a persistent additive disturbance.

Proposition 3.6. *For any N, Algorithm 3.3 ensures that (3.10) satisfies $Fx_t + Gu_t \leq 1$ for all $t = 0, \ldots, \infty$ and guarantees that $x_t \to \mathcal{S}$ exponentially fast as $t \to \infty$.*

Tube MPC of this type has been extended to handle more general dynamics with additive disturbances, such as the nonlinear dynamics $x_{t+1} = f(x_t, u_t) + w_t$, and to moving horizon estimation and output feedback control [MR09]. More general tube parameterisations have also been considered, such as the 'homothetic' parameterisation in [RKFC12].

The key generalisation to these methods that we will make in this thesis is that we develop more sophisticated ways to apply constraints to the tube cross-sections than using the tightening parameters $\bar{f}$ and $\bar{v}$. This will lead to much more variety in the class of set parameterisations that can be used for the tube and make the method applicable to a wider class of problems, in particular systems described by difference inclusions and with parameter variations.

3.5 Robust MPC for parametric uncertainty

A very different uncertainty description is obtained if, instead of assuming a linear system with an additive disturbance, we allow the matrices of the linear system to be functions of an unknown time-varying parameter $\theta_t \in \Theta$, giving the LDI:

$$x_{t+1} = A(\theta_t)x_t + B(\theta_t)u_t \tag{3.18a}$$

$$y_t = C(\theta_t)x_t + D(\theta_t)u_t \tag{3.18b}$$

It is often the case in applications that θ is a nonlinear function of the states and then (3.18) can be used to conservatively represent a wide variety of nonlinear dynamics.

3.5.1 FIR models

Some of the earliest techniques to handle parametric uncertainty in MPC used finite impulse response models

$$y_t = \theta^T \underline{u}_t \tag{3.19}$$

where the notation $\underline{u}_t$ denotes a vector containing the previous M inputs for some positive integer M. These form a subclass of systems (3.18) corresponding to the case that the A matrix is nilpotent and there is uncertainty in either $B(\theta)$ or $C(\theta)$, and were first introduced in [CM87] for the robust MPC of system subject to input constraints. The problem considered was the constrained minimisation of the maximal deviation from a reference trajectory.

This was later improved upon in [ZM93] where guarantees of robust stability were provided, then extended to consider systems in a norm-bounded model set in [BCH98]. As pointed out in the introduction to [Löf03b], the interesting characteristic of these latter formulations is that the worst-case uncertainty is accounted for analytically, leading to efficient robust control algorithms.

3.5.2 LMI methods

Possibly the first receding horizon control technique for (3.18) appears in [KBM94] and [KBM96], in which an SDP is formulated for the cases that the uncertainty set Θ is either polytopic or given by a structured norm-bounded uncertainty, and the system must satisfy linear constraints on the state and the input. At each time t, this optimisation computes a stabilising linear feedback law

$$u^k = Kx^k \tag{3.20}$$

that ensures the current state lies in a level set of a quadratic Lyapunov function that is a subset of the constraint sets. A standard change of variables from [BEGFB94] is used to render the resulting problem convex. Due to the ellipsoidal bounding, the algorithm can be quite conservative in terms of constraint handling (this can be seen in Figure 9 of [KBM96], where the input only reaches a peak value of around 0.14 despite inputs of a much larger magnitude being allowed under the constraint $|u| \leq 1$). An improvement to the algorithm for the case of polytopic uncertainty is given in [CGM02] where, instead of the use of a single quadratic Lyapunov function, a Lyapunov function is associated with each vertex of the uncertainty.

These ideas can also be extended to LPV systems, where the parameter θ_t can be measured, and so is available to the controller at time t. This was originally done in

[LA00], which considers the optimisation of parameter dependent (or gain-scheduled) state feedback laws:

$$u^k = K(\theta^k)x^k \tag{3.21}$$

The approach was later improved upon in [WSS04] where a parameter-dependent Lyapunov function was introduced to reduce conservatism. Despite the various improvements made on the method of [KBM94], all of these approaches rely on bounding the state in ellipsoidal control-invariant sets online, and hence they are somewhat conservative when used to apply polyhedral constraints. This will be seen later in the thesis when this class of methods is compared with tube MPC techniques.

A different class of LMI optimisation methods is given in [Löf03b] for systems that have uncertainty only in the $B(\theta)$ matrix, or for systems with LFT uncertainty, when it is desired to minimise a quadratic cost function subject to constraints. It is shown that, by using an epigraph formulation of a minimax optimisation problem, a minimax MPC optimisation can be conservatively approximated using a semidefinite programming relaxation based on the S-procedure. The resulting MPC is quite effective when dealing with input constrained systems, but as noted in [Löf03b, Section 11.3], can lead to a large number of LMI constraints when dealing with state constraints and LFT uncertainties.

3.5.3 Prestabilised predictions

An alternative to optimising feedback laws online is to construct perturbations on a fixed feedback law, so that the input is parametrised as

$$u^k = Kx^k + c^k \tag{3.22}$$

and only the parameter c^k is determined online. This is analogous to the parameterisation of feedback policies used in tube MPC for additive disturbances. Although this is more conservative in general than optimising over the feedback matrix K, it may lead to more straightforward online optimisations. Following [KRS00], the input parameterisation (3.22) leads to an autonomous formulation of the predicted dynamics

(3.18) as

$$\begin{bmatrix} x^{k+1} \\ \underline{c}^{k+1} \end{bmatrix} = \begin{bmatrix} A(\theta^k) + B(\theta^k)K & B(\theta^k)S \\ 0 & T \end{bmatrix} \begin{bmatrix} x^k \\ \underline{c}^k \end{bmatrix} \tag{3.23}$$

where $\underline{c}^t$ is a vector containing the predicted values of c^t over the prediction horizon, S is a matrix such that $c^t = S\underline{c}^t$ and T is a shift matrix so that $\underline{c}^{t+1} = T\underline{c}^t$. It is possible to bound the maximum value of a predicted cost for these dynamics using a convex quadratic function, and minimise this bound online. Representing the autonomous dynamics (3.23) in shortened form as $\zeta^{k+1} = \Psi\zeta^k$, we can find such an upper bound by finding a matrix W satisfying

$$W - \Psi^T W \Psi \succeq \bar{Q} \tag{3.24}$$

where $\bar{Q}$ is a matrix such that $(\zeta^k)^T \bar{Q} \zeta^k$ gives a stage cost for the autonomous system (3.23). Minimising the trace of the matrix W subject to (3.24) is an SDP, and typically leads to a good bound.

We can then build an MPC controller for (3.23) with guaranteed stability properties by finding an invariant set $\mathcal{Z}$ of this autonomous system, and ensuring that the initial state and input lie within this set. This leads to an online optimisation:

$$\begin{aligned} \underset{\underline{c}_t}{\text{minimise}} \quad & \zeta_t^T W \zeta_t \\ \text{subject to} \quad & \zeta_t \in \mathcal{Z} \end{aligned}$$

In [KRS00] an ellipsoidal invariant set is considered, leading to an online optimisation with a single ellipsoidal constraint on x_t and $\underline{c}_t$. An interesting variation on this approach is given in [CK05] where the matrices S and T are optimised offline to give the largest ellipsoidal constraint set. These methods are quite computationally efficient as most of the necessary computation, namely computing the cost bound and ellipsoidal invariant set using SDP, is carried out offline. But once again, the use of ellipsoidal bounding can make constraint handling very conservative.

As a remedy for this conservatism with respect to constraints, polyhedral admissible sets for (3.23) as used in [PRSDM05c], which uses the maximal admissible set in particular. This is less conservative with respect to constraints and typically increases the size of the feasible region of the resulting MPC optimisation. Unfortunately, the MAS of (3.23) has a number of constraints that grows exponentially with the length N of the prediction horizon, so that this approach rapidly becomes impractical as N is increased. An extension to this work is found in [PRSDM05d], which considers a method of interpolating between fixed feedback gains in order to give an increased region of attraction of the resulting MPC controller.

Tube MPC ideas have also been extended to LDI systems in [ECK12a] and [RC13]. In both papers, the optimisation considers control policies that are perturbations of a fixed feedback and in both cases the focus is on polytopic uncertainty. The advantage of these tube formulations is that the online computation is tractable, being a QP with a number of variables and constraints that grows only linearly with N. In addition the use of polyhedral, rather than ellipsoidal, bounding allows the closed-loop system to operate closer to constraint boundaries.

The difference between [ECK12a] and [RC13] lies in the tube parameterisation used, in that the latter paper considers a restrictive formulation where a fixed set is translated and scaled around the state space. In contrast, the former considers sets of the form $\{Vx \leq \alpha\}$ with α a variable in the online optimisation. This leads to quite large optimisation problems in general, though entails only a little conservatism. Both methods are actually special cases of the MPC controllers that we develop in Chapter 5 of this thesis, where we demonstrate that it is possible to use a wide variety of different tube parameterisations and handle arbitrary uncertainty sets. This will allow us to consider the possibility of designing set parameterisations, and therefore tube MPC controllers, suited to specific applications.

3.6 Stochastic MPC

We now consider stochastic techniques that can take into account information that may be available on the distribution of uncertainty in an MPC problem. Research interest in stochastic methods is not confined to MPC, and there exist many control analysis and design methods in the literature that exploit a stochastic uncertainty description in various ways. A good overview is available in the review paper [CDT11] and the book [TCD12].

An early paper to consider stochastic uncertainty descriptions in MPC is [BSW$^+$02], which considers a linear state-space system with state and input constraints, where the system is subject to an additive stochastic disturbance. Expected-value cost functions are handled by minimising an empirical mean obtained by sampling and constraints by including a loss function in the cost function to penalise constraint violations. In the associated thesis [Bat04], the author notes that this is necessary if the uncertainty is unbounded, as in that case it is impossible to satisfy hard constraints on state variables.

An expected-value cost is also present in [lPMBA05], which additionally considers parametric stochastic uncertainty. Samples of the uncertainty are used to generate a 'scenario tree' of future possible states. One drawback of this is that the number of generated scenarios grows exponentially with the length of the prediction horizon, so that we may expect the methods to become intractable as N increases. One particularly interesting feature is that in closed-loop the cost function need only decrease in expectation, so that the system is stable in a weaker sense than the usual sense in robust MPC, where the cost is usually required to decrease monotonically for all uncertainty realisations.

3.6.1 Chance-constrained MPC

A recent development in stochastic MPC is the introduction of methods that can handle chance constraints of the form:

$$\mathbb{P}\{Fx_t \leq b\} \geq 1 - \varepsilon \tag{3.25}$$

This problem was first considered in [SN99] in which finite impulse response models were used and the chance-constrained problem solved by exploiting an easily-solvable special case of the constraint (3.25) – when considering an individual chance constraint with normally-distributed uncertainty. The problem becomes substantially more difficult when joint constraints are considered, accordingly [VHSB01] considers state space models with a disturbance input of zero-mean Gaussian white noise and a joint chance constraint and derives a conservative LMI optimisation scheme. An improved method for the same problem appears in [BO09], in which the joint chance constraint is simplified using Boole's inequality to introduce risk parameters ε_i corresponding to each chance constraint. It is shown that the problem of finding the optimal parameters ε_i is convex and can be solved concurrently with the MPC optimisation.

A general method of MPC for additive stochastic disturbances is presented in [OJM08], in which an affine disturbance feedback policy is used with a conservative convex approximation of the chance constraint. This was later used in a successful control design for building climate control [OPJ$^+$10]. Also considering additive disturbances, [KCRC10] shows that when considering individual chance constraints, the problem can be solved efficiently by performing a univariate integration offline and incorporating a tightening parameter into the MPC. This offline integration does not introduce any conservatism with regard to constraint handling, and the assumption of finitely supported uncertainty along with a 'one-step-ahead' chance constraint allows for a recursive feasibility guarantee. Similar ideas are used in [KGCO11] to give a controller using affine disturbance feedback, and a tube MPC controller is given in [CKRC11], both for the same problem. Also using affine disturbance feedback, the more difficult problem of distributionally robust constrained control of a linear system with a stochastic disturbance is solved in [VPKGM13], which assumes that only the mean and variance of the uncertainty distribution are known.

Noting that a strict guarantee of recursive feasibility can be a restrictive condition for chance constraints, [KGOJ14] presents a framework in which the average number of constraint violations over time is controlled and satisfies some asymptotic bound. With similar motivations, [LADT15] considers relaxing the recursive feasibility requirement for the constraints over the prediction horizon, but then recovering a recursive feasibility guarantee by imposing a constraint on the first predicted state. This gives a larger feasible region, at the possible expense of having to perform difficult computations of reachable and invariant sets offline.

An early paper to consider parametric uncertainty in a chance-constrained MPC scheme was [KCT04], in which a specific sustainable development problem was considered. This was later refined into a stochastic MPC formulation valid for FIR models in [CCK06], which had guaranteed recursive feasibility and stability properties due to the inclusion of a terminal constraint and an assumption that the uncertainty was finitely supported. Later work by the same authors considered the use of ellipsoidal invariant sets computed offline for an autonomous formulation of the system dynamics, which allows chance constraints to be handled using a single convex online optimisation [CKW09a, CKW09b], although the use of ellipsoidal sets may itself introduce conservatism. More recently, an approach was presented using polyhedral tubes and using sampling to handle constraints approximately [ECK12b]. The stochastic tube MPC methods of this thesis can be viewed as a natural progression of this latter paper. Many of these techniques can also be found in the recent book [KC15].

Confidence bounds from the stochastic programming literature on sampled optimisation problems were applied to MPC of an LPV system in [CF13], although the method appears to be somewhat conservative with respect to the constraints in the example given. A less conservative controller using sampling is given in [SFFM14], which builds on an extension to the confidence results for sampled programs that is given in [SFM13]. Further improvements appear in [ZGS+15], which shows that the required number of samples can often be reduced by considering the structure of the uncertainty, and in [GZM+16], which extends the approach to nonconvex problems. Although yielding good performance, these approaches assume recursive feasibility of the optimisation problem rather than enforcing it, and the need for resampling at each time instant appears to make any kind of recursive feasibility guarantee (and hence, constraint satisfaction guarantee in closed-loop) impossible. This is one of the issues that we will address in this thesis, as we show that stochastic MPC controllers using online sampling of constraints can be recursively feasible.

3.7 Explicit MPC

It was shown in [BMDP00], [BMDP02] and [BBM+02] that the solution to the constrained MPC problem for linear systems can be written explicitly as a piecewise affine function, and computed using the technique of multiparameteric programming. This is an appealing idea as it suggests that it is possible to precompute the controller ahead of time, and implement it by doing a simple 'lookup' of the needed affine control function for a particular value of x_t, which can be implemented efficiently using a binary search tree [TJB03]. This does nothing to simplify the optimisation problem if it is difficult to solve, but it does give the luxury of solving the problem offline where computation time is cheap.

This class of methods has been extended to more general systems than linear ones, firstly for mixed logical dynamical systems in [BBM00], and later to uncertain linear systems in [BBM03], where both parameter uncertainty and additive disturbances are considered. The LMI-based algorithm of [KBM96] is given an explicit formulation in [lPMBF04] where it is solved using approximate multiparametric techniques.

One great advantage of multiparametric optimisation techniques is that, as they find an optimal objective function and optimal decision variable in an explicit form, they can be used to solve the value iteration arising in discrete-time dynamic programming [Bel57]. This technique is presented in [Die04] and [Ber05] for instance, and extended to hybrid systems in [BBBM05].

This approach to computing an optimal constrained control has also been applied to systems with parametric uncertainty. In [BLM12] it is shown that the control of LPV systems can be accomplished by considering a dynamic programming optimisation that can be solved by multiparametric programming. Another recent paper [KSFD13] uses these methods to minimise a quadratic cost when the uncertainty is limited to the matrix $A(\theta)$ and the other system parameters are known.

The general drawback of these approaches is that the number of polyhedral regions that must be considered to compute the explicit solution may quickly become very large for longer prediction horizons and for systems with many states. This is a difficulty, as even when the resulting controllers can be computed the large storage requirements for the solution may render implementation impossible. For this reason we do not consider explicit techniques further in this thesis, but we note that in principle the controllers we develop could be computed using a multiparametric QP solver when the uncertainty is polytopic.

Part II

Robust Tube MPC

4 Robust inclusions

4.1 Introduction

One of the more promising recent approaches to constraint handling in nonlinear robust MPC is by the construction of a prediction tube [MKVWF11]. In tube MPC the controller conservatively bounds the future states of the system in polytopic sets, which can greatly simplify constraints yet retains strong theoretical properties such as recursive feasibility and closed-loop stability [MSR05].

In this chapter, a method is presented of satisfying set inclusions in optimisations based on the theory of linear programming duality. This will allow us to construct flexible parameterisations of prediction tubes for systems with parametric uncertainty. This can be contrasted with the analysis based around Minkowski sums and differences considered in [MSR05], which is more suited to the case of additive uncertainty and simple parameterisations of the tube cross-sections. The new approach relies on being able to choose, offline, a matrix function '$H(\theta)$' that is used to define robust constraints on some parameters α that specify a polyhedron.

The question that naturally arises is what offline choice of H is best. Surprisingly, when applying an inclusion like $\mathcal{P}[V,g] \subseteq \mathcal{P}[F,b]$ it turns out that there often exist choices of H that do not restrict the sets $\mathcal{P}[V,g]$ that satisfy this inclusion. This is possible partly because there is often redundancy in the parameterisation of the sets, such that it is possible for $\mathcal{P}[V,g_1]$ and $\mathcal{P}[V,g_2]$ to represent the same set even when $\alpha \neq \beta$, and conservatism in the feasible set of g need not imply conservatism with regard to the sets $\mathcal{P}[V,g]$ and the inclusion $\mathcal{P}[V,g] \subseteq \mathcal{P}[F,b]$.

Although this conservatism has been known about for some time, and some heuristic methods of designing H known based on minimising 1-norms [ECK12a, FKC13], this chapter includes the first formal analysis of this issue. As a result, many of the concepts and results, in particular all results of section 4.3, are entirely novel.

4.1.1 Notation

It will be useful to regard the dynamics of an uncertain system to be defined by a set-valued function $\Phi : \mathbb{R}^n \to P(\mathbb{R}^n)$. With a little abuse of notation, we define images and preimages of a set $\mathcal{A} \subseteq \mathbb{R}^n$ under this function as:

$$\Phi(\mathcal{A}) = \{x \in \mathbb{R}^n : \exists z \in \mathcal{A}, x \in \Phi(z)\}$$

$$\Phi^{-1}(\mathcal{A}) = \{x \in \mathbb{R}^n : \forall z \in \Phi(x), z \in \mathcal{A}\}$$

The set $\mathcal{A}$ will usually be a polyhedron. Wherever possible we avoid working with vertex representations of polyhedra, due to the inefficiency of such representations in high-dimensional spaces, instead considering polyhedra as the intersections of half-spaces. We write

$$\mathcal{P}[V, g] = \{x \in \mathbb{R}^n |\ Vx \leq g\}$$

so the rows of the matrix V give the facet normals of the polyhedron and g the inner products of those normals with any point on the corresponding facet.

For tube MPC, we will often need to stipulate that a particular polyhedron satisfies some robust linear constraints for all values of a parameter θ. Accordingly, we introduce a similar notation to the above to specify a set defined by robust constraints

$$\mathcal{R}_\Theta[F, b] = \{x \in \mathbb{R}^n |\ F(\theta)x \leq b(\theta),\ \ \forall\theta \in \Theta\}$$

where we now have functions $F : \Theta \to \mathbb{R}^{\dim(b)\times n}$ and $b : \Theta \to \mathbb{R}^{\dim(b)}$ from some 'uncertainty space' Θ. As this uncertainty space is arbitrary, we may assume these mappings are affine with no loss of generality.

Assumption 4.1. *The functions $F(\theta)$ and $b(\theta)$ are affine in θ.*

These functions define a polyhedron $\mathcal{P}[F(\theta), b(\theta)]$ for any fixed $\theta \in \Theta$, and it is immediate that $\mathcal{R}_\Theta[F, b]$ can be expressed as an intersection over this class of polyhedra.

Lemma 4.1. *The robust constraint set $\mathcal{R}_\Theta[F, b]$ can be expressed as*

$$\mathcal{R}_\Theta[F, b] = \bigcap_{\theta\in\Theta} \mathcal{P}[F(\theta), b(\theta)]$$

the intersection of all polytopes corresponding to values of $\theta \in \Theta$*.*

Being an intersection of polyhedra, $\mathcal{R}_\Theta[F, b]$ is convex but not itself polyhedral in general. Using the theory of support and gauge functions, it is possible to express this set explicitly as a convex set of constraints. We have

$$\mathcal{R}_\Theta[F, b] = \{x \in \mathbb{R}^n | \ F(\theta)x \leq b(\theta)\}$$

and if we assume (without loss of generality) a linear parameterisation for each row of $F(\theta)$ and $b(\theta)$ as $f_i(\theta) = \bar{f}_i + F_i\theta$ and $b_i(\theta) = \bar{b}_i + B_i\theta$ then these constraints can be expressed as

$$(\bar{f}_i + F_i\theta)^T x \leq \bar{b}_i + B_i\theta, \quad \forall\theta \in \Theta, \forall i$$

which, after a rearrangement, is:

$$\bar{f}_i^T x + \sup_{\theta\in\Theta} \theta^T (F_i^T x - B_i^T) \leq \bar{b}_i, \quad \forall\theta \in \Theta, \forall i$$

Using the definition of the support function, this is

$$\bar{f}_i^T x + \sigma_\Theta(F_i^T x - B_i^T) \leq \bar{b}_i, \quad \forall i$$

which can be written equivalently using the gauge function of the polar set of Θ:

$$\bar{f}_i^T x + \gamma_{\Theta^*}(F_i^T x - B_i^T) \leq \bar{b}_i, \quad \forall i$$

This last expression is equivalent to

$$\bar{f}_i^T x + \lambda_i \leq \bar{b}_i, \quad F_i^T x - B_i^T \in \lambda_i\Theta^*, \quad \forall i$$

which is a convex set of constraints for fixed λ_i, and simplifies to convenient forms for many standard uncertainty sets Θ. For instance, in the case that Θ is the unit 2-norm ball we have the second-order cone constraints:

$$\bar{f}_i^T x + \|F_i^T x - B_i^T\|_2 \leq \bar{b}_i, \quad \forall i$$

4.2 Conditions for robust inclusions

We now develop results that will allow us to enforce certain subset relations in the online MPC optimisation, and hence design a prediction tube. We recall a corollary of Farkas' Lemma that can be found in [BM07].

Lemma 4.2. *The polyhedral inclusion* $\mathcal{P}[V, g] \subseteq \mathcal{P}[F, b]$ *holds if and only if there exists a matrix* $H \in \mathbb{R}^{\dim(b)\times\dim(g)}$ *satisfying:*

$$H \geq 0, \;\; HV = F, \;\; Hg \leq b$$

This is a useful condition in its own right, and can be used in isolation to develop tube MPC constraints for systems with polytopic uncertainty [ECK12a, FKC13]. To deal with parametric uncertainty that may lie in more general convex sets, we would like a similar condition for inclusions of the form $\mathcal{P}[V, g] \subseteq \mathcal{R}_\Theta[F, b]$. This is fairly straightforward by considering Lemma 4.2 pointwise on Θ, yielding the following result.

Lemma 4.3. *The robust inclusion* $\mathcal{P}[V, g] \subseteq \mathcal{R}_\Theta[F, b]$ *holds if and only if there exists a function* $H : \Theta \to \mathbb{R}^{\dim(b)\times\dim(g)}$ *satisfying*

$$H(\theta) \geq 0, \;\; H(\theta)V = F(\theta), \;\; H(\theta)g \leq b(\theta)$$

for all $\theta \in \Theta$.

Although intuitively satisfying by analogy with Lemma 4.2, this is of little direct use as there is no indication of what form the function $H(\theta)$ might take. Fortunately, it turns out that if $F(\theta)$ and $b(\theta)$ are affine in θ, then $H(\theta)$ will also admit such an affine parameterisation. We state this as a theorem and note that for any basis for the vector space of uncertain $(F(\theta), b(\theta))$, we can find a corresponding basis of the vector space of uncertain $H(\theta)$.

Theorem 4.4. *Denoting the ith element of* θ *as* θ_i*, if* $F(\theta)$ *and* $b(\theta)$ *satisfy*

$$(F(\theta), b(\theta)) = \left(\bar{F}, \bar{b}\right) + \sum_{i=1}^{q} \left(F^{[i]}, b^{[i]}\right)\theta_i \tag{4.1}$$

and the inclusion $\mathcal{P}[V, g] \subseteq \mathcal{R}_\Theta[F, b]$ holds, then there exists $H(\theta)$ satisfying the conditions of Lemma 4.3 in the form

$$H(\theta) = \bar{H} + \sum_{i=1}^{q} H^{[i]}\theta_i \tag{4.2}$$

where $\bar{H}$ and $H^{[i]}$ satisfy:

$$\bar{H} \geq 0, \;\; \bar{H}V = \bar{F}$$
$$H^{[i]} \geq 0, \;\; (H^{[i]} - \bar{H})V = F^{[i]}$$

If we consider the parameterisation row-wise we find a straightforward corollary of the above that eliminates the explicit use of the summation sign, and is somewhat more convenient to work with when considering robust optimisation problems.

Corollary 4.5. *If the ith row of $F(\theta)$ and $b(\theta)$ can be expressed as*

$$f_i(\theta) = \bar{f}_i + F_i\theta, \quad b_i(\theta) = \bar{b}_i + B_i\theta \tag{4.3}$$

then if the inclusion $\mathcal{P}[V, g] \subseteq \mathcal{R}_\Theta[F, b]$ holds, there exists $H(\theta)$ satisfying the conditions of Lemma 4.3 with rows in the form

$$h_i(\theta) = \bar{h}_i + H_i\theta$$

where $\bar{h}_i$ and H_i satisfy:

$$\bar{h}_i \geq 0, \;\; \bar{h}_i^T V = \bar{f}_i^T$$
$$H_i \geq 0, \;\; H_i^T V = F_i^T$$

Together, Lemma 4.3 and Theorem 4.4 yield a set of robust constraints that are necessary and sufficient conditions for the set inclusion $\mathcal{P}[V, g] \subseteq \mathcal{R}_\Theta[F, b]$. Whether these inequalities can be solved efficiently depends greatly on the form of the uncertainty set Θ, so we now turn our attention to structured uncertainty of a few common types.

4.2.1 Polytopic uncertainty

Suppose that the uncertainty set Θ is a polytope

$$\Theta = \mathrm{Co}\{\theta^{(j)} |\ j = 1, 2, \ldots, p\}$$

then application of Lemma 4.3 and 4.4 yields the following:

Proposition 4.6. *The robust inclusion $\mathcal{P}[V, g] \subseteq \mathcal{R}_\Theta[F, b]$ holds if and only if there exist $H^{(j)}$ satisfying*

$$H^{(j)} \geq 0, \ \ H^{(j)}V = F(\theta^{(j)}), \ \ H^{(j)}g \leq b(\theta^{(j)})$$

for all $j = 1, 2, \ldots, p$.

The constraints of Proposition 4.6 are linear in $H^{(j)}$, hence it is possible to test if $\mathcal{P}[V, g] \subseteq \mathcal{R}_\Theta[F, b]$ by solving a single linear program.

4.2.2 Norm-bounded uncertainty

We now suppose that the set Θ is a p-norm ball, so that

$$\Theta = \{\theta |\ \|\theta\|_p \leq 1\}$$

and we will make use of the row-wise affine parameterisation (4.3), so that we have:

$$f_i(\theta) = \bar{f}_i + F_i\theta, \quad b_i(\theta) = \bar{b}_i + B_i\theta$$

Lemma 4.3 and Theorem 4.4 then lead to the following result.

Proposition 4.7. *If the parameterisation (4.3) holds and $\Theta = \{\theta |\ \|\theta\|_p \leq 1\}$, then $\mathcal{P}[V, g] \subseteq \mathcal{R}_\Theta[f^T, b]$ if and only if, for all i, there exist $\bar{h}_i$, H_i satisfying*

$$\bar{h}_i^T g + \|H_i^T g + B_i^T\|_{p^*} \leq \bar{b}_i$$

where $(p^*)^{-1} + p^{-1} = 1$ *so that we are working with the dual norm and*

$$\bar{h}_i \geq 0,\ \bar{h}_i^T V = \bar{f}_i^T$$
$$H_i \geq 0,\ H_i^T V = F_i^T$$

for all $i = 1, 2, \ldots, q$.

If $p = 1$ or $p = \infty$ then $p^* = \infty$ or $p^* = 1$ respectively and this leads to a linear programming problem. For $p = 2$ the norm is self-dual, and hence finding an $H(\theta)$ to verify the inclusion $\mathcal{P}[V, g] \subseteq \mathcal{R}_\Theta[F, b]$ requires solving a second-order cone program.

4.2.3 General uncertainty

In the general case, that is for any set Θ, we have the following result.

Proposition 4.8. *If the parameterisation (4.3) holds then* $\mathcal{P}[V, g] \subseteq \mathcal{R}_\Theta[f^T, b]$ *if and only if, for all* i*, there exist* $\bar{h}_i$*,* H_i *satisfying*

$$\bar{h}_i^T g + \lambda_i \leq \bar{b}_i, \quad H_i^T g - B_i^T \in \lambda_i \Theta^*$$

where Θ^* *is the polar dual of* Θ *and*

$$\bar{h}_i \geq 0,\ \bar{h}_i^T V = \bar{f}_i^T$$
$$H_i \geq 0,\ H_i^T V = F_i^T$$

for all $i = 1, 2, \ldots, q$.

Of course, this result is only useful for sets Θ that have an easily-representable polar set Θ^*. However, this is a fairly varied class, allowing us to handle all polyhedra, ellipsoids, and sets defined as the intersection of symmetric cones such as the second order or semidefinite cones.

4.2.4 Characterising robustly invariant polyhedra

As a simple application of the inclusion conditions presented, we show that they can be used to test for robust invariance of polyhedra for parameter uncertain systems.

Suppose the state $x \in \mathbb{R}$ of a dynamical system has dynamics

$$x(t+1) = A(\theta_t)x(t) + w(\theta_t) \tag{4.4}$$

where $\theta_t \in \Theta$ for each $t \geq 0$, and we have affine functions $A : \Theta \to \mathbb{R}^{n\times n}$ and $w : \Theta \to \mathbb{R}^n$. Recall that a robustly invariant set is one that contains its own image under the mapping given by the uncertain dynamics. Now consider a polyhedron $\mathcal{P}[V, g]$ and note that, if $\Phi : \mathbb{R}^n \to P(\mathbb{R}^n)$ is a set-valued function

$$\Phi(x) = \{A(\theta)x + w(\theta) |\ \theta \in \Theta\}$$

representing the uncertain dynamics (4.4), we can write the condition for robust invariance as:

$$\Phi(\mathcal{P}[V, g]) \subseteq \mathcal{P}[V, g]$$

By taking the preimage of both sides, this is equivalent to

$$\mathcal{P}[V, g] \subseteq \Phi^{-1}(\mathcal{P}[V, g]) = \mathcal{R}_\Theta[VA, g - Vw]$$

and hence the polyhedron is robustly positively invariant if and only if the subset relation $\mathcal{P}[V, g] \subseteq \mathcal{R}_\Theta[VA, g - w]$ holds. Application of Lemma 4.3 gives a condition that, in the spirit of [Bit88], characterises the class of robustly invariant polyhedral sets for the affine difference inclusion (4.4).

Proposition 4.9. *The polyhedron $\mathcal{P}[V, g]$ is robustly positively invariant for (4.4) if and only if there is an affine function $H : \Theta \to \mathbb{R}^{\dim(g)\times\dim(g)}$ satisfying*

$$H(\theta) \geq 0,\ \ H(\theta)V = VA(\theta),\ \ H(\theta)g + Vw(\theta) \leq g$$

for all $\theta \in \Theta$.

Propositions 4.6 and 4.7 can be used to give more specific conditions for robustly invariant polytopes under either polytopic or norm-bounded uncertainty. In particular, Proposition 4.6 (for polytopic uncertainty) yields a condition equivalent to those found in [BM07].

Proposition 4.10. *If* $\Theta = Co\{\theta^{(j)}|\ j = 1, 2, \ldots, q\}$*, then* $\mathcal{P}[V, g]$ *is robustly positively invariant for (4.4) if and only if there exist* $H^{(j)}$ *satisfying*

$$H^{(j)} \geq 0, \;\; H^{(j)}V = VA(\theta^{(j)}), \;\; H^{(j)}g + w(\theta^{(j)}) \leq g$$

for all $i = 1, 2, \ldots, q$*.*

Using Lemma 4.3 and Theorem 4.4, we can extend this result to give convex conditions giving a complete characterisation of invariant polyhedra for any uncertainty set Θ. For simplicity in stating the result, we introduce an affine parameterisation of each row of the product $VA(\theta)$ as

$$[VA(\theta)]_i = \bar{a}_i + A_i\theta$$

and assume also that we have, for the additive disturbance $w(\theta)$

$$w(\theta) = \bar{w} + W\theta$$

and for each row i of the affine $H(\theta)$:

$$[H(\theta)]_i = \bar{h}_i + H_i\theta$$

Proposition 4.11. *The polyhedron* $\mathcal{P}[V, g]$ *is robustly positively invariant for (4.4) if and only if, for all rows* i*, there exist* $\bar{h}_i$*,* H_i *satisfying*

$$\bar{h}_i^T g + \bar{w}^T v_i + \lambda_i \leq g_i, \quad H_i^T g + W^T v_i \in \lambda_i \Theta^*$$

where v_i^T *is the* i*th row of* V *and:*

$$\bar{h}_i \geq 0, \;\; \bar{h}_i^T V = \bar{a}_i$$
$$H_i \geq 0, \;\; H_i^T V = A_i^T$$

Recalling that an output admissible set is one that is both invariant and a subset of some 'feasible' set, it is straightforward to extend the previous results to give conditions characterising output admissible sets for systems with parameter uncertainty by enforcing another subset relation using similar techniques.

4.3 Parametrised polyhedra and tubes

In tube MPC, the controller solves an optimisation to build a ***prediction tube***, which is a sequence of sets (the ***tube cross-sections***) containing the future system states. This set sequence is analogous to the predictions in deterministic MPC, and gives a systematic way of applying constraints to the uncertain future state. The power of the technique is due to the fact that these sets need not be fixed ahead of time, but rather are considered as 'variables' in the controller's optimisation problem.

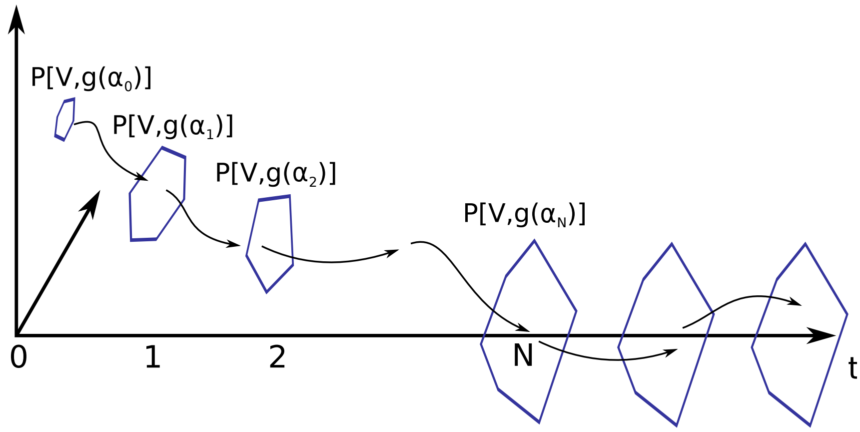

Figure 4.1: The tube concept

We will consider polyhedra $\mathcal{P}[V, g(\alpha)] \subset \mathbb{R}^n$ parametrised by a variable $\alpha \in \mathbb{R}^{\dim(\alpha)}$ via an affine ***parameterisation function*** $g : \mathbb{R}^{\dim(\alpha)} \to \mathbb{R}^{\dim(g)}$. A typical example is the ***homothetic parameterisation***

$$g(\alpha) = g((z, \beta)) = Vz + \beta\bar{g}$$

for $z \in \mathbb{R}^n$ and $\beta \in \mathbb{R}^+$ that translates and scales a polyhedron $\{x \in \mathbb{R}^n |\ Vx \leq \bar{g}\}$. This parameterisation is considered in [RKFC12] for instance, though without explicitly stating g. Another example is

$$g(\alpha) = \alpha$$

which is used in [ECK12a] and [FKC15]. This parameterisation lacks a name, but since the elements of α correspond to the inner products of the facet normals with points on the corresponding facets, we shall refer to it as the ***inner product parameterisation***.

Considering a sequence of parameters α_k for each prediction time $k \geq t$ in an MPC problem yields a sequence of polyhedral sets. Suppose the predicted state of the system satisfies some difference inclusion

$$x_{t+1} \in \Phi(x_t)$$

for a set-valued function $\Phi : \mathbb{R}^n \to P(\mathbb{R}^n)$, then, introducing predicted states x^k, we have $x^k \in \mathcal{P}[V, g(\alpha^k)]$ for all $k \geq t$ if $x_t \in \mathcal{P}[V, g(\alpha^t)]$ and

$$\Phi(\mathcal{P}[V, g(\alpha^k)]) \subseteq \mathcal{P}[V, g(\alpha^{k+1})]$$

for all $k \geq t$. This sequence of tube cross-sections is infinite, but we can neatly truncate it in practice by supposing that there is some terminal set at time $t+N$ that is robustly invariant, so that

$$\Phi(\mathcal{P}[V, g(\alpha^{t+N})]) \subseteq \mathcal{P}[V, g(\alpha^{t+N})]$$

and this final tube cross-section will contain the predictions x^k for all $k \geq t+N$. This is illustrated in Figure 4.1.

In the case of dynamics like (4.4), it is straightforward to consider the preimage under the uncertain mapping and apply Lemma 4.3 to find a condition for the inclusion to hold. In this case, we find that there must exist a $H(\theta)$ such that

$$H(\theta) \geq 0, \;\; H(\theta)V = VA(\theta), \;\; H(\theta)g(\alpha^k) + Vw(\theta) \leq g(\alpha^{k+1})$$

whereupon we hit a stumbling block of sorts: If the tube MPC optimisation is to be convex, we cannot optimize simultaneously over both α and $H(\theta)$. Hence for α to be a variable, we must fix $H(\theta)$ ahead of time such that it satisfies $H(\theta) \geq 0, \;\; H(\theta)V = VA(\theta)$ and then apply the constraint $H(\theta)g(\alpha^k)+w(\theta) \leq g(\alpha^{k+1})$ online. For feasibility with respect to constraints, we usually also require an inclusion of the form

$$\mathcal{P}[V, g(\alpha^k)] \subseteq \mathcal{R}_\Theta[F, b]$$

for all k, for which the same caveats apply.

This is potentially problematic, as by choosing $H(\theta)$ offline we may be restricting the class of possible tube cross-sections $\mathcal{P}[V, g(\alpha^k)]$ to a proper subset of those satisfying the inclusion relation, and hence making the tube MPC controller conservative. But remarkably, we shall see that this usually does not occur if we are careful in our choice of $H(\theta)$. We first consider an illustrative example of how this can be the case, and then give a more formal analysis of the issue.

4.3.1 An example of conservatism

Consider a polyhedron $\mathcal{P}[V, \alpha] \subseteq \mathbb{R}^2$ where $\alpha \in \mathbb{R}^3$ is a free parameter and the facet normals are given by the matrix

$$V = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \end{bmatrix}$$

and consider using Lemma 4.2 to apply the inclusion $\mathcal{P}[V, \alpha] \subseteq \mathcal{P}[F, 1]$ where $F = [1 \;\; 1]$. There are many possible choices of $H \geq 0$ satisfying $HV = F$, but we concentrate on two of them in particular:

$$H_1 = [1 \;\; 1 \;\; 0], \quad H_2 = [0 \;\; 0 \;\; 1]$$

From Lemma 4.2, any α satisfying either $H_1\alpha \leq 1$ or $H_2\alpha \leq 1$ will also satisfy the inclusion $\mathcal{P}[V, \alpha] \subseteq \mathcal{P}[F, 1]$. But which of these is the better choice? Consider the parameter

$$\alpha^{(1)} = [5 \;\; 5 \;\; 1]^T$$

for which it is clear that the constraint $H_1\alpha \leq 1$ is not satisfied. Despite the constraint not holding, it is easy to verify that $\mathcal{P}[V, \alpha^{(1)}] \subseteq \mathcal{P}[F, 1]$. Worse, there is no choice of α with $H_1\alpha \leq 1$ that gives this polyhedron, so that we can say definitively that H_1 leads to conservatism in applying the inclusion.

Now consider H_2. The constraint $H_2\alpha \leq 1$ can be simplified to $\alpha_3 \leq 1$. Again it is possible to find a parameter, say

$$\alpha^{(2)} = [1\ \ 1\ \ 5]^T$$

for which the polyhedron satisfies $\mathcal{P}[V, \alpha^{(2)}] \subseteq \mathcal{P}[F, 1]$ but the constraint $H_2\alpha^{(2)} \leq 1$ does not hold. However, there is a very important difference from the case with H_1; if we were to replace the parameter $\alpha^{(2)}$ by

$$\beta^{(2)} = [1\ \ 1\ \ 1]^T$$

then the constraint $H_2\alpha \leq 1$ holds and $\mathcal{P}[V, \alpha^{(2)}] = \mathcal{P}[V, \beta^{(2)}]$, so that the polyhedron is unchanged by this substitution. In fact, it is the case that for any polytope $\mathcal{P}[V, \alpha]$ satisfying the inclusion, there will exist a parameter β such that $\mathcal{P}[V, \alpha] = \mathcal{P}[V, \beta]$ satisfying $H_2\beta \leq 1$ that can be found simply by setting the third element of the parameter to 1. Hence, even though $H_2\alpha \leq 1$ is a sufficient (and not necessary) condition on α for the inclusion to hold, it does not actually restrict the class of polytopes to a subset of those that satisfy $\mathcal{P}[V, \alpha] \subseteq \mathcal{P}[F, 1]$. We will say that H_2 is ***tight*** with respect to this inclusion, reflecting the fact that it gives the best possible bounding on the parameter α for the inclusion to hold.

4.3.2 Conservatism of inclusion conditions

The previous example reveals that it is not necessary for any parametrised polytope $P[V, g(\alpha)]$ to have a unique representation in terms of some parameter α. This is somewhat tiresome from an explanatory perspective as we must make statements (as in the previous example) like "for any α satisfying the inclusion, there exists a β such that $Hg(\beta) \leq 1$" and prohibits us from stating that $Hg(\alpha) \leq 1$ 'if and only if' the inclusion holds. As a remedy to this semantic awkwardness, we consider equivalence classes A of parameters such that $\alpha, \beta \in A$ implies $\mathcal{P}[V, g(\alpha)] = \mathcal{P}[V, g(\beta)]$ and write $\alpha \sim \beta$ for the corresponding equivalence relation. We will write $\mathcal{P}[V, g(A)]$ for the unique polytope given by the equivalence class A, which will simplify discussion.

Suppose now we know some H satisfying $H \geq 0$, $HV = F$. For any equivalence class of parameters A, it is clear that $\mathcal{P}[V, g(A)] \subseteq \mathcal{P}[F, b]$ if there is $\alpha \in A$ with

$H\alpha \leq b$. We shall call H 'tight' precisely when the converse also holds, so that this becomes an 'if and only if' relation.

Definition 4.1. *Suppose that $\mathcal{P}[V, g(A)] \subseteq \mathcal{P}[F, b]$ if and only if there exists $\alpha \in A$ with $Hg(\alpha) \leq b$. Then we say H is* **tight** *for this inclusion with parameterisation function $g(\alpha)$.*

Remark 4.12. The above definition is worth some careful consideration, as it is quite subtle. A particularly interesting point is that if we consider the set of parameters α that satisfy the inclusion $\mathcal{P}[V, g(\alpha)] \subseteq \mathcal{P}[F, b]$, this set is nonconvex in general (it is a union of polyhedra). Despite this nonconvexity, if a tight H exists then we can replace this nonconvex condition with the convex constraint $Hg(\alpha) \leq b$ without any conservatism.

This property occurs trivially when there is only one choice of H satisfying $H \geq 0$ and $HV = F$, as a result of the condition in Lemma 4.2 that there must exist an H with the given properties if the inclusion condition holds.

Proposition 4.13. *If there is a unique H^* satisfying $H^* \geq 0$ and $H^*V = F$ then it is tight for $\mathcal{P}[V, g(A)] \subseteq \mathcal{P}[F, b]$ for any parameterisation function $g(\alpha)$.*

The next theorem gives a necessary and sufficient condition for a given H to be tight for an inclusion, and in the process of doing so classifies all V and F that admit a tight choice of H. For simplicity it is stated first for the case that $F = f^T$ is a row vector so that the containing polyhedron is a half-space, and then generalised.

Theorem 4.14. *A row vector $\tilde{h}^T$ satisfying $\tilde{h}^T \geq 0$ and $\tilde{h}^T V = f^T$ is tight for the inclusion $\mathcal{P}[V, g(A)] \subseteq \mathcal{P}[f^T, b]$ if and only if for all A, $\mathcal{P}[V, g(A)]$ is a subset of the cone $\{x \in \mathbb{R}^n | \ V_i x \leq V_i x^*, \ i \in \mathcal{I}\}$, for any $x^* \in \arg\max_x \{f^T x | \ x \in \mathcal{P}[V, g(A)]\}$, where $\mathcal{I}$ contains the indices of the nonzero elements of $\tilde{h}$.*

Theorem 4.15. *A matrix H satisfying $H \geq 0$ and $HV = F$ is tight for the inclusion $\mathcal{P}[V, g(A)] \subseteq \mathcal{P}[F, b]$ if and only if every row H_i is tight considering the inclusion $\mathcal{P}[V, g(A)] \subseteq \mathcal{P}[F_i, b_i]$.*

A direct consequence of Theorem 4.14 is that there is a unique choice of tight H with fewest nonzero elements. This is because $\tilde{h}^T V$ can always be written as $\sum_{i \in I} \tilde{h}_i V_i$

for some $\mathcal{I}$ with $|\mathcal{I}| \leq n$, and the condition of Theorem 4.14 will then still hold. These theorems have some other interesting consequences that are worth stating as independent results, as they will motivate choices of V when designing tube parameterisations for MPC, and give an indication of when we can expect to have conservatism in a tube MPC scheme that uses the inclusion conditions in this chapter.

Proposition 4.16. *If each row of F is a positive multiple of a row in V so that for all i there exists a j with $F_i = bV_j$ for some $b > 0$, then there exists a tight choice of H for the inclusion $\mathcal{P}[V, g(A)] \subseteq \mathcal{P}[F, b]$.*

This proposition explains the 'tightness' of H_2 in the motivating example at the start of this section. It is this structure that we shall try to emulate when designing the V matrix for our tube MPC controllers, a topic that we shall return to shortly. The next corollary is an obvious consequence.

Corollary 4.17. *If the rows of F are a subset of the rows of V, there exists a tight choice of H for $\mathcal{P}[V, g(A)] \subseteq \mathcal{P}[F, b]$.*

As the definition of a tight H depends on the tube parameterisation function $g(\alpha)$, it should come as no surprise that some parameterisations always admit tight choices of H. In particular, the homothetic parameterisation has this property, and we shall see soon that this allowed the authors of [RKFC12] to apply this tube parameterisation without considering the kind of conservatism that we are investigating here.

Theorem 4.18. *For the homothetic parameterisation function*

$$g(\alpha) = g(z, \beta) = Vz + \beta\bar{g}$$

there exists a tight H for any inclusion $\mathcal{P}[V, g(A)] \subseteq \mathcal{P}[F, b]$.

Interestingly, it is possible to show that the convex cone of Theorem 4.14 can always be constructed in a Euclidean space of dimension one or two.

Proposition 4.19. *If $\mathcal{P}[V, g(A)] \subset \mathbb{R}^1$ or $\mathcal{P}[V, g(A)] \subset \mathbb{R}^2$ then there exists a tight H for $\mathcal{P}[V, g(A)] \subseteq \mathcal{P}[F, b]$ for any $\mathcal{P}[F, b]$.*

There is an obvious extension to higher dimensions whenever it is possible to consider a two dimensional subspace containing the facet normals in F.

Corollary 4.20. *If the rows of F are coplanar, and for all i there exist j, k such that $F_i = aV_j + bV_k$ with $a, b > 0$, there exists a tight H for $\mathcal{P}[V, g(A)] \subseteq \mathcal{P}[F, b]$.*

Finally, we note that in $\mathbb{R}^3$ there do exist choices of V and F that lead to compact polyhedra but for which it is impossible to construct the cone of Theorem 4.14. The simplest example this author can think of is when $Vx \leq 1$ defines a square-based pyramid with the base defined by the plane $x_3 = -1$, $F = f^T = [0, 0, 1]$, and the inner product parameterisation is used. Since we almost certainly want to be able to consider control problems where the state dimension is 3 or greater, this implies that we should be careful in our choice of V when formulating the constraints for tube MPC, if possible ensuring that it contains positive multiples of the rows of F. This idea will be revisited near the end of the chapter.

4.3.3 Conservatism in robust inclusions

So far we have only considered polyhedral inclusions of the form

$$\mathcal{P}[V, g(A)] \subseteq \mathcal{P}[F, b]$$

and not the robust equivalent:

$$\mathcal{P}[V, g(A)] \subseteq \mathcal{R}_\Theta[F, b]$$

It is fairly straightforward to extend Definition 4.1 to the robust case when considering this inclusion.

Definition 4.2. *Suppose that $\mathcal{P}[V, g(A)] \subseteq \mathcal{R}_\Theta[F, b]$ if and only if there exists $\alpha \in A$ with $H(\theta)g(\alpha) \leq b(\theta)$ for all $\theta \in \Theta$. Then we say H is* **tight** *for this robust inclusion with parameterisation function $g(\alpha)$.*

From this definition, there is an obvious necessary and sufficient condition for $H(\theta)$ to be tight for a robust inclusion.

Theorem 4.21. *The matrix function $H(\theta)$ is tight for the inclusion $\mathcal{P}[V, g(A)] \subseteq \mathcal{R}_\Theta[F, b]$ if and only if it is tight for the inclusion $\mathcal{P}[V, g(A)] \subseteq \mathcal{P}[F, b]$ for each fixed $\theta \in \Theta$.*

We must be careful with the above result because there is no stipulation that this tight $H(\theta)$ must be affine – in fact, it will be piecewise affine in general. We desire $H(\theta)$ to be affine in θ so that we can use it in the constraints of an MPC optimization, and next result gives a practical sufficient condition for the existence of an affine $H(\theta)$.

Theorem 4.22. *If $F(\theta)$ and $b(\theta)$ are given by the affine parameterisation (4.1), and the rows of V include positive multiples of the rows of $\bar{F}$ and $F^{[i]}$ for all i, then there is a tight choice of $H(\theta)$ that is an affine function of θ in the form (4.2).*

This completes our existence results for tight choices of H. We now digress to consider a link between our inclusion conditions and the 'constraint-tightening' parameters used in other formulations of tube MPC, before considering the design of tube parameterisations.

4.3.4 Tube MPC using tightened constraints

In many other formulations of tube MPC, like those found in the book [MR09] or in papers like [MSR05] and [RKFC12], it is common to consider the deviations of the system trajectories around some nominal state z. Constraints on z are applied that are more restrictive than those on x to account for the uncertainty. We now show an equivalence between this approach and the inclusion conditions we have developed.

Consider the polyhedral inclusion

$$\mathcal{P}[V, g(A)] \subseteq \mathcal{P}[F, b]$$

where the parameterisation function g is the homothetic $g(z, \beta) = Vz + \beta\bar{g}$ with the assumption that $\bar{g} > 0$ and $\beta > 0$, so that z lies in the interior of the set. Applying our inclusion conditions, we must choose $H \geq 0$, $HV = F$ and

$$Hg(\alpha) = HVz + \beta H\bar{g} \leq b$$

but after substitution using $HV = F$ we find

$$Fz \leq b - \beta H\bar{g}$$

and as $H \geq 0$, $\bar{g} > 0$ and $\beta > 0$ the term $\beta H\bar{g}$ is nonnegative. Hence it is a parameter that makes the constraints on z more stringent to account for the spread of the set $\mathcal{P}[V, g(A)]$ around this point. As we know that a tight H exists for a homothetic set, Definition 4.1 implies that $\beta H\bar{g}$ gives a tight bound on z for the inclusion to hold, justifying the terminology somewhat. In the case that only additive uncertainty is present in a system, the state can be decomposed into decoupled nominal and uncertain parts so that $x = z + e$ [MSR05]. The above form of the constraints is very convenient in this case as it results from considering a Minkowski summation $\{z\} \oplus S$ where $e \in S$, and it is trivial to see how this set will evolve in time if we know deterministic dynamics for z.

However, we did not take this viewpoint from the beginning as in the parameter-uncertain case no convenient decoupling of the state into $x = z + e$ exists (in general, e will have a dependence on z - see [CCKE14] for an example). In addition, it is not immediately obvious how to generalise the tightening parameter $\beta H\bar{g}$ when using tube parameterisations more complex than the homothetic one.

4.3.5 Choosing $H(\theta)$

We now consider how to make an offline choice of H in practice. Often a particular subset relation might not admit a tight choice of H, so we will give a design procedure that recovers a tight H when it is available and gives an intuitively reasonable choice otherwise. Before doing so, it is worth asking if this is really the choice of H that we want: what if there were a sparse choice of H that would lead to faster MPC optimisations? The next result reveals a pleasing coincidence, that the tight H matrices are sparse.

Proposition 4.23. *If H is tight, then each row H_j satisfies $\|H_j\|_0 \leq n$ where $n = \dim(x)$.*

Perhaps the best method of choosing H is to use the conditions in the previous sections to choose a tight H directly when this is possible (such as when the rows of V contain multiples of the rows of F). We will typically advocate this for MPC, since it is usually possible to choose V so that this occurs for most inclusions in the problem. We will return to this point in the next section.

If this is not possible, then whether or not a tight H exists, a sensible way to approach the design of each row $H_j = h^T$ is to try to minimise $h^T g(\alpha)$ in some way. As the set is unknown ahead of time, this can be thought of as a strategic game against an adversary who chooses an equivalence class A of α, and we can consider the minimax solution

$$h^* = \arg \min_{h\in\mathcal{H},\alpha\in A} \max_{A\in\mathcal{A}} h^T g(\alpha)$$

where $\mathcal{H} = \{h \geq 0,\ \ h^T V = f^T\}$ and $\mathcal{A}$ is the set of all equivalence classes A such that the desired inclusion holds. We could perhaps restrict the domain of g to a set S so that each equivalence class A becomes a singleton $\{\arg\min_{\alpha\in A} h^T g(\alpha)\}$ on S (this can be easier than it sounds; for instance the example in section 4.3.1 requires $S = \{\alpha \in \mathbb{R}^{\dim(g)} |\ \ \alpha_3 \leq 1\}$ for any h). In principle we could then solve the minimax problem:

$$h^* = \arg \min_{h\in\mathcal{H}} \max_{\alpha\in S} h^T g(\alpha)$$

It is straightforward to show by contradiction that this recovers a tight choice of h if it exists for the inclusion.

Theorem 4.24. *The minimax optimization yields a tight H if one exists.*

For some parameterisation functions g we can solve this exactly; for instance in the homothetic case the A are singletons and we have

$$h^* = \arg \min_{h\in\mathcal{H}} \max_{(z,\beta)\in S} (h^T V z + \beta h^T \bar{g}) = \arg \min_{h\in\mathcal{H}} \max_{(z,\beta)\in S(h)} (f^T z + \beta h^T \bar{g})$$

and the term $f^T z$ does not depend on h, so we can simplify to

$$h^* = \arg \min_{h\in\mathcal{H}} \max_{(z,\beta)\in S(h)} \beta h^T \bar{g} = \arg \min_{h\in\mathcal{H}} h^T \bar{g}$$

from which we can find h^* by linear programming.

In other instances this optimisation can be prohibitively difficult as the set $S(h)$ may have a complex dependence on h. In general, this minimax approach leads to a combinatorial search as the representation of S requires enumeration of the possible solutions of $h \geq 0,\ \ h^T V = f^T$, which is a finite set if we restrict ourselves to solutions for which $\|h\|_0 \leq n$ (in light of Proposition 4.23). While this is guaranteed to find a tight h if it exists, it seems quite impractical.

A simpler possibility is to avoid the maximisation step altogether, choosing some 'nominal' value $\alpha^* \in \mathcal{A}$, and then solving

$$h^* = \arg\min_{h\in\mathcal{H}}\ h^T g(\alpha^*)$$

which is itself a linear programming problem.

Algorithm 4.1 Suboptimal design of H

---

for $i = 1 : size(V, 1)$ **do**
 $H(i, :) = \arg\min_{h\in\mathcal{H}}\ h^T g(\alpha^*)$
end for

This option seems to work very well in practice, giving a tight choice of H in many instances when one exists (as can be verified using the conditions in the preceding sections). It is often the case that there is a natural choice of α^* in a given MPC problem, for instance one corresponding to a large invariant set. The resulting H has some desirable properties, including sparsity.

Theorem 4.25. *Algorithm 4.1 produces rows H_j with at most $n = \dim(x)$ nonzero elements in each row.*

So far we have only considered H and not the $H(\theta)$ required to enforce a robust inclusion like $\mathcal{P}[V, g] \subseteq \mathcal{R}_\Theta[F, b]$. Similar concepts apply when designing $\bar{H}$ and $H^{[i]}$ in the affine parameterisation (4.2); we can consider, in principle, the minimax problem

$$h^*(\theta) = \arg\min_{h(\theta)\in\mathcal{H},\alpha\in A}\ \max_{A\in\mathcal{A},\theta\in\Theta}\ h^T(\theta) g(\alpha)$$

where the adversary now chooses θ as well as α. This time, in the case of a homothetic tube it simplifies to

$$h^*(\theta) = \arg \min_{h(\theta)\in\mathcal{H}} \max_{\theta\in\Theta} h^T(\theta)\bar{g} = \arg \min_{\bar{h},h^{[i]}} \max_{\theta\in\Theta} \left(\bar{h} + \sum_{i=1}^{q} h^{[i]}\theta_i\right)^T \bar{g}$$

where we require $\bar{h}_j \geq 0$, $\bar{h}V = \bar{f}$ and $h^{[i]} \geq 0$, $(h^{[i]} - \bar{h})V = f^{[i]}$. This is equivalent to a robust linear program, which can be solved depending on the form of Θ; it is a linear program for polytopic Θ and a second-order cone program for ellipsoidal Θ for example.

Again this full minimax design of $H(\theta)$ is impractical when the set parameterisation is not homothetic, so we give a suboptimal method of design for a fixed α^* analogous to Algorithm 4.1.

Algorithm 4.2 Suboptimal design of H(θ)

for $i = 1 : size(V, 1)$ **do**
  $(\bar{H}(i,:), H^{[i]}(i,:)) = \arg \min_{\bar{h},h^{[i]}} \max_{\theta\in\Theta} \left(\bar{h} + \sum_{i=1}^{q} h^{[i]}\theta_i\right)^T g(\alpha^*)$
  subject to $\bar{h}_j \geq 0$, $\bar{h}V = \bar{f}$, $h^{[i]} \geq 0$, $(h^{[i]} - \bar{h})V = f^{[i]}$
end for

This tends to lead to good results in practice, and sparse choices of the resulting matrix $H(\theta)$ for each fixed θ, while still involving only a fixed number of robust linear programming problems in its implementation.

4.4 State tube design

We now deal with the question of how to design the matrix V and tube parameterisation functions $g(\alpha)$ for use in tube MPC. Typically the choice of g involves making a trade off between complexity (the number of variables and constraints) and conservatism of the resulting tube MPC controller, and this will depend on the intended application. On the other hand, designing V is a more straightforward matter in that there is usually a natural way to construct V for a particular problem that allows exploitation of the various tightness properties already discussed.

4.4.1 Choosing V

Recalling the discussion at the start of section 4.3, in a tube MPC problem we typically want to apply feasibility conditions such as:

$$\mathcal{P}[V, g(\alpha_k)] \subseteq \mathcal{P}[F, b] \quad \text{or} \quad \mathcal{P}[V, g(\alpha_k)] \subseteq \mathcal{R}_\Theta[F, b]$$

Considering these feasibility relations, Proposition 4.16 and Theorem 4.22 suggest that we should include in V the rows of F or, in the robust case, $\bar{F}$ and $F^{[i]}$ (which is a basis for a subspace containing $F(\theta)$). This will ensure there is a simple choice of H or $H(\theta)$ that gives a tight bound on this inclusion.

In the tube MPC problem we will also have preimage conditions such as

$$\mathcal{P}[V, g(\alpha_k)] \subseteq \mathcal{R}_\Theta[VA, g(\alpha_{k+1}) - VBu_k]$$

which are based the uncertain system dynamics $x_{k+1} = A(\theta)x_k + B(\theta)u_k$, for example. For simplicity, assume that the set Θ is polytopic with vertices $\theta^{(j)}$ and denote $A^{(j)} = A(\theta^{(j)})$. Proposition 4.16 or Theorem 4.22 then suggests that V should include the rows of $V\bar{A}$ and $VA^{(i)}$, but this is not generally possible as it implies an infinite number of rows in V: F, $FA^{(i)}$, $FA^{(j)}A^{(i)}$ etc. for all i, j, and so on. In practice, the constraints implied by these rows of V will tend to be redundant later in the sequence as long as the system is stable, so we can truncate this sequence at some point and take a subset of these multiples without introducing much conservatism.

It is no coincidence that the elements of this sequence correspond precisely to the preimages of the facet normals in F under the uncertain mapping given by $A(\theta)$. Hence they correspond to the facet normals of the maximal admissible set of the system under the constraint given by the feasibility condition. In the next chapter, we will see that it is a requirement for recursive feasibility of a tube MPC controller that the tube cross-sections can represent constraint admissible sets of the system, and therefore, it seems prudent to choose V to correspond to the constraint matrix for a large, polyhedral invariant set of the system. Algorithms for finding the maximal admissible set, in the case that the parameter uncertainty is polytopic, can be found in [PRSDM05b] and

[BM07]. Unfortunately the maximal such set can often be quite complex for systems with parametric uncertainty. This issue was partially addressed in [PRSDM05a] where a procedure was given for reducing the number of faces of an invariant polyhedron while giving only a minor reduction in volume.

In the same spirit, we now present a simple algorithm that produces reduced-complexity admissible polyhedra for parameter uncertain systems – Algorithm 4.3. It is based on the algorithm of [PRSDM05b], but modified with a heuristic that tightens constraints when a particular constraint has many 'child' constraints. In contrast to the algorithm of [PRSDM05a], it can also handle additive uncertainty if the uncertainty is included in the polytopic uncertainty description, so that we have:

$$x_{t+1} = \tilde{A}x_t + w_t, \;\; (\tilde{A}, w_t) \in \mathrm{Co}(\tilde{A}^{(j)}, w^{(j)}), \;\; j = 1, \ldots, r \tag{4.5}$$

For simplicity we assume that the constraint is given as a state constraint in the form $\tilde{F}x \leq 1$. Two constants, a 'threshold' parameter γ and a 'tightening' parameter δ control how aggressively the algorithm removes constraints.

Algorithm 4.3 Reduced complexity admissible sets

$i = 0$; $k = \mathrm{length}(\tilde{F}) + 1$; $V = \tilde{F}$; $g = \mathrm{ones}(k, 1)$;
while $i < k$ **do**
  $V_{\mathrm{child}} = []$; $g_{\mathrm{child}} = []$; $n_{\mathrm{child}} = 0$;
  for $j = 1 : r$ **do** ▷ Check for child facets to add
    if $\{x \mid Vx \leq g\} \not\subseteq \{x \mid V(i,:)\tilde{A}^{(j)}x \leq g(i) - V(i,:)w^{(j)}\}$ **then**
      $V_{\mathrm{child}} = [V_{\mathrm{child}}; V(i,:)\tilde{A}^{(j)}]$
      $g_{\mathrm{child}} = [g_{\mathrm{child}}; g(i) - V(i,:)w^{(j)}]$;
      $n_{\mathrm{child}} = n_{\mathrm{child}} + 1$
    end if
  end for
  if $n_{\mathrm{child}} \geq 2$ **then** ▷ Tighten constraints if many children
    $V_{\mathrm{mean}} = \mathrm{mean}(V_{\mathrm{child}}, 1)$;
    if $\max_j \|V_{\mathrm{child}}(j,:) - V_{\mathrm{mean}}\| \leq \gamma$ **then**
      $V_{\mathrm{child}} = (1 + \delta)V_{\mathrm{child}}$;
    end if
  end if
  $V = [V; V_{\mathrm{child}}]$, $g = [g; g_{\mathrm{child}}]$
  Check $\{x \mid Vx \leq g\}$ for redundancies and remove them
end while

Remark 4.26. Implementation of Algorithm 4.3 is subject to many of the same comments as the algorithm of [PRSDM05b]: The subset relation can be checked efficiently by solving a linear programming problem, and a procedure is required for removing redundant rows from the matrix V. Checking the subset relation is typically the slowest step, and an efficient implementation will typically require using an LP solver to solve a feasibility (rather than optimization) problem. It is also important to implement some kind of 'garbage collection' procedure to periodically remove redundancies. We refer the reader to [PRSDM05b] for a detailed analysis of such procedures, and note that like the algorithm of [PRSDM05a], Algorithm 4.3 has guaranteed convergence for small enough values of the parameters γ and δ.

We note the following correctness result for Algorithm 4.3.

Theorem 4.27. *If Algorithm 4.3 terminates, the resulting set* $\{x \in \mathbb{R}^n | Vx \leq g\}$ *is feasible and positively invariant under the dynamics of (4.5).*

An important consideration when choosing the parameters γ and δ in Algorithm 4.3 is that there exists a trade-off between the volume of the obtained admissible set and its complexity. As the number of rows of V directly affects the number of constraints in the tube MPC optimisation, it is generally preferable to reduce the number of rows of V in order to obtain a smaller optimisation. But the final tube cross-section usually forms the terminal set for the predictive controller, so that a larger (and potentially more complex) terminal set will increase the volume of the controller's feasible set. In practice, this tends to mean that we reach a point of diminishing returns as we increase γ and δ.

4.4.2 Choosing $g(\alpha)$

In contrast to the design of V, the choice of parameterisation function $g(\alpha)$ tends to be more subjective and we must trade-off competing objectives. Broadly speaking, this is because including more degrees of freedom in any given tube parameterisation will relax the constraints applied in the online optimisation and so reduce the conservatism of the controller, but doing so will introduce more free variables into the online optimisation and give longer computation times.

We have already introduced, in section 4.3, the homothetic tube parameterisation originating in [RKFC12]

$$g_k(\alpha_k) = Vz_k + \beta_k g \tag{4.6}$$

where $\alpha_k = [z_k \beta_k]^T$ with $z_k \in \mathbb{R}^n$, $\beta_k \in \mathbb{R}$. This parameterisation is quite efficient in terms of the size of the resulting online optimisation, since it requires only $n+1$ variables per prediction time. We also gave the much more flexible inner product parameterisation

$$g_k(\alpha_k) = \alpha_k \tag{4.7}$$

where $\alpha_k \in \mathbb{R}^{\dim(g)}$. Note that there is no requirement that the elements of α_k be positive in general here, as the polyhedron $\mathcal{P}[V, g(\alpha)]$ need not contain the origin (of course, this is the natural state of affairs during transients).

We introduce one more parameterisation that can strike a trade-off between (4.6) and (4.7) by virtue of including them as special cases. Let $\alpha_k = [z_k \ \beta_k^{(1)} \ \cdots \ \beta_k^{(s)}]^T$ with $z_k \in \mathbb{R}^n$, $\beta_k^{(i)} \in \mathbb{R}$ for each i and let

$$g_k(\alpha_k) = Vz_k + \sum_{i=1}^{s} \beta_k^{(i)} g_i \tag{4.8}$$

so that the resulting set can be represented by a Minkowski sum as $\{z_k\} \oplus \bigoplus_{i=1}^{s} \beta_k^{(i)} \{x \in \mathbb{R}^n | \ Vx \leq g_i\}$. In the example at the end of this chapter, we show that with a little work at the design stage, it is sometimes possible for parameterisations of this type to outperform both (4.6) and (4.7).

Remark 4.28. It is not generally required to use the same parameterisation for all prediction times k in a tube MPC controller. We shall see in the next chapter that, due to recursive feasibility considerations, the function g_k must be able to reproduce any set given by g_{k+1} for any k. Generally this means that a parameterisation with many degrees of freedom can be used early in the prediction horizon, and a simpler parameterisation with fewer degrees of freedom used near the end. For instance if the prediction horizon was $N = 10$, it would be possible to use a inner product g_k for $k = t$ to $t+2$, a homothetic parameterisation from $t+3$ to $t+9$, and perhaps no parameterisation at all for $k = t+10$ so that the terminal set is effectively fixed. This can sometimes be exploited in the design of controllers to greatly reduce the number of variables in the online optimisation.

A sensible design process for g is to start by comparing the performance and feasibility region of the homothetic and inner product parameterisations. If the feasible region or performance of the resulting controller is improved as a result of the increased number of degrees of freedom in the inner product parameterisation, then the designer can attempt to find a parameterisation that emulates this effect with fewer free variables.

4.5 Design example

To end the chapter we introduce a small design example of a system with a two dimensional state space. We apply feasibility and preimage inclusions as discussed at the start of Section 4.3 and plot typical tubes resulting from the homothetic and inner product parameterisations. The system studied is a separately-excited DC motor taken from [Bla91] which has the continuous time dynamics

$$\dot{x}(t) = \begin{bmatrix} -0.07 & -0.86(1+q_1) \\ 0.06(1+q_1) & -q_2 \end{bmatrix} x(t) + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u(t)$$

where the uncertain parameters $q_{1,2}$ satisfy the inequalities $-0.2 \le q_1 \le 0.2$ and $0.0085 \le q_2 \le 0.5$, resulting in a polytopic description for the uncertain system with 4 vertices. These dynamics were discretised using an Euler scheme with a sampling time of $T = 0.1s$, and the state constraint

$$\begin{bmatrix} -1 \\ -1 \end{bmatrix} \le x_k \le \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

applied along with an input constraint $-10 \le u_k \le 10$.

This system was prestabilised with the state-feedback gain $K = [-0.9534 \;\; -0.5776]$ which was found by minimising a quadratic bound on an LQR-type cost with $Q = \text{diag}(1, 1)$ and $R = 1$ as described in [BEGFB94]. Admissible polyhedra were then found using Algorithm 4.3 using a range of γ and δ parameters. The results of changing these threshold and tightening parameters can be seen in Figures 4.2 and 4.3. It is apparent that the algorithm can greatly reduce the number of facets in the admissible set without losing much volume, as intended. From inspection of these plots, $\gamma = 0.005$ and $\delta = 0.04$ was chosen as a good trade off between complexity and volume of the terminal set, giving:

$$V = \begin{bmatrix} -2.3661 & -1.1175 \\ 2.3661 & 1.1175 \\ 0.0846 & 3.5359 \\ -0.0782 & -3.4280 \\ 0.0974 & 3.5233 \\ -0.0910 & -3.4162 \end{bmatrix}, \quad g = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix},$$

By assuming an input of the form $u_k = Kx_k + c_k$, inclusions were formulated as in section 4.3 to define a tube containing the state, using both the homothetic parameterisation (4.6) and the inner product parameterisation (4.7). The resulting constraints on the tube were formed using Yalmip, with a prediction horizon length of $N = 5$. The feasible set of initial conditions of the resulting constraints can be seen in Figures 4.4 and 4.5 along with an example of a feasible tube starting from the initial condition $x_0 = [-0.59 \;\; -0.27]^T$. It is immediately obvious that using the inner product parameterisation gives a much larger feasible region. After some experimentation, it was found that using the alternative parameterisation of the form (4.8)

$$g_k(\alpha_k) = Vz_k + \beta_k^{(1)} g_1 + \beta_k^{(2)} g_2$$

where

$$g_1 = [1 \;\; 1 \;\; 0 \;\; 0 \;\; 0 \;\; 0]^T, \quad g_2 = [0 \;\; 0 \;\; 1 \;\; 1 \;\; 1 \;\; 1]^T$$

gives comparable results to the inner product parameterisation, with the feasibility regions of the two being indistinguishable at the scale of Figure 4.5. However it achieves this using only 4 variables to parametrise the tube at each prediction time rather than 6, leading to a significant reduction in the number of free variables needed. This would likely be a considerable advantage if using this tube parameterisation to form a predictive controller.

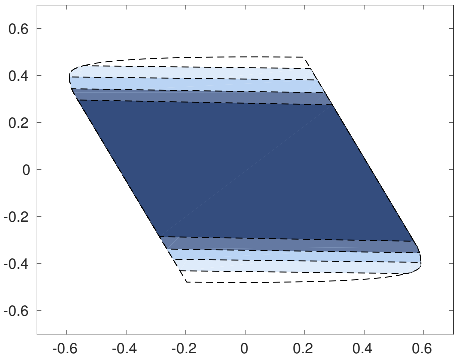

Figure 4.2: Effect of changing γ

γ	δ	# of facets
0.000	0.02	64
0.002	0.02	24
0.004	0.02	14
0.006	0.02	12
0.008	0.02	9

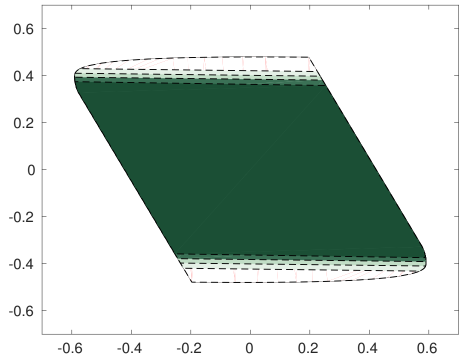

Figure 4.3: Effect of changing δ

γ	δ	# of facets
0.005	0.00	64
0.005	0.01	30
0.005	0.02	12
0.005	0.03	9
0.005	0.04	6

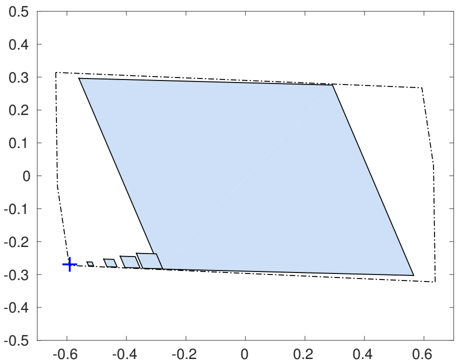

Figure 4.4: Homothetic tube example

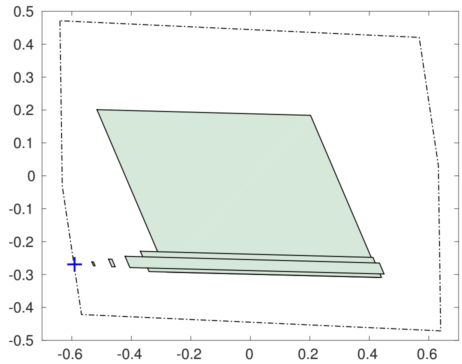

Figure 4.5: Inner product tube example

4.A Proofs

Lemma 4.1

This follows immediately from the definitions of $\mathcal{P}[\cdot,\cdot]$ and $\mathcal{R}_\Theta[\cdot,\cdot]$.

Lemma 4.2

With F_i and b_i the ith row and element of F and b respectively, consider the linear program in inequality form:

$$\begin{aligned} \text{maximise} \quad & F_i x \\ \text{subject to} \quad & Vx \leq g \end{aligned}$$

All points within $\mathcal{P}[V, g]$ will satisfy $F_i x \leq b_i$ if and only if $F_i x^* \leq b_i$, with x^* the solution of this LP. The dual LP in standard form is:

$$\begin{aligned} \text{minimise} \quad & g^T h_i \\ \text{subject to} \quad & V^T h_i = F_i, \;\; h_i \geq 0 \end{aligned}$$

Since $\mathcal{P}[V, g]$ has a nonempty interior (otherwise the theorem is trivial) strong duality holds by Slater's condition and all $x \in \mathcal{P}[V, g]$ satisfy $F_i x \leq b_i$ if and only if $g^T h_i^* \leq b_i$. The result follows by defining H to be the matrix with h_i^* as its ith row for all i and noting that the original inclusion now holds if and only if $g^T h_i^* \leq b_i$ for all i.

Lemma 4.3

From Lemma 4.1 the inclusion holds if and only if

$$\mathcal{P}[V, g] \subseteq \bigcap_{\theta \in \Theta} \mathcal{P}[F, b]$$

and application of Lemma 4.2 immediately gives the result.

Theorem 4.4

The proof is by construction. From Lemma 4.3 the matrix function $H(\theta)$ satisfies:

$$H(\theta) \geq 0, \;\; H(\theta)V = \bar{F} + \sum_{i=1}^{q} F^{[i]}\theta_i$$

and putting $\theta = 0$ immediately gives

$$H(0) \geq 0, \;\; H(0)V = \bar{F}$$

suggesting that we define $\bar{H} = H(0)$. If we now choose $\theta = e_i$ such that the ith element $\theta_i = 1$ and all other elements are zero we find

$$H(e_i) \geq 0, \;\; H(e_i)V = \bar{F} + F^{[i]}$$

and subtracting $H(0)V$ from the equation on the right gives

$$(H(e_i) - \bar{H})V = F^{[i]}$$

suggesting that we can define $H^{[i]} = H(e_i)$ for all i. This completes the proof, as $H(\theta)$ for any other value of θ can now be defined using equation (4.2) and seen to satisfy the conditions of Lemma 4.3 by linearity.

Proposition 4.6

Follows directly from substituting the affine parameterisation of Theorem 4.4 into the result of Lemma 4.3.

Proposition 4.7

Lemma 4.3 and the uncertainty parameterisation lead to the robust constraint:

$$(\bar{h}_i + H_i\theta)^T g \leq \bar{b}_i + B_i\theta, \quad \forall\theta : \;\; \|\theta\|_p \leq 1$$

A simple rearrangement gives

$$\bar{h}_i^T g + \theta^T (H_i^T g - B_i^T) \leq \bar{b}_i, \quad \forall \theta : \; \|\theta\|_p \leq 1$$

which is equivalent to:

$$\bar{h}_i^T g + \sup_{\|\theta\|_p \leq 1} \left\{ \theta^T (H_i^T g - B_i^T) \right\} \leq \bar{b}_i$$

Recalling the definition of the dual norm as $\|y\|_{p^*} = \sup_{\|x\|_p} x^T y$, this is identical to the conic constraint

$$\bar{h}_i^T g + \|H_i^T g - B_i^T\|_{p^*} \leq \bar{b}_i$$

concluding the proof.

Proposition 4.8

Lemma 4.3 and the uncertainty parameterisation lead to the robust constraint:

$$(\bar{h}_i + H_i \theta)^T g \leq \bar{b}_i + B_i \theta, \quad \forall \theta \in \Theta$$

A rearrangement gives

$$\bar{h}_i^T g + \theta^T (H_i^T g - B_i^T) \leq \bar{b}_i, \quad \forall \theta \in \Theta$$

which, using the definition of the support function, is equivalent to:

$$\bar{h}_i^T g + \sigma_\Theta (H_i^T g - B_i^T) \leq \bar{b}_i$$

Recalling that the support function is equal to the gauge function of the polar set yields

$$\bar{h}_i^T g + \lambda_i \leq \bar{b}_i, \quad H_i^T g - B_i^T \in \lambda_i \Theta^*$$

concluding the proof.

Proposition 4.13

If $\mathcal{P}[V, g(\beta)] \subseteq \mathcal{P}[F, b]$ then by Lemma 4.2 there exists a H satisfying the given conditions with $Hg(\beta) \leq b$. As there is only one H satisfying these conditions, we must have $H = H^*$ and $H\alpha \leq b$ by choosing $\alpha = \beta$.

Theorem 4.14

The proof is by construction. Suppose $\mathcal{P}[V, g(\beta)] \subseteq \mathcal{P}[f^T, b]$ and define $a_i = g_i(\beta)$ for all $i \notin \mathcal{I}$ and $a_i = \min\{V_i x^*, g_i(\beta)\}$ otherwise. Then, recalling that $\tilde{h}_j = 0\ \forall j \notin \mathcal{I}$, we have $\tilde{h}^T a = \sum_{i\in\mathcal{I}} \tilde{h}_i a_i \leq \sum_{i\in\mathcal{I}} \tilde{h}_i V_i x^* = \tilde{h}^T V x^* = f^T x^* \leq b$. So $\tilde{h}^T g(\alpha) \leq b$ as required. Since $a \leq g_i(\beta)$ for all i, any x with $Vx \leq a$ must also satisfy $Vx \leq g(\beta)$, hence $\mathcal{P}[V, a] \subseteq \mathcal{P}[V, g(\beta)]$. On the other hand, the condition that $\mathcal{P}[V, g(\beta)]$ lies in the given cone guarantees that in setting $a_i = \min\{V_i x^*, g_i(\beta)\}, i \in \mathcal{I}$ no points have been removed from the set and hence $\mathcal{P}[V, g(\beta)] \subseteq \mathcal{P}[V, a]$. Combining these two subset relations implies $\mathcal{P}[V, a] = \mathcal{P}[V, g(\beta)]$. This proves the 'if' part of the theorem.

Suppose that there exists a β such that $\mathcal{P}[V, \beta]$ is not a subset of the given cone, consider $c = \max_x\{f^T x|\ Vx \leq g(\beta)\}$ and define $\gamma = bg(\beta)/c$. Then $\max_x\{f^T x|\ Vx \leq \gamma\} = b$ so that $\mathcal{P}[V, \gamma] \subseteq \mathcal{P}[f^T, b]$. As $\mathcal{P}[V, g(\beta)]$ was not a subset of the given cone, neither is $\mathcal{P}[V, \gamma]$ a subset of the cone $\{x \in \mathbb{R}^n|\ V_i x \leq V_i x',\ i \in \mathcal{I}\}$ where $x' \in \arg\max_x\{f^T x|\ Vx \leq \gamma\}$, implying that there exists a point $\tilde{x}$ in $\mathcal{P}[V, \gamma]$ for which $V_j \tilde{x} > V_j x'$ for some $j \in \mathcal{I}$ and hence $\gamma_j > V_j x'$. As x' is also in $\mathcal{P}[V, \gamma]$ we have $\gamma \geq Vx'$, and together with $\gamma_j > V_j x'$ this implies that $\tilde{h}^T \gamma > \tilde{h}^T V x' = f^T x' = b$ so that $\tilde{h}^T$ cannot be tight.

Theorem 4.15

Considering Definition 4.1 we can choose b to have a finite value in the ith element and ∞ in all the others, and since the conditions of Definition 4.1 must apply for all b this implies that the ith row of any tight H must be tight. Conversely, if all the rows of H are tight, then we find $H\alpha \leq b$ if $\mathcal{P}[V, \beta] \subseteq \mathcal{P}[F, b]$ as this inclusion gives $h_i^T \alpha \leq b_i$ for all i.

Proposition 4.16

Due to the positive multiple condition, there is an H satisfying $H \geq 0$, $HV = F$ with a single element in each row. The convex cone condition of Theorem 4.14 is thus trivially satisfied, since any polytope is a subset of the half spaces defined by its facets.

Theorem 4.18

This is easiest to prove by a geometric argument: As the homothetic parameterisation simply translates and scales a set around the state space, the cone condition of Theorem 4.14 will always hold.

Proposition 4.19

In $\mathbb{R}^1$, the convex cone of Theorem 4.14 is a half space and hence the condition of the theorem is trivially satisfied. In $\mathbb{R}^2$, such a cone can always be constructed by computing $\cos^{-1}(V_i^{(1)})$ for all i and choosing the facet for which $\cos^{-1}(V_i^{(1)}) - \cos^{-1}(f)$ is minimised but positive and the facet for which $\cos^{-1}(V_i^{(1)}) - \cos^{-1}(f)$ is maximised but negative.

Theorem 4.21

If $H(\theta)$ is tight, then if $H(\theta)\alpha \leq b(\theta)$ there is a choice of β such that $\mathcal{P}[V, \beta] \subseteq \mathcal{R}_\Theta[F, b]$. For fixed θ, this implies tightness by Definition 4.1. Conversely, if we have a tight choice of H_θ for each θ then this defines a (piecewise affine) function $H(\theta)$ that is tight for the robust inclusion.

Theorem 4.22

We proceed by construction. Let $\psi(i, j)$ give the index of the row in V containing $F_j^{[i]}$ and $\psi(0, j)$ give the row containing $\bar{F}_j$. Then we can choose each row of $H(\theta)$ by setting the $\psi(0, j)$'th element in that row to be 1 and the $\psi(i, j)$'th element to be θ_i. Suppose the subset relation $\mathcal{P}[V, g(\beta)] \subseteq \mathcal{R}_\Theta[F, b]$ holds implying both $\mathcal{P}[V, g(\beta)] \subseteq \mathcal{P}[\bar{F}, \bar{b}]$ and $\mathcal{P}[V, g(\beta)] \subseteq \mathcal{P}[F^{[i]}, b^{[i]}]$. Then define $\alpha_j = \max_x\{V_j x | Vx \leq g(\beta)\}$ with x_j^* the argument for which this maximum is attained. Clearly $\mathcal{P}[V, g(\alpha)] = \mathcal{P}[V, g(\beta)]$ and we have:

$$H_j(\theta)g(\alpha) = g(\alpha_{\psi(0,j)}) + \sum_i g(\alpha_{\psi(i,j)})\theta_i = V_{\psi(0,j)}x^*_{\psi(0,j)} + \sum_i V_{\psi(i,j)}x^*_{\psi(i,j)}\theta_i$$

$$= \bar{F}_j x^*_{\psi(0,j)} + \sum_i F_j^{[i]} x^*_{\psi(i,j)}\theta_i \leq \bar{b}_j + \sum_i b_j^{[i]}\theta_i = b_j(\theta)$$

The inequality follows from $x^*_{\psi(0,j)} \in \mathcal{P}[V, g(\alpha)] \subseteq \mathcal{P}[\bar{V}, \bar{g}]$ and $x^*_{\psi(i,j)} \in \mathcal{P}[V, g(\alpha)] \subseteq \mathcal{P}[V^{[i]}, g^{[i]}]$. As this holds for all rows j, the proof is complete.

Proposition 4.23

From Theorem 4.14, any row h is tight if and only if the set $\mathcal{P}[V, g(\beta)]$ is contained in the cone $\{x \in \mathbb{R}^n |\ V_i x \leq V_i x^*(\beta),\ \ i \in \mathcal{I}\}$, for any

$$x^*(\beta) \in \arg\max_x \{f^T x |\ Vx \leq g(\beta)\}$$

for all β, where $\mathcal{I}$ contains the indices of the nonzero elements of h. Consider $\gamma_i(\beta) = V_i x^*(\beta)$ for all $i \in \mathcal{I}$ and $\gamma_i(\beta) = \infty$ otherwise. Then as $\mathcal{P}[V, g(\beta)]$ is contained within the given cone, the active set of $\max_x \{f^T x |\ V_i x \leq \gamma_i(\beta), \forall i\}$ coincides with the nonzero elements of h for any choice of β. Hence there can be at most n nonzero elements of h in total.

Theorem 4.24

If H_i^* is not tight then $H_i^* g(\alpha^*) > b_i$ for some $\alpha^* \in S(h)$. By definition, any tight H_i will have $H_i g(\alpha) \leq b_i$ for all $\alpha \in S(h)$, hence if a tight H_i exists it contradicts H_i^* being the minimax solution unless H_i^* is also tight.

Theorem 4.25

If $m = \dim(H_j)$, then noting that the optimisation of Algorithm 4.1 has n equality constraints, at least $m-n$ inequality constraints will be active at the solution. Therefore at least $m - n$ elements of H_j are zero.

Theorem 4.27

As the algorithm is initialised with $\{x \in \mathbb{R}^n | Vx \leq g\} = \{x \in \mathbb{R}^n | \tilde{F}x \leq 1\}$, and the only removed constraints are redundancies, it is clear that the feasibility of the set is preserved throughout execution of the algorithm. If the algorithm terminates then, from the subset condition, we have

$$\{x \in \mathbb{R}^n | Vx \leq g\} \subseteq \{x \in \mathbb{R}^n | V\tilde{A}^{(j)}x \leq g(i) - V(i,:)w^{(j)}\}$$

for all i, implying that the resulting set is a subset of its preimage under the dynamics. Hence it is positively invariant.

5 Tube MPC of LDI systems

5.1 Introduction

In control theory literature there are several MPC techniques that have been developed for use with systems that have parametric, or multiplicative, uncertainty. An early approach relying on LMI optimisations appears in [KBM94], and more recent publications have considered improvements to the technique [CFF04] as well as moving some of the computational effort offline in order to give simpler online optimisations [KRS00, PRSDM05c]. Unfortunately, to robustly handle constraints these methods either rely on a very conservative bounding of the current state (usually in an ellipsoidal invariant set) or, for polytopic uncertainty, involve a number of constraints that grows exponentially with the length of the prediction horizon. Tube MPC provides an alternative to these methods by giving a tractable way to bound the predicted state in polyhedral sets that can be determined by the controller as part of its optimisation.

The theory of polyhedral inclusions developed in the previous chapter is now applied to form tube MPC controllers suitable for difference inclusion systems. Tubes of this type have been presented before for systems with multiplicative disturbances [ECK12a, FKC13] but the treatment here has several key differences and improvements:

- There is no requirement for the uncertainty to be polytopic, instead it can lie in any convex set;
- The system dynamics can include additive disturbance terms as well as parametric uncertainties;
- The parameterisation of the prediction tube can be defined by any affine function, and hence can be tailored to a given application;
- The terminal conditions used are less restrictive, and an intuitive 'reproducibility' condition is presented that guarantees strong recursive feasibility;

Considered together, these innovations result in a very versatile theoretical framework for robust MPC of differential inclusions. As in the case of deterministic MPC, the number of constraints in the online optimisation increases only linearly with the length of the prediction horizon. A design and simulation example is given, in which the presented techniques are shown to greatly outperform MPC controllers that use ellipsoidal invariant sets and LMI optimisations to tackle these problems.

5.2 Problem formulation

Denoting the state of a discrete-time system by x_t, and two outputs by y_t and z_t, we will consider the uncertain dynamics

$$x_{t+1} = A(\theta_t)x_t + B(\theta_t)u_t + w(\theta_t) \tag{5.1a}$$

$$y_t = C(\theta_t)x_t + D(\theta_t)u_t \tag{5.1b}$$

$$z_t = F(\theta_t)x_t + G(\theta_t)u_t \tag{5.1c}$$

where the system matrices and disturbance are affine functions of an unknown parameter θ that lies within a bounded set $\theta_t \in \Theta$. Speaking precisely, it may be more accurate to call such a system an affine difference inclusion rather than a 'linear' one, but this terminology does not seem to be standard. Hence we will refer to (5.1) as an LDI, noting that it includes truly linear difference inclusions as a special case when $w = 0$.

Remark 5.1. It is notable that uncertainty defined by a linear fractional transform (LFT) can also be cast in the form (5.1) by a suitable state augmentation. LFT uncertainty leads to system dynamics

$$x_{t+1} = A_x x_t + B_p p_t + B_x u_t \tag{5.2a}$$

$$q_t = C_q x_t + C_p p_t + D_q u_t \tag{5.2b}$$

$$p_{t+1} = \Delta_p(\theta_t)q_t, \;\; \theta_t \in \Theta \tag{5.2c}$$

from which uncertain dynamics in the form (5.1) can be found if the state x_t is augmented with the signal p_t, after substituting (5.2b) into (5.2c). See, for example, the definition of norm-bounded LDIs in [BEGFB94].

We will assume an observability condition on the output y_t, which will be required to prove stability of the system later in the chapter. This is natural, as y_t will appear in the cost function to be minimised by the online MPC controller, and we cannot hope to stabilise any unobservable modes by feedback on this output. The notion of observability that we require here is fairly weak - we assume only that we have observability for some θ in the uncertainty set.

Assumption 5.1. *There exists some $\theta \in \Theta$, such that the pair $(C(\theta), A(\theta))$ is observable and $D(\theta)^T D(\theta)$ is positive definite.*

Remark 5.2. This assumption ensures that the cost functions to be minimised by receding horizon control we will develop are positive definite functions of x and u. The observability condition can usually be relaxed to a detectability requirement such as $(A(\theta), C(\theta))$ being a detectable pair for some θ, or zero-state detectability (as defined in [KG96]) if the LDI is being used to approximate a nonlinear system. In such cases, the stability proofs of this chapter imply convergence of y_t rather than of x_t, and conclusions about the long term behaviour of x_t must be drawn from the detectability condition. See [MR09, Section 2.4.7].

The control objective is to make the output y_t small in some appropriate sense (such as bounding its l_2 norm) while satisfying a constraint on the output z_t so that

$$z_t = F(\theta_t)x_t + G(\theta_t)u_t \leq 1 \tag{5.3}$$

for all times t. This condition can be viewed as a robust linear constraint as, noting that θ_t is not available to the controller at time t, it must hold for all $\theta_t \in \Theta$.

In light of [BTN98, Proposition 2.1], it is possible to replace nonconvex disturbance sets with their convex hulls without any conservatism if the constraints in a robust optimisation problem are concave in the uncertain parameter. As we are assuming an affine parameter dependence here, we can assume without any loss of generality that Θ is convex.

Assumption 5.2. *The uncertainty set Θ is compact and convex.*

To obtain an online optimisation that is finite and tractable, it is necessary to choose some parameterisation of the predicted system inputs u_t^k that involve a finite number of decision variables. We choose to optimise a sequence of perturbations on a fixed feedback law, so that we choose c^k and set:

$$u^k = Kx^k + c^k \tag{5.4}$$

The change from subscripts to superscripts here is meant as a simple reminder that we are now working with predicted, and not actual quantities. We require that the state feedback K is stabilising for the system.

Assumption 5.3. *The state dynamics (5.1) are quadratically stabilised by the state feedback $u^k = Kx^k$.*

Remark 5.3. We assume here that stabilising state feedbacks are already known for the system in question. A useful way to choose such a feedback, that gives an l_2-gain of less than γ from w_t to y_t, is to maximise trace(Q) subject to $Q \succeq 0$ and

$$\begin{bmatrix} Q & Q\Phi(\theta)^T & Q\Delta(\theta) & Q\tilde{w}(\theta)^T \\ \Phi(\theta)Q & Q & 0 & 0 \\ \Delta(\theta)Q & 0 & I & 0 \\ \tilde{w}(\theta)Q & 0 & 0 & \gamma^{-2}I \end{bmatrix} \succeq 0, \quad \forall \theta \in \Theta \tag{5.5}$$

where $\tilde{w}(\theta) = [w(\theta)^T \;\; 0]^T$ and we have

$$\Phi(\theta)Q = \begin{bmatrix} A(\theta) & w(\theta) \\ 0 & 1 \end{bmatrix} Q + \begin{bmatrix} B(\theta) \\ 0 \end{bmatrix} \tilde{K}Q$$

$$\Delta(\theta)Q = \begin{bmatrix} C(\theta) & 0 \end{bmatrix} Q + D(\theta)\tilde{K}Q$$

where $\tilde{K} = [K \;\; 0]$. The standard change of variables $Y = \tilde{K}Q$ reveals (5.5) as a robust linear matrix inequality in Q and Y. This gives a robust SDP that can be solved by either vertex enumeration (for polytopic uncertainty) or by finding a suitable inner approximation of the feasible set as in [EGOL98]. Alternative methods for designing state feedback controllers can be found in [BEGFB94] for instance.

The predicted dynamics of the system (5.1) with inputs given by (5.4) can be expressed in the convenient form

$$x^{k+1} = \tilde{A}(\theta^k)x^k + B(\theta^k)c^k + w(\theta^k) \tag{5.6a}$$

$$y^k = \tilde{C}(\theta^k)x^k + D(\theta^k)c^k \tag{5.6b}$$

$$z^k = \tilde{F}(\theta^k)x^k + G(\theta^k)c^k \tag{5.6c}$$

where we define $\tilde{A}(\theta) = A(\theta)+B(\theta)K$, $\tilde{C}(\theta) = C(\theta)+D(\theta)K$ and $\tilde{F}(\theta) = F(\theta)+G(\theta)K$.

5.3 Cost function evaluation

In robust MPC, we would generally like to minimise some cost function that reflects the 'worst-case' evolution of the system. For this, minimax MPC formulations are often advocated, but it is generally difficult to evaluate minimax cost functions directly when parametric uncertainty is present in a system. Recalling the approaches in [Löf03b], we first develop a quadratic cost function that serves as a conservative upper bound on a minimax cost resembling those found in $\mathcal{H}_\infty$ control. Later in the chapter, this will be seen to lead to an l_2 stability guarantee of the closed-loop system.

In case a stronger stability guarantee is sought, we additionally give a cost function that penalises the deviation of y_t from that expected if the state x_t were in some small invariant set of the LDI (5.1). This resembles a minimisation of the l_2 norm of the system output and, although it results in a somewhat artificial cost function, it can lead to an exponential guarantee of convergence to the target set in closed-loop.

5.3.1 Satisfying an l_2-gain bound

If we introduce a vector of input predictions $\underline{c}_t$, and its 'tail' $\tilde{c}_{t+1}$ as

$$\underline{c}_t = \begin{bmatrix} c_t^t \\ c_t^{t+1} \\ \vdots \\ c_t^{t+N-1} \end{bmatrix}, \quad \tilde{c}_{t+1} = \begin{bmatrix} c_t^{t+1} \\ \vdots \\ c_t^{t+N-1} \\ 0 \end{bmatrix}$$

and define matrices T and S such that $c_t^t = S\underline{c}_t$ and $\tilde{c}_{t+1} = T\underline{c}_t$, the predicted state dynamics of the system (5.6) can be succinctly expressed in the form:

$$\begin{bmatrix} x^{k+1} \\ \underline{c}^{k+1} \\ 1 \end{bmatrix} = \begin{bmatrix} \tilde{A}(\theta^k) & B(\theta^k)S & w(\theta^k) \\ 0 & T & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x^k \\ \underline{c}^k \\ 1 \end{bmatrix} \tag{5.7a}$$

$$y^k = \begin{bmatrix} \tilde{C}(\theta^k) & D(\theta^k)S & 0 \end{bmatrix} \begin{bmatrix} x^k \\ \underline{c}^k \\ 1 \end{bmatrix} \tag{5.7b}$$

This can be written more simply as $\zeta^{t+1} = \Psi(\theta^t)\zeta^t$ and $y^t = \Lambda(\theta^t)\zeta^t$ with the augmented state vector ζ^k and matrices $\Psi(\theta)$, $\Lambda(\theta)$ defined by comparison with (5.7):

$$\Psi(\theta) = \begin{bmatrix} \tilde{A}(\theta^k) & B(\theta^k)S & w(\theta^k) \\ 0 & T & 0 \\ 0 & 0 & 1 \end{bmatrix}, \qquad \Lambda(\theta) = \begin{bmatrix} \tilde{C}(\theta^k) & D(\theta^k)S & 0 \end{bmatrix}$$

We can use this autonomous formulation of the predicted dynamics to develop a quadratic bound on a cost function that, when minimised in an online optimisation, implies a bound on the l_2-gain from w_t to y_t. A sufficient condition for such a bound to hold is given in the following proposition, which exploits the autonomous prediction dynamics (5.7) to form a Lyapunov matrix inequality that bounds the output l_2 norm by a quadratic form $\zeta_t^T W \zeta_t$, which can be minimised online by the controller.

Proposition 5.4. *If there exists $W \succ 0$ satisfying*

$$W - \Psi(\theta)^T W \Psi(\theta) \succeq \Lambda(\theta)^T \Lambda(\theta) - \gamma^2 v(\theta) v(\theta)^T, \quad \forall \theta \in \Theta \tag{5.8}$$

where $v(\theta)^T = [0 \ 0 \ w(\theta)^T]$ then for any disturbance sequence of finite l_2 norm, the predicted l_2 norm of the output can be bounded as:

$$\sum_{k=t}^{\infty} \|y^k\|^2 \leq \zeta_t^T W \zeta_t + \gamma^2 \sum_{k=t}^{\infty} \|w^k\|^2 \tag{5.9}$$

Remark 5.5. Bounds such as the one given above are needed for efficient minimisation of a minimax quadratic cost function in MPC of a parameter-uncertain system since the explicit minimisation of

$$\max_{\theta^k \in \Theta} \left\{ \sum_{k=t}^{\infty} \|y^k\|^2 \right\}$$

is difficult in general (in fact, the associated decision problem is NP-hard). This is obvious if the substitution $y^k = C(\theta^k)x^k$ is made since this cost then becomes

$$\max_{\theta^k \in \Theta} \left\{ \sum_{k=t}^{\infty} \|C(\theta^k)x^k\|^2 \right\}$$

which is a positive semidefinite quadratic function of θ^k. It is well-known that maximisation of a positive semidefinite quadratic function has an NP-complete decision version, as the related decision problem is a polynomial-time reduction of the Boolean satisfiability problem - see [MK87].

By inserting $W^{-1}W$ into the second term, the quadratic matrix inequality (5.8) can be rearranged to

$$W - \Psi(\theta)^T W W^{-1} W \Psi(\theta) - \Lambda(\theta)^T \Lambda(\theta) + \gamma^2 v(\theta) v(\theta)^T \succeq 0, \quad \forall \theta \in \Theta$$

and applying the Schur Complement Lemma yields a linear matrix inequality:

$$\begin{bmatrix} W & \Psi(\theta)^T W & \Lambda(\theta)^T & v(\theta) \\ W\Psi(\theta) & W & 0 & 0 \\ \Lambda(\theta) & 0 & I & 0 \\ v(\theta)^T & 0 & 0 & \gamma^{-2} I \end{bmatrix} \succeq 0, \quad \forall \theta \in \Theta \tag{5.10}$$

For design purposes, an effective strategy to ensure tightness of the bound of Proposition 5.4 is to minimise trace(W) subject to (5.10) and $W \succ 0$ for some fixed γ. This is a robust semidefinite programming problem that is difficult in general, but can be solved suboptimally by using an inner approximation of the feasible region [EGOL98]. One important case which is solvable exactly is when the uncertainty set Θ is polytopic, as the robust SDP can then be solved by vertex enumeration.

It should be noted that the inclusion of the additive disturbance $w(\theta)$ as a function of θ in both (5.8) and (5.10) is quite deliberate, as it takes into account any codependence between $(A(\theta), B(\theta))$ and $w(\theta)$ (in the sense that they may both depend on the same components of θ). This would not be the case, for instance, if (5.8) were replaced with the quadratic matrix inequality

$$\begin{bmatrix} W & 0 \\ 0 & 0 \end{bmatrix} - \begin{bmatrix} \Psi(\theta)^T & 0 \\ I & I \end{bmatrix} \begin{bmatrix} W & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \Psi(\theta) & I \\ 0 & I \end{bmatrix} \succeq \begin{bmatrix} \Lambda(\theta)^T\Lambda(\theta) & 0 \\ 0 & -\gamma^2 I \end{bmatrix}$$

which, by pre- and post- multiplying by $[\zeta_t^T w_t^T]$ and its transpose, would also give an l_2-gain bound of γ^2, but might give a worse bound in (5.9) if $(A(\theta), B(\theta))$ and $w(\theta)$ are codependent.

5.3.2 Penalising deviations from a target set

An alternative to minimising the bound (5.9) is to attempt to drive the system to some target set by penalising deviations of the state from that set. Suppose that Ω is some small invariant set for the dynamics (5.6a) with $c_t = 0$. This could be the minimal invariant set or some approximation to it, and algorithms for computing such sets are considered in [KRK+05] for example. We can then consider minimising an upper bound on the cost

$$\min_{s^t \in \Omega} \sum_{k=t}^{\infty} \|C(\theta^k)(x^k - s^k) + D(\theta^k)(u^k - Ks^k)\|^2 \qquad (5.11)$$

where we constrain the predicted s^k to satisfy $s^{k+1} = A(\theta^k)s^k + B(\theta^k)Ks^k + w(\theta^k)$. Noting that we specified Ω to be an invariant set for these dynamics, we have $s^k \in \Omega$ for all $k \geq t$.

For convenience in evaluating the cost, we now define an augmented state vector

$$\phi_t = \begin{bmatrix} x_t - s_t \\ \underline{c}_t \end{bmatrix}$$

and similarly to the previous section we consider autonomous dynamics of the controlled system, leading to the following dynamics. Note that the subtraction of s_{t+1} in ϕ_{t+1}

cancels the effect of w_t on the subsequent state.

$$\begin{bmatrix} x_{t+1} - s_{t+1} \\ \underline{c}_{t+1} \end{bmatrix} = \begin{bmatrix} \tilde{A}(\theta_t) & B(\theta_t)S \\ 0 & T \end{bmatrix} \begin{bmatrix} x_t - s_t \\ \underline{c}_t \end{bmatrix} \tag{5.12a}$$

$$y_k = \begin{bmatrix} \tilde{C}(\theta_t) & D(\theta_t)S \end{bmatrix} \begin{bmatrix} x_t - s_t \\ \underline{c}_t \end{bmatrix} \tag{5.12b}$$

This can be written more simply as $\phi_{t+1} = \Psi(\theta_t)\phi_t$ and $y_t = \Lambda(\theta_t)\zeta_t$ where the matrices $\Psi(\theta)$, $\Lambda(\theta)$ are defined by comparison with (5.12). Similar to the bound on output l_2 norm in the previous section, these autonomous dynamics can be exploited to find Lyapunov-type matrix inequalities that give the cost bound using a quadratic form.

Proposition 5.6. *If there exists $X \succ 0$ satisfying*

$$X - \Psi(\theta)^T X \Psi(\theta) \succeq \Lambda(\theta)^T \Lambda(\theta), \quad \forall \theta \in \Theta \tag{5.13}$$

then the predicted cost (5.11) can be bounded as:

$$\sum_{k=t}^{\infty} \|C(\theta^k)(x^k - s^k) + D(\theta^k)(u^k - Ks^k)\|^2 \leq \phi_t^T X \phi_t \tag{5.14}$$

By inserting $X^{-1}X$ into the second term, the Lyapunov inequality (5.13) can be written as

$$X - \Psi(\theta)^T X X^{-1} X \Psi(\theta) - \Lambda(\theta)^T \Lambda(\theta) \succeq 0, \quad \forall \theta \in \Theta$$

applying the Schur Complement Lemma yields a linear matrix inequality

$$\begin{bmatrix} X & \Psi(\theta)^T X & \Lambda(\theta)^T \\ X\Psi(\theta) & X & 0 \\ \Lambda(\theta) & 0 & I \end{bmatrix} \succeq 0, \quad \forall \theta \in \Theta \tag{5.15}$$

and once again, a good strategy to apply this bound in practice is to minimise trace(X) subject to this LMI and $X \succeq 0$.

5.4 Constraint handling using tubes

We now introduce functions $g_t^k(\alpha^k)$, which we call the ***tube parameterisation functions***, that will be used to define polytopes that contain the predicted system state x_t^k. The variables α^k will be a free variable in an online MPC optimisation.

Assumption 5.4. *The functions g_t^k are affine in α^k.*

The assumption that the tube parameterisation functions are affine ensures that inequality constraints such as $Vx^k \leq g^k(\alpha^k)$ are themselves affine in α^k, which will later allow us to express the online MPC optimisation as a convex programming problem.

Consider the prediction dynamics of the system (5.6) for a sequence of control perturbations c_k under the control (5.4).

$$x^{k+1} = \tilde{A}(\theta^k)x^k + B(\theta^k)c^k + w(\theta^k) \tag{5.16}$$

For later convenience, we now define a set-valued function $\Phi : \mathbb{R}^n \times \mathbb{R}^p \to P(\mathbb{R}^n)$ representing these controlled dynamics as

$$\Phi(x, c) = \{\tilde{A}(\theta)x + B(\theta)c + w(\theta) |\ \theta \in \Theta\} \tag{5.17}$$

and suppose that there exists a matrix of facet normals V, that defines polyhedral sets $\mathcal{P}[V, g^k(\alpha^k)]$ that will be used to contain predicted states of the system. Abusing notation, we will write $\Phi\left(\mathcal{P}[V, g^k(\alpha^k)], c\right)$ for the image of this polyhedron under the set-valued mapping. We assume that the chosen parameterisation is capable of covering $\mathbb{R}^n$, so that there are no states that cannot be placed within a polyhedron.

Assumption 5.5. *The matrix V and the functions g_t^k are chosen so that, for all t and k and for all $x \in \mathbb{R}^n$ there exists α such that $x \in \mathcal{P}[V, g_t^k(\alpha)]$.*

It is now possible to give a formal definition of a state tube as a sequence of polytopes that contain the predicted states under the uncertain dynamics.

Definition 5.1. *A sequence of sets $\mathcal{P}[V, g_t^k(\alpha_t^k)]$ for $k = t, \ldots, t+N$ is called a* **state**

tube *if* $x_t \in \mathcal{P}[V, g_t^t(\alpha_t^t)]$ *and:*

$$\Phi\left(\mathcal{P}[V, g^k(\alpha^k)], c^k\right) \subseteq \mathcal{P}[V, g^{k+1}(\alpha^{k+1})] \tag{5.18}$$

The members of this sequence are called **tube cross-sections**.

The relation in this definition can be written equivalently using a preimage as

$$\mathcal{P}[V, g^k(\alpha^k)] \subseteq \Phi^{-1}(\mathcal{P}[V, g^{k+1}(\alpha^{k+1})], c^k)$$

so that (5.18) is equivalent to the robust inclusion:

$$\mathcal{P}[V, g^k(\alpha^k)] \subseteq \mathcal{R}_\Theta[V\tilde{A}, g^{k+1}(\alpha^{k+1}) - VBc^k + Vw] \tag{5.19}$$

To ensure satisfaction of the constraint (5.3), the tube cross-section at time k will also be required to satisfy:

$$\mathcal{P}[V, g^k(\alpha^k)] \subseteq \mathcal{R}_\Theta[\tilde{F}, 1 - Gc^k] \tag{5.20}$$

The MPC problem can then be viewed as the minimisation of some cost function over the parameters α^k and c^k subject to the constraints (5.18) and (5.20). Currently these constraints are not in a form that can be exploited by standard optimisation solvers, but we can use the conditions of Chapter 4 to remedy this.

5.4.1 Inclusion and feasibility conditions

We now give a necessary and sufficient condition for the inclusion (5.18), a direct application of Lemma 4.3, which ensures that the sequence of sets $\mathcal{P}[V, g_k(\alpha)]$ is a state tube in the sense of Definition 5.1.

Proposition 5.7. *If there exists a matrix function* $\Gamma(\theta) \geq 0$ *such that*

$$\Gamma(\theta)V = V\tilde{A}(\theta) \tag{5.21a}$$

$$\Gamma(\theta)g^k(\alpha^k) + VB(\theta)c^k + Vw(\theta) \leq g^{k+1}(\alpha^{k+1}) \tag{5.21b}$$

for all $\theta \in \Theta$*, then (5.18) holds.*

Remark 5.8. In the case that $w(\theta)$ and $A(\theta), B(\theta)$ depend on different components of θ, a considerable simplification is available in (5.21b). To make this more precise, consider a decomposition $\theta = (\theta_a, \theta_b)$ such that the uncertainty set is a Cartesian product $\Theta = \Theta_a \times \Theta_b$. If we have $w(\theta) = w_a(\theta_a)$ and $A(\theta) = A_b(\theta_b)$, $B(\theta) = B_b(\theta_b)$ for some functions w_a, A_b, B_b then we can introduce a quantity $\bar{v} = \max_{\theta_a \in \Theta_a} V w_a(\theta_a)$ and replace the third term in (5.21b) with this parameter without introducing any conservatism.

In addition to the tube containing the predicted states, it is also a requirement that the tube cross-sections satisfy the constraints. This can be ensured by another application of Lemma 4.3.

Proposition 5.9. *If there exists a matrix function $\Delta(\theta) \geq 0$ such that*

$$\Delta(\theta)V = \tilde{F}(\theta) \tag{5.22a}$$

$$\Delta(\theta)g^k(\alpha^k) + G(\theta)c^k \leq 1 \tag{5.22b}$$

for all $\theta \in \Theta$, then (5.20) holds.

These Propositions give sufficient conditions for (5.18) and (5.20) in the form of the inequality constraints (5.22b) and (5.21b). From the discussion in Section 4.3, these matrix functions $\Gamma(\theta)$ and $\Delta(\theta)$ must be fixed offline in order that these inequality constraints result in a convex online optimisation.

5.4.2 Terminal conditions

The typical method used to ensure recursive feasibility in MPC is to place the final predicted state in some safe set, called a ***terminal set***, defined by a terminal constraint on x_N such as:

$$F^N x^N \leq h^N \tag{5.23}$$

It is straightforward to find a condition ensuring that the final tube-cross section $\mathcal{P}[V, g^N(\alpha^N)]$ satisfies this terminal constraint. However, this would not give any guarantee, in general, of recursive feasibility of MPC formulations using the tube, as it would not guarantee the existence of a parameter α^N at subsequent times. Instead, we consider enforcing the terminal condition

$$\Phi(\mathcal{P}[V, g^N(\alpha^N)], 0) \subseteq \mathcal{P}[V, g^N(\alpha^N)] \tag{5.24}$$

that is, that the final tube cross-section must itself be an invariant set for $c_N = 0$. This means that, when repeating the optimisation at time $t+1$, it will be possible to choose $\alpha_t^N = \alpha_{t+1}^{N+1}$.

Proposition 5.10. *If there exists a matrix function* $\Gamma(\theta) \geq 0$ *such that*

$$\Gamma(\theta)V = V\tilde{A}(\theta) \tag{5.25a}$$

$$\Gamma(\theta)g^N(\alpha^N) + Vw(\theta) \leq g^N(\alpha^N) \tag{5.25b}$$

for all $\theta \in \Theta$ *then (5.24) holds.*

This condition will still not be enough to ensure recursive feasibility and hence constraint satisfaction generally. However, in Section 5.4.4 it will be seen that there is an intuitive condition on the functions g^k defining the tube parameterisation that does guarantee recursive feasibility with this terminal condition.

Remark 5.11. In order to ensure feasibility of the resulting online optimisation problems, it is important to ensure that there exists an α^N such that (5.25b) holds for the choice of $\Gamma(\theta)$ made offline. This will always occur if there is an α^N for which the final tube cross-section is robustly invariant and a tight choice of $\Gamma(\theta)$ is used, due to the results in Section 4.3. If a tight $\Gamma(\theta)$ is not available for the inclusion (5.24), then the constraint (5.25b) for the invariant α^N should be applied when designing $\Gamma(\theta)$ using the algorithms in Section 4.3.5.

5.4.3 Tube constraints

Now we use the constraints of the previous two sections to give a set of constraints on c_k and α that ensure the predicted states satisfy constraints (5.3) and ensure $Vx^k \leq g^k(\alpha^k)$, with the final tube cross-section $\mathcal{P}[V, g_t^N(\alpha_t^N)]$ a robustly invariant set.

Theorem 5.12. *If* $\Gamma(\theta)$ *and* $\Delta(\theta)$ *satisfy (5.21a) and (5.22a) and there exist* α_t^k *sat-*

isfying, for all $\theta \in \Theta$ and $k = t, \ldots, N-1$, the constraints

$$Vx^t \leq g^t(\alpha^t) \tag{5.26a}$$

$$\Delta(\theta)g^k(\alpha^k) + G(\theta)c^k \leq 1 \tag{5.26b}$$

$$\Delta(\theta)g^{t+N}(\alpha^{t+N}) \leq 1 \tag{5.26c}$$

$$\Gamma(\theta)g^k(\alpha^k) + VB(\theta)c^k + Vw(\theta) \leq g^{k+1}(\alpha^{k+1}) \tag{5.26d}$$

$$\Gamma(\theta)g^{t+N}(\alpha^{t+N}) + Vw(\theta) \leq g^{t+N}(\alpha^{t+N}) \tag{5.26e}$$

then the control policy (5.4) satisfies constraints (5.3) and (5.24) and guarantees $Vx^k \leq g^k(\alpha^k)$ and $Vx^k \leq g^N(\alpha^N)$ for $k \geq N$ under (5.16).

Remark 5.13. For implementation, the robust constraints (5.26) must be expressed as constraints that can be handled by a standard solver. This is straightforward when Θ is a polytope or an ellipsoid, as the results of Propositions 4.6 and 4.7 in Chapter 4 are directly applicable. In practice, for these and more complicated uncertainties, software exists to automatically convert these robust constraints to convex conic constraints. In particular the robust convex programming module of Yalmip can do this automatically – see [Lö04] and [Lö12].

For convenience, we introduce the notation $(\underline{c}_t, \underline{\alpha}_t) \in \mathcal{T}(x_t)$ to mean that the sequences $\underline{c}_t = (c_t^t, \ldots, c_t^{N-1})$ and $\underline{\alpha}_t = (\alpha_t^t, \ldots, \alpha_t^N)$ satisfy constraints (5.26a-e).

Definition 5.2. *Define the* **tube constraint set** *by*

$$\mathcal{T}(x_t) = \{(\underline{c}, \underline{\alpha}) | \ \underline{c} \text{ and } \underline{\alpha} \text{ satisfy (5.26) with } x^t = x_t\} \tag{5.27}$$

The constraint set implicitly defines regions of the state space by considering all x for which there exists c and α such that $(c, \alpha) \in T(x)$.

Definition 5.3. *The set $\mathcal{X}(\mathcal{T}) \subseteq \mathbb{R}^n$ is defined by*

$$\mathcal{X}(\mathcal{T}) = \{x \in \mathbb{R}^n | \ \exists(\underline{c}, \underline{\alpha}) \in \mathcal{T}(x)\}$$

and will be referred to as the **feasible set** *of the tube.*

The term 'feasible set' is chosen to coincide with the usual meaning of the term with regard to MPC optimisations. It is the region of the state space in which an MPC law using the tube can ensure constraint satisfaction.

An important aspect of any MPC controller is some guarantee of constraint satisfaction in closed-loop. Key to this is the condition that any feasible control policy must lead to a feasible problem at the next point in time. This recursive feasibility is a property of the constraints and hence, in tube MPC, is a property of the tube parameterisation.

Definition 5.4. *A tube constraint set $\mathcal{T}$ with feasible set $\mathcal{X}(\mathcal{T})$ is* **strongly recursively feasible** *if and only if $\mathcal{X}(\mathcal{T})$ is invariant under the closed-loop system dynamics (5.16), that is*

$$\Phi(\mathcal{X}(\mathcal{T}), c) \subseteq \mathcal{X}(\mathcal{T})$$

for any feasible c.

5.4.4 Reproducible tubes

The terminal constraint (5.25) implies that the final cross-section $\mathcal{P}[V, g^N(\alpha^N)]$ is positively invariant, and hence that the predicted state will be contained in the tube for all time. But in general this will not be enough to ensure recursive feasibility of the constraints (5.26) as it does not provide a guarantee that, given $(\underline{c}_t, \underline{\alpha}_t) \in \mathcal{T}(x_t)$, we can choose $(\underline{c}_{t+1}, \underline{\alpha}_{t+1}) \in \mathcal{T}(x_{t+1})$ at the following time. In linear MPC schemes, it is common to prove recursive feasibility constructively by considering the 'tail' of the input sequence and appending a known stabilising feedback law. For the tube MPC presented, we could take $c^N_{t+1} = 0$ to exploit the fact that the final tube cross-section is invariant for $u^N = Kx^N$. This will provide a feasible $\underline{c}_{t+1}$ if we can also ensure there is $\underline{\alpha}_{t+1}$ recreates the same tube cross-sections for times $t+1, \ldots, N+1$ as was given by $\underline{\alpha}_t$. This motivates the following restriction on the parameterisation functions g^k_t.

Definition 5.5. *A tube parameterisation defined by the functions $g^k_t(\alpha)$ is* **t+1 reproducible** *if, for all t, for any $\underline{\alpha}_t$ there exists $\underline{\alpha}_{t+1}$ such that $g^{N+1}_{t+1}(\alpha^{N+1}_{t+1}) = g^N_t(\alpha^N_t)$ and $g^k_{t+1}(\alpha^k_{t+1}) = g^k_t(\alpha^k_t)$ for all $k = t, \ldots, t+N$.*

This condition is natural given that we would like to be able to give a constructive proof for recursive feasibility, as it simply requires that the tube parameterisation can reproduce the same sequence of tube-cross sections at the subsequent time. Along with the terminal constraints (5.25), this reproducibility condition ensures strong recursive feasibility.

Theorem 5.14. *If the tube parameterisation is $t+1$-reproducible then the tube constraint set $\mathcal{T}$ as defined in (5.2) is strongly recursively feasible.*

For the majority of applications, any constraints on the system state and input do not vary with time and so there is no need to allow the tube parameterisation functions g_t^k to vary arbitrarily with t. Considering the form of the tube constraints (5.26), it will be convenient if the parameterisation satisfies the condition $g_t^k = g_{t+1}^k$ for each $k = t, \ldots, t+N$ since the constraints in the online optimisation will differ between times t and $t+1$ otherwise, which may make implementation difficult. The next definition formalises this, giving a condition that ensures (5.26) is identical for all t.

Definition 5.6. *A tube parameterisation defined by the functions g_t^k is* **t-invariant** *if $g_t^k = g_{t+1}^k$ for all $t \geq 0$ and all $k \geq 0$.*

As t-invariance will be such a common requirement in implementations of tube MPC controllers, it seems prudent to discover what affect it may have on the t+1 reproducibility condition of Definition 5.5. In particular, t-invariance implies that $g_t^{k+1} = g_{t+1}^k$, which immediately gives the following result after a substitution into Definition 5.5.

Theorem 5.15. *If a tube parameterisation given by functions g^k is t-invariant, then it is t+1 reproducible if and only if for any α^k there exists α^{k-1} such that $g^{k-1}(\alpha^{k-1}) = g^k(\alpha^k)$ for all $k = t+1, \ldots, N$.*

We will refer to the condition of Theorem 5.15 as '$k-1$ reproducibility', and can summarise the result by saying that a tube parameterisation is $k-1$ reproducible if it is t-invariant and $t+1$ reproducible. This result says that if we would like to construct a tube parameterisation that can reproduce itself at subsequent 't' (hence giving recursive feasibility), while giving identical constraints (5.26), then it must also be able to reproduce itself at earlier prediction times. This can be understood intuitively by noting that it implies the tube parameterisation can be 'simpler' (depending on fewer parameters α) at time $k+1$ than at time k, but not the other way around. For example, this means we cannot choose a more complex parameterisation for the terminal tube cross-section than the others. We have already commented on this in Remark 4.28 of the previous chapter.

5.5 Algorithms and stability

Now that we have the tube constraints (5.26) along with the conditions required for them to be recursively feasible, these can be combined with the quadratic cost function bounds to give MPC optimisations. Accordingly, we now simply assume that we have some recursively feasible tube formulation available to us.

Assumption 5.6. *The tube formulation $\mathcal{T}$ is strongly recursively feasible.*

From the discussion in the previous section and Section 4.4 of the previous chapter, we can form such a tube by choosing V to correspond to the face normals of some invariant set of the system (chosen using Algorithm 4.3) and parameterisation functions g^k that satisfy the condition of Theorem 5.15.

5.5.1 l_2-stability

We first consider using the bound of Proposition 5.4 to give a cost function that can be used along with the tube $\mathcal{T}$. We remind the reader of the definition of ζ_t

$$\zeta_t = \begin{bmatrix} x_t \\ \underline{c}_t \\ 1 \end{bmatrix}$$

as this augmented state vector is used in the statement of the online algorithm.

The offline design procedure is summarised in Algorithm 5.1 and the online MPC algorithm is stated in Algorithm 5.2. The optimisation solved at each time t is, for

Algorithm 5.1 Offline, LDI Robust Tube MPC with l_2 norm bound

1. Find a stabilising K by maximising trace(Q) subject to $Q \succeq 0$ and (5.5).
2. Find V such that $\mathcal{P}[V, \bar{g}]$ is robustly invariant for some $\bar{g}$, using Algorithm 4.3.
3. Choose an affine function $g(\alpha)$ to parameterise a prediction tube, for example by using a formulation from Section 4.4.2.
4. Form matrix functions $\Gamma(\theta)$ and $\Delta(\theta)$ satisfying $\Gamma(\theta)V = V\tilde{A}(\theta)$ and $\Delta(\theta)V = \tilde{F}(\theta)$ by using Algorithm 4.2.
5. Find W satisfying the conditions of Proposition 5.4, by minimising trace(W) subject to (5.10) and $W \succ 0$.

Algorithm 5.2 Online, LDI Robust Tube MPC with l_2 norm bound

At each time t, solve

$$\begin{array}{ll} \underset{\underline{c}_t, \underline{\alpha}_t}{\text{minimise}} & \zeta_t^T W \zeta_t \\ \text{subject to} & (\underline{c}_t, \underline{\alpha}_t) \in \mathcal{T}(x_t) \end{array}$$

and apply the control $u_t = Kx_t + c_t^t$.

---

instance, a quadratic program in the case that Θ is a polytope, or a second order cone program when Θ is defined by a 2-norm bound, due to the robust constraints (5.26). We can readily prove an l_2 stability result for the closed-loop system using this algorithm.

Theorem 5.16. *Under the control of Algorithm 5.2 the output of the LDI (5.1) satisfies*

$$\sum_{k=0}^{t} \|y_k\|^2 \leq \zeta_0^T W \zeta_0 + \gamma^2 \sum_{k=0}^{t} \|w_k\|^2$$

for all t if the optimisation is initially feasible.

A consequence of this result is that the l_2 norm of the output will be finite if the l_2 norm of w_t is also finite. If the l_2 norm of w_t is finite, by taking limits we have

$$\sum_{k=0}^{\infty} \|y_t\|^2 \leq \zeta_0^T W \zeta_0 + \gamma^2 \sum_{k=0}^{\infty} \|w_t\|^2$$

which shows clearly that the induced l_2 norm of the closed-loop system is less than γ^2 and we have $x_t \to 0$ as $t \to \infty$ from basic calculus. In the case that w_t does not have finite l_2 norm, but instead has a finite average power, we can divide through by t before taking limits to find

$$\lim_{t\to\infty} \frac{1}{t} \sum_{k=0}^{t} \|y_t\|^2 \le \gamma^2 \lim_{t\to\infty} \frac{1}{t} \sum_{k=0}^{t} \|w_t\|^2$$

so that the output y_t is also a finite power signal and the system scales this seminorm by at most a factor of γ^2.

5.5.2 Exponential stability to a set

The stability result of the preceding section is not particularly strong if the l_2 norm of the disturbance w_t is not finite, as in the case of a signal of nonzero, finite average power. But as we shall now see, it is possible to get a stronger stability result in this case by using the cost of Proposition 5.6 that is zero on some (hopefully small) invariant set Ω, which can be found using the methods in [KRK+05] for example.

Assumption 5.7. *Set Ω is invariant under the dynamics $s_{t+1} = \tilde{A}(\theta)s_t + w_t$.*

Recalling that we defined an augmented state

$$\phi_t = \begin{bmatrix} x_t - s_t \\ \underline{c}_t \end{bmatrix}$$

we will include s_t as a free variable in the optimisation problem. The resulting optimisation will therefore have an optimal cost that is nonzero only when $x_t \notin \Omega$, and otherwise will choose $\underline{c}_t = 0$ so the input $u_t = Kx_t$ is applied.

The immediate advantage of this controller is that it allows for a guarantee that the state converges to Ω exponentially fast as $t \to \infty$, regardless of θ_t. This resembles exponential stability results in tube MPC formulations for additive disturbances, for instance [MSR05] and [RKFC12], and is proven using similar methods.

Theorem 5.17. *Under the control of Algorithm 5.4 the state x_t approaches Ω exponentially if the optimisation is feasible initially. More specifically, we have*

$$\phi_t^T X \phi_t \le \delta^{\frac{t}{n}-1} \phi_0^T X \phi_0$$

Algorithm 5.3 Offline, LDI Robust Tube MPC with target set

1. Find a stabilising K by maximising trace(Q) subject to $Q \succeq 0$ and (5.5).
2. Find V such that $\mathcal{P}[V, \bar{g}]$ is robustly invariant for some $\bar{g}$, using Algorithm 4.3.
3. Choose an affine function $g(\alpha)$ to parameterise a prediction tube, for example by using a formulation from Section 4.4.2.
4. Form matrix functions $\Gamma(\theta)$ and $\Delta(\theta)$ satisfying $\Gamma(\theta)V = V\tilde{A}(\theta)$ and $\Delta(\theta)V = \tilde{F}(\theta)$ by using Algorithm 4.2.
5. Find X satisfying the conditions of Proposition 5.6, by minimising trace(X) subject to $X \succ 0$ and (5.15).

Algorithm 5.4 Online, LDI Robust Tube MPC with target set

At each time t, solve

$$\begin{array}{ll} \underset{\underline{c}_t, \underline{\alpha}_t, s_t}{\text{minimise}} & \phi_t^T X \phi_t \\ \text{subject to} & (\underline{c}_t, \underline{\alpha}_t) \in \mathcal{T}(x_t) \\ & s_t \in \Omega \end{array}$$

and apply the control $u_t = Kx_t + c_t^t$.

for some $0 \leq \delta < 1$, *so that both* $x_t - s_t$ *and* $\underline{c}_t$ *converge exponentially fast to zero in closed-loop.*

A technical complication in the proof is that we have only assumed that the system is observable for some θ (Assumption 5.1), and therefore the stage cost is not necessarily positive definite even though the cost function is. In particular this means that the cost function need not decrease immediately but might stay at a constant value for up to n steps. Indeed, if the stage cost function were positive definite we could make the slightly stronger claim

$$\phi_t^T X \phi_t \leq \delta^{\frac{t}{n}} \phi_0^T X \phi_0$$

for the same δ.

This exponential stability result holds true despite the fact that w_t may have an infinite l_2 norm. In fact, it is a straightforward extension of Theorem 5.17 to show that if w_t has a finite l_2 norm, the set Ω is reached in finite time, as in that case $x_t \to 0$ asymptotically.

5.5.3 Implementation and sparsity

We conclude the section with some comments on the implementation of Algorithms 5.2 and 5.4, especially with regard to the tube constraints $(\underline{c}_t, \underline{\alpha}_t) \in \mathcal{T}(x_t)$ which are given in equation (5.26). For simplicity we focus on the case that Θ is polytopic so that we can consider the matrices $G(\theta)$, $\Delta(\theta)$ etc. to be defined by their values at the vertices of the polytope, which we write as $G^{(i)}$, $\Delta^{(i)}$ respectively. The online optimisation can hence be expressed as a QP.

Although it varies a little depending on the software used, most QP solvers will accept inequality constraints in the form $A_c z_c \leq b_c$ where z_c is the free variable to be determined by the solver and the matrix A_c and vector b_c are supplied by the user. For the tube MPC presented in this chapter, z_c will contain elements corresponding to α^k and c^k for different values of k. Consider expressing the initial condition (5.26a), feasibility conditions (5.26b) and the inclusion constraints (5.26d) in this form. For simplicity, take the tube parameterisation function g as the identity as in the inner product parameterisation.

The corresponding matrix A_c and vector b_c have the form:

$$
\begin{bmatrix}
-I & 0 & 0 & 0 & \cdots \\
\Delta^{(1)} & G^{(1)} & 0 & 0 & \cdots \\
\Delta^{(2)} & G^{(2)} & 0 & 0 & \cdots \\
\vdots & \vdots & \vdots & \vdots & \cdots \\
\Delta^{(q)} & G^{(q)} & 0 & 0 & \cdots \\
\Gamma^{(1)} & VB^{(1)} & -I & 0 & \cdots \\
\Gamma^{(2)} & VB^{(2)} & -I & 0 & \cdots \\
\vdots & \vdots & \vdots & \vdots & \cdots \\
\Gamma^{(q)} & VB^{(q)} & -I & 0 & \cdots \\
0 & 0 & \Delta^{(1)} & G^{(1)} & \cdots \\
0 & 0 & \Delta^{(2)} & G^{(2)} & \cdots \\
\vdots & \vdots & \vdots & \vdots & \cdots \\
0 & 0 & \Delta^{(q)} & G^{(q)} & \cdots \\
\vdots & \vdots & \vdots & \vdots & \ddots
\end{bmatrix}
\begin{bmatrix}
\alpha^t \\ c^t \\ \alpha^{t+1} \\ c^{t+1} \\ \vdots
\end{bmatrix}
\leq
\begin{bmatrix}
-Vx_t \\ 1 \\ 1 \\ \vdots \\ 1 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 1 \\ 1 \\ \vdots \\ 1 \\ \vdots
\end{bmatrix}
$$

Here we have considered only the constraints corresponding to $k = t$ and the feasibility constraints at $k = t + 1$ for brevity, but it is immediately obvious that the constraint matrix A_c will be quite large but sparse with a 'banded' structure. In fact, we know also from Proposition 4.25 in the previous chapter that there are at most n nonzero elements in $\Gamma^{(i)}$ and $\Delta^{(i)}$ for each i.

This all means that the online optimisation problems will have a very sparse structure that should be exploited by the solver used in the implementation. For the simulation example in the following section, the general purpose solver Gurobi was used, which does exploit sparsity in the constraints. Although not considered in this thesis, it seems likely that specialised solvers could be written to take advantage of the sparse, banded nature of the constraint matrices when solving this optimisation problem.

5.6 Simulation example

To demonstrate the efficacy of the presented tube MPC, especially with respect to constraint handling, we now give a comparison with the approach of the widely-cited publication [KBM94] using an example system from the same paper. The system in question consists of two unit masses joined by a spring, with a control input given by a force on one of the masses. Considering an Euler discretisation with sample time $0.2s$, and assuming that the spring constant is unknown and possibly time-varying with a value between 0.5 and 20 gives uncertain dynamics

$$x_{t+1} = [\theta_t A^{(1)} + (1 - \theta_t)A^{(2)}]x_t + Bu_t$$

where $\theta \in [0, 1]$ and the matrices $A^{(1)}$, $A^{(2)}$ and B are given by:

$$A^{(1)} = \begin{bmatrix} 1 & 0 & 0.2 & 0 \\ 0 & 1 & 0 & 0.2 \\ -0.1 & 0.1 & 1 & 0 \\ 0.1 & -0.1 & 0 & 1 \end{bmatrix}, \quad A^{(2)} = \begin{bmatrix} 1 & 0 & 0.2 & 0 \\ 0 & 1 & 0 & 0.2 \\ -2 & 2 & 1 & 0 \\ 2 & -2 & 0 & 1 \end{bmatrix}$$

$$B = [0 \;\; 0 \;\; 0.2 \;\; 0]^T$$

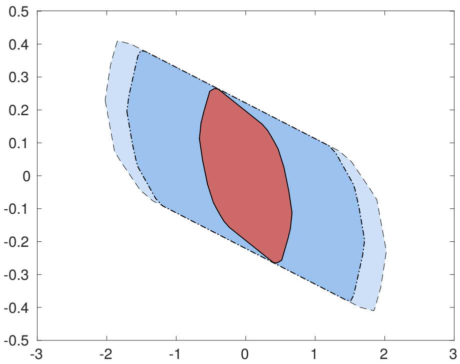

Figure 5.1: Terminal set comparison

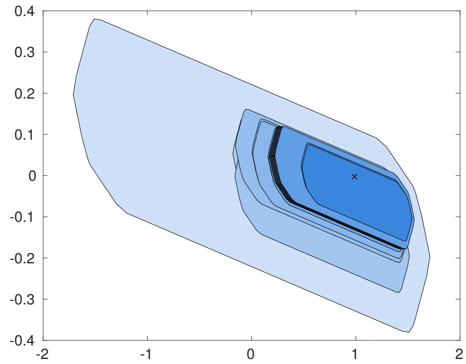

Figure 5.2: Optimal tube at $t = 0$

For the simulation the uncertain parameter θ_t was taken to be time-varying as

$$\theta_t = \frac{1}{2} - \frac{1}{2}\sin\frac{t}{2}$$

and the initial condition was set to $x_1 = x_2 = 1$, $x_3 = x_4 = 0$ as in [KBM94] corresponding to an initial perturbation on the two position states and zero initial velocity for both masses. The controllers were designed to regulate the system to the origin by, at each sample time, minimising the quadratic bound of Proposition 5.4 on the LQR-type stage cost

$$V_t = \sum_{k=t}^{\infty} \left(x_k^T x_k + u_k^2 \right)$$

while satisfying an input constraint $-1 \leq u_k \leq 1$.

The first step in the design of the tube MPC controller was to choose a prestabilising state feedback K. Initially, one was found by minimising a bound on an LQR-type cost with $Q = \mathrm{diag}(1, 1, 1, 1)$ and $R = 1$ using the LMI (5.5). However, the volume of the maximal admissible set was found to be rather small in this case, with 382 facets; Figure 5.1 shows the projection of the MAS onto the x_1 and x_3 axes in red. This would be restrictive for the tube MPC controller.

Physical intuition suggests that an optimal controller that assigns a greater penalty to the velocity states x_3 and x_4 will lead to a larger maximal admissible set, as it will tend to increase the damping of the resulting prestabilised system. Also, assigning a greater penalty to the input will give a larger feasible region in state space due to the input constraint. Indeed this is is the case: Figure 5.1 shows the MAS obtained when using K found by minimising a cost bound with $Q = \text{diag}(1, 1, 20, 20)$ and $R = 10$ in light blue. This polyhedron had 286 facets, which would give a large number of constraints in the tube MPC optimisation, so it was reduced using Algorithm 4.3 with $\gamma = 10$, $\delta = 0.02$ yielding the dark blue polyhedron, which has only 86 facets.

This polyhedral set was used to define V and g for the tube MPC. After designing Γ and Δ using Algorithm 4.1 and choosing a prediction horizon of $N = 20$, this resulted in a total of 3740 linear constraints in the tube MPC problem. Although this number might seem large, it is worth noting that a naive approach to enumerating all possible uncertainty combinations would lead to $2 \times 2^{20} = 2097152$ constraints in this case.

Given the relatively large number of rows in the chosen V, the homothetic tube parameterisation (4.6) was chosen to reduce the number of decision variables in the online optimisation. A cost matrix X was defined using Proposition 5.6 and the right-hand side of (5.14) used as a cost function, noting that in this case we can take $\Omega = \{0\}$ and hence set $s_t = 0$ in the online optimisation as there is no additive disturbance present. By Theorem 5.17 the closed-loop system is exponentially stable and the state will converge to the origin in closed loop. A typical sequence of tube cross-sections is shown in Figure 5.2, again projected onto the x_1 and x_3 coordinates. Note that the initial condition $x_1 = x_2 = 1$, $x_3 = x_4 = 0$ is in fact outside the final tube cross-section and outside the maximal admissible set, even though its projection is within the projection of the MAS as seen in the figure.

Table 5.1: Comparison of MPC optimisations

	Method of [KBM94]	Tube MPC
Optimisation type	SDP	QP
Solver	SeDuMi	Gurobi
Algorithm	Primal-dual interior point	Dual simplex
# Variables	20	126
# Linear Constraints	1	3740
# SDP Constraints	(13x13)x2, (5x5)x2	0
Sparsity factor	1	0.082
Average solver time /ms	108.3	84.3
Closed-loop cost	76.79	50.32

Simulations of the closed-loop system were performed on a laptop with a (mobile) Intel Core i7 processor and 16GB of RAM, using MATLAB R2015b running on Debian Linux. The running times and closed loop costs of the simulation are shown in Table

5.1, along with several other values of interest. Closed-loop responses resulting from the two controllers are given in Figures 5.3-5.8. It is immediately apparent that the tube MPC approach is less conservative, getting much closer to input constraint bounds.

The QP solver was not warm-started and it spent approximately 70% of the time for each solve finding an initial feasible point, so that the running time reported could likely be reduced by a factor of 3 if the solver were warm-started from the recursively feasible 'tail' of the preceding solution. Even without warm-starting, it is notable that the tube MPC controller achieves a 34% reduction in closed-loop cost while requiring 22% less running time than the controller of [KBM94], despite the larger number of variables and constraints in the QP.

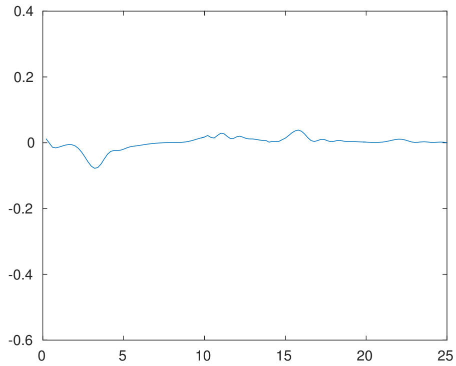

Figure 5.3: Input using [KBM94]

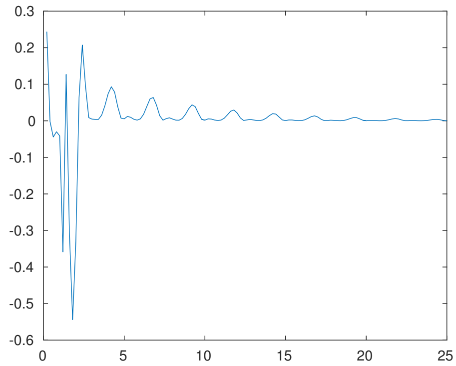

Figure 5.4: Input, tube MPC

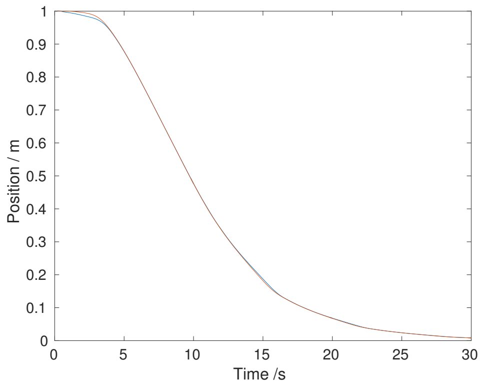

Figure 5.5: Position using [KBM94]

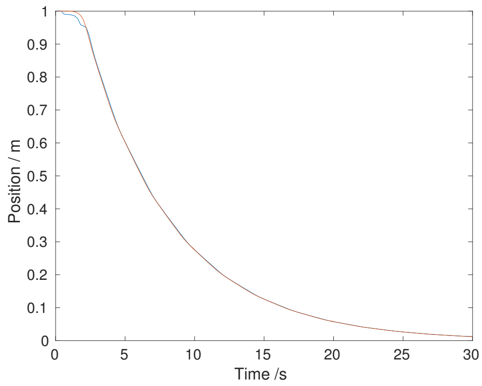

Figure 5.6: Position, tube MPC

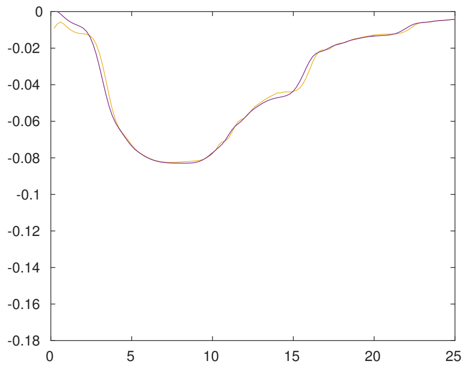

Figure 5.7: Velocity using [KBM94]

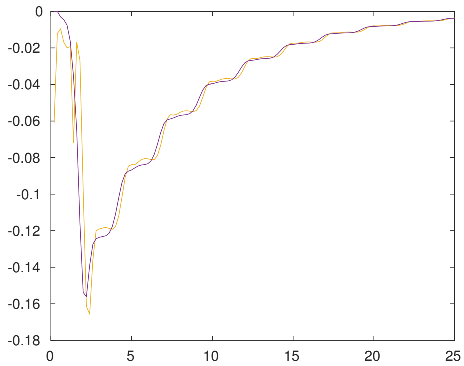

Figure 5.8: Velocity, tube MPC

5.A Proofs

Proposition 5.4

Multiplying (5.8) on the right and left by $\zeta^k = [(x^k)^T \;\; (\underline{c}^k)^T \;\; 1]^T$ and its transpose gives

$$(\zeta^k)^T W \zeta^k - (\zeta^k)^T \Psi(\theta)^T W \Psi(\theta) \zeta^k \geq (\zeta^k)^T \Lambda(\theta)^T \Lambda(\theta) \zeta^k - \gamma^2 (\zeta^k)^T v(\theta) v(\theta)^T \zeta^k$$

for all $\theta \in \Theta$. Because θ^k lies in the uncertainty set Θ we can invoke the definitions of y^k and v^k to find

$$(\zeta^k)^T W \zeta^k - (\zeta^{k+1})^T W \zeta^{k+1} \geq (y^k)^T y^k - \gamma^2 (w^k)^T w^k$$

and summing over $k = t, t+1, \ldots, \infty$ gives, after replacing $\zeta^t = \zeta_t$:

$$\zeta_t^T W \zeta_t - \lim_{k \to \infty} (\zeta^k)^T W \zeta^k \geq \sum_{k=t}^{\infty} \|y^k\|^2 - \gamma^2 \sum_{k=t}^{\infty} \|w^k\|^2$$

Since the limit must be nonnegative and the right-hand side is bounded below if the l_2 norm of w^t is finite, the limit must be finite. As it is nonnegative, we can write

$$\zeta_t^T W \zeta_t \geq \sum_{k=t}^{\infty} \|y^k\|^2 - \gamma^2 \sum_{k=t}^{\infty} \|w^k\|^2$$

from which the bound (5.9) follows after a rearrangement of the terms.

Proposition 5.6

Multiplying (5.13) on the right and left by $\zeta^k = [(x^k)^T - (s^k)^T \;\; (\underline{c}^k)^T \;\; 1]^T$ and its transpose gives

$$(\phi^k)^T X \phi^k - (\phi^k)^T \Psi(\theta)^T X \Psi(\theta) \phi^k \geq (\phi^k)^T \Lambda(\theta)^T \Lambda(\theta) \phi^k$$

for all $\theta \in \Theta$. Because $\phi^k \in \Theta$ we can write, using the definition of y^k and ϕ^{k+1}

$$(\phi^k)^T X \phi^k - (\phi^{k+1})^T X \phi^{k+1} \geq \|C(\theta^k)(x^k - s^k) + D(\theta^k)(u^k - K s^k)\|^2$$

and summing over $k = t, t+1, \ldots, \infty$ and putting $\phi^t = \phi_t$ gives

$$\phi_t^T X \phi_t - \lim_{k\to\infty} (\phi^k)^T W \phi^k \geq \sum_{k=t}^{\infty} \|C(\theta^k)(x^k - s^k) + D(\theta^k)(u^k - Ks^k)\|^2$$

and as the limit is nonnegative, we can write

$$\phi_t^T W \phi_t \geq \sum_{k=t}^{\infty} \|C(\theta^k)(x^k - s^k) + D(\theta^k)(u^k - Ks^k)\|^2$$

which is the claimed bound (5.14).

Proposition 5.7

By taking the preimage of both sides, (5.18) is equivalent to

$$\mathcal{P}[V, g_k(\alpha)] \subseteq \Phi_k^{-1}\left(\mathcal{P}[V, g_{k+1}(\alpha)]\right)$$

which holds if and only if:

$$\mathcal{P}[V, g_k(\alpha)] \subseteq \mathcal{R}_\Theta[V\tilde{A}, g_{k+1}(\alpha) - V(Bc_k + w_k)]$$

The result follows by application of Lemma 4.3.

Proposition 5.9

Follows immediately from applying Lemma 4.3 to (5.20).

Proposition 5.10

By taking the preimage of both sides, (5.24) is equivalent to

$$\mathcal{P}[V, g_N(\alpha_N)] \subseteq \Phi_k^{-1}\left(\mathcal{P}[V, g_N(\alpha_N)]\right)$$

which holds if and only if

$$\mathcal{P}[V, g_N(\alpha)] \subseteq \mathcal{R}_\Theta[V\tilde{A}, g_N(\alpha) - Vw_k]$$

The result follows by application of Lemma 4.3.

Theorem 5.12

The initial condition (5.26a) is equivalent to $x_0 \in \mathcal{P}[V, g_0(\alpha_0)]$. By Proposition 5.7, the inclusion conditions (5.26d) imply the subset condition (5.18) for all $k = 0, \dots, N$ which gives $x_k \in \mathcal{P}[V, g_k(\alpha_k)]$ for all $k = 0, \dots, N$ by induction. Also (5.26b) and (5.26c) imply satisfaction of the constraints (5.3) and (5.24) by Propositions 5.9 and 5.10.

Theorem 5.14

From Definition 5.4, recursive feasibility is equivalent to requiring $\Phi(\mathcal{X}, c_t^t) \subseteq \mathcal{X}$ for any $(\underline{c}_t, \underline{\alpha}_t) \in \mathcal{T}(x_t)$. This invariance condition will be satisfied if it is possible to find $(\underline{c}_{t+1}, \underline{\alpha}_{t+1}) \in \mathcal{T}(x_{t+1})$, so considering (5.26) we require

$$V x_{t+1}^{t+1} \le g_{t+1}^{t+1}(\alpha_{t+1}^{t+1}) \tag{5.28a}$$

$$\Delta(\theta) g_{t+1}^k(\alpha_{t+1}^k) + G(\theta) c_{t+1}^k \le 1, \quad k = t+1, \dots, t+N \tag{5.28b}$$

$$\Delta(\theta) g_{t+1}^{t+N+1}(\alpha_{t+1}^{t+N+1}) \le 1 \tag{5.28c}$$

$$\Gamma(\theta) g_{t+1}^k(\alpha_{t+1}^k) + VB(\theta) c_{t+1}^k + Vw(\theta) \le g^{k+1}(\alpha_{t+1}^{k+1}), \quad k = t+1, \dots, t+N \tag{5.28d}$$

$$\Gamma(\theta) g_{t+1}^{t+N+1}(\alpha_{t+1}^{t+N+1}) + Vw(\theta) \le g_{t+1}^{t+N+1}(\alpha_{t+1}^{t+N+1}) \tag{5.28e}$$

for any $x_{t+1} \in \Phi(x_t, c_t^t)$. We may choose $c_{t+1}^k = c_t^k$ for $k = t+1, \dots, N-2$ and $c_{t+1}^{N-1} = 0$, and there exist α_{t+1}^k satisfying constraints (5.28b-e) because g_t^k is t+1 reproducible. It only remains to verify (5.28a), but again this is trivial by backwards reproducibility since $g_{t+1}^{t+1}(\alpha_{t+1}^{t+1}) = g_t^{t+1}(\alpha_t^{t+1})$ and $\Phi_k(x_0, c_0) \subseteq \mathcal{P}[V, g_1(\alpha)]$ from Theorem 5.12.

Theorem 5.15

This follows immediately by substitution of $g_t^{k+1} = g_{t+1}^k$ resulting from the t-invariance condition into $g_{t+1}^k(\alpha_{t+1}^k) = g_t^k(\alpha_t^k)$ from Definition 5.5.

Theorem 5.16

From the bound of Proposition 5.4 we can write:

$$\zeta_t^T W \zeta_t - \zeta_t^T \Psi(\theta)^T W \Psi(\theta)\zeta_t \geq \|\Lambda(\theta)\zeta_t\|^2 - \gamma^2 \|w(\theta)\|^2, \quad \forall \theta \in \Theta$$

By recursive feasibility of $\mathcal{T}$, $\Psi(\theta)\zeta_t$ is feasible at time $t+1$ for any θ. Hence after optimising at $t+1$ we will have $\zeta_{t+1}^T W \zeta_{t+1} \leq \zeta_t^T \Psi(\theta_t)^T W \Psi(\theta_t)\zeta_t$ and so:

$$\zeta_k^T W \zeta_k - \zeta_{k+1}^T W \zeta_{k+1} \geq \|y_k\|^2 - \gamma^2 \|w_k\|^2$$

Summing over the dummy variable $k = 0, \ldots, t$ and rearranging gives

$$\zeta_0^T W \zeta_0 - \zeta_t^T W \zeta_t + \gamma^2 \sum_{k=0}^{t} \|w_k\|^2 \geq \sum_{k=0}^{t} \|y_k\|^2$$

which implies the bound claimed in the theorem as the second term is nonpositive.

Theorem 5.17

From (5.13) we have, by pre- and post- multiplying by ϕ_t

$$(\phi^{t+1})^T X \phi^{t+1} - \phi_t^T X \phi_t \leq -\|\Lambda(\theta^t)\phi_t\|^2, \quad \forall \theta^t \in \Theta \tag{5.29}$$

where $\phi^{t+1} = \Psi(\theta^t)\phi^t$ is the predicted value of ϕ at time $t+1$. By similarly post- and pre- multiplying by ϕ^k and its transpose for $k = t, \ldots, t+n-1$ and summing the results we find:

$$(\phi^{t+n})^T X \phi^{t+n} - \phi_t^T X \phi_t \leq -\sum_{k=t}^{t+n-1} \|\Lambda(\theta^k)\phi^k\|^2, \quad \forall \theta^k \in \Theta, k = t, \ldots, t+n-1$$

The observability condition of Assumption 5.1 implies that the right-hand side is a positive definite function of ϕ_t, hence we can find some $c_1, c_2 > 0$ such that $c_1\|\phi_t\|^2 \leq \phi_t^T X \phi_t \leq c_2\|\phi_t\|^2$ and $\sum_{k=t}^{t+n-1} \|\Lambda(\theta^k)\phi^k\|^2 \geq c_1\|\phi_t\|^2$ for all θ^k, $k = t, \ldots, t+n-1$.

Hence from the above we have

$$(\phi^{t+n})^T X \phi^{t+n} - \phi_t^T X \phi_t \leq -\frac{c_1}{c_2} \phi_t^T X \phi_t$$

and from recursive feasibility we know that $(\phi^{t+n})^T X \phi^{t+n} \geq \phi_{t+n}^T X \phi_{t+n}$, so after a rearrangement we find

$$\phi_{t+n}^T X \phi_{t+n} \leq \left(1 - \frac{c_1}{c_2}\right) \phi_t^T X \phi_t$$

and in particular this implies

$$\phi_t^T X \phi_t \leq \left(1 - \frac{c_1}{c_2}\right)^{\frac{t}{n}} \phi_0^T X \phi_0$$

for $t = 0, n, 2n, \ldots$. Moreover from (5.29) we have $\phi_{t+1}^T X \phi_{t+1} \leq \phi_t^T X \phi_t$ and so we can write

$$\phi_t^T X \phi_t \leq \delta^{\frac{t}{n}-1} \phi_0^T X \phi_0$$

for all $t > 0$, where $\delta = 1 - \frac{c_1}{c_2} < 1$, as claimed.

6 Tube MPC of LPV-A systems

6.1 Introduction

A difficulty with robust model predictive control in the presence of uncertainty is that the control policy must have a feedback component for efficient disturbance rejection. This stands in stark contrast to deterministic MPC, where it is sufficient to optimise over open-loop control sequences. Computing optimal feedback policies under constraints is a difficult problem in general, so usually the policies considered are restricted to some simple class such as perturbations on a linear state feedback, as in [KRS00], [MSR05] and the tube MPC of the previous chapter, or affine functions of past additive disturbances [Gou07].

In this chapter, we extend the tube MPC to cover the class of linear parameter-varying (LPV) systems where the uncertain parameters do not appear in the input mapping (termed LPV-A systems in [BLM12]), allowing optimisation over a class of control policies that are affine functions of the uncertain parameter at each point in time. When dealing with such systems, it is a common assumption that the current value of some uncertain parameter is available to the controller at any particular time, and as such it seems desirable to formulate controllers that exploit this information in their prediction structure.

MPC schemes that find parameter-dependent controls for LPV systems are not novel, but they generally require either conservative ellipsoidal bounding of the predicted state [LA00, WSS04], or are intractable for long horizons due to exponential growth in the number of constraints required [CFF02, PRSDM05c]. As in the case of LDI systems in the previous chapter, we will see that it is possible to strike a more favourable trade-off between conservatism and complexity using polyhedral tubes. We make a couple of important changes to exploit knowledge of the parameter θ_t:

- We restrict the feedback policy used for the predicted input to be of the form

$u^k = K(\theta^k)x^k + c(\theta^k)$ where $K(\theta)$ is a gain-scheduled but predetermined state feedback and $c(\theta)$, which is chosen online, is an affine function of θ;

- The cost function used online is a quadratic upper bound on the minimax cost that takes into account the dependence of $K(\theta)$ and $c(\theta)$ on θ;

As we will also assume that the uncertainty in the system is polytopic, we will formulate the online optimisation as a quadratic program that can readily be handled using standard solvers. Moreover, the number of variables and constraints in the online optimisation scale linearly with increasing state dimension and prediction horizon length, as when using linear MPC.

6.2 Problem formulation

We consider the Linear Parameter Varying (LPV-A) system with dynamics

$$x_{t+1} = A(\theta_t)x_t + Bu_t \tag{6.1a}$$

$$y_t = C(\theta_t)x_t + Du_t \tag{6.1b}$$

$$z_t = F(\theta_t)x_t + Gu_t \tag{6.1c}$$

where the matrices $A(\theta_t), F(\theta_t)$ are affine in the parameter θ_t which, in contrast to the system considered in Chapter 5, is available to the controller at time t. As before the parameter is contained in a compact set, but we now assume that the set in question is the unit simplex $\mathcal{S}^{p-1} \subset \mathbb{R}^p$ and that $(A(\theta), B(\theta)) = \sum_i (A^{(i)}, B^{(i)})\theta_i$ so that all the state space matrices lie within a known convex hull. This is not strictly necessary, and we could consider more general uncertainties as in the previous chapter, but polytopic uncertainty is so common in the literature when considering LPV systems that we instead choose to concentrate on this special case.

Assumption 6.1. *There exists some $\theta \in \mathcal{S}^{p-1}$, such that the pair $(C(\theta), A(\theta))$ is observable. The matrix D^TD is positive definite.*

We will once again require this observability condition for stability, as we will seek to minimise the size of the output y_t in the sense of l_2-norm while robustly satisfying a constraint on the output z_t:

$$z_t = F(\theta_t)x_t + Gu_t \leq 1 \tag{6.2}$$

Similarly to the approach we followed in developing the tube MPC for LDIs in the preceding chapter, we choose to optimise a sequence of perturbations on a fixed feedback law, so that the predicted input is of the form

$$u^k = K(\theta^k)x^k + c^k(\theta^k) \tag{6.3}$$

where as before the superscript notation reflects our notion of a predicted, rather than actual, quantity. The matrix $K(\theta^k)$ defines a gain-scheduled state feedback chosen offline, and $c(\theta^k)$ is an affine function of θ^k that is chosen online by the MPC controller. Hence we parametrise $c^k(\theta^k)$ as, say

$$c^k(\theta^k) = \sum_{i=1}^{p} c^{(k,i)}\theta_i^k \tag{6.4}$$

and consider the $c^{(k,i)}$ as free variables in an optimisation problem at each prediction time $k \geq t$. Of course, there is no point doing this for the current time, where the parameter θ_t is already known, so we could also consider c^t as an optimisation variable in its own right without requiring an affine dependence on θ. For notational consistency however we will assume that we use this affine dependence at the current time also, with the implicit assumption that we can make use of the currently known value of θ_t where necessary.

This scheduled control perturbation will give the controller some measure of 'feedback' in the prediction structure to changing values of θ_t. Of course, future values of θ_t are not known at a particular time, but a suitable interpretation of this control parameterisation is that the controller is 'planning' its control in the knowledge that it the parameter θ_t will be known once time t is reached. This has some similarities to the disturbance-affine MPC formulations of [GKM06] and the associated thesis [Gou07] in that the predictions are dependent on quantities that are currently unknown.

We assume that $K(\theta)$ is quadratically stabilising, in order to develop quadratic bounds on the cost function later.

Assumption 6.2. *The state dynamics (6.1) are quadratically stabilised by the state feedback* $u_t = K(\theta)x_t$*.*

Remark 6.1. The design of a gain-scheduled feedback can usually be accomplished by solving optimisation problems involving linear matrix inequalities; a good overview is available in the papers [AGB95] and [AG95]. For our present purposes, an acceptable state feedback controller $K(\theta)$ can be found by maximising trace(S), subject to $S \succ 0$ and

$$\begin{bmatrix} S & SA_i^T + Y_i^T B^T & SC_i^T & Y_i^T D^T \\ A_i S + BY_i & C_i S & 0 & 0 \\ C_i S & 0 & I & 0 \\ DY_i & 0 & 0 & I \end{bmatrix} \succeq 0, \quad \forall i = 1, \ldots, p \tag{6.5}$$

then setting $K_i = Y_i S^{-1}$ and taking the state feedback law as $K(\theta) = \sum_i K_i \theta_i$. This gives a quadratically stabilising gain-scheduled state feedback.

This choice of input parameterisation naturally leads to predicted dynamics

$$x^{k+1} = \tilde{A}(\theta^k)x^k + Bc^k(\theta^k) \tag{6.6a}$$

$$y^k = \tilde{C}(\theta^k)x_t + Dc^k(\theta^k) \tag{6.6b}$$

$$z^k = \tilde{F}(\theta^k)x_t + Gc^k(\theta^k) \tag{6.6c}$$

where we have $\tilde{A}(\theta^k) = A(\theta^k) + BK(\theta^k)$, $\tilde{F}(\theta^k) = F(\theta^k) + GK(\theta^k)$, and where the unknown parameter is contained in the simplex $\theta^k \in \mathcal{S}^{p-1}$. As far as the predicted constraints are concerned, it would be fairly simple to instead consider a sequence of increasing uncertainty sets that vary with prediction time. This could give better constraint handling in the common case that a bound on the maximum rate of variation of the uncertain parameter is known, a common assumption when using LPV systems. For simplicity though, we restrict ourselves to the more general case that θ_t can vary arbitrarily quickly within the simplex.

6.3 Cost function evaluation

As in the previous chapter we will now use an autonomous (or 'lifted') form of the prediction dynamics of the system in order to develop a quadratic bound on the l_2-norm of the predicted output y^k. This autonomous formulation will be different from the one used in the LDI case due to the dependence of the control perturbation $c^k(\theta^k)$ on θ^k.

Because we have not included an additive disturbance in the dynamics, there will be no distinction, as in the last chapter, between developing a cost that bounds the l_2-norm of the output signal and one that penalises the deviations from the minimal invariant set. This is because the minimal invariant set of the LPV system described will be the singleton containing the origin. It is a fairly straightforward exercise to extend the results we will develop in this chapter to that case also, but in the interest of repeating ourselves as little as possible, we do not bother to do so here.

6.3.1 Bounding the output l_2-norm

Consider the concatenation of the (predicted) free control variables $c^{(k,i)}$ into a vector d^k as

$$d^k = \begin{bmatrix} c^{(k,1)} \\ c^{(k,2)} \\ \cdots \\ c^{(k,q)} \end{bmatrix}$$

so that d^k defines the entire gain-scheduled control perturbation at prediction time k. Considering vectors d_t^k with their natural interpretation as the predicted d^k at time t, we introduce $\underline{d}_t$, and a 'tail' $\tilde{d}_{t+1}$ as

$$\underline{d}_t = \begin{bmatrix} d_t^t \\ d_t^{t+1} \\ \vdots \\ d_t^{t+N-1} \end{bmatrix}, \quad \tilde{d}_{t+1} = \begin{bmatrix} d_t^{t+1} \\ \vdots \\ d_t^{t+N-1} \\ 0 \end{bmatrix}$$

where there is a near-perfect analogy with the quantities $\underline{c}_t$ and $\tilde{c}_{t+1}$ of the LDI systems of the last chapter except that we are now considering the whole gamut of controls corresponding to vertices of the uncertainty simplex.

As in the LDI uncertainty case, we can define a shift matrix T such that $\tilde{\underline{d}}_{t+1} = T\underline{d}_t$. The definition of a quantity analogous to S is somewhat more complex due to the gain-scheduling. Consider $S^{(1)} = [I \ \ 0\ldots]$, $S^{(2)} = [0 \ \ I\ldots]$ etc. up to $S^{(q)}$, so that the matrix $S^{(i)}$ has the identity in its ith block, and let:

$$S(\theta) = \sum_{i=1}^{q} \theta_i S^{(i)}$$

Conveniently, we now have $c_t(\theta_t) = S(\theta_t)\underline{d}_t$ and similarly for the predicted quantities, so that the predicted state dynamics of the system (6.1) under the scheduled feedback law can be stated autonomously as:

$$\begin{bmatrix} x^{k+1} \\ \underline{d}^{k+1} \end{bmatrix} = \begin{bmatrix} \tilde{A}(\theta^k) & BS(\theta^k) \\ 0 & T \end{bmatrix} \begin{bmatrix} x^k \\ \underline{d}^k \end{bmatrix} \tag{6.7a}$$

$$y^k = \begin{bmatrix} \tilde{C}(\theta^k) & D \end{bmatrix} \begin{bmatrix} x^k \\ \underline{d}^k \end{bmatrix} \tag{6.7b}$$

This can be written more simply as $\eta^{k+1} = \Psi(\theta^k)\eta^k$ and $y^k = \Lambda(\theta^k)\eta^k$ where the augmented state vector η^k and matrices $\Psi(\theta)$, $\Lambda(\theta)$ are defined by comparison with (6.7). For convenience, we also introduce the matrices corresponding to the vertices of the uncertainty as $\Psi^{(i)}$, $\Lambda^{(i)}$ etc. A quadratic bound on output l_2 norm can now be established using Lyapunov-type matrix inequalities, as in the following proposition.

Proposition 6.2. *If there exists $W \succ 0$ satisfying,*

$$W - (\Psi^{(i)})^T W \Psi^{(i)} \succeq (\Lambda^{(i)})^T \Lambda^{(i)}, \quad \forall i = 1, \ldots, p \tag{6.8}$$

the predicted l_2 norm of the output can be bounded as:

$$\sum_{k=t}^{\infty} \|y^k\|^2 \leq \eta_t^T \Lambda(\theta_t)^T \Lambda(\theta_t)\eta_t + \eta_t^T \Psi(\theta_t)^T W \Psi(\theta_t)\eta_t \tag{6.9}$$

It is worth noting that (6.9) exploits the knowledge of the current scheduling parameter θ_t. To solve the linear matrix inequality (6.8), a good strategy (in the sense that it yields a tight bound) is to minimise trace(W) subject to (6.8) and $W \succ 0$. This is readily reformulated into a standard form semidefinite programming problem, a process which can be automated using Yalmip (or an equivalent modelling language) and carried out as part of the design of the MPC controller.

Remark 6.3. The autonomous formulation given here is novel and has uses beyond simply providing a cost bound for our tube MPC. The procedure of forming a prediction tube that we carry out in the next section can be seen as a method of forming an efficient representation of some approximation to the maximal admissible set of the autonomous system (6.2). In fact, any other invariant set of that system could be used in place of a tube for constraint handling. This means that Algorithm 4.3 could be used to form a polyhedral invariant set of the system and an MPC law formed using this a constraint for the one-step ahead augmented state η^{t+1}, in the spirit of [PRSDM05c]. Or a suitable LMI condition could be formulated to give an ellipsoidal invariant set, similarly to [KRS00].

6.4 Constraint handling using tubes

We again introduce ***tube parameterisation functions*** $g_t^k(\alpha)$ that will give polyhedra containing the predicted system state as $\{x \in \mathbb{R}^n |\ Vx \leq g^k(\alpha^k)\}$. To ensure a well-behaved tube formulation. we make the standard assumptions:

Assumption 6.3. *The functions g_t^k are affine in α.*

Assumption 6.4. *The matrix V and the functions g_t^k are chosen so that, for all t and k and for all $x \in \mathbb{R}^n$ there exists α such that $x \in \mathcal{P}[V, g_t^k(\alpha)]$.*

In addition, we will make the following two assumptions that will ensure recursive feasibility of the tube formulation. The explanation and background to these assumptions can be found in Sections 5.4.2 and 5.4.4.

Assumption 6.5. *There exists α^N such that $\{x \in \mathbb{R}^n |\ Vx \leq g^N(\alpha^N)\}$ is positively invariant under the dynamics (6.6a) with $c^k = 0$.*

Assumption 6.6. *The tube parameterisation is t-invariant, that is we have $g_t^k = g_0^k$ for all $k \geq 0$ and all $t \geq 0$.*

Assumption 6.7. *The tube parameterisation is $k-1$-reproducible, that is for any α^k there exists α^{k-1} such that $g^{k-1}(\alpha^{k-1}) = g^k(\alpha^k)$ for all $k = t+1, \ldots, N$.*

We will proceed, in concept, similarly to the previous chapter where LDI systems were considered. Hence we will develop constraints on the tube parameters $\underline{\alpha}_t$ that ensure, for predicted dynamics denoted by a suitable set-valued function $\Phi(\cdot, d^k)$ analogous to (5.17), inclusion conditions

$$\Phi\left(\mathcal{P}[V, g^k(\alpha^k)], d^k\right) \subseteq \mathcal{P}[V, g^{k+1}(\alpha^{k+1})] \tag{6.10}$$

as well as a terminal invariance condition

$$\Phi\left(\mathcal{P}[V, g^N(\alpha^N)], 0\right) \subseteq \mathcal{P}[V, g^N(\alpha^N)] \tag{6.11}$$

and conditions to ensure feasibility of the predicted state:

$$\mathcal{P}[V, g^k(\alpha^k)] \subseteq \mathcal{R}_\Theta[\tilde{F}, 1 - Gc^k] \tag{6.12}$$

As for LDI systems, we will see that the $k-1$-reproducibility condition on the tube parameterisation is enough to ensure recursive feasibility along with the terminal invariance condition (6.11).

6.4.1 Tube constraints

Taking the preimage of both sides of equation (6.10) gives the robust inclusion

$$\mathcal{P}[V, g^k(\alpha^k)] \subseteq \mathcal{R}_\Theta[V\tilde{A}, g^{k+1}(\alpha^{k+1}) - Bc^k]$$

and doing likewise with (6.11) yields:

$$\mathcal{P}[V, g^k(\alpha^k)] \subseteq \mathcal{R}_\Theta[V\tilde{A}, g^{k+1}(\alpha^{k+1}) - Bc^k]$$

Using the fact that $\theta \in \mathcal{S}^{p-1}$, it is straightforward to rewrite these robust set inclusions as a set of polyhedral inclusions. Hence we find that (6.10) becomes

$$\mathcal{P}[V, g^k(\alpha^k)] \subseteq \mathcal{P}[V\tilde{A}^{(i)}, g^{k+1}(\alpha^{k+1}) - Bc^{(k,i)}], \quad i = 1, \ldots, p \tag{6.13}$$

and (6.11) becomes

$$\mathcal{P}[V, g^k(\alpha^k)] \subseteq \mathcal{P}[V\tilde{A}^{(i)}, g^{k+1}(\alpha^{k+1}) - Bc^{(k,i)}], \quad i = 1, \ldots, p \tag{6.14}$$

and finally (6.12) becomes:

$$\mathcal{P}[V, g^k(\alpha^k)] \subseteq \mathcal{P}[\tilde{F}^{(i)}, 1 - Gc^{(k,i)}], \quad i = 1, \ldots, p \tag{6.15}$$

These conditions can be converted to constraints on the parameters α using Lemma 4.2. This naturally leads to the propositions:

Proposition 6.4. *If there exist matrices $\Gamma^{(i)} \geq 0$ such that*

$$\Gamma^{(i)} V = V\tilde{A}^{(i)} \tag{6.16a}$$

$$\Gamma^{(i)} g^k(\alpha^k) + VBc^{(k,i)} \leq g^{k+1}(\alpha^{k+1}) \tag{6.16b}$$

for all $i = 1, \ldots, p$, then (6.10) holds.

Proposition 6.5. *If there exist matrices $\Gamma^{(i)} \geq 0$ such that*

$$\Gamma^{(i)} V = V\tilde{A}^{(i)} \tag{6.17a}$$

$$\Gamma^{(i)} g^N(\alpha^N) \leq g^N(\alpha^N) \tag{6.17b}$$

for all $i = 1, \ldots, p$, then (6.11) holds.

Proposition 6.6. *If there exist matrices $\Delta^{(i)} \geq 0$ such that*

$$\Delta^{(i)} V = \tilde{F}^{(i)} \tag{6.18a}$$

$$\Delta^{(i)} g^k(\alpha^k) + Gc^{(k,i)} \leq 1 \tag{6.18b}$$

for all $i = 1, \ldots, p$, then (6.12) holds.

The matrices $\Gamma^{(i)}$ and $\Delta^{(i)}$ can be chosen during the design of the MPC controller, using the algorithms in Section 4.3.5. To these constraints, we add the initial condition $Vx^t \leq g^t(\alpha^t)$ to ensure the current state is within the first tube cross-section. At the first prediction time $k = t$ the constraints of the preceding propositions can be simplified considerably due to the knowledge of the current θ_t. This leads us to define constraints for the whole tube

$$
\begin{aligned}
& Vx^t \leq g^t(\alpha^t) && && \text{(6.19a)}\\
& \Delta(\theta_t)g^t(\alpha^t) + Gc(\theta_t) \leq 1 && && \text{(6.19b)}\\
& \Delta^{(i)}g^k(\alpha^k) + Gc^{(k,i)} \leq 1, && i = 1, \ldots, p && \text{(6.19c)}\\
& \Delta^{(i)}g^{t+N}(\alpha^{t+N}) \leq 1, && i = 1, \ldots, p && \text{(6.19d)}\\
& \Gamma(\theta_t)g^t(\alpha^t) + VBc(\theta_t) \leq g^{t+1}(\alpha^{t+1}) && && \text{(6.19e)}\\
& \Gamma^{(i)}g^k(\alpha^k) + VBc^{(k,i)} \leq g^{k+1}(\alpha^{k+1}), && i = 1, \ldots, p && \text{(6.19f)}\\
& \Gamma^{(i)}g^{t+N}(\alpha^{t+N}) \leq g^{t+N}(\alpha^{t+N}), && i = 1, \ldots, p && \text{(6.19g)}
\end{aligned}
$$

where (6.19b) and (6.19e) depend on the (known) value of θ_t in order to exploit the knowledge of that parameter.

We introduce the notation $(\underline{d}_t, \underline{\alpha}_t) \in \mathcal{T}(x_t, \theta_t)$ to mean that the free variables $\underline{d}_t$ and $\underline{\alpha}_t$ satisfy (6.19) for a given initial condition x_t and current system parameter θ_t. That is we have

$$
\mathcal{T}(x_t, \theta_t) = \{(\underline{d}, \underline{\alpha}) |\ \underline{d} \text{ and } \underline{\alpha} \text{ satisfy (5.26) with } x^t = x_t\} \tag{6.20}
$$

in analogy with Definition 5.2. The assumptions we made on the reproducibility properties of the tube parameterisation (i.e. Assumptions 6.5–6.7) immediately lead us to a recursive feasibility result for these constraints.

Theorem 6.7. *The tube constraint set $\mathcal{T}$ as defined by (6.20) is strongly recursively feasible.*

6.5 Algorithm and stability

We now present an MPC algorithm that minimises the cost bound of (6.2) while ensuring constraint satisfaction using the prediction tube. We remind the reader of the definition of η_t, as it appears in the statement of the MPC optimisation:

$$\eta_t = \begin{bmatrix} x^k \\ \underline{d}^k \end{bmatrix}$$

There are no novelties in the statement of the algorithm, which appears as Algorithm 6.2; all the relevant theory has been covered in the preceding parts of the chapter. The offline design procedure is summarised in Algorithm 6.1.

Algorithm 6.1 Offline, LPV Robust Tube MPC

1. Find a stabilising $K(\theta)$ by maximising trace(S), subject to $S \succ 0$ and 6.5.
2. Find V such that $\mathcal{P}[V, \bar{g}]$ is robustly invariant for some $\bar{g}$, using Algorithm 4.3.
3. Choose an affine function $g(\alpha)$ to parameterise a prediction tube, for example by using a formulation from Section 4.4.2.
4. Form matrices $\Gamma^{(i)}$ and $\Delta^{(i)}$ satisfying $\Gamma^{(i)}V = V\tilde{A}^{(i)}$ and $\Delta^{(i)}V = \tilde{F}^{(i)}$ by using Algorithm 4.1.
5. Find X satisfying the conditions of Proposition 5.6, by minimising trace(W) subject to (6.8) and $W \succ 0$.

Algorithm 6.2 LPV Robust Tube MPC

At each time t, solve

$$\begin{aligned} \underset{\underline{d}_t, \underline{\alpha}_t}{\text{minimise}} \quad & \eta_t^T \Lambda(\theta_t)^T \Lambda(\theta_t)\eta_t + \eta_t^T \Psi(\theta_t)^T W \Psi(\theta_t)\eta_t \\ \text{subject to} \quad & (\underline{c}_t, \underline{\alpha}_t) \in \mathcal{T}(x_t, \theta_t) \end{aligned}$$

and apply the control $u_t = K(\theta_t)x_t + c_t(\theta_t)$.

Algorithm 6.2 requires the solution of a quadratic program at each time t, so that it can readily be implemented using standard solvers. If possible, the algorithm should be warm-started using the result of the previous optimisation. This is typically possible if solving the resulting QP using an active-set method as the 'tail' $(\tilde{\underline{d}}_{t+1}, \tilde{\underline{\alpha}}_{t+1})$ of the previous solution can be used as a guess, and we know that it is feasible from Theorem 6.7. The recursive feasibility property and form of the cost function as a bound on the infinite-horizon l_2-norm of y_t also leads us directly to an exponential stability result.

Theorem 6.8. *Under the control of Algorithm 6.2 the state x_t approaches the origin* *exponentially if the optimisation is feasible initially. More precisely*

$$\eta_t^T W \eta_t \leq \delta^{\frac{t}{n}-1} \eta_0^T W \eta_0$$

for some $0 \leq \delta < 1$*, so* x_t *and* $\underline{d}_t$ *converge exponentially to zero in closed-loop.*

Remark 6.9. If there exits a θ such that $C(\theta)^T C(\theta)$ is positive definite in Assumption 6.1 (which is clearly much stronger than the observability condition, as it implies that $C(\theta)^T C(\theta)$ is $n \times n$ and full-rank) we could claim

$$\eta_t^T W \eta_t \leq \delta^{\frac{t}{n}} \eta_0^T W \eta_0$$

for the same δ. This is readily seen by considering the proof of Theorem 6.8, noting that the extra -1 in the exponent is required as it may initially be up to n time steps before the cost decreases if only observability is assumed.

6.6 Simulation example

We now consider a design and simulation example to illustrate the advantages of the approach over more conservative ellipsoidal methods of handling uncertainty. We compare the presented tube MPC with the method given in [LA00] which, similarly to the tube MPC of this chapter, both exploits knowledge of the currently known θ_t and performs an optimisation over a gain-scheduled predicted system input (albeit of the form $K(\theta)$ rather than $K(\theta) + c(\theta)$). Unlike the presented tube MPC it relies on bounding the state in an ellipsoidal admissible set. As in the simulation example of the previous chapter, we shall see that this makes the constraint handling over the prediction horizon rather conservative in comparison to using a variable tube.

For the comparison, we use the same system as that used in the example in [LA00], namely an LPV system with two uncertainty vertices for the A matrix, such that we have $A^{(1)}$, $A^{(2)}$ and B matrices and initial condition x_0

$$A^{(1)} = \begin{bmatrix} 0.2730 & 0.0660 & 0.3021 & -0.5012 \\ 0.2717 & 0.4416 & 0.5602 & -0.7123 \\ 0.3051 & -0.7865 & 0.7651 & -0.3121 \\ 0.7962 & -0.1452 & 0.5231 & -0.9345 \end{bmatrix}, \quad B = \begin{bmatrix} 0.2300 \\ 0.2601 \\ 0.1213 \\ 1.3452 \end{bmatrix}$$

$$A^{(2)} = \begin{bmatrix} 0.2093 & -0.1981 & 0.2394 & 0.5671 \\ 0.2717 & 0.4598 & 0.5602 & 1.3782 \\ -0.4700 & 0.6700 & -0.8600 & -1.2400 \\ 0.3456 & -0.6312 & -1.4594 & 1.8936 \end{bmatrix}, \quad x_0 = \begin{bmatrix} -0.3964 \\ 0.4377 \\ -1.0905 \\ 1.1137 \end{bmatrix}$$

and the uncertain matrix $A(\theta)$ is given by the expression:

$$A(\theta) = \theta A^{(1)} + (1-\theta)A^{(2)}$$

The system also has a single output given by $y_t = (x_t)_2$. That is, it is equal to the second state variable. The control objective is to minimise the worst-case value of

$$J_t = \sum_{k=t}^{\infty}(x_k^T Q x_k + u_k^T R u_k)$$

where $Q = \mathrm{diag}(1,1,1,1)$, $R = 1$, and while satisfying $|u_t| \le 2$ and $|y_t| \le 1.4$.

The LMI (6.5) was used to design a stabilising, gain-scheduled state feedback $K(\theta)$ that minimises the quadratic bound on the cost function. To construct the matrix V required for the tube parameterisation, Algorithm 4.3 was used with $\gamma = 0$, $\delta = 0$ in order to give the maximal invariant set, which has 26 facets in this case. The inner-product parameterisation (4.7) was chosen to define the parameterisation functions as $g(\alpha^k) = \alpha^k$ over the entire prediction horizon. The prediction horizon was taken as $N = 3$, which is the minimal value for which the initial condition x_0 is part of the feasible set of the tube MPC.

By Theorem 6.8 this will lead to a closed-loop system that is exponentially stable with the state converging to the origin in closed loop. An example of an optimal prediction tube is shown in Figures 6.3 and 6.4, projected onto the first and last two

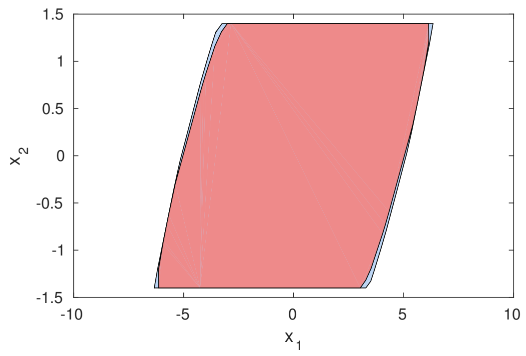

Figure 6.1: Feasible set comparison

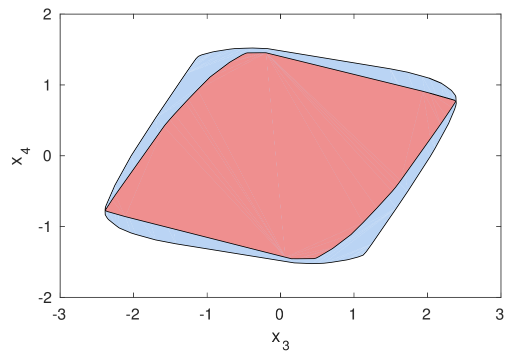

Figure 6.2: Feasible set comparison

Table 6.1: Comparison of MPC optimisations

	Method of [LA00]	Tube MPC
Optimisation type	SDP	QP
Solver	SeDuMi	Gurobi
Algorithm	Primal-dual interior point	Dual simplex
# Variables	25	111
# Linear Constraints	2	266
# SDP Constraints	(13x13)x2, (10x10), (5x5)x2	0
Sparsity factor	1	0.04
Average solver time /ms	91.8	10.2
Closed-loop cost	14.80	11.01

state variables respectively. Also shown in Figures 6.1 and 6.2 is a comparison of the feasible regions of the controller when using a gain-scheduled control perturbation $c(\theta)$ (in blue), and when gain-scheduling of this parameter is not used (in red) as in the tube MPC given in the last chapter for LDI systems. It is clear that introducing an affine dependence of the control on the scheduling parameter improves the feasible region.

Simulations of the closed-loop system were performed on a laptop with a mobile Intel Core i7 processor and 16GB of RAM, using MATLAB R2015b running on Debian Linux. The modelling language Yalmip was used with the solvers SeDuMi and Gurobi to solve the generated convex optimisation problems. The running times, closed-loop costs, number of constraints and variables, and sparsity factors of the two formulations are shown in Table 6.1. To give a fair comparison with [LA00], we choose θ to have the same time-variation as in that paper, that is a sigmoid function centered at $t = 10$:

$$\theta = 1 - \frac{1}{1 - e^{(t-10)}}$$

The closed-loop responses can be seen in Figures 6.5-6.10. The tube MPC approach is visibly less conservative, getting closer to the input constraint bound and having smaller peak values of input and output, as well as a closed-loop cost that is 25% lower. Also, the computation time is reduced by almost an order of magnitude, being a factor of 9 lesser when using the tube MPC.

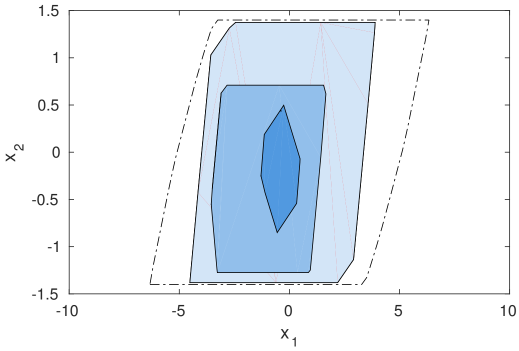

Figure 6.3: Optimal tube at $t = 2$

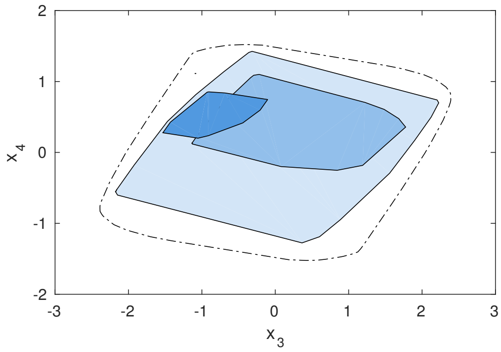

Figure 6.4: Optimal tube at $t = 2$

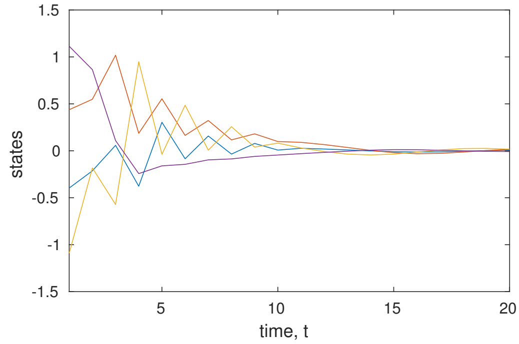

Figure 6.5: States using [LA00]

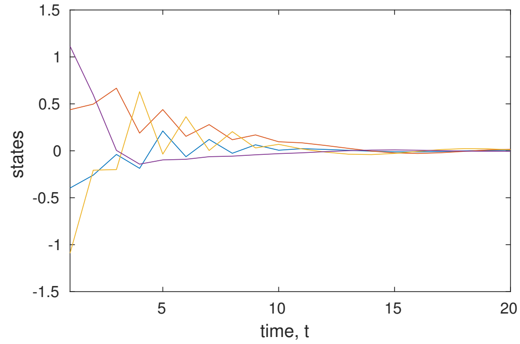

Figure 6.6: States, tube MPC

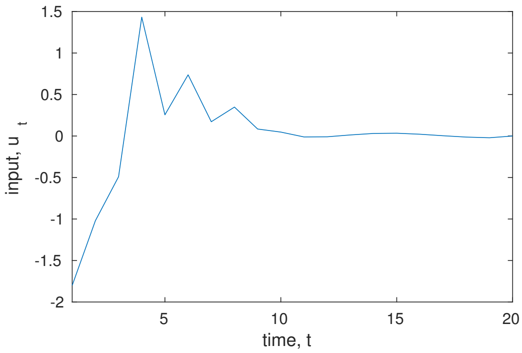

Figure 6.7: Input using [LA00]

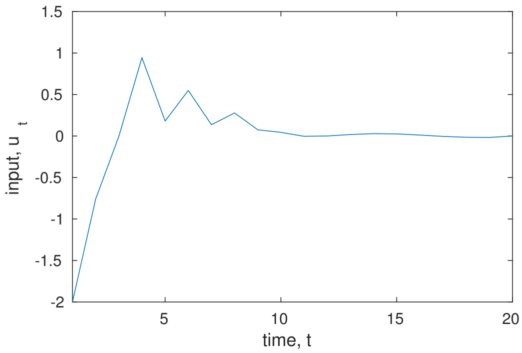

Figure 6.8: Input, tube MPC

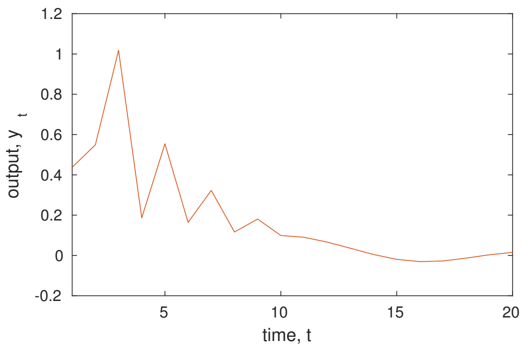

Figure 6.9: Output using [LA00]

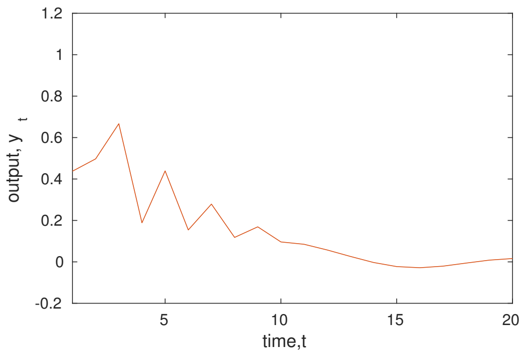

Figure 6.10: Output, tube MPC

6.A Proofs

Proposition 6.2

Multiplying (6.8) on the left and right by $\eta^k = [(x^k)^T \ (\underline{c}^k)^T]$ and its transpose gives

$$(\eta^k)^T W \eta^k - (\eta^k)^T \Psi(\theta)^T W \Psi(\theta) \eta^k \geq (\eta^k)^T \Lambda(\theta)^T \Lambda(\theta) \eta^k$$

for all $\theta \in \Theta$. Because θ^k lies in this uncertainty set we obtain, after invoking the definitions of y^k and η^{k+1}

$$(\eta^k)^T W \eta^k - (\eta^{k+1})^T W \eta^{k+1} \geq (y^k)^T y^k$$

and summing over $k = t+1, \ldots, \infty$ gives:

$$(\eta^{t+1})^T W \eta^{t+1} - \lim_{k \to \infty} (\eta^k)^T W \eta^k \geq \sum_{k=t+1}^{\infty} \|y^k\|^2$$

As the limit is nonnegative, we discover the bound:

$$(\eta^{t+1})^T W \eta^{t+1} \geq \sum_{k=t+1}^{\infty} \|y^k\|^2$$

Now, we need only note that $\eta^{t+1} = \Psi(\theta_t)\eta_t$ and $y_t = \Lambda(\theta_t)\eta_t$ to conclude that

$$\eta_t^T \Lambda(\theta_t)^T \Lambda(\theta_t) \eta_t + \eta_t^T \Psi(\theta_t)^T W \Psi(\theta_t) \eta_t \geq \sum_{k=t}^{\infty} \|y^k\|^2$$

which is precisely the bound that was claimed in the proposition.

Proposition 6.4

Follows immediately by applying Lemma 4.2 to (6.13) for each i.

Proposition 6.5

Follows immediately by applying Lemma 4.2 to (6.14) for each i.

Proposition 6.6

Follows immediately by applying Lemma 4.2 to (6.15) for each i.

Theorem 6.7

From Definition 5.4, recursive feasibility is equivalent to $\Phi(\mathcal{X}, c_t^t(\theta_t)) \subseteq \mathcal{X}$ for any $(\underline{d}_t, \underline{\alpha}_t) \in \mathcal{T}(x_t, \theta_t)$. This condition will be satisfied if we can find $(\underline{d}_{t+1}, \underline{\alpha}_{t+1}) \in \mathcal{T}(x_{t+1}, \theta_{t+1})$, that is if we can satisfy the constraints

$$Vx^{t+1} \leq g^{t+1}(\alpha^{t+1}) \tag{6.21a}$$

$$\Delta(\theta_{t+1})g^{t+1}(\alpha^{t+1}) + Gc(\theta_{t+1}) \leq 1 \tag{6.21b}$$

$$\Delta^{(i)}g^k(\alpha^k) + Gc^{(k,i)} \leq 1, \qquad i = 1, \ldots, p \tag{6.21c}$$

$$\Delta^{(i)}g^{t+N+1}(\alpha^N) \leq 1, \qquad i = 1, \ldots, p \tag{6.21d}$$

$$\Gamma(\theta_{t+1})g^{t+1}(\alpha^{t+1}) + VBc(\theta_{t+1}) \leq g^{t+2}(\alpha^{t+2}) \tag{6.21e}$$

$$\Gamma^{(i)}g^k(\alpha^k) + VBc^{(k,i)} \leq g^{k+1}(\alpha^{k+1}), \qquad i = 1, \ldots, p \tag{6.21f}$$

$$\Gamma^{(i)}g^{t+N+1}(\alpha^{t+N+1}) \leq g^{t+N+1}(\alpha^{t+N+1}), \qquad i = 1, \ldots, p \tag{6.21g}$$

for any $x_{t+1} \in \Phi_k(x_t, c_t^t(\theta_t)$. If we choose $d_{t+1}^k = d_t^k$ for $k = t+1, \ldots, N-2$ and $d_{t+1}^{N-1} = 0$, then there exist α_{t+1}^k satisfying (6.21b-6.21g) trivially by Assumptions 6.6 and 6.7, noting that if $\Delta^{(i)}g^{t+1}(\alpha^{t+1}) + Gc^{(t+1,i)}$ holds for all i, then it must hold for any particular $\theta_{t+1} \in \mathcal{S}^{p-1}$. Similarly, there exists α^{t+1} such that (6.21a) holds from the same assumptions and the fact that $x_{t+1} \in \Phi_k(x_t, c_t^t(\theta_t))$.

Theorem 6.8

From (6.8) we have, by pre- and post- multiplying by η_t

$$(\eta^{t+1})^T W \eta^{t+1} - \eta_t^T W \eta_t \leq -\|\Lambda(\theta^t)\eta_t\|^2, \quad \forall \theta^t \in \Theta \tag{6.22}$$

where $\eta^{t+1} = \Psi(\theta^t)\eta^t$. By similarly post- and pre- multiplying by η^k and its transpose for $k = t, \dots, t+n-1$ and summing the results we find:

$$(\eta^{t+n})^T W \eta^{t+n} - \eta_t^T W \eta_t \leq - \sum_{k=t}^{t+n-1} \|\Lambda(\theta^k)\eta^k\|^2, \quad \forall \theta^k \in \Theta, k = t, \dots, t+n-1$$

The observability condition of Assumption 6.1 implies that the right-hand side is a positive definite function of η_t, hence we can find $c_1, c_2 > 0$ such that $c_1\|\eta_t\|^2 \leq \eta_t^T W \eta_t \leq c_2\|\eta_t\|^2$ and $\sum_{k=t}^{t+n-1} \|\Lambda(\theta^k)\eta^k\|^2 \geq c_1\|\eta_t\|^2$ for all θ^k, $k = t, \dots, t+n-1$. From the above we have

$$(\eta^{t+n})^T W \eta^{t+n} - \eta_t^T W \eta_t \leq -\frac{c_1}{c_2}\eta_t^T W \eta_t$$

and from recursive feasibility we know that $(\eta^{t+n})^T W \eta^{t+n} \geq \eta_{t+n}^T W \eta_{t+n}$, so after rearranging we find

$$\eta_{t+n}^T W \eta_{t+n} \leq \left(1 - \frac{c_1}{c_2}\right) \eta_t^T W \eta_t$$

and this implies

$$\eta_t^T W \eta_t \leq \left(1 - \frac{c_1}{c_2}\right)^{\frac{t}{n}} \eta_0^T W \eta_0$$

for $t = 0, n, 2n, \dots$. Moreover from (6.22) we have $\eta_{t+1}^T W \eta_{t+1} \leq \eta_t^T W \eta_t$ and

$$\eta_t^T W \eta_t \leq \delta^{\frac{t}{n}-1} \eta_0^T W \eta_0$$

for all $t > 0$, where $\delta = 1 - \frac{c_1}{c_2} < 1$, as claimed.

Part III

Stochastic Tube MPC

7 Sample-admissible sets

7.1 Introduction

Many MPC algorithms rely on the use of positively invariant terminal sets to ensure feasibility of the optimisation for all time. In a similar fashion, the tube MPC controllers considered in chapters 5 and 6 require the final tube cross-section to be a robustly invariant set satisfying constraints. Constraint-admissible sets are therefore invaluable in MPC formulations, as well as being useful in their own right in the analysis and control of constrained systems.

The theory of the maximum constraint-admissible set for linear time-invariant systems was given in [GT91]. This was extended to consider systems with bounded disturbances in [KG98], and more recent papers have considered other extensions such as nonlinear systems [KM00] or parametric uncertainty [PRSDM05b]. A good overview of the use of such sets in model predictive control can be found in the thesis [Ker01], and for other control applications in the survey paper [Bla99]. More recently, there has also been some interest in minimum robustly invariant sets for use as a 'target set' in robust MPC, and useful methods for constructing robustly invariant approximations to that set appeared in [RKKM05] and were extended to the case of a linear difference inclusion in [KRK$^+$05]. The book [BM07] contains many results concerning set robust invariance as it pertains to control of uncertain systems.

In this chapter, we consider the problem of finding the maximum admissible set (MAS) for a linear difference inclusion subject to a chance constraint on the state. This has direct application to chance-constrained MPC of stochastic systems, where a chance-constraint can be applied on the 'one-step-ahead' value of the state variable to retain recursive feasibility. We shall see that, under some standard assumptions, the MAS in this case is given by a finite number of chance constraints. Although this set shares many theoretical properties with its counterpart in the deterministic case, there are two major practical drawbacks:

- The chance constraints that represent the MAS are very difficult to determine numerically;
- Sampling the constraints that represent the MAS yields a polyhedron that is not robustly invariant in general;

The second point is a serious problem if sample approximations are to be used to handle chance constraints for implementation of a stochastic model predictive controller, as recursive feasibility generally relies on the robust invariance of the terminal set. As a remedy, we consider a subclass of the admissible sets that are representable by constraints that satisfy an robust invariance condition pointwise over the whole sample space. We will show that when such a condition is satisfied, sample approximations yield positively-invariant polyhedral sets. We shall also show that, as with constraint admissible sets, there exists a well-defined maximum and minimum element of this subclass with respect to subset relations. Conveniently, the largest set in this class can be determined using existing robust algorithms for computing admissible sets, thus circumventing the former problem as well as the latter one. Finally, we suggest a simple algorithm for constructing robustly invariant polyhedral approximations based on sampling.

7.1.1 Notation and preliminaries

For analysis purposes, we will be considering dynamics defined by a set-valued function $\Phi : \mathbb{R}^n \to P(\mathbb{R}^n)$. Recall that images and preimages of this function are defined, with a little notational abuse, as:

$$\Phi(\mathcal{A}) = \{x \in \mathbb{R}^n : \exists z \in \mathcal{A}, x \in \Phi(z)\}$$

$$\Phi^{-1}(\mathcal{A}) = \{x \in \mathbb{R}^n : \forall z \in \Phi(x), z \in \mathcal{A}\}$$

Such a set-valued function is said to be upper semicontinuous if the preimage of any closed set is closed and lower semicontinuous if the preimage of any open set is open. When both of these conditions hold, we say that it is continuous. The image and preimage operations are inverses of each other, so if k-fold repeated images and preimages are denoted by $\Phi^k(\mathcal{A})$ and $\Phi^{-k}(\mathcal{A})$ respectively, we have the composition law

$\Phi^p(\Phi^q(\mathcal{A})) = \Phi^{p+q}(\mathcal{A})$ for all integers p, q of the same sign. When considering images of preimages and vice versa, we have $\mathcal{A} \subseteq \Phi^{-1}(\Phi(\mathcal{A}))$ and $\Phi(\Phi^{-1}(\mathcal{A})) \subseteq \mathcal{A}$. These operations distribute over unions of sets, and will 'distribute' over intersections of sets if we replace the equality with a subset relation, so that:

$$\Phi\left(\bigcap \mathcal{A}_\alpha\right) = \bigcap \Phi(\mathcal{A}_\alpha), \quad \Phi\left(\bigcup \mathcal{A}_\alpha\right) \subseteq \bigcup \Phi(\mathcal{A}_\alpha)$$

For preimages, we get equality in both cases:

$$\Phi^{-1}\left(\bigcap \mathcal{A}_\alpha\right) = \bigcap \Phi^{-1}(\mathcal{A}_\alpha), \quad \Phi^{-1}\left(\bigcup \mathcal{A}_\alpha\right) = \bigcup \Phi^{-1}(\mathcal{A}_\alpha)$$

The set valued functions we consider are often defined as the convex hull of a finite set of points:

$$\Phi(x) = \mathrm{Conv}\left(\{\Phi_i(x) : \ \ i = 1, \ldots, m\}\right)$$

For such functions, the image and preimage operations take the simplified forms,

$$\Phi(\mathcal{A}) = \bigcup_{i=1}^{m} \Phi_i(\mathcal{A}), \quad \Phi^{-1}(\mathcal{A}) = \bigcap_{i=1}^{m} \Phi_i^{-1}(\mathcal{A})$$

and it is immediate from these expressions that continuity of the functions Φ_i implies continuity of Φ in the sense defined above. We introduce the following 'multi-index' notation for compositions of the vertex functions $\Phi_i(x)$. If $M = \{i, j, \ldots, m\}$ then a finite composition can be written as:

$$\Phi_M(x) = \Phi_i(\Phi_j(\ldots \Phi_m(x) \ldots))$$

The set of all possible combinations of k indices in the range $1, \ldots, m$ will be denoted $\mathcal{I}_k$. For convenience we also define $\Phi_\emptyset(x) = x$, that is, setting M as the empty set gives the identity function.

For any sequence of sets $\mathcal{A}_k, k = 0, 1, 2, \ldots$ we define:

$$\liminf_{k\to\infty} \mathcal{A}_k = \bigcup_{N=0}^{\infty} \bigcap_{k \geq N} \mathcal{A}_k$$

$$\limsup_{k\to\infty} \mathcal{A}_k = \bigcap_{N=0}^{\infty} \bigcup_{k\geq N} \mathcal{A}_k$$

If these two limits are equal, say to $\mathcal{A}_\infty$, then we define this as the limit of the sequence and write

$$\mathcal{A}_\infty = \lim_{k\to\infty} \mathcal{A}_k$$

which will allow us to investigate the asymptotic properties of set sequences.

In comparison to defining limits using a Hausdorff metric, these notions of limits and continuity allow for statements such as $\lim_{k\to\infty} \mathcal{A}_k = \mathbb{R}^n$ when a set diverges and covers the entire space. In contrast, such a set would not converge in the Hausdorff metric, since it is generally defined on subsets of a compact space. In most instances the distinction will be unimportant, since whenever the limit in the sense of a Hausdorff metric exists it will be equal to the limit given here, but it is important to note that there is no implicit assumption that the sets we describe are compact.

7.2 Admissible sets

The system we consider is the autonomous parameter-uncertain system

$$x_{t+1} = A(\theta_t)x_t + w(\theta_t) \tag{7.1a}$$

$$z_t = F(\omega_t)x_t + v(\omega_t) \tag{7.1b}$$

where $\theta_t \in \Theta$ for all t. We consider the output z_t to be subject to a chance constraint

$$\mathbb{P}\{F(\omega_t)x_t + v(\omega_t) \leq 1\} \geq 1 - \epsilon \tag{7.2}$$

where $F(\omega)$ and $v(\omega)$ are affine functions of a stochastic process ω_t that takes values in Ω. We make the assumption that the process ω_t has independent values and is identically distributed at each time t. For this reason, we will often write simply ω in place of ω_t when the time index is not relevant.

Assumption 7.1. *The random variables ω_t are independent and identically distributed for each t.*

Remark 7.1. Stochastic systems under chance constraints can be considered by defining $F(\omega)$ and $v(\omega)$ such that z_k is the predicted state one time step ahead, that is $z_k = x_{k+1}$. One can then consider the problem of robustly satisfying this one-step-ahead chance constraint. We will reconsider this viewpoint briefly in the following chapter when we consider stochastic MPC with a similar chance constraint.

Remark 7.2. By substituting $\tilde{A}(\theta) = A(\theta) + B(\theta)K$ or $\tilde{A}(\theta) = A(\theta) + BK(\theta)$ (and similarly for $F_k(\omega)$), the results of this chapter immediately extend to LDI and LPV-A systems controlled by linear or parameter-varying state feedbacks, respectively. The behaviour of systems under dynamic feedback controllers (rather than static controllers) can be handled by augmenting the system state to include the controller state, so that the resulting dynamics become representable by a state feedback and these substitutions applied.

In almost all cases of general interest, boundedness of the uncertainty set Θ is a necessary condition for robustly positively invariant sets of the system (7.1) to exist. In order to obtain a closed-form representation of the robustly invariant sets and suggest algorithms, we will make the stronger assumption here that Θ is polytopic.

Assumption 7.2. *The uncertainty set Θ is a polytope.*

Define a set-valued function $\Phi : \mathbb{R}^n \to P(\mathbb{R}^n)$ representing the dynamics (7.1):

$$\Phi(x) = \bigcup_{\theta\in\Theta} \Phi_\theta(x) = \bigcup_{\theta\in\Theta} \{A(\theta)x + w(\theta)\} \tag{7.3}$$

With a little abuse of notation, images and preimages of sets $\mathcal{X} \subseteq \mathbb{R}^n$ will be denoted $\Phi(\mathcal{X})$ and $\Phi^{-1}(\mathcal{X})$. If we define a function $\Phi_\theta(x) = A(\theta)x + w(\theta)$, then images and preimages under the uncertain mapping Φ can be written quite conveniently in terms of unions and intersections over θ:

$$\Phi(\mathcal{X}) = \bigcup_{\theta\in\Theta} \Phi_\theta(\mathcal{X}), \quad \Phi^{-1}(\mathcal{X}) = \bigcap_{\theta\in\Theta} \Phi_\theta^{-1}(\mathcal{X})$$

Continuity of Φ_θ implies that $\Phi_\theta^{-1}(\mathcal{X})$ will be a closed set if $\mathcal{X}$ is closed. As intersections of closed sets are also closed, the preimage is closed. Hence the set-valued mapping Φ is upper semicontinuous. In similar fashion we can see that preimages of convex sets $\mathcal{X}$ will be convex themselves, since $\Phi_\theta^{-1}(\mathcal{X})$ will then be convex and we have an intersection of convex sets.

We now consider the set of points in $\mathbb{R}^n$ satisfying (7.2), which will be denoted $\mathcal{Z}_0$. This set is closed, since the probability measure of (7.2) must be continuous from above.

Definition 7.1. *The* **feasible set** $\mathcal{Z}_0$ *is defined as*

$$\mathcal{Z}_0 = \{x \in \mathbb{R}^n | \ \mathbb{P}\{F(\omega)x + v(\omega) \leq 1\} \geq 1 - \epsilon\} \tag{7.4}$$

and is the set of all points in state space that satisfy (7.2).

With this definition, it is now straightforward to define what is meant by an admissible set of (7.1); it is one that is robustly positively invariant and satisfies constraints.

Definition 7.2. *A set* $\mathcal{A}$ *is* **admissible** *if* $\Phi(\mathcal{A}) \subseteq \mathcal{A}$ *and* $\mathcal{A} \subseteq \mathcal{Z}_0$.

For the subsequent discussion we will assume that the system (7.1) satisfies an asymptotic stability condition, so that all trajectories of the system tend to a given set in the fullness of time. This ensures that there exists at least one admissible set.

Assumption 7.3. *There exists a bounded set* $\mathcal{X}_\infty \subseteq \mathcal{Z}_0$ *such that*

$$\mathcal{X}_\infty = \lim_{k\to\infty} \Phi^k(\{\mathcal{X}_0\})$$

for any bounded $\mathcal{X}_0 \subseteq \mathbb{R}^n$.

Remark 7.3. When no additive disturbance is present, this stability condition is equivalent to the notion of absolute stability of a difference inclusion given in [Gur95] in the case that $\mathcal{X}_\infty = \{0\}$, that is that all trajectories of the LDI approach 0. It is also equivalent to the joint spectral radius of the matrices $A(\theta)$ being less than one, since this is a necessary and sufficient condition for stability in that case [BM07, Theorem 6.25].

We define maximum and minimum admissible sets as those that contain, or are contained within, all other admissible sets.

Definition 7.3. *The set $\mathcal{A}_{\max}$ is the* **maximum admissible set** *(MAS) of (7.1) if it is admissible and $\mathcal{A} \subseteq \mathcal{A}_{\max}$ for any admissible $\mathcal{A}$. Similarly $\mathcal{A}_{\min}$ is the* **minimum admissible set** *(mAS) of (7.1) if it is admissible and $\mathcal{A}_{\min} \subseteq \mathcal{A}$ for any admissible $\mathcal{A}$.*

The class of admissible sets has an interesting algebraic structure in that it is closed under the operations of set union and intersection. This will be quite an important property in subsequent results, so we briefly provide a proof.

Proposition 7.4. *Any union or intersection of admissible sets is admissible.*

Noting that '$\subseteq$' gives a partial order on the class of admissible sets, these closure properties immediately lead to an existence and uniqueness result for the maximum and minimum admissible sets.

Proposition 7.5. *System (7.1) has unique maximum and minimum admissible sets.*

Remark 7.6. Algebraically speaking, the closure properties above and the existence of maximum and minimum elements can be summarised by stating that the admissible sets of (7.1) form a bounded lattice with the operations of union and intersection. This algebraic structure of invariant sets is fairly well known in the study of dynamical systems - see e.g. [KMV13, Section 2.2] for a technical treatment of the topic. Our definition of the limit of sets fits well within this framework, being an order-theoretic definition rather than the more conventional approach in constrained control of convergence in Hausdorff metric.

The next result states, perhaps unsurprisingly, that $\mathcal{X}_\infty$ is the minimum admissible set.

Proposition 7.7. *The set $\mathcal{X}_\infty$ of Assumption 7.3 is the minimum admissible set of (7.1).*

As the expression for the minimum admissible set does not depend on the constraint set $\mathcal{Z}_0$, the presence of a chance constraint in the problem has no bearing on the form of the set. If the uncertainty set Θ is polytopic, then [KRK+05] gives an algorithm that can approximate the minimum admissible set of the LDI (7.1) with a polytope to any given degree of accuracy. This polytope is itself admissible.

By contrast, the maximum admissible set will have a form that depends on the set $\mathcal{Z}_0$, and so the MAS may be very different in the presence of a chance constraint than it would be with linear constraints. This is the problem we consider in detail in the next section.

7.2.1 The maximum admissible set

We now develop a closed-form expression for the maximal admissible set of (7.1) under the chance constraint. Use the image and preimage operations to define the sets $\mathcal{Z}_k$ for any integer k by

$$\mathcal{Z}_k = \Phi^{-k}(\mathcal{Z}_0) \tag{7.5}$$

so that $\Phi(\mathcal{Z}_k) = \mathcal{Z}_{k-1}$ and $\Phi^{-1}(\mathcal{Z}_k) = \mathcal{Z}_{k+1}$ for all $k \in \mathbb{Z}$. We further define $\mathcal{S}_k$ for $k \geq 0$ to be the intersection of all $\mathcal{Z}_j$ with $0 \leq j \leq k$:

$$\mathcal{S}_k = \bigcap_{j=0}^{k} \mathcal{Z}_j = \bigcap_{j=0}^{k} \Phi^{-j}(\mathcal{Z}_0) \tag{7.6}$$

The following theorem establishes that the limit $\lim_{k\to\infty} \mathcal{S}_k$ exists and is the MAS of (7.1).

Theorem 7.8 (Maximum Admissible Set)**.** *The limit* $\lim_{k\to\infty} \mathcal{S}_k$ *exists, is equal to* $\mathcal{S}_\infty$ *and is the maximum admissible set for the system (7.1) under the chance constraint (7.2).*

We now investigate some properties of the maximum robustly invariant set, namely convexity, compactness, and finite determination. From the form of $\mathcal{S}_\infty$, it is evident that the MAS will be convex or compact if $\mathcal{Z}_0$ is convex or compact, respectively. This implies that any condition (such as logconcavity of the distribution of $F(\omega), v(\omega)$) that implies convexity of the chance constraint (7.2) also implies convexity of the MAS.

Proposition 7.9 ($\mathcal{S}_k$ inherits convexity, compactness)**.** *If* $\mathcal{Z}_0$ *is convex or compact then* $\mathcal{S}_k$ *is convex or compact respectively, for all* $k = 1, 2, \dots, \infty$.

The definition of $\mathcal{S}_\infty$ contains an infinite number of chance constraints. However the next theorem on finite determination shows that only a finite number of these constraints are required if the minimum robustly invariant set lies in the interior of $\mathcal{Z}_0$.

Theorem 7.10 (Finite Determination). *If $\mathcal{S}_k$ is compact for some finite k and $\mathcal{X}_\infty \subseteq$ int$(\mathcal{Z}_0)$ then $\mathcal{S}_N = \mathcal{S}_\infty$ for all $N \geq K$, for some sufficiently large but finite K.*

7.2.2 Representation using chance constraints

Recalling that the set Θ is polytopic by Assumption 7.2, we now investigate the chance constraints defining the maximal admissible set $\mathcal{S}_\infty$. Let $\Theta = Co\{\theta^{(i)},\ i = 1, \ldots, r\}$, and for convenience denote $A(\theta^{(i)})x + w(\theta^{(i)}) = \Phi_i(x)$. We introduce the following 'multi-index' notation for compositions of the vertex functions $\Phi_i(x)$, which is similar to the notation used in [BM07]. If $M = \{i, j, \ldots, m\}$ then a finite composition can be written as:

$$\Phi_M(x) = \Phi_i(\Phi_j(\ldots \Phi_m(x)\ldots))$$

The set of all possible combinations of k indices in the range $1, \ldots, m$ will be denoted $\mathcal{I}_k$. For convenience we also define $\Phi_\emptyset(x) = x$, that is, setting M as the empty set gives the identity function. Using this notation, the set $\mathcal{Z}_k$ for $k \geq 0$ is given by a finite number of chance constraints

$$\mathcal{Z}_k = \{x \in \mathbb{R}^n |\ \mathbb{P}\{F(\omega)\Phi_M(x) + v(\omega) \leq 1\} \geq 1 - \epsilon,\ \ \forall M \in \mathcal{I}_k\}$$

so that the sets $\mathcal{S}_k$ can be represented similarly as

$$\mathcal{S}_k = \left\{ x \in \mathbb{R}^n \,\middle|\, \mathbb{P}\{F(\omega)\Phi_M(x) + v(\omega) \leq 1\} \geq 1 - \epsilon,\ \ \forall M \in \bigcap_{j=0}^{k} \mathcal{I}_j \right\}$$

and if $\mathcal{S}_\infty$ is finitely determined in the sense of Theorem 7.10, the maximum admissible set is given by a finite number of chance constraints. Defining

$$\mathcal{S}_\mathcal{J} = \{x \in \mathbb{R}^n |\ \mathbb{P}\{F(\omega)\Phi_M(x) + v(\omega) \leq 1\} \geq 1 - \epsilon,\ \ \forall M \in \mathcal{J}\}$$

then, if $\mathcal{S}_\infty$ is finitely determined, $\mathcal{S}_\mathcal{J} = \mathcal{S}_\infty$ for some finite set $\mathcal{J}$. It is quite possible that many of the indices in $\bigcap_{j=0}^{k} \mathcal{I}_j$ might correspond to redundant constraints, so that in general we have $\mathcal{J} \subseteq \bigcap_{j=0}^{N} \mathcal{I}_j$ for some N.

Remark 7.11. Consider the special case in which the matrix $F(\omega) = f^T$ consists of a single row and is not random. In that case, the constraint defining $\mathcal{X}_0$ can be written as

$$\mathbb{P}\{f^T x \leq 1 - v(\omega)\} \geq 1 - \epsilon$$

which is equivalent to

$$f^T x \leq 1 - P_v^{-1}(1 - \epsilon)$$

where $P_v(.)$ denotes the distribution function of v. Then $\mathcal{S}_\infty$ is an intersection of polytopes and is polytopic if $\mathcal{S}_N = \mathcal{S}_\infty$ for some finite N. A particular case is considered in [KCRC10] where Θ is a singleton, so that there is no parameter uncertainty in the system, but the MAS is also polytopic whenever Θ and $\mathcal{Z}_0$ are.

Now consider the problem of deciding if $\mathcal{S}_\mathcal{J} = \mathcal{S}_\infty$ for a given N. This equality holds if and only if $\Phi_i(\mathcal{S}_\mathcal{J}) \subseteq \mathcal{S}_\mathcal{J}$ for all $i - 1, \ldots, m$. For any i, this is equivalent to showing that the optimal objective of the following optimisation problem is greater than $1 - \epsilon$ for all $M \in \mathcal{J}$:

$$\begin{aligned} &\underset{x}{\text{minimise}} && \mathbb{P}\{F(\omega)\Phi_M(\Phi_i(x)) + v(\omega) \leq 1\} \\ &\text{subject to} && \mathbb{P}\{F(\omega)\Phi_M(x) + v(\omega) \leq 1\} \geq 1 - \epsilon \end{aligned}$$

If this optimisation is tractable, then we can find the maximum robustly invariant set by the following generic algorithm, which gives an index set $\mathcal{J}$ defining the MAS.

Algorithm 7.1 MAS index set

1. Initialise $\mathcal{J} = \{\emptyset\}$
2. For each $M \in \mathcal{J}$:
   (a) For each $i = 1, \ldots, m$:
      i. If $\Phi_i(\mathcal{S}_\mathcal{J}) \not\subseteq \mathcal{S}_\mathcal{J}$, append $\{M, i\}$ to $\mathcal{J}$

As the loop in step 2 can only terminate if $\Phi_i(\mathcal{S}_\mathcal{J}) \subseteq \mathcal{S}_\mathcal{J}$ for all $i = 1, \ldots, m$ (as otherwise, extra elements are added to $\mathcal{J}$) we see that the algorithm will only terminate if a robustly invariant set has been found. As $\mathcal{S}_\infty \subseteq \mathcal{S}_\mathcal{J}$ for any finite $\mathcal{J}$, this can only occur if $\mathcal{S}_\mathcal{J} = \mathcal{S}_\infty$. This will occur after a finite number of iterations if the MAS is finitely determined, as it is clear from the form of the algorithm that it will give all nonredundant constraints in $\mathcal{S}_N$ for any finite N if it does not terminate.

Proposition 7.12. *If $\mathcal{S}_\infty$ is finitely determined, Algorithm 7.1 terminates in a finite amount of time with $\mathcal{S}_\mathcal{J} = \mathcal{S}_\infty$.*

A major drawback is that for most problems involving chance constraints it is unlikely that checking $\Phi_i(\mathcal{S}_\mathcal{J}) \subseteq \mathcal{S}_\mathcal{J}$ corresponds to a convex optimisation problem, and so this might be difficult if the state dimension is not small. Indeed in the case that $\mathcal{Z}_0$ is a convex set the domain $\mathcal{S}_\mathcal{J}$ will be convex but the objective function is quasiconcave, and so this is not a convex optimisation problem unless the objective is linear (as in Remark 7.11).

Another practical issue with the MAS is that, in applications such as MPC, it may be difficult to evaluate the constraint functions $\mathbb{P}\{F(\omega)\Phi_M(x) + v(\omega) \leq 1\}$. This issue is often resolved by taking a sample approximation, so that the chance constraints are replaced by linear constraints $F(\omega^{(j)})\Phi_M(x) + v(\omega^{(j)}) \leq 1$ for some sufficiently large subset δ_a of a multisample $\delta = \{\omega^{(1)}, \omega^{(2)}, \ldots, \omega^{(N)}\}$, leading to an approximation:

$$\mathcal{S}_\mathcal{J}(\delta) = \left\{x \in \mathbb{R}^n \middle| \; F(\omega^{(j)})\Phi_M(x) + v(\omega^{(j)}) \leq 1, \;\; \forall \omega^{(j)} \in \delta_a, \;\; \forall M \in \mathcal{J}\right\}$$

There is then no guarantee that the resulting sampled constraints define a robustly invariant set, so that $\Phi(\mathcal{S}_\mathcal{J}(\delta)) \not\subseteq \mathcal{S}_\mathcal{J}(\delta)$ in general. Practically, it would be better if we could find a set that preserves the robust invariance property when sample approximations are considered. For these reasons, in the next section we turn our attention to finding admissible approximations of the MAS that we can compute using more tractable methods, and that preserve the robust invariance property under sampling.

7.3 Sample-admissible sets

We now construct approximations to the MAS that are defined by chance constraints, but that can be found without chance-constrained programming and that give robustly invariant sets when sample approximations are taken. For simplicity in stating some of the results, we will assume that the set Ω is polytopic. Unbounded uncertainty is obviously of practical interest, so we briefly describe (in Remark 7.13 below) an extension to this case.

Assumption 7.4. *The set Ω is a polytope.*

For any point $\omega \in \Omega$ in the sample space, we can define a corresponding polytopic feasible set in state space given by:

$$\hat{\mathcal{Z}}_0(\omega) = \{x \in \mathbb{R}^n | \ F(\omega)x + v(\omega) \leq 1\} \tag{7.7}$$

The set $\hat{\mathcal{Z}}_0(\omega)$ can be thought of as a set-valued random variable, that is, a measurable function mapping Ω into $P(\mathbb{R}^n)$. Note in particular that with this interpretation we have

$$\mathcal{Z}_0 = \{x \in \mathbb{R}^n | \ \mathbb{P}\{x \in \hat{\mathcal{Z}}_0(\omega)\} \geq 1 - \epsilon\}$$

so that membership of $\mathcal{Z}_0$ is equivalent to membership of $\hat{\mathcal{Z}}_0(\omega)$ for a set of ω of sufficiently large measure.

We will consider robustly invariant sets constructed by taking intersections and unions of $\hat{\mathcal{S}}_{\mathcal{J}}(\omega)$ for different $\omega^{(i)}$ from a multisample δ, where $\hat{\mathcal{S}}_{\mathcal{J}}(\omega)$ is defined analogously to $\mathcal{S}_{\mathcal{J}}$ as:

$$\hat{\mathcal{S}}_{\mathcal{J}}(\omega) = \{x \in \mathbb{R}^n | \ F(\omega)\Phi_M(x) + v(\omega) \leq 1, \ \forall M \in \mathcal{J}\}$$

As the class of robustly invariant sets is closed under unions and intersections, the resulting sets will be robust invariant if $\mathcal{S}_{\mathcal{J}}(\omega^{(i)})$ is for each $\omega^{(i)} \in \delta$. We will see this latter condition can be ensured by constructing maximal admissible sets under robust, rather than chance, constraints, so that in carrying out this procedure we will not have to consider any chance-constrained programming problems. We first define a subclass of the admissible sets that is well-behaved with respect to sample approximations using any multisample δ.

Definition 7.4. *The set $\mathcal{A}$ given by*

$$\mathcal{A} = \{x \in \mathbb{R}^n | \ \mathbb{P}\{x \in \hat{\mathcal{A}}(\omega)\} \geq 1 - \epsilon\} \tag{7.8}$$

is **sample-admissible** *if, for all $\omega \in \Omega$, $\Phi(\hat{\mathcal{A}}(\omega)) \subseteq \hat{\mathcal{A}}(\omega)$ and $\hat{\mathcal{A}}(\omega) \subseteq \hat{\mathcal{Z}}_0(\omega)$.*

Remark 7.13. If dealing with unbounded stochastic uncertainty in ω, the condition $\Phi(\hat{\mathcal{A}}(\omega)) \subseteq \hat{\mathcal{A}}(\omega)$ over all Ω in the above is likely too strong to make these sets of any practical use. In that case, this condition can be relaxed so that the robust invariance property need only hold for ω in the multisample δ. Much of the resulting theory is identical. Note however that we must still assume that Θ is bounded, as otherwise no robustly invariant sets will exist in general.

From the definition of $\mathcal{A}$, we can alternatively write it as a union and intersection over subsets of the sample space of measure at least $1-\epsilon$. Let $\mathcal{C} = \{E \subseteq \Omega |\ \mathbb{P}(E) \geq 1-\epsilon\}$ and then:

$$\mathcal{A} = \{x \in \mathbb{R}^n |\ \mathbb{P}\{x \in \hat{\mathcal{A}}(\omega)\} \geq 1-\epsilon\} = \bigcup_{E \in \mathcal{C}} \bigcap_{\omega \in E} \hat{\mathcal{A}}(\omega) \tag{7.9}$$

This expression reveals that any sample-admissible set is also admissible, from the closure of the admissible sets under union and intersection, justifying our earlier claim that this forms subclasses of the admissible sets. If we consider a sample approximation of such a set

$$\hat{\mathcal{A}}(\delta) = \left\{ x \in \mathbb{R}^n \middle|\ x \in \hat{\mathcal{A}}(\omega^{(j)}),\ \ \forall \omega^{(j)} \in \delta_a,\ \ \delta_a \subseteq \delta,\ \ |\delta_a| \geq r \right\}$$

then we can express this as

$$\hat{\mathcal{A}}(\delta) = \bigcup_{\{\delta_a : |\delta_a| \geq r\}} \bigcap_{\omega^{(j)} \in \delta_a} \hat{\mathcal{A}}(\omega^{(j)})$$

so that the set given by the sample approximation will itself satisfy the robust invariance condition $\Phi(\hat{\mathcal{A}}(\delta)) \subseteq \hat{\mathcal{A}}(\delta)$ from the closure properties.

It would be convenient if, like the admissible sets, the class of sample-admissible sets were closed under unions and intersection, as it would imply the existence of a maximum and minimum sample-admissible set under the partial ordering '$\subseteq$'. But this cannot be true. To see this, we must simply note that an intersection of sample-admissible sets will be defined by two chance constraints in general, and there is no guarantee that these are equivalent to a single chance constraint in the form (7.8).

To circumvent this difficulty, we (borrowing terminology from the theory of lattices and boolean algebras) define alternative 'join' and 'meet' operations that respect the partial ordering '$\subseteq$' by considering unions and intersections pointwise on the sample space. Hence define the ***join*** of a family of sample-admissible sets as

$$\bigvee_{\alpha} \mathcal{A}_\alpha = \{x \in \mathbb{R}^n | \ \mathbb{P}\{x \in \bigcup_{\alpha} \hat{\mathcal{A}}_\alpha(\omega)\} \geq 1-\epsilon\} \tag{7.10}$$

and the ***meet*** of a family as

$$\bigwedge_{\alpha} \mathcal{A}_\alpha = \{x \in \mathbb{R}^n | \ \mathbb{P}\{x \in \bigcap_{\alpha} \hat{\mathcal{A}}_\alpha(\omega)\} \geq 1-\epsilon\} \tag{7.11}$$

and we then note that the resulting algebra of sets is closed under the operations of join and meet. It is a bounded lattice, just like the admissible sets with the union and intersection operations.

Proposition 7.14. *The join and meet of any family of sample-admissible sets is sample-admissible.*

Noting that, for all α, we have $\mathcal{A}_\alpha \subseteq \bigvee_\alpha \mathcal{A}_\alpha$ and $\bigwedge_\alpha \mathcal{A}_\alpha \subseteq \mathcal{A}_\alpha$, allows us to prove the existence of a unique maximum and minimum sample-admissible set from the closure properties.

Proposition 7.15. *The maximum and minimum sample-admissible sets of (7.1) exist and are unique.*

7.3.1 The maximum sample-admissible set

Because sample-admissible sets are constructed from random sets that are admissible pointwise on the sample space, it is straightforward to give an expression for the maximum sample-admissible set. We first define $\hat{\mathcal{S}}_\infty(\omega)$ as the maximal admissible set corresponding to the random feasible set $\hat{\mathcal{Z}}_0(\omega)$ under the dynamics (7.1a). From the definition in the previous section, the set $\hat{\mathcal{S}}_\mathcal{J}(\omega)$ can be expressed as an intersection of preimages

$$\hat{\mathcal{S}}_\mathcal{J}(\omega) = \bigcap_{M \in \mathcal{J}} \Phi_M^{-1}(\hat{\mathcal{Z}}_0(\omega)) \tag{7.12}$$

and there will exist some finite $\mathcal{J}$ such that $\hat{\mathcal{S}}_\mathcal{J}(\omega) = \hat{\mathcal{S}}_\infty(\omega)$.

Recalling that $\mathcal{C} = \{E \subseteq \Omega |\ \mathbb{P}(E) \geq 1 - \epsilon\}$, define:

$$\mathcal{T}_\infty = \{x \in \mathbb{R}^n |\ \mathbb{P}\{x \in \hat{\mathcal{S}}_\infty(\omega)\} \geq 1 - \epsilon\} = \bigcup_{E \in \mathcal{C}} \bigcap_{\omega \in E} \hat{\mathcal{S}}_\infty(\omega) \tag{7.13}$$

We claim that $\mathcal{T}_\infty$ is the maximal sample-admissible set of (7.1).

Theorem 7.16 (Maximum Sample Admissible Set). *The set $\mathcal{T}_\infty$ is the maximum sample-admissible set for the system (7.1).*

Despite being defined by an uncountable union of sets, the representation $\mathcal{T}_\infty = \{x \in \mathbb{R}^n |\ \mathbb{P}\{x \in \hat{\mathcal{S}}_\infty(\omega)\} \geq 1 - \epsilon\}$ reveals that $\mathcal{T}_\infty$ is a closed set, as $\hat{\mathcal{S}}_\infty(\omega)$ is closed (it is an intersection of closed sets, by upper semicontinuity of Φ) and the measure $\mathbb{P}$ must be continuous from above. We know $\mathcal{T}_\infty \subseteq \mathcal{S}_\infty$ by admissibility of $\mathcal{T}_\infty$, so that the maximal sample-admissible set will be compact if $\mathcal{S}_\infty$ is. Finally, we note that the set will be finitely determined if $\hat{\mathcal{S}}_\infty(\omega)$ is for all $\omega \in \Omega$, so that Theorem 7.10 applies here in a pointwise sense.

7.3.2 Representation using chance constraints

We again exploit the polytopic representation $\Theta = Co\{\theta^{(i)},\ i = 1, \ldots, r\}$, so that we may consider the chance constraints defining the maximal sample-admissible set $\mathcal{T}_\infty$. Then the set $\mathcal{T}_k$, analogously to $\mathcal{S}_k$, can be defined using a joint chance constraint as:

$$\mathcal{T}_k = \left\{ x \in \mathbb{R}^n \,\middle|\, \mathbb{P}\{F(\omega)\Phi_M(x) + v(\omega) \leq 1,\ \forall M \in \bigcap_{j=0}^{k} \mathcal{I}_j\} \geq 1 - \epsilon \right\}$$

Similarly we can also define $\mathcal{T}_\mathcal{J}$:

$$\mathcal{T}_\mathcal{J} = \{x \in \mathbb{R}^n |\ \mathbb{P}\{F(\omega)\Phi_M(x) + v(\omega) \leq 1,\ \forall M \in \mathcal{J}\} \geq 1 - \epsilon\}$$

The key thing to notice in these definitions is that the '$\forall M \in \mathcal{J}$' is part of the event that we are considering the probability of. Hence this is a joint chance constraint, whereas the maximal admissible set is defined by many individual chance constraints.

An index set $\mathcal{J}$ giving $\mathcal{T}_\mathcal{J} = \mathcal{T}_\infty$ is not difficult to find if we know the vertices of the polytope Ω, since then we only need to find index sets $\mathcal{J}^{(i)}$ corresponding to admissible sets of the vertex constraints $F(\omega^{(i)})x + v(\omega^{(i)}) \leq 1$ of that polytope and set $\mathcal{J} = \bigcup_i \mathcal{J}^{(i)}$. This can be done using any algorithm for finding maximal admissible sets for a linear constraint. This leads to a systematic procedure (Algorithm 7.2) for finding the index set $\mathcal{J}$. The main part of the algorithm, step 2a, can be carried out using an existing robust MAS algorithm like Algorithm 4.3, perhaps with a few simple modifications to keep track of the index set and update it on each iteration.

Algorithm 7.2 Maximum sample-admissible index set

1. Initialise $\mathcal{J} = \{\emptyset\}$
2. For each $i = 1, \ldots, m$:
   (a) Compute $\mathcal{J}^{(i)}$ such that $\hat{\mathcal{S}}_{\mathcal{J}^{(i)}}(\omega^{(i)}) = \hat{\mathcal{S}}_\infty(\omega^{(i)})$
3. Set $\mathcal{J} = \bigcup_{i=1}^m \mathcal{J}^{(i)}$

Remark 7.17. It is possible to extend Algorithm 7.2 to consider reduced-complexity sets analogously to Algorithm 4.3, if we keep track of constraint-tightening parameters along with the indices in $\mathcal{J}$.

Once the correct index set $\mathcal{J}$ has been computed, the set $\mathcal{T}_\infty$ can be used in one of two ways. It could be sampled directly, say online by an MPC controller, to give a robustly invariant sample approximation. Alternatively, we could consider such sample approximations offline and try to find a simpler representation of the sample approximation as a polyhedron. We now briefly consider this latter approach.

7.3.3 Approximation by robustly invariant polyhedra

We can exploit the 'pointwise in ω' robust invariance property of the sample-admissible sets to construct useful polyhedral robustly invariant sets for chance-constrained systems, by explicitly computing unions and intersections of $\hat{\mathcal{S}}_{\mathcal{J}}(\omega)$ for some finite number of samples. As $\hat{\mathcal{S}}_{\mathcal{J}}(\omega)$ is a robustly invariant set for each $\omega \in \Omega$, unions or intersections of thse sets will also be robustly invariant from Proposition 7.4. Hence the resulting sets are robustly invariant, in that they are positively invariant for the uncertain system for any $\theta \in \Theta$, and feasible for the constraints in the sense that $\hat{\mathcal{S}}_{\mathcal{J}}(\omega) \subseteq \hat{\mathcal{Z}}_0(\omega)$

for each ω. Of course, the resulting sets may or may not be subsets of $\mathcal{Z}_0$ depending on the choice of samples used in the sample approximation.

Recalling the definition of $\mathcal{T}_\infty$, we can write

$$\mathcal{T}_\infty = \{x \in \mathbb{R}^n | \ \mathbb{P}\{x \in \hat{\mathcal{S}}_\mathcal{J}(\omega)\} \geq 1 - \epsilon\}$$

and, choosing a multisample $\delta = \{\omega^{(1)}, \omega^{(2)}, \ldots, \omega^{(N)}\}$, this expression suggests we can form sample approximations as

$$\hat{\mathcal{T}}_\infty(\delta) = \left\{x \in \mathbb{R}^n \middle| \ x \in \mathcal{S}_\mathcal{J}(\omega^{(j)}), \ \forall \omega^{(j)} \in \delta_a, \ \delta_a \subseteq \delta, \ |\delta_a| \geq r\right\}$$

for some appropriately chosen N and r. We can write this sample approximation as a union of polyhedra, by considering a union over all possible δ_a:

$$\hat{\mathcal{T}}_\infty(\delta) = \bigcup_{\{\delta_a \subseteq \delta: \ |\delta_a| \geq r\}} \left\{x \in \mathbb{R}^n \middle| \ x \in \mathcal{S}_\mathcal{J}(\omega^{(j)}), \ \forall \omega^{(j)} \in \delta_a\right\}$$

There is no guarantee that this set is convex, but fortunately, its convex hull is also a robustly invariant set, so that we can alternatively consider its convex hull if we would like to recover a polyhedron at the end of the process.

For all but small values of N, this is likely to be a union over many thousands or even millions of sets, as the number of possible δ_a is combinatorial. However, there is no need to consider them all if all we want is a robustly invariant set, so we advocate simply choosing some fixed number M of them at random. This leads to the following procedure.

Algorithm 7.3 Robustly invariant polyhedral sample approximations

1. For each $i = 1, \ldots, M$:
   (a) Choose $\delta^{(i)}$, a random subset of r samples from δ
2. Calculate the convex hull of $\bigcup_{i=1}^{M} \left\{x \in \mathbb{R}^n \middle| \ x \in \hat{\mathcal{S}}_\mathcal{J}(\omega^{(j)}), \ \forall \omega^{(j)} \in \delta^{(i)}\right\}$

Computation of the polytopes can be carried out by using existing software, such as the Multi Parametric Toolbox (MPT) [KGBM04]. For instance in the third version of MPT, step 2 can be carried out by converting each polyhedron to a vertex representation and then defining a new polyhedron from the resulting list of vertices. This is typically a computationally intensive process due to the need to compute vertex representations of polyhedra, but this computation need only be carried out offline (rather than online in MPC, for instance).

7.4 Design example

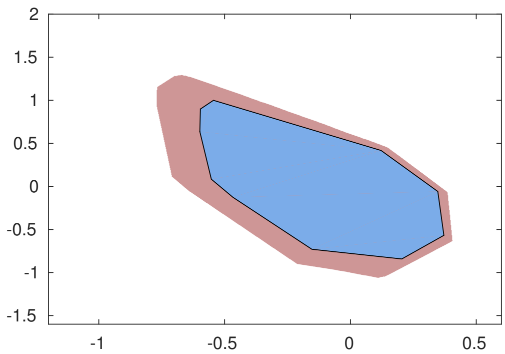

Figure 7.1: Robust vs. sample-admissible

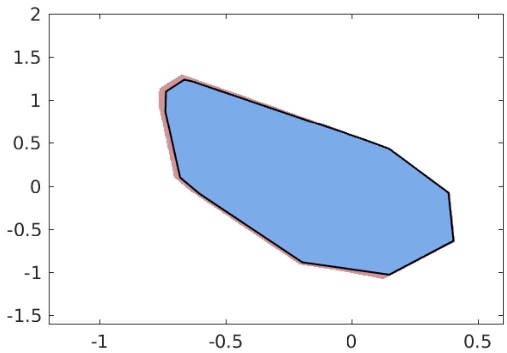

Figure 7.2: Polyhedral approximation

Consider dynamics given by $x_{t+1} = A(\omega)x_t$ with a random matrix defined as $A(\omega) = \sum_{i=1}^{3} A^{(i)}\omega_i$, with $\omega_i \in [0,1]$ uniformly distributed random variables for $i = 1,2,3$. Hence ω lies in the unit cube in $\mathbb{R}^3$, and A is contained in a corresponding polytope with 8 vertices. Define matrices $A^{(i)}$ and F as:

$$A^{(1)} = \begin{bmatrix} 0.37 & 0.21 \\ 0.75 & 0.65 \end{bmatrix}, \quad A^{(2)} = \begin{bmatrix} 0.32 & 0.03 \\ 1.36 & 0.75 \end{bmatrix}$$

$$A^{(3)} = \begin{bmatrix} 0.43 & -0.33 \\ 1.05 & 0.79 \end{bmatrix}, \quad F = \begin{bmatrix} 1 & 1 \\ -1 & 2 \\ 1.5 & -1 \\ -1.5 & 0 \end{bmatrix}$$

This system was subject to a one-step-ahead chance constraint $\mathbb{P}\{FA(\omega) \leq 1\} \geq 0.9$. The maximal sample-admissible set for this chance constraint was computed using Algorithm 7.2, yielding a multi-index set $\mathcal{J}$ with 10 entries. The chance constraint defining this maximal sample-admissible set was plotted by gridding over the state space and evaluating the probability at each point by sampling. The result is shown in Figure 7.1 with the robust MAS for comparison purposes, with the robust MAS shown in blue and the maximal sample-admissible set in red. The maximum sample-admissible set is 24.6% larger in this case.

Figure 7.2 shows a robustly invariant set computed using Algorithm 7.3 in blue. In red, this figure also shows the set defined by the constraints

$$\mathbb{P}\{FA(\omega)\Phi_M(x) \leq 1\} \geq 0.9, \quad \forall M \in \mathcal{J}$$

and since these are a subset of the constraints defining the MAS of the system, this is an outer approximation to the MAS. This outer approximation is indistinguishable from the maximum sample-admissible set shown in Figure 7.1 at this scale, demonstrating that in this case the difference in volume between the MAS $\mathcal{S}_\infty$ and the maximum sample-admissible set $\mathcal{T}_\infty$ is negligible. It is also clear that the polyhedral approximation is similar in volume to the MAS for this example. This polyhedral approximation was computed using Algorithm 7.3 using $N = 50$, $M = 50$ and $r = 45$ (that is, it is the convex hull of the union of 50 sets defined by satisfying 45 randomly chosen constraints from a multisample of 50). It has 13 vertices and 13 facets.

7.A Proofs

Proposition 7.4

Let $\mathcal{A}_\alpha$ be a family of admissible sets indexed by α. Then we have

$$\Phi\left(\bigcap_\alpha \mathcal{A}_\alpha\right) \subseteq \bigcap_\alpha \Phi(\mathcal{A}_\alpha) \subseteq \bigcap_\alpha \mathcal{A}_\alpha$$

and $\bigcap_\alpha \mathcal{A}_\alpha \subseteq \mathcal{Z}_0$. The proof for unions is similar as we can write

$$\Phi\left(\bigcup_\alpha \mathcal{A}_\alpha\right) = \bigcup_\alpha \Phi(\mathcal{A}_\alpha) \subseteq \bigcup_\alpha \mathcal{A}_\alpha$$

and of course $\bigcup_\alpha \mathcal{A}_\alpha \subseteq \mathcal{Z}_0$ once again.

Proposition 7.5

As the subset relation partially orders the class of admissible sets, there must exist at least one maximal element and at least one minimal element under this ordering. Because the class of admissible sets is closed under union and intersection, taking the union of all maximal elements and the intersection of all minimal elements gives two admissible sets that, respectively, contain or are contained within all other admissible sets. Uniqueness follows by noting that if there were two maxima (or minima) $\mathcal{A}$ and $\mathcal{B}$ then both $\mathcal{A} \subseteq \mathcal{B}$ and $\mathcal{B} \subseteq \mathcal{A}$ would have to hold by definition, so that $\mathcal{A} = \mathcal{B}$.

Proposition 7.7

We first prove that the given set is robustly invariant. Considering the definition of limit inferior, then taking the image and applying the distrubution properties of images over unions and intersections gives

$$\Phi(\mathcal{D}_\infty) = \Phi\left(\bigcup_{N=0}^{\infty} \bigcap_{k \geq N} \Phi^k(\{0\})\right) = \bigcup_{N=0}^{\infty} \Phi\left(\bigcap_{k \geq N} \Phi^k(\{0\})\right) \subseteq \bigcup_{N=0}^{\infty} \bigcap_{k \geq N} \Phi^{k+1}(\{0\})$$

and the set on the right is easily seen to be a subset of $\mathcal{X}_\infty$ as

$$\bigcup_{N=0}^{\infty} \bigcap_{k \geq N} \Phi^{k+1}(\{0\}) = \bigcup_{N=1}^{\infty} \bigcap_{k \geq N} \Phi^{k}(\{0\}) \subseteq \bigcup_{N=0}^{\infty} \bigcap_{k \geq N} \Phi^{k}(\{0\})$$

so that we have proven the robust invariance property $\Phi(\mathcal{X}_\infty) \subseteq \mathcal{X}_\infty$. Since $\mathcal{X}_\infty \subseteq \mathcal{Z}_0$, it is an admissible set. If $\mathcal{A}$ is any admissible set, then by Assumption 7.3 it must contain $\mathcal{X}_\infty$ as otherwise the robust invariance condition $\Phi\mathcal{A} \subseteq \mathcal{A}$ would be contradicted. Hence $\mathcal{X}_\infty$ is the minimum admissible set.

Theorem 7.8

We first calculate $\liminf_{k\to\infty} \mathcal{S}_k$ and $\limsup_{k\to\infty} \mathcal{S}_k$:

$$\liminf_{k\to\infty} \mathcal{S}_k = \bigcup_{N=0}^{\infty} \bigcap_{k \geq N} \bigcap_{j=0}^{k} \mathcal{Z}_j = \bigcup_{N=0}^{\infty} \bigcap_{j=0}^{\infty} \mathcal{Z}_j = \bigcap_{j=0}^{\infty} \mathcal{Z}_j = \mathcal{S}_\infty$$

$$\limsup_{k\to\infty} \mathcal{S}_k = \bigcap_{N=0}^{\infty} \bigcup_{k \geq N} \bigcap_{j=0}^{k} \mathcal{Z}_j = \bigcap_{N=0}^{\infty} \bigcap_{j=0}^{N} \mathcal{Z}_j = \bigcap_{j=0}^{\infty} \mathcal{Z}_j = \mathcal{S}_\infty$$

This proves the existence of the limit. To prove admissibility, we use the distribution property of images over intersections, so that the image of $\mathcal{S}_\infty$ under the mapping Φ is

$$\Phi(\mathcal{S}_\infty) = \Phi\left(\bigcap_{k=0}^{\infty} \mathcal{Z}_k\right) \subseteq \bigcap_{k=0}^{\infty} \Phi(\mathcal{Z}_k) \subseteq \bigcap_{k=-1}^{\infty} \mathcal{Z}_k \subseteq \mathcal{S}_\infty$$

giving $\Phi(\mathcal{S}_\infty) \subseteq \mathcal{S}_\infty$. Because $\mathcal{S}_\infty \subseteq \mathcal{Z}_0$ also, it is admissible.

To prove maximality, consider any admissible set $\mathcal{A}$. By definition $\Phi(\mathcal{A}) \subseteq \mathcal{A}$ and $\mathcal{A} \subseteq \mathcal{Z}_0$, implying by induction that $\Phi^k(\mathcal{A}) \subseteq \mathcal{Z}_0$ for $k = 1, 2, \ldots$. Taking the preimage under Φ of both sides of this relation k times yields $\mathcal{A} \subseteq \Phi^{-k}(\mathcal{Z}_0) = \mathcal{Z}_k$ for all $k = 1, 2, \ldots$. Hence $\mathcal{A} \subseteq \mathcal{S}_\infty$ for any admissible $\mathcal{A}$ and therefore $\mathcal{S}_\infty$ is the maximum such set.

Proposition 7.9

Taking the preimage of a convex set under an affine mapping preserves convexity [BV04, Section 2.3.2], and $\mathcal{Z}_k$ is an intersection of such preimages for nonnegative k. Hence if $\mathcal{Z}_0$ is a convex set then so is $\mathcal{Z}_k$ for any $k > 0$. The definitions of $\mathcal{S}_k$ and $\mathcal{S}_\infty$ then imply that they are intersections of convex sets, therefore they are also convex.

If $\mathcal{Z}_0$ is compact, then $\mathcal{S}_k$ is bounded due to the relation $\mathcal{S}_k \subseteq \mathcal{Z}_0$. Because $\mathcal{Z}_0$ is closed, upper semicontinuity of Φ implies that the set $\mathcal{Z}_k$ is also closed for any k. Hence $\mathcal{S}_k$ is closed for any k as it is an intersection of closed sets. Because it is closed and bounded, it is compact.

Theorem 7.10

Since $\mathcal{S}_k$ is compact and $\mathcal{X}_\infty$ is in the interior of $\mathcal{Z}_0$, Assumption 7.3 implies that there must exist a finite integer $K \geq k$ such that

$$\Phi^{K+1}(\mathcal{S}_k) \subseteq \mathcal{Z}_0.$$

Taking the pre-image $K+1$ times gives

$$\mathcal{S}_k \subseteq \Phi^{-K-1}(\mathcal{Z}_0) = \mathcal{Z}_{K+1}$$

and since $K \geq k$ implies $\mathcal{S}_K \subseteq \mathcal{S}_k$, we therefore have $\mathcal{S}_K \subseteq \mathcal{Z}_{K+1}$. Furthermore $\mathcal{S}_{K+1} = \Phi^{-1}(\mathcal{S}_K) \cap \mathcal{Z}_0$ by definition, so $\mathcal{S}_K \subseteq \mathcal{Z}_{K+1}$ implies $\mathcal{S}_{K+1} \subseteq \Phi^{-1}(\mathcal{Z}_{K+1}) = \mathcal{Z}_{K+2}$ and hence by induction we obtain

$$\mathcal{S}_N \subseteq \mathcal{Z}_{N+1} \quad \forall N \geq K. \tag{7.14}$$

Finally, by definition we also have that $\mathcal{S}_{N+1} = \mathcal{S}_N \cap \mathcal{Z}_{N+1}$, so (7.14) implies $\mathcal{S}_{N+1} = \mathcal{S}_N$ for all $N \geq K$, and hence $\mathcal{S}_N = \mathcal{S}_\infty$ for all $N \geq K$ by induction.

Proposition 7.14

Let $\mathcal{A}_\alpha$ be a family of sample-admissible sets. Taking the meet implies that we are taking intersections pointwise on the sample space, so that for any $\omega \in \Omega$ we have

$$\Phi\left(\bigcap_\alpha \mathcal{A}_\alpha(\omega)\right) \subseteq \bigcap_\alpha \Phi(\mathcal{A}_\alpha(\omega)) \subseteq \bigcap_\alpha \mathcal{A}_\alpha(\omega)$$

and $\bigcap_\alpha \mathcal{A}_\alpha(\omega) \subseteq \hat{\mathcal{Z}}_0(\omega)$, so that the set given by the meet $\bigwedge_\alpha \mathcal{A}_\alpha$ is sample-admissible. For the join we consider unions on the sample space

$$\Phi\left(\bigcup_\alpha \mathcal{A}_\alpha(\omega)\right) = \bigcup_\alpha \Phi(\mathcal{A}_\alpha(\omega)) \subseteq \bigcup_\alpha \mathcal{A}_\alpha(\omega)$$

and $\bigcup_\alpha \mathcal{A}_\alpha(\omega) \subseteq \hat{\mathcal{Z}}_0(\omega)$. As these relations hold for all $\omega \in \Omega$, $\bigvee_\alpha \mathcal{A}_\alpha$ is sample-admissible.

Proposition 7.15

The subset relation of inclusion partially orders the sample-admissible sets, so there must exist a nonempty family of maximal elements $\mathcal{A}_\alpha$ and a nonempty family of minimal elements $\mathcal{B}_\alpha$ under this ordering. Taking the join $\bigvee_\alpha \mathcal{A}_\alpha$ and the meet $\bigwedge_\alpha \mathcal{B}_\alpha$ now gives two sample-admissible sets that, respectively, contain or is contained within all other sample-admissible sets. Uniqueness follows by noting that if there are two maxima (or minima) $\mathcal{A}_1$ and $\mathcal{A}_2$ then both $\mathcal{A}_1 \subseteq \mathcal{A}_2$ and $\mathcal{A}_2 \subseteq \mathcal{A}_1$ hold, so that $\mathcal{A}_1 = \mathcal{A}_2$.

Theorem 7.16

Theorem 7.8 implies that $\mathcal{T}_\infty$ is a sample-admissible set, so we need only prove maximality. Suppose that $\mathcal{A}$ is another sample-admissible set. For all $\omega \in \Omega$ we have $\hat{\mathcal{A}}(\omega) \subseteq \hat{\mathcal{S}}_\infty(\omega)$ since the latter set is maximal. As unions and intersections preserve subset relations this implies

$$\bigcup_{E\in\mathcal{C}} \bigcap_{\omega\in E} \hat{\mathcal{A}}(\omega) \subseteq \bigcup_{E\in\mathcal{C}} \bigcap_{\omega\in E} \hat{\mathcal{S}}_\infty(\omega)$$

so that $\mathcal{A} \subseteq \mathcal{T}_\infty$ as required.

8 Tube MPC of stochastic LDIs

8.1 Introduction

The tube MPC of Chapter 5 is now extended to cover stochastic systems, in which it is desired to minimise a bound on an expected-value cost and the predicted state of the system may be subject to chance constraints. Such constraints are important in applications where it may be too stringent to specify that a constraint must be satisfied for all possible uncertainties (that is, robustly) such as building climate control [OPJ+10] or in the control of structures subject to fatigue constraints [ECK15].

Unfortunately, despite their widespread applicability, handling chance constraints in MPC is difficult as such constraints rarely have a convenient form that can be exploited by a standard optimisation solver. Worse, in many cases they may not even be convex. For these reasons, much recent research work has gone into developing MPC controllers that handle chance constraints by online sampling techniques. These often exploit recent results from the stochastic programming literature (such as those found in [Cal10, CG11]) on sample approximations to give a bound on the probability that the original chance constraints are satisfied, if a given number of samples are used in the sample approximation. These methods have the advantage that the sampled constraints are typically convex.

Examples of the application of such techniques to MPC of system with parameter uncertainty can be found in [SFFM14] for instance. Unfortunately, due to the nature of the sampling process, and the necessity of resampling at each time for a confidence result to hold, there are typically no guarantees of recursive feasibility for such controllers. This could be a serious drawback of these methods, as the confidence results only give a bound on the probability of constraint satisfaction if the current optimisation is feasible (in [SFFM14], feasibility at every time is taken as an assumption, for instance) and say nothing about the likelihood of infeasibility.

In this chapter, we show that it is possible to apply chance constraints to MPC problems using a prediction tube, and crucially that the resulting optimisations can be handled by sampling in such a way that recursive feasibility is retained. This means that it is possible to give a strict guarantee on satisfaction of the sampled constraints. It also means that the usual closed-loop stability guarantees can apply. In addition, if the chance constraint is only applied at the first prediction time, and robust constraints applied thereafter, then confidence results analogous to those in [Cal10, CG11] will apply. In this case it is also possible to handle coloured noise in the uncertain parameter by drawing samples from conditional distributions.

As well as having these strong theoretical properties, the algorithm that we present in this chapter is quite efficient in that it typically only requires around $3-6$ times the computation time of an equivalent robust MPC controller. This is achieved by using a fast algorithm for sample removal that relies on a heuristic of sorting slack variables from a convex optimisation problem.

8.2 Problem Formulation

Once again we consider a Linear Differential Inclusion (LDI) system

$$x_{t+1} = A(\omega_t)x_t + B(\omega_t)u_t + w(\omega_t) \tag{8.1a}$$

$$y_t = C(\omega_t)x_t + D(\omega_t)u_t \tag{8.1b}$$

$$z_t = F(\omega_t)x_t + v(\omega_t) \tag{8.1c}$$

where the system matrices are assumed to be affine functions of a random variable $\omega \in \Omega$ so that, for instance,

$$(A(\omega), B(\omega), w(\omega)) = (\bar{A}, \bar{B}, 0) + \sum_{i=1}^{r}(A^{[i]}, B^{[i]}, w^{[i]})\omega_i$$

where the random variable lies within a compact set. Unlike in Chapter 5, we will make use of the probability distribution of the unknown parameter ω to improve performance of the controller.

Assumption 8.1. *The set Ω is compact and convex.*

We make the simplifying assumption that ω_t is independent at each time and has identical distribution for all t. When evaluating a cost function for MPC, we will require knowledge of its covariance matrix, so we also assume that this is available.

Assumption 8.2. *The random variables ω_t are independent and identically distributed, with mean zero and known variance.*

Remark 8.1. Coloured noise in the additive term w can usually be handled with a state augmentation, so that the above assumption only restricts the uncertainty in the system matrices to be independent at each time.

The control objective will be to minimise a bound on the expected-value l_2 norm of the output y_t, that is:

$$\sum_{t=0}^{\infty} \mathbb{E}\|y_t\|^2$$

As before, the output z_t is subject to a constraint and we seek to develop a prediction tube to handle that constraint in an online MPC optimisation. However the constraint is now probabilistic in nature:

$$\mathbb{P}_t\{z_k \leq 1\} \geq 1 - \varepsilon \tag{8.2}$$

Here, $\mathbb{P}_t$ is the conditional probability at time t, the time at which the MPC controller is forming predictions.

Remark 8.2. The constraint (8.2) must be satisfied robustly for all possible evolutions of the system, so that we seek to robustly satisfy this probabilistic constraint. It is possible to consider probabilistic constraints on the uncertain state of the system by encoding these as a one-step-ahead type constraint in 8.2, for instance by choosing $C(\omega) = FA(\omega)$, $D(\omega) = GB(\omega)$ and $v_k = Fw_k$. This approach allows for strong guarantees of recursive feasibility for systems with compact uncertainty sets.

As in Chapter 5, we will assume that the system is quadratically stabilisable by a feedback matrix K.

Assumption 8.3. *The state dynamics (8.1a) are quadratically stabilised by the feedback $u_t = Kx_t$.*

Remark 8.3. [BEGFB94] contains specialised methods for designing state feedbacks for stochastic systems in chapter 9. Alternatively, any of the robust techniques mentioned in the earlier chapters of the thesis can also be used to find a suitable K.

This allows us to parametrise the predicted system inputs as:

$$u^k = Kx^k + c^k \tag{8.3}$$

so that once again we are considering perturbations on a fixed state feedback. The following assumption on observability and positive definiteness ensures that the cost functions we will use in MPC optimisations are positive definite in both x and u. This is required to prove stability of the presented MPC controllers in closed-loop.

Assumption 8.4. *There exists $S \subseteq \Omega$ with $\mathbb{P}(S) > 0$ such that, on S, $(C(\omega), A(\omega))$ is observable and $D(\omega)^T D(\omega)$ is positive definite.*

Remark 8.4. Assumption 8.4 could be equivalently stated as "(C, A) is an observable pair with probability greater than 0". The extra precision in specifying S is simply to help in the proofs at the end of the chapter.

8.3 Cost function evaluation

As with the robust MPC for LDI systems, we introduce a vector of input predictions $\underline{c}_t$, and its 'tail' $\tilde{c}_{t+1}$ as

$$\underline{c}_t = \begin{bmatrix} c_t^t \\ c_t^{t+1} \\ \vdots \\ c_t^{t+N-1} \end{bmatrix}, \quad \tilde{c}_{t+1} = \begin{bmatrix} c_t^{t+1} \\ \vdots \\ c_t^{t+N-1} \\ 0 \end{bmatrix}$$

and define matrices T and S such that $c_t^t = S\underline{c}_t$ and $\tilde{c}_{t+1} = T\underline{c}_t$. The predicted state dynamics (8.1a) can then be expressed in the familiar autonomous form

$$\begin{bmatrix} x^{k+1} \\ \underline{c}^{k+1} \\ 1 \end{bmatrix} = \begin{bmatrix} \tilde{A}(\omega^k) & B(\omega^k)S & w(\omega^k) \\ 0 & T & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x^k \\ \underline{c}^k \\ 1 \end{bmatrix} \tag{8.4a}$$

$$y^k = \begin{bmatrix} \tilde{C}(\omega^k) & D(\omega^k)S & 0 \end{bmatrix} \begin{bmatrix} x^k \\ \underline{c}^k \\ 1 \end{bmatrix} \tag{8.4b}$$

which we write more simply as $\zeta^{k+1} = \Psi(\omega^t)\zeta^t$ and $y^t = \Lambda(\omega^t)\zeta^t$, where the augmented state vector ζ^k and matrices $\Psi(\omega)$, $\Lambda(\omega)$ are defined by comparison with (8.4).

8.3.1 Satisfying an l_2-gain bound

If the l_2 norm of the discrete time stochastic process y_t is defined as

$$\sum_{k=0}^{\infty} \mathbb{E}\|y_k\|^2$$

then it is possible to bound this norm and give bounds on the induced l_2-gain of the system, using similar methods to those we used in the robust MPC case.

Proposition 8.5. *If there exists $W \succ 0$ satisfying the Lyapunov inequality*

$$W - \mathbb{E}[\Psi(\omega)^T W \Psi(\omega)] \succeq \mathbb{E}[\Lambda(\omega)^T \Lambda(\omega)] - \gamma^2 \mathbb{E}[\mu(\omega)\mu(\omega)^T] \tag{8.5}$$

where $\mu(\omega) = [0\ \ 0\ \ w(\omega)]^T$, then the conditional expectation of the l_2 norm of the predicted output y_t can be bounded as

$$\sum_{k=t}^{\infty} \mathbb{E}_t\|y^k\|^2 \leq \zeta_t^T W \zeta_t + \gamma^2 \sum_{k=t}^{\infty} \mathbb{E}_t\|w^k\|^2 \tag{8.6}$$

for all $t = 0, 1, \ldots, \infty$.

Remark 8.6. To solve the inequality (8.5) we must make use of the fact that the system matrices are affine in ω. If we have, for example,

$$(\Psi(\omega), \Lambda(\omega), \mu(\omega)) = (\bar{\Psi}, \bar{\Lambda}, 0) + \sum_{i=1}^{r} (\Psi^{[i]}, \Lambda^{[i]}, v^{[i]})\omega_i$$

then we may rewrite (8.5) as

$$W - \bar{\Psi}^T W \bar{\Psi} - \sum_{i,j} (\Psi^{[i]})^T W \Psi^{[j]} \mathbb{E}[\omega_i \omega_j] \succeq \bar{\Lambda}^T \bar{\Lambda} + \sum_{i,j} (\Lambda^{[i]})^T \Lambda^{[i]} \mathbb{E}[\omega_i \omega_j] - \gamma^2 \sum_{i,j} v^{[i]} v^{[i]} \mathbb{E}[\omega_i \omega_j] \quad (8.7)$$

which is a linear matrix inequality in W. As usual, we can minimise the trace of W subject to this LMI to get a tight bound.

8.4 Constraint handling using tubes

As in the earlier parts of the thesis dealing with robust tube MPC, the constraint handling will use a prediction tube to bound the future state of the system. To accomplish this, we will first introduce an extension to the results of Chapter 4 to allow for set inclusions involving chance constraints.

8.4.1 Chance-constrained inclusions

For convenience, we introduce a notation for the set defined by a chance constraint, analogous to $\mathcal{P}[\cdot,\cdot]$ for a polyhedron and $\mathcal{R}_\Theta[\cdot,\cdot]$ for a robust constraint. Define

$$\mathcal{C}[F(\omega), b(\omega), \varepsilon] = \{x \in \mathbb{R}^n |\ \mathbb{P}\{F(\omega)x \le b(\omega)\} \ge 1 - \varepsilon\}$$

where $V(\omega)$ and $g(\omega)$ are affine functions of the random variable ω.

Lemma 8.7. *The chance constraint set $\mathcal{C}[F(\omega), b(\omega), \varepsilon]$ can be expressed as*

$$\mathcal{C}[F(\omega), b(\omega), \varepsilon] = \bigcup_{\mathcal{E} \in \mathcal{M}} \bigcap_{\omega \in \mathcal{E}} \mathcal{P}[F(\omega), b(\omega)]$$

where $\mathcal{M} = \{\mathcal{E} \subseteq \Omega |\ \mathbb{P}(\mathcal{E}) \ge 1 - \varepsilon\}$, the class of all subsets of Ω of $\mathbb{P}$-measure at least $1 - \varepsilon$.

We can now give a necessary and sufficient condition for $\mathcal{P}[V,g] \subseteq \mathcal{C}[F(\omega), b(\omega), \varepsilon]$ involving only linear and chance constraints.

Lemma 8.8. *$\mathcal{P}[V,g] \subseteq \mathcal{C}[F(\omega), b(\omega), \varepsilon]$ if and only if there exists a function $H(\omega)$ such that:*

$$H(\omega) \geq 0, \;\; H(\omega)V = F(\omega), \;\; \mathbb{P}\{H(\omega)g \leq b(\omega)\} \geq 1 - \varepsilon$$

This result is insightful but as with Lemma 4.3, which is its counterpart for robust inclusions, not very useful without knowing the specific form of the function $H(\omega)$. This is no hindrance here, however, as Theorem 4.4 from Chapter 4 holds without any modification, meaning that once again we can exploit affine parameterisations of the uncertainty. To this end we (denoting the *ith* element of ω as ω_i) assume that we have:

$$(F(\omega), b(\omega)) = (\bar{F}, \bar{b}) + \sum_{i=1}^{r} (F^{[i]}, b^{[i]})\omega_i \tag{8.8}$$

Applying Theorem 4.4 to the conditions of Lemma 8.8 leads directly to the next result.

Proposition 8.9. *If the parameterisation (8.8) holds then $\mathcal{P}[V,g] \subseteq \mathcal{C}[F(\omega), b(\omega), \varepsilon]$ if and only if there exist $\bar{H}$ and $H^{[i]}$ satisfying*

$$\mathbb{P}\left\{\left(\bar{H} + \sum_{i=1}^{q} H^{[i]}\theta_i\right) g^{(1)} \leq \bar{g} + \sum_{i=1}^{q} g^{[i]}\theta_i\right\} \geq 1 - \varepsilon$$

where

$$\begin{aligned} &\bar{H} \geq 0, \;\; \bar{H}V = \bar{V} \\ &H^{[i]} \geq 0, \;\; (H^{[i]} - \bar{H})V = V^{[i]} \end{aligned}$$

Proposition 8.9 says that the decision problem "Is $\mathcal{P}[V,g] \subseteq \mathcal{C}[F(\omega), b(\omega), \varepsilon]$?" can be expressed as a chance-constrained optimisation. If $H^{[i]} = 0$ for all i, then this can be recast as a linear program. If the uncertainty is normally-distributed, it is equivalent to a second-order cone program. Otherwise, this is likely to be intractable in most cases; indeed, evaluating a function like $\mathbb{P}\{(\bar{f}^T + \omega^T)x \leq 1\}$ is known to be NP-hard even in the case when ω is uniformly distributed in a box. However various conservative methods for finding a feasible point are available that will lead to sufficient (and not necessary) conditions for the set inclusion, such as solving sample approximations. This is the approach we take when applying these results to tube MPC.

8.4.2 Applying the chance constraint

Once again, we will introduce ***tube parameterisation functions*** $g_t^k(\alpha)$ that will specify tube cross-sections containing the predicted system state. We make the usual assumptions to ensure that the tube parameterisation is well defined and to ensure recursive feasibility. The explanation and background to these assumptions can be found in Sections 5.4.2 and 5.4.4 if required.

Assumption 8.5. *The functions g_t^k are affine in α.*

Assumption 8.6. *The matrix V and the functions g_t^k are chosen so that, for all t and k and for all $x \in \mathbb{R}^n$ there exists α such that $x \in \mathcal{P}[V, g_t^k(\alpha)]$.*

Assumption 8.7. *There exists α^N such that $\{x \in \mathbb{R}^n | \ Vx \leq g^N(\alpha^N)\}$ is positively invariant under the dynamics (8.1a) with $c^k = 0$.*

Assumption 8.8. *The tube parameterisation is t-invariant, that is we have $g_t^{t+j} = g_0^j$ for all $j = 1, \dots, N$.*

Assumption 8.9. *The tube parameterisation is $k-1$-reproducible, that is for any α^k there exists α^{k-1} such that $g^{k-1}(\alpha^{k-1}) = g^k(\alpha^k)$ for all $k = t+1, \dots, N$.*

Remark 8.10. Assumption 8.7 has implications for the design of V, as for the robust tube MPC (see Section 4.4.1). However, if it is possible to construct an invariant approximation to the maximum sample admissible set using Algorithm 7.3 from the previous chapter, then this can be used to define V rather than Algorithm 4.3. This gives the advantage of a larger terminal set for the resulting tube MPC that takes into account the stochastic nature of the constraints, and may increase the feasible region considerably.

To this framework we must add constraints on α that ensure satisfaction of the chance constraints (8.2). The following is a direct analog of Proposition 5.9 dealing with chance constraints. That is, it is a necessary and sufficient condition for the inclusion

$$\mathcal{P}[V, g^k(\alpha^k)] \subseteq \mathcal{C}[F(\omega^k), 1 - v(\omega^k), \varepsilon] \tag{8.9}$$

that provides an inequality constraint on the tube parameter α.

Proposition 8.11. *If, and only if, there exists a matrix function $\Delta(\omega) \geq 0$ such that*

$$\Delta(\omega)V = F(\omega) \tag{8.10a}$$

$$\mathbb{P}\{\Delta(\omega)g^k(\alpha) + v(\omega) \leq 1\} \geq 1 - \varepsilon \tag{8.10b}$$

for all k then (8.9) holds for all k.

As in the formulation of a tube for robust MPC, whether or not this constraint can be applied online will depend greatly on the form of the uncertainty (in this case, its probability distribution). Later in the chapter, we will consider sample approximation techniques to give a set of linear constraints corresponding to samples of ω.

8.4.3 Chance-constrained tubes

It is now possible to give a full set of constraints defining a tube for the system with the probabilistic constraints. Unsurprisingly they are quite similar to those used for the robust tube, except that they contain a chance constraint given by Proposition 8.11. Handling the chance constraints in the online optimisation is likely to be computationally intensive, so we introduce a parameter M so that chance constraints are only considered up to time $t + M$, and then robust constraints are used for the rest of the prediction horizon. This provides another design parameter that can be used to improve performance and yields the constraints:

$$Vx^t \leq g^t(\alpha^t) \tag{8.11a}$$

$$\mathbb{P}\{\Delta(\omega)g^k(\alpha^k) + v(\omega) \leq 1\} \geq 1 - \varepsilon, \quad k = t \ldots t + M - 1 \tag{8.11b}$$

$$\Delta(\omega)g^k(\alpha^k) + v(\omega) \leq 1, \quad \forall \omega \in \Omega, \;\; k = t + M \ldots t + N - 1 \tag{8.11c}$$

$$\Delta(\omega)g^{t+N}(\alpha^{t+N}) \leq 1, \quad \forall \omega \in \Omega \tag{8.11d}$$

$$\Gamma(\omega)g^k(\alpha^k) + VB(\omega)c^k + Vw(\omega) \leq g^{k+1}(\alpha^{k+1}), \quad \forall \omega \in \Omega \tag{8.11e}$$

$$\Gamma(\omega)g^{t+N}(\alpha^{t+N}) + Vw(\omega) \leq g^{t+N}(\alpha^{t+N}), \quad \forall \omega \in \Omega \tag{8.11f}$$

The assumptions we made on the reproducibility properties of the tube parameterisation (i.e. Assumptions 8.7–8.9) immediately lead us to a recursive feasibility result for these constraints.

Theorem 8.12. *The tube constraints (8.11) are strongly recursively feasible for the system (8.7) under the control policy (8.9).*

Remark 8.13. Handling these chance constraints in an online optimisation is nontrivial and will generally require either sampling or a conservative bound of some kind. In the event that the random variables are normal and $\Delta(\omega)$ contains only a single row, the chance constraint can be represented exactly by a second order cone constraint, for instance. If the uncertainty is only additive (in $v(\omega)$) then it is possible to integrate over the uncertainty distribution offline to represent the constraint as a linear one, as in [KCRC10]. Joint constraints can be conservatively approximated using Boole's inequality (simply, a value ε_i is assigned to each individual chance constraint and their sum must not exceed ε).

The constraints (8.11) give constraints that could in principle be used in an online optimisation to give a recursively feasible MPC controller. Unfortunately, the chance constraints involved are difficult to handle and may not even be convex in general. For this reason, we now consider using sampling methods to approximately represent the chance constraints by considering linear constraints.

8.4.4 Sampled constraints

Sampling provides a very general method to handle chance constraints. In this section we show that it is possible to formulate a recursively feasible tube using sampled constraints. We assume that samples $\omega^{\{k,i\}}$ are drawn from the distribution of ω, and that constraints are applied for some subset of these samples $\mathcal{I}^k \subseteq \{1, 2, \ldots, N_s\}$ at each time k. These index sets of 'active' constraints must contain at least r elements, leading to a constraint

$$\Delta(\omega^{\{k,i\}})g^k(\alpha^k) + v(\omega^{\{k,i\}}) \leq 1, \;\; \forall i \in \mathcal{I}^k \tag{8.12}$$

but where the index set $\mathcal{I}^k$ itself can be considered as a decision variable that must satisfy a constraint $|\mathcal{I}^k| \geq r$.

Later in the chapter, fast approximate solution methods for these sampled MPC problem will be developed. For these algorithms, it will be beneficial to have a form of the constraints that includes slack variables corresponding to each sampled constraint. Hence we rewrite the sampled constraint (8.12) as

$$\Delta(\omega^{\{k,i\}})g^k(\alpha^k) + v(\omega^{\{k,i\}}) + s^{\{k,i\}} = 1 \tag{8.13}$$

with a condition on the slacks

$$s^{\{k,i\}} \geq 0, \;\; \forall i \in \mathcal{I}^k, \;\; |\mathcal{I}^k| \geq r \tag{8.14}$$

This will form the basis of a heuristic method of constraint removal aimed at quickly finding a suboptimal solution to a sampled optimisation problem, the idea being to remove the constraints with the smallest slacks. With these slack variables $s_k^{\{k,i\}}$ the tube constraints (8.11) take the form:

$$V x^t \leq g^t(\alpha^t) \tag{8.15a}$$

$$\Delta(\omega^{\{k,i\}})g^k(\alpha^k) + v(\omega^{\{k,i\}}) + s^{\{k,i\}} = 1, \quad k = t \dots t+M-1 \tag{8.15b}$$

$$\Delta(\omega)g^k(\alpha^k) + v(\omega) \leq 1, \quad \forall \omega \in \Omega, \;\; k = t+M \dots t+N-1 \tag{8.15c}$$

$$\Delta(\omega)g^{t+N}(\alpha^{t+N}) + v(\omega) \leq 1, \quad \forall \omega \in \Omega \tag{8.15d}$$

$$\Gamma(\omega)g^k(\alpha^k) + VB(\omega)c^k + Vw(\omega) \leq g^{k+1}(\alpha^{k+1}), \quad \forall \omega \in \Omega \tag{8.15e}$$

$$\Gamma(\omega)g^{t+N}(\alpha^{t+N}) + Vw(\omega) \leq g^{t+N}(\alpha^{t+N}), \quad \forall \omega \in \Omega \tag{8.15f}$$

$$s^{\{k,i\}} \geq 0, \;\; \forall i \in \mathcal{I}^k, \;\; |\mathcal{I}^k| \geq r \tag{8.15g}$$

As in the robust case, we introduce a simpler notation for the feasible region of these constraints as $(\underline{c}_t, \underline{\alpha}_t, \underline{s}_t) \in \mathcal{T}_M(x_t, \mathcal{I})$, where for simplicity we have introduced the set of index sets $\mathcal{I} = \{\mathcal{I}^t, \mathcal{I}^{t+1}, \dots, \mathcal{I}^{t+M-1}\}$ and a vector $\underline{s}$ containing all slacks. Hence we have:

$$\mathcal{T}_M(X, \mathcal{I}) = \{(\underline{c}, \alpha, \underline{s}) | \;\; c_k, \, \alpha \text{ and } \underline{s} \text{ satisfy (8.15) with the given } x^t = x_t, \, \mathcal{I}\}$$

We have not specified any particular order for the slack variables in $\underline{s}$, but we will not need to when stating the algorithm as we will work directly with the individual slacks $s^{\{k,i\}}$, and the above is simply a notational convenience.

As ever, we require that the constraints (8.15) should be recursively feasible. The sampling and dependence on the index sets $\mathcal{I}^k$ makes this a little more complex than it was for robust MPC. Fortunately, it will be enough to show that these constraints are recursively feasible for fixed values of the index sets, in other words when we have $\mathcal{I}^k_{t+1} = \mathcal{I}^k_t$ for all $k = t, \ldots, t+M-1$ (using our usual convention that the superscript is prediction time, and the subscript is the actual time at which we carry out the optimisation). For this to be possible, we require that samples are reused between time steps, and we make this explicit with the following assumption.

Assumption 8.10. *The samples used are fixed for each prediction time k in the sense that $\omega_t^{\{k,i\}} = \omega_{t+1}^{\{k,i\}}$ for all t, k and i.*

In practice, the above assumption means that new samples can only be introduced at prediction time $t+M-1$, and thereafter must be reused until the current time passes that point (when they can be discarded as they are no longer needed). Introducing fresh samples in this manner is recommended to ensure that in closed-loop the effect of any particular 'bad' set of samples chosen cannot persist forever.

8.5 Algorithm and stability

Now that we have a bound on output l_2 norm from the quadratic function in Proposition 8.5, along with the sampled constraints (8.15), we can state MPC optimisations. In principle we would like to minimise $\zeta^T W \zeta$ where W satisfies (8.5), subject to the constraint $(\underline{c}_t, \underline{\alpha}_t, \underline{s}_t) \in \mathcal{T}_M(x_t, \mathcal{I})$. This leads to a notional MPC algorithm when $\mathcal{I}$ is considered fixed.

The immediate problem with using Algorithm 8.2 to solve the MPC problem is that the index set $\mathcal{I}$ is fixed in this optimisation, and if we try to optimise over $\mathcal{I}$ as well we are left with a difficult combinatorial problem. It is possible to recast this optimisation as a binary mixed-integer program with a convex continuous relaxation, that can be solved exactly by a branch and bound algorithm [FCK14]. But this is quite inefficient

Algorithm 8.1 Offline, Stochastic sampled MPC

1. Find a stabilising K, for example by maximising trace(Q) subject to $Q \succeq 0$ and (5.5).
2. Find V such that $\mathcal{P}[V, \bar{g}]$ is robustly invariant for some $\bar{g}$, using Algorithm 4.3 or Algorithm 7.3.
3. Choose an affine function $g(\alpha)$ to parameterise a prediction tube, for example by using a formulation from Section 4.4.2.
4. Form matrix functions $\Gamma(\omega)$ and $\Delta(\omega)$ satisfying $\Gamma(\omega)V = V\tilde{A}(\omega)$ and $\Delta(\omega)V = \tilde{F}(\omega)$ by using Algorithm 4.2.
5. Find W by minimising trace(W) subject to $W \succ 0$ and (8.7).

Algorithm 8.2 Online, Stochastic sampled MPC

With W satisfying (8.5), solve:

$$\begin{array}{ll} \underset{\alpha_t, \underline{c}_t, \underline{s}_t}{\text{minimise}} & \zeta_t^T W \zeta_t \\ \text{subject to} & (\underline{c}_t, \underline{\alpha}_t, \underline{s}_t) \in \mathcal{T}_M(x_t, \mathcal{I}) \end{array}$$

as a solution method, as the solution time required for MIPs scales badly with the number of integer variables they contain (it is, after all, an NP-hard problem to check feasibility of mixed-integer constraints).

As the sampled tube constraints are an approximation to the chance constraints that we wish to satisfy, it seems counterproductive to expend computation time finding the optimal $\mathcal{I}$ when, at best, this provides only an approximate solution to the chance-constrained program we would like to solve. So we will be satisfied with a faster, suboptimal method of solving this sampled problem, if the obtained solutions are close to the optimal ones in practice.

8.5.1 Heuristic sample removal

In [CG11], a 'greedy' constraint removal algorithm is applied to quickly find a suboptimal feasible solution. The approach we advocate here can be seen as a modification of this method, using a heuristic based on slack variables in the constraint functions to rapidly identify samples that should be removed from $\mathcal{I}$ to improve the value of the cost function in Algorithm 8.2.

The procedure begins by setting $\mathcal{I}^k$ to be equal to the solutions found at the previous time step, except for the index set for prediction time $t + M - 1$ which is initialised as $\mathcal{I}^{t+M-1} = \{1, \ldots, N_s\}$, and samples are drawn from the distribution of ω for this prediction time. The MPC optimisation is run with these fixed $\mathcal{I}^k$, which is a convex optimisation. The slack variables are then sorted using an efficient algorithm such as quicksort, and the $N_s - r$ samples with the lowest values of the slacks at each time step k removed from $\mathcal{I}^k$. This can be repeated several times to iterate towards a suboptimal solution, and the procedure terminated when a previous choice of $\mathcal{I}$ is repeated.

Algorithm 8.3 Stochastic MPC, heuristic constraint removal

At each time t:

1. Choose samples ω_t^{t+M-1} and set $\mathcal{I}_t^{t+M-1} = \{1, \ldots, N_s\}$
2. Set $\mathcal{I}_t^k = \mathcal{I}_{t-1}^k$ and $\mathcal{J}^k = \{1, 2, \ldots, N_s\}$ for $k = t, \ldots, t + M - 2$,
3. Run Algorithm 8.2 to give c_t^*, $\underline{s}_t$,
4. For each $k = t, \ldots, t + N$:
   (a) Set $\mathcal{J}^k = \mathcal{J}^k \cap \{i| \; s_k^{\{i\}} \geq 0\}$,
   (b) Sort $\mathcal{J}^k$ in descending order of $\min(s_k^{\{i\}})$,
   (c) Set $\mathcal{L}^k = \mathcal{J}^k[1 \ldots r]$,
5. If $\mathcal{L}^k \neq \mathcal{I}^k$ for some k then set $\mathcal{I}^k = \mathcal{L}^k$ for all k and go to step 2,
6. Apply the control $u_t = Kx_t + c_t^*$ where c_t^* is the solution from step 2.

The stopping criterion in Algorithm 8.3 is to check if the same sample set is repeated. Clearly if this occurs, then there is little point repeating the optimisation as any further solutions will be identical to the current one (assuming that the convex optimisation is solved to optimality in step 2). On the other hand, it is not immediately obvious that this repetition should ever occur. We now claim that it does and that Algorithm 8.3 terminates in polynomial time with a solution no worse than the one found initially for $\mathcal{I}_t = \mathcal{I}_{t-1}$. Recall (from [Cal10], for example) the following definition of a support constraint in a convex optimisation.

Definition 8.1. *A constraint in a convex optimisation is known as a* **support constraint** *if removing it improves the optimal cost.*

It is an intuitive fact that step 4 of Algorithm 8.3 removes support constraints from the problem if it is possible to do so, so that at each iteration of the algorithm, either the cost decreases and the cardinality $|\mathcal{J}_k|$ decreases for some k or the solution will be identical to that obtained at the previous iteration and the algorithm will terminate. This implies a bound on the maximum number of possible iterations of the algorithm.

Theorem 8.14. *Algorithm 8.3 terminates after at most $M(N_s - r) + 1$ iterations and the value of ζ for the final solution satisfies:*

$$\zeta_t^T W \zeta_t^T \leq \tilde{\zeta}_t^T W \tilde{\zeta}_t$$

where $\tilde{\zeta}_t$ is the solution of Algorithm 8.2 with $\mathcal{I}_t^k = \mathcal{I}_{t-1}^k$ for all k (that is, the same index sets as the previous timestep). Furthermore, if the optimisation in step 2 is feasible the first time it is performed, it remains feasible forever.

In practice, the bound of $M(N_s - r) + 1$ iterations is typically very conservative. In the example that we will consider later in this chapter, this procedure typically converges to a good solution in around 3-6 iterations. The sorting procedure is very fast, so that the resulting algorithm has a solution time of about 3-6 times that required for the convex optimisation. This means that it produces a solution within an order of magnitude of the time required to solve the corresponding robust MPC problem.

Nonetheless it may be desired to provide an upper limit to the number of iterations that the algorithm will perform, since the computation time available to perform MPC will usually be upper bounded by the sample time of the system. In that case the termination condition in step 4 of Algorithm 8.3 can be modified so that it will stop once a maximum number of iterations is reached. This could be useful in applications where the time between samples is short, as the algorithm could even be stopped after a time limit. The condition in Theorem 8.14 that the cost cannot increase still holds in this instance, a fact that is crucial to proving stability of the closed-loop system for this controller.

8.5.2 l_2-stability

The recursive feasibility of Theorem 8.14 implies that the value of $\zeta_0^T W \zeta_0$ gives a bound on the output l_2 norm, similarly to the robust MPC of Chapter 5.

Theorem 8.15. *Under the control of Algorithm 8.3, the output of (8.1) satisfies the bound*

$$\sum_{k=0}^{t} \mathbb{E}\|y_k\|^2 \leq \zeta_0^T W \zeta_0 + \gamma^2 \sum_{k=0}^{t} \mathbb{E}\|w_k\|^2$$

for all t if the optimisation is initially feasible.

As in the deterministic case considered in Chapter 5, a simple consequence of Theorem 8.15 is an l_2 stability result. This is because we can take the limit $t \to \infty$ to find

$$\sum_{k=0}^{\infty} \mathbb{E}\|y_t\|^2 \leq \zeta_0^T W \zeta_0 + \gamma^2 \sum_{k=0}^{\infty} \mathbb{E}\|w_t\|^2$$

whenever these infinite sums are finite. This implies in particular that

$$\sum_{k=t}^{\infty} \mathbb{E}\|\Lambda(\omega) x_k\|^2 \to 0$$

as $t \to \infty$. The observability of Assumption 8.4 ensures that $\sum_{k=t}^{\infty} \|\Lambda(\omega) x^k\|^2$ is positive definite in x, so we can conclude that x_t converges to 0 in the sense of functions converging in $L_2(\Omega)$.

On the other hand, if w_t does not have finite l_2 norm, we can divide through by t before taking limits to find

$$\lim_{t \to \infty} \frac{1}{t} \sum_{k=0}^{t} \mathbb{E}\|y_t\|^2 \leq \gamma^2 \lim_{t \to \infty} \frac{1}{t} \sum_{k=0}^{t} \mathbb{E}\|w_t\|^2$$

so that the system scales this seminorm by at most a factor of γ^2, analogously to the behaviour of Algorithm 5.2 for robust MPC when the disturbance input is a finite power signal.

8.5.3 Confidence of chance constraint satisfaction

From recursive feasibility, it is possible to guarantee that at least r of the sampled constraints are satisfied for all time. However, the purpose of sampling is to approximately solve a chance-constrained MPC problem, so if possible we would like to relate the solution of the sampled problem to the chance-constrained one. For chance-constrained optimisations, several results are available that imply that solutions obtained from sampled programs satisfy the chance constraint with a high probability [Cal10, CG11]. These results were extended to multistage problems with multiple chance constraints in [SFM13] and applied in the context of MPC in [SFFM14], in which the expected number of constraint violations is shown to satisfy a bound in closed-loop. But this was achieved at the expense of resampling at each time step, making any recursive feasibility guarantee impossible.

In contrast, for recursive feasibility we have required reuse of the samples between times t in Assumption 8.10. Unfortunately, this means that the results of [Cal10, CG11, SFM13] do not apply here in general as it implies that the samples used at each time step are not independent if we are considering conditional probability distributions at time t. This follows from the dependence between the random variable x_t and the samples that were used in the optimisation at previous times. To add to the complexity, the state x_t in closed loop is not Markovian when samples are reused, so there is an additional distinction between conditioning on time t and conditioning on the state x_t that is not present in the case of resampling. This makes it difficult to claim anything meaningful about the probability of violating the chance constraint in closed-loop operation of the controller.

On the other hand, the situation becomes much simpler when $M = 1$, since then no sample reuse is needed and a new set of samples can be used at each time step. When $M = 1$, so that sampling is used only a single step ahead and the remaining constraints are applied robustly for the rest of the prediction horizon, the following confidence result follows immediately from the results in [SFM13].

Proposition 8.16. *If $M = 1$ the solution of Algorithm 8.3 at time t satisfies*

$$\mathbb{P}\{V_t > \varepsilon\} \leq \begin{pmatrix} r+\mu+1 \\ r \end{pmatrix} \sum_{i=0}^{r+\mu-1} \begin{pmatrix} N_s \\ i \end{pmatrix} \epsilon^i (1-\epsilon)^{N_s - i}$$

where

$$V_t = \mathbb{P}_t\{F(\omega_t)x_t + v(\omega_t) > 1\}$$

and μ is the 'support rank' of this chance constraint as defined in [SFM13].

There are both theoretical and practical advantages of this MPC scheme with $M = 1$. Theoretically, the confidence result of Proposition 8.16 holds in closed loop, along with the recursive feasibility and stability guarantees of Theorems 8.14 and 8.15. To the best of the author's knowledge, there is no MPC algorithm in the existing literature using online sampling that gives such confidence guarantees as well as a guarantee of recursive feasibility for a system with stochastic parametric uncertainty.

8.6 Simulation example

We now present the simple example from [FCK16] to show the advantages of a stochastic approach to constraints when multiple sources of uncertainty are present. Defining nominal (expected-value) system matrices

$$A_0 = \begin{bmatrix} -1.9 & -1.4 \\ 0.7 & 0.5 \end{bmatrix}, \quad B_0 = \begin{bmatrix} 1 \\ -0.25 \end{bmatrix}.$$

we take the matrices $A(\omega)$ and $B(\omega)$ and disturbance vector $w(\omega)$ as

$$A(\omega) = A_0 + \Delta_A^{(1)}\omega_1 + \Delta_A^{(2)}\omega_2 + \Delta_A^{(3)}\omega_3 \tag{8.16}$$

$$B(\omega) = B_0 + \Delta_B^{(1)}\omega_4 + \Delta_B^{(2)}\omega_5 \tag{8.17}$$

$$w(\omega) = w^{(1)}\omega_6 + w^{(2)}\omega_7 \tag{8.18}$$

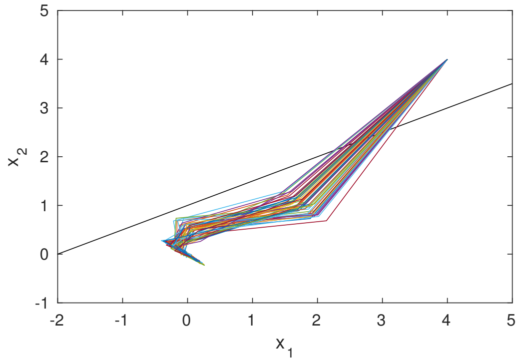

Figure 8.1: Robust constraint handling

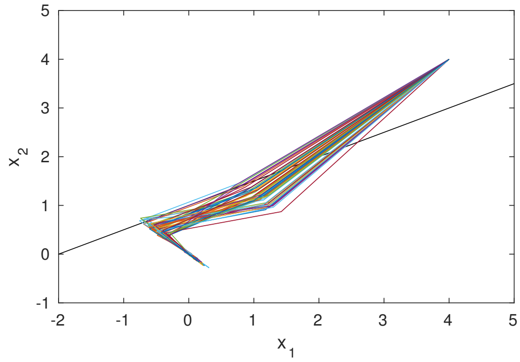

Figure 8.2: Sampled constraint handling

where $\omega_1, \ldots, \omega_7$ are independent and uniformly distributed on the interval $[0, 1]$, and:

$$\Delta_A^{(1)} = \begin{bmatrix} 0.01 & 0.05 \\ -0.05 & -0.01 \end{bmatrix}, \quad \Delta_A^{(2)} = \begin{bmatrix} -0.01 & -0.05 \\ 0 & -0.01 \end{bmatrix}$$
$$\Delta_A^{(3)} = \begin{bmatrix} 0 & 0 \\ 0.05 & 0.02 \end{bmatrix}, \quad \Delta_B^{(1)} = \begin{bmatrix} 0.03 \\ -0.02 \end{bmatrix}$$
$$\Delta_B^{(2)} = \begin{bmatrix} -0.03 \\ 0.02 \end{bmatrix}, \quad w_1 = \begin{bmatrix} 0.2 \\ -0.2 \end{bmatrix}, \quad w_2 = \begin{bmatrix} -0.2 \\ 0.2 \end{bmatrix}$$

The range of ω is therefore the unit hypercube with 128 vertices. This is a quite inefficient way of representing the uncertainty, and it may be the case that using a norm-bounded uncertainty description would be a better choice practically, but it should at least present a challenging test case due to the large number of constraints it implies.

The cost matrices and a probabilistic constraint were defined as

$$Q = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad R = 1, \quad \mathbb{P}_t\{[\ -0.5 \quad 1\ ]x(t+1) \leq 1\} \geq 0.9.$$

and the cost of Proposition 8.5 was used with $\gamma = 1$.

The stochastic MPC algorithm of this chapter, and its robust analog from Chapter 5, were both applied to the system with an initial condition $x_0 = [4 \quad 4]^T$, prediction horizon $N = 4$ and taking $N_s = 250$ samples of ω with $M = 1$. The feedback gain K was chosen to be the LQ-optimal for the pair (A_0, B_0): $K = [1.31 \ \ 0.97]$, and V was constructed by finding the maximal robustly invariant set respecting the constraint

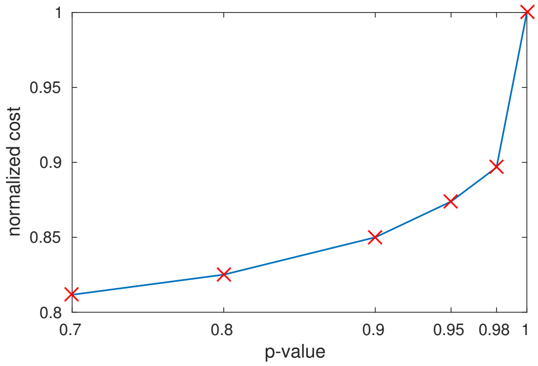

Figure 8.3: Cost improvements

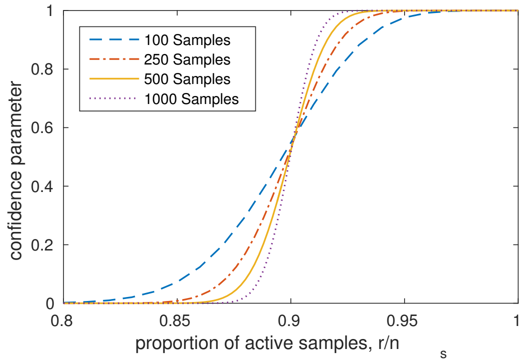

Figure 8.4: Confidence parameters

Table 8.1: Comparison of MPC optimisations

	Robust tube MPC	Stochastic tube MPC
Optimisation type	QP	Repeated QP
Solver	Gurobi	Algorithm 8.3 / Gurobi
Algorithm	Dual simplex	Dual simplex
# Variables	94	94 + 250
# Linear Constraints	11534	11532
# Sampled Constraints	–	250
% Constraints satisfied	100	94
Sparsity factor	0.005	0.005
Average solver time /ms	67.8	131
Closed-loop cost	244.1	208.3

$[-0.5\ \ 1]x \leq 1$, which is defined by 18 linear inequalities. The support rank of the chance constraint is 1 and applying Proposition 8.16 gives the relationship between the confidence parameter and r shown in Figure 8.4 for $\varepsilon = 0.1$. In this case, the chance constraint was applied with 99% confidence, mandating that $r = 233$ samples must satisfy the sampled constraint when solving the MPC problem.

The resulting state-space trajectories are shown in Figures 8.1 and 8.2. It is clear that using robust MPC in an example such as this tends to be very conservative. This is a result of having several independent sources of uncertainty: even though each is uniformly distributed, the one-step ahead probababilty density function resulting from them will be centre-weighted and quickly fall to a negligible value some distance from its centroid.

Table 8.1 compares the closed loop properties of the robust and stochastic implementations of the approach for 500 realizations of model uncertainty. The mean closed loop cost is 15% lower in the stochastic case than the robust. The proportion of trajectories satisfying constraints (94.0%) implies some conservatism in the stochastic MPC. However this is expected from the confidence value of 99%, which requires $r/N_s = 0.932$ from Figure 8.4.

It should be noted that improvements in closed-loop cost are achieved even for large values of p. Figure 8.3 shows the mean cost given by the stochastic MPC algorithm relative to the robust MPC when the chance constraint is applied at different probability levels, with the constraint applied with a confidence of 99% in each case. The improvement in closed-loop cost is significant even when $p = 0.98$.

8.A Proofs

Proposition 8.5

From (8.5), by multiplying right and left by ζ^k, we have:

$$(\zeta^k)^T W \zeta^k - (\zeta^k)^T \mathbb{E}[\Psi(\omega^k)^T W \Psi(\omega^k)]\zeta^k \geq \\ (\zeta^k)^T \mathbb{E}[\Lambda(\omega^k)^T \Lambda(\omega^k)]\zeta^k - \gamma^2 (\zeta^k)^T \mathbb{E}[\mu(\omega^k)\mu(\omega^k)^T]\zeta^k$$

Taking conditional expectations, noting that ζ^k and ω^k are independent from Assumption 8.2, and substituting from the definitions of v, Ψ and Λ yields

$$\mathbb{E}_t[(\zeta^k)^T W \zeta^k] - \mathbb{E}_t[(\zeta^{k+1})^T W \zeta^{k+1}] \geq \mathbb{E}_t\|y^k\|^2 - \gamma^2 \mathbb{E}_t\|w^k\|^2$$

which can be summed over $k = t, t+1, \ldots, \infty$ to give, after substituting $\zeta^t = \zeta_t$:

$$\zeta_t^T W \zeta_t - \lim_{k\to\infty} \mathbb{E}_t[(\zeta^k)^T W \zeta^k] \geq \sum_{k=t}^{\infty} \mathbb{E}_t\|y^k\|^2 - \gamma^2 \sum_{k=t}^{\infty} \mathbb{E}_t\|w^k\|^2$$

Since the limit is nonnegative, we can write

$$\zeta_t^T W \zeta_t \geq \sum_{k=t}^{\infty} \mathbb{E}_t\|y^k\| - \gamma^2 \sum_{k=t}^{\infty} \mathbb{E}_t\|w^k\|^2$$

from which the bound (8.6) follows after a rearrangement.

Lemma 8.8

First we prove the ‘if’ part of the theorem. If the conditions on $H(\omega)$ hold, then there exists some subset of the sample space $\mathcal{E} \subseteq \Omega$ of $\mathbb{P}$-measure at least $1 - \varepsilon$ such that

$$H(\omega) \geq 0, \;\; H(\omega)V = F(\omega), \;\; H(\omega)g \leq b(\omega)$$

on $\mathcal{E}$. By Lemmata 4.1 and 4.3 in Chapter 4 this implies

$$\mathcal{P}[V, g] \subseteq \bigcap_{\omega\in\mathcal{E}} \mathcal{P}[F(\omega), b(\omega)] \subseteq \bigcup_{\mathcal{E}\in\mathcal{M}} \bigcap_{\omega\in\mathcal{E}} \mathcal{P}[F(\omega), b(\omega)]$$

from which $\mathcal{P}[V, g] \subseteq \mathcal{C}[F(\omega), b(\omega), \varepsilon]$ follows by Lemma 8.7.

Now we consider the 'only if' part of the theorem, proceeding by construction of $H(\omega)$. The inclusion in the statement of the theorem implies

$$\mathbb{P}\{\mu_j(\omega) \le b_j(\theta), \forall j\} \ge 1 - \varepsilon \tag{8.19}$$

where $b_j(\omega)$ is the jth element of $b(\omega)$ and:

$$\mu_j(\omega) = \max_x \{F_j(\omega)x | \; Vx \le g\} \tag{8.20}$$

with $F_j(\omega)$ the jth row of $F(\omega)$. Note that as the solution of this LP is continuous and piecewise affine in $V_j(\omega)$ [BBM+02, Theorem 4], $\mu_j(\omega)$ is a measurable function, as compositions of continuous functions are continuous and continuity implies measurability [CK04, Section 3.3]. Taking the dual of the Linear Program defining $\mu_j(\omega)$, noting that if the polyhedron is nonempty we have strong duality by Slater's condition, gives:

$$\mu_j(\omega) = \min_h \{hg | \; hV = F_j(\omega), h \ge 0\} \tag{8.21}$$

Now let $h_j^*(\omega)$ denote an optimal solution to this dual LP, and let $H(\omega)$ be the matrix which has $h_j^*(\omega)$ in the jth row. $H(\omega)$ satisfies $H(\omega) \ge 0$ and $H(\omega)V = F(\omega)$ from (8.21) and $\mathbb{P}[H(\omega)g \le b(\omega)] \ge 1 - \varepsilon$ follows from (8.19) noting that $H(\omega)g$ is the vector with elements μ_j. Finally, we note that $H(\omega)$ is a measurable function and hence a random variable itself: measurability follows from the facts that $h_j^*(\omega)$ is a piecewise affine function of F_j [BBM+02, Theorem 4], that compositions of continuous functions are continuous and continuity implies measurability [CK04, Section 3.3].

Proposition 8.11

Follows immediately by applying Lemma 8.8 to (8.9).

Theorem 8.12

The result follows by verifying that for any possible x_t and a given feasible $\underline{\alpha}$ and $\underline{c}_t$, the constraints at the next time step can be satisfied by constructing $\tilde{c}_{t+1}$ with $\tilde{c}_{t+1}^k = c_t^k$

for all k except $\tilde{c}_{t+1}^{t+N} = 0$ and a $\tilde{\alpha}_{t+1}$ chosen so that $g_{t+1}^k(\tilde{\alpha}_{t+1}^k) = g_t^k(\alpha_t^k)$ for all k except $g_{t+1}^{t+N+1}(\tilde{\alpha}_{t+1}^{t+N+1}) = g_t^{t+N}(\alpha_t^{t+N})$.

Theorem 8.14

We will call a support constraint that corresponds to an element of the index set $\mathcal{I}^k$ ***removable*** if the corresponding index is present in $\mathcal{J}_k$ and $|\mathcal{J}_k| > r$ (as in such an instance it is possible to choose $\mathcal{L}_k$ so that it does not include the support constraint). Note that at each iteration, due to Assumption 8.10, the optimisation in step 3 contains only constraints that were satisfied in the previous solution (by step 4). Hence any solution to that optimisation will be feasible for the constraints in the optimisation on the next iteration of the algorithm, and also when the algorithm is begun at the next time t.

Now suppose there are no removable support constraints, then the solution of the problem at the next iteration will have an equal cost $J(\zeta)$. Assuming that some tie-break rule exists to distinguish solutions with equal values of $\min(s_k^{\{i\}})$, then because the current solution ζ is feasible at the next iteration, the solution at the next iteration will be equal to the current one. This implies that $\mathcal{L}_k = \mathcal{I}^k$ for all k after the next iteration, so that the algorithm terminates.

Conversely, if there are removable support constraints, then as the optimisation in step 3 was solved to optimality we will have $\min(s_k^{\{i\}}) = 0$ for at least one support constraint, so that it will not be present in $\mathcal{L}_k$. Hence that support constraint is removed from $\mathcal{I}^k$ for the next iteration and the cost $J(\zeta)$ must decrease in step 3. As the support constraint will surely be violated in the solution to that optimisation, the cardinality of $\mathcal{J}_k$ will decrease by at least one. This can only occur a maximum of $M(N_s - r)$ times before there are no more removable constraints, and then of course the algorithm will terminate after the following iteration.

Theorem 8.15

From the bound of Proposition 5.4 we can write:

$$\zeta_k^T W \zeta_k - \mathbb{E}[\zeta_k^T \Psi(\omega)^T W \Psi(\omega)\zeta_k] \geq \mathbb{E}\|\Lambda(\omega)\zeta_t\|^2 - \gamma^2 \mathbb{E}\|w(\omega)\|^2, \quad \forall \omega \in \Omega$$

By the recursive feasibility result of Theorem 8.14, $\Psi(\omega)\zeta_t$ is feasible at time $t+1$ for any ω. Hence we will have $\mathbb{E}[\zeta_{k+1}^T W \zeta_{k+1}] \leq \mathbb{E}[\zeta_k^T \Psi(\theta_k)^T W \Psi(\theta_k)\zeta_k]$ so:

$$\zeta_k^T W \zeta_k - \mathbb{E}[\zeta_{k+1}^T W \zeta_{k+1}] \geq \mathbb{E}\|y_k\|^2 - \gamma^2 \mathbb{E}\|w_k\|^2$$

Summing over the dummy variable $k = 0, \ldots, t$ and rearranging gives

$$\zeta_0^T W \zeta_0 - \mathbb{E}[\zeta_t^T W \zeta_t] + \gamma^2 \sum_{k=0}^{t} \mathbb{E}\|w_k\|^2 \geq \sum_{k=0}^{t} \mathbb{E}\|y_k\|^2$$

which implies the bound of the theorem as the second term is nonpositive.

9 Robust time average constraints

9.1 Introduction

The chance constraints of the previous chapter present many difficulties from a computational perspective, as in general they are difficult to evaluate and may even be nonconvex. Although sampling methods can deal with these problems fairly effectively, they are not ideal since solving such sample approximations tends to entail a considerably more complex algorithm than a simple convex optimisation. In addition, any guarantee of satisfaction of the original chance constraints will be probabilistic in nature, and there is a small but finite probability that the solution will not satisfy the constraints.

In many applications, chance constraints arise because there is some undesirable event that is not permitted to occur 'too often'. A good example is the structural constraints found in the control of wind turbines [HBP07, ECK15]. In this application it is usually desired to limit damage to the turbine caused by structural fatigue. This naturally leads to a limit on the time-average number of violations of a state constraint. Such a constraint could be enforced by applying a chance constraint as in the last chapter, but in many ways this probabilistic condition is a stronger one than the original bound on the time-average of a number of violations. This raises the question of whether the stochastic MPC problem becomes any easier when we consider bounds on numbers of constraint violations, rather than chance constraints.

In this chapter, we consider applying a robust constraint on the number of allowed violations of a given state constraint in closed-loop, in any given time period. The tube MPC formulation considered has two versions of the state constraint - a 'strict' version and a 'relaxed' version. We show that by allowing the controller to choose dynamically between the strict and relaxed versions of the constraints in operation, it is possible to construct a controller that robustly satisfies a constraint on the number of violations of a state constraint. For example, one could specify that a maximum of 10 constraint violations is allowed in any period of 100 contiguous sampling times. This has several advantages over chance constraints:

- The online optimisation only requires applying robust constraints, which is a convex problem in many cases;
- Constraints can be relaxed in response to past satisfaction of constraints, giving some 'feedback' to constraint violations in closed-loop;

In the literature, there already exist approaches to MPC that consider bounding the time average number of violations in closed-loop. Perhaps the most interesting proposal is found in [KGOJ14], where the cumulative value of a loss function is considered over time and constrained, with a mechanism included to gradually 'forget' past constraint violations. This leads to controllers that asymptotically satisfy a bound on this quantity. In contrast, the tube MPC controllers that we present here satisfy a bound robustly, allowing for a somewhat stronger constraint satisfaction result in closed-loop.

9.1.1 Motivating discussion

In some stochastic control problems, it is desirable to apply a constraint on the time-average of the number of violations of a given constraint in closed-loop. As mentioned earlier, constraints of this sort appear quite naturally as the fatigue constraints arising in the control of the blade pitch of wind turbines [HBP07, ECK15] where they result from an application of Miner's rule to limit the number of allowable stress cycles.

If we introduce a quantity M_t that is equal to one when there is a violation of a constraint $F(\omega_t)x + v(\omega_t) \leq 1$, and zero when there is not, we can express a constraint on the expected time average as:

$$\lim_{T\to\infty} \frac{1}{T} \sum_{t=0}^{T} \mathbb{E}[M_t] \leq 1 - \epsilon \tag{9.1}$$

Now consider the design of an MPC controller that ensures this bound in closed-loop operation. If we wished to use the techniques of the preceding chapter, we could strengthen (9.1) to a chance constraint

$$\mathbb{E}[M_t] = \mathbb{P}\{M_t = 1\} \leq 1 - \epsilon \tag{9.2}$$

which will ensure (9.1) by the law of large numbers. This gives a method to approach the problem, but chance-constrained programming is hard and it might well be the case that in replacing (9.1) with (9.2) we have made the problem significantly more difficult to solve.

This chapter considers a different strengthening of the time-average constraint (9.1). Instead of replacing it with a probabilistic condition, we choose a finite value of T and consider robustly satisfying the condition

$$\frac{1}{T}\sum_{k=t}^{t+T} M_t \leq 1 - \epsilon \tag{9.3}$$

for all values of t. It is clear that this latter condition ensures (9.1) as, for any positive integer l it implies

$$\frac{1}{lT}\left(\sum_{k=t}^{t+T} M_t + \sum_{k=t+T}^{t+2T} M_t + \ldots + \sum_{k=(l-1)T}^{t+lT} M_t\right) \leq 1 - \epsilon$$

and taking expectations and considering the limit as $l \to \infty$, we recover the time-average constraint (9.1). The point here is that applying the constraint (9.3) for all t may be much easier than applying a chance constraint like (9.2). Indeed, in this chapter we show that this can be carried out by solving a single convex optimisation at each time t, and the resulting controller is recursively feasible and stable.

If it is desired to limit the amount of constraint violation, we can also consider applying a constraint

$$\frac{1}{T}\sum_{k=t}^{t+T} L(F(\omega_t)x + v(\omega_t) - 1) \leq 1 - \epsilon \tag{9.4}$$

for all t where the loss function L is convex and equal to zero whenever its argument is nonpositive.

9.2 Problem formulation

For the most part, the problem we consider is identical to the one in the previous chapter. Hence we again consider the stochastic LDI system

$$x_{t+1} = A(\omega_t)x_t + B(\omega_t)u_t + w(\omega_t) \tag{9.5a}$$

$$y_t = C(\omega_t)x_t + D(\omega_t)u_t \tag{9.5b}$$

$$z_t = F(\omega_t)x_t + v(\omega_t) \tag{9.5c}$$

with the system matrices affine functions of a random variable $\omega \in \Omega$ so that, for instance,

$$(A(\omega), B(\omega), w(\omega)) = (\bar{A}, \bar{B}, 0) + \sum_{i=1}^{r}(A^{[i]}, B^{[i]}, w^{[i]})\omega_i$$

and where Ω is a compact set. All of the standing assumptions of the previous chapter, such as observability, stabilisation by a state feedback K, compactness of the uncertainty set Ω, and the properties of the tube parameterisation, hold here as well. This will save us from repetition. Similarly, we will use the cost function of Proposition 8.5 as a quadratic bound on the expected value of the predicted l_2-norm of the output y_t.

Instead of a chance constraint, we will consider applying a robust constraint on the number of violations of a certain constraint in a given time period. Hence we define a indicator random variable for the event $z_t \not\leq 1$ as

$$M_t = \begin{cases} 1 & \text{if } z_t \not\leq 1 \\ 0 & \text{otherwise} \end{cases} \tag{9.6}$$

and we will require that in closed-loop the controlled system robustly satisfies the constraint

$$\sum_{k=t}^{t+N_c} M_k \leq \bar{M} \tag{9.7}$$

for some N_c and $\bar{M}$, for all t. The mechanism through which this is possible is by allowing the MPC controller to 'choose' between applying a robust constraint to the predictions

$$F(\omega)x^k + v(\omega) \leq 1, \quad \forall \omega \in \Omega \tag{9.8}$$

and a relaxed version

$$F(\omega)x^k + v(\omega) \leq 1 + r, \quad \forall \omega \in \Omega \tag{9.9}$$

where $r \geq 0$ is some positive vector.

Remark 9.1. The exact form of the relaxed constraint is unimportant as long as any prediction x^k satisfying (9.8) satisfies it as well. So we could consider applying $F_0 x^k + v_0 \leq 1$ for some nominal value of $F(\omega), v(\omega)$ for instance, or even the sampled constraints of the previous chapter.

9.3 Constraint handling using tubes

As in previous chapters, we will consider using a prediction tube and applying (9.8) and (9.9) to the tube cross-sections with constraints

$$\Delta(\omega)g(\alpha^k) + v(\omega) \leq 1, \quad \forall \omega \in \Omega \tag{9.10}$$

and:

$$\Delta(\omega)g(\alpha^k) + v(\omega) \leq 1 + r, \quad \forall \omega \in \Omega \tag{9.11}$$

These two conditions can be combined into a single constraint if a binary variable b^k is introduced to distinguish between the strict and relaxed forms of the constraint. This leads us to

$$\Delta(\omega)g(\alpha^k) + v(\omega) \leq 1 + b^k r, \quad \forall \omega \in \Omega \tag{9.12}$$

where $b^k \in \{0, 1\}$ will be determined before running the MPC optimisation at each sampling time (that is, it is ***not*** to be considered as a decision variable in a mixed-integer optimisation). This leads to constraints for the tube:

$$Vx^t \leq g^t(\alpha^t) \tag{9.13a}$$

$$\Delta(\omega)g^k(\alpha^k) + v(\omega) \leq 1 + b^k r, \quad \forall \omega \in \Omega \tag{9.13b}$$

$$\Delta(\omega)g^{t+N}(\alpha^{t+N}) \leq 1, \quad \forall \omega \in \Omega \tag{9.13c}$$

$$\Gamma(\omega)g^k(\alpha^k) + VB(\omega)c^k + Vw(\omega) \leq g^{k+1}(\alpha^{k+1}), \quad \forall \omega \in \Omega \tag{9.13d}$$

$$\Gamma(\omega)g^{t+N}(\alpha^{t+N}) + Vw(\omega) \leq g^{t+N}(\alpha^{t+N}), \quad \forall \omega \in \Omega \tag{9.13e}$$

As usual we will introduce a more compact notation for these constraints as $(\underline{c}_t, \underline{\alpha}_t) \in \mathcal{T}(x_t, \underline{b}_t)$ where $\underline{b}_t = [b_t \;\; b_{t+1} \;\; \dots \;\; b_{t+N-1}]$ and the other symbols have their usual meanings.

If we wish to consider the constraint (9.4) which includes the loss function on the quantity of constraint violation, that is

$$\frac{1}{T}\sum_{k=t}^{t+T} L(F(\omega_t)x + v(\omega_t) - 1) \leq 1 - \epsilon$$

then we may relax the condition that b^k is a binary variable. If we instead require $0 \leq b^k \leq 1$ for all k, we can reformulate the tube constraints with this condition included and consider optimising over b^k as well as c^k and α^k, subject to the constraints

$$Vx^t \leq g^t(\alpha^t) \tag{9.14a}$$

$$\Delta(\omega)g^k(\alpha^k) + v(\omega) \leq 1 + b^k r, \quad \forall\omega \in \Omega \tag{9.14b}$$

$$\Delta(\omega)g^{t+N}(\alpha^{t+N}) \leq 1, \quad \forall\omega \in \Omega \tag{9.14c}$$

$$\Gamma(\omega)g^k(\alpha^k) + VB(\omega)c^k + Vw(\omega) \leq g^{k+1}(\alpha^{k+1}), \quad \forall\omega \in \Omega \tag{9.14d}$$

$$\Gamma(\omega)g^{t+N}(\alpha^{t+N}) + Vw(\omega) \leq g^{t+N}(\alpha^{t+N}), \quad \forall\omega \in \Omega \tag{9.14e}$$

$$L_s^k + \frac{1}{T}\sum_{l=t}^{k} L(rb^l) \leq 1 - \epsilon \tag{9.14f}$$

$$0 \leq b^k \leq 1 \tag{9.14g}$$

here, we have

$$L_s^k = \frac{1}{T}\sum_{l=k-T}^{t} L(F(\omega_l)x_l + v(\omega_l) - 1)$$

which can be easily computed by the controller before solving an optimisation, if it stores the past values of the loss function at each time. We will write constraints (9.14) in the brief form $(\underline{c}_t, \underline{\alpha}_t, \underline{b}_t) \in \mathcal{T}_b(x_t)$. They are convex if the loss function L is.

9.4 Algorithms and stability

To satisfy the robust bound (9.7) on constraint violations in closed-loop, the controller must be aware of the number of previous constraint violations and when they occurred. We will assume, therefore, that the controller stores up to the last N_c values of M_t. This is fairly simple in implementation terms, as these values are simply binary quantities and need only require a single bit of storage each.

For each prediction time k, the controller must choose b_k based on the sum of constraint violations in the preceding N_c sample times (and the possible future constraint violations over the prediction horizon). This means that we would like to ensure

$$\sum_{i=k-N_c}^{t-1} M_i + \sum_{i=t}^{k-1} b^i \leq \bar{M}$$

for all prediction times k over the prediction horizon. This suggests the following simple procedure for choosing b_k for each k: starting with $k = t$, evaluate the sum on the left-hand-side of the above expression. If the result is less than $\bar{M}$ then the number of violations in the worst case up to that prediction time is less than $\bar{M}$, so we can safely choose $b_k = 1$. Otherwise, choose $b_k = 0$. We can continue in this way to choose b_k for $k = t+1, \ldots, t+N-1$ and once we have them all, solve the tube MPC optimisation using the constraints $\mathcal{T}(x_t, \underline{b}_t)$.

Repeating the sum over M_i and previously chosen values of b^k seems somewhat inefficient, and we can simplify the procedure if we introduce a variable $\underline{v}^k$ that keeps track of the 'worst-case' number of potential constraint violations up to prediction time k. If we set

$$\underline{v}^k = \sum_{i=k-N_c}^{t-1} M_i + \sum_{i=t}^{k-1} b^i$$

then subsequent values of $\underline{v}^k$ satisfy:

$$\bar{v}^k = \bar{v}^{k-1} - M_{k-1-N_c} + b^{k-1}$$

This suggests an MPC algorithm (Algorithm 9.1) at each t where the values b^k are computed in a pre-computation step and then a tube MPC optimisation solved using the combination of strict and relaxed constraints. We omit the offline design procedure, as it is identical to Algorithm 8.1.

Algorithm 9.1 Violation bound MPC

1. Calculate $\bar{v}^t = \sum_{i=t-N_c}^{t-1} M_i$
2. For each $k = t, \ldots, t+N-1$
   (a) Calculate $\bar{v}^k = \bar{v}^{k-1} - M_{k-1-N_c} + b^{k-1}$
   (b) If $\bar{v}^k < \bar{M}$, set $b^k = 1$. Otherwise $b_k = 0$.
3. Set $\underline{b}_t = [b^t \;\; b^{t+1} \;\; \ldots \;\; b^{t+N-1}]$ and solve:

$$\begin{array}{ll} \underset{\alpha_t, \underline{c}_t}{\text{minimise}} & \zeta_t^T W \zeta_t \\ \text{subject to} & (\underline{c}_t, \underline{\alpha}_t) \in \mathcal{T}(x_t, \underline{b}_t) \end{array}$$

4. Apply $u_t = K x_t + c_t$ to the system.

Algorithm 9.1 is simple and almost as fast as the robust MPC problem considered in Chapter 5, as steps 1 and 2 only require simple arithmetic and comparison operations and can typically be carried out much more quickly than the optimisation in step 3. A key fact about the policy in step 2 for choosing b^k is that in closed-loop the value of b_t^k corresponding to a fixed prediction time k can only increase, and not decrease.

Lemma 9.2. *For any $k = t, \ldots, t+N-1$ we have $b_{t+1}^k \geq b_t^k$ in closed loop for all t if the optimisation in step 3 of Algorithm 9.1 is feasible at time t.*

An immediate consequence of Lemma 9.2 is that the constraints for any given prediction time can only be relaxed, and not tightened, in closed-loop operation of the algorithm. From the usual recursive feasibility properties of the tube MPC, this implies that the optimisation in step 3 of Algorithm 9.1 remains feasible for all time if it is feasible initially. This means that (9.7) holds in closed-loop.

Theorem 9.3. *Under the control of Algorithm 9.1 we have*

$$\sum_{k=t}^{t+N_c} M_k \leq \bar{M}$$

for all $t = 0, \ldots, \infty$ if the optimisation is initially feasible.

In the case that we wish to satisfy a robust bound involving the loss function, that is (9.4), we can use a similar algorithm.

Algorithm 9.2 Loss bound MPC

---

1. For $k = t, \dots, t+N-1$, calculate $L_s^k = \frac{1}{T}\sum_{l=k-T}^{t} L(F(\omega_l)x_l + v(\omega_l) - 1)$
2. With $\underline{b}_t = [b^t \;\; b^{t+1} \;\; \dots \;\; b^{t+N-1}]$, solve:

$$\begin{array}{ll} \underset{\alpha_t, \underline{c}_t}{\text{minimise}} & \zeta_t^T W \zeta_t \\ \text{subject to} & (\underline{c}_t, \underline{\alpha}_t, \underline{b}_t) \in \mathcal{T}_b(x_t) \end{array}$$

3. Apply $u_t = Kx_t + c_t$ to the system.

We can similarly give a recursive feasibility and correctness result for Algorithm (9.2).

Theorem 9.4. *Under the control of Algorithm 9.2, the optimisation remains feasible if it is initially feasible, and we have*

$$\frac{1}{T}\sum_{k=t}^{t+T} L(F(\omega_t)x + v(\omega_t) - 1) \leq 1 - \epsilon$$

for all $t = 0, \dots, \infty$ *in this case.*

As we have recursive feasibility guarantees, the stability result of the last chapter (Theorem 8.15) applies here also, with no modification. Hence we can state:

Theorem 9.5. *Under the control of Algorithm 9.1 or 9.2, the output of (8.1) satisfies the bound*

$$\sum_{k=0}^{t} \mathbb{E}\|y_k\|^2 \leq \zeta_0^T W \zeta_0 + \gamma^2 \sum_{k=0}^{t} \mathbb{E}\|w_k\|^2$$

for all $t = 0, \dots, \infty$ *if the optimisation is initially feasible.*

9.5 Simulation example

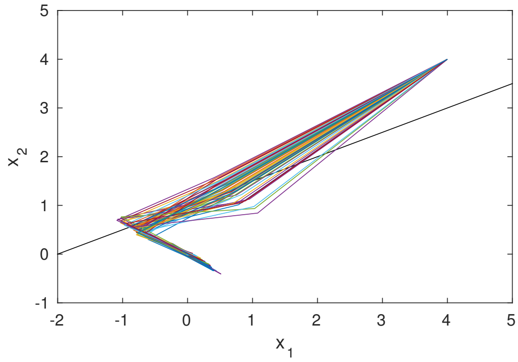

Figure 9.1: Sampled constraints

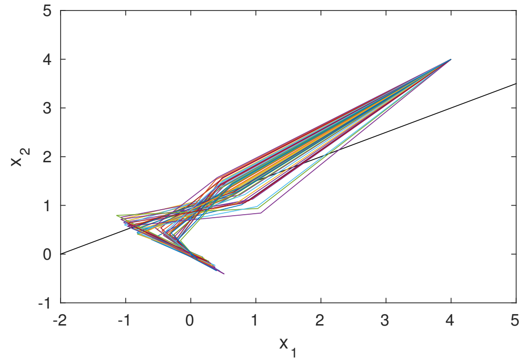

Figure 9.2: Time-average constraints

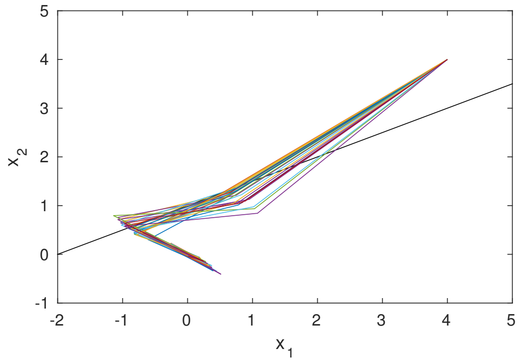

Figure 9.3: t=1 constraint satisfied

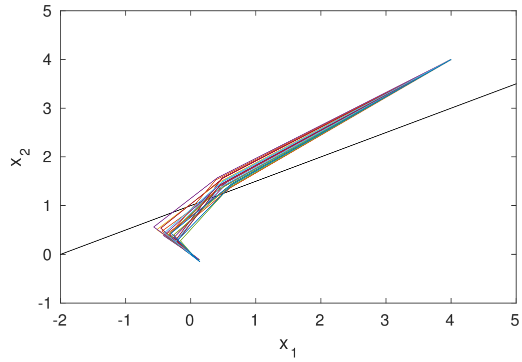

Figure 9.4: t=1 constraint violated

For simplicity, we reuse the example system of Chapter 8. This will have the advantage that we can easily compare the results of the present algorithm with the sampling approach used previously. The system has nominal matrices

$$A_0 = \begin{bmatrix} -1.9 & -1.4 \\ 0.7 & 0.5 \end{bmatrix}, \quad B_0 = \begin{bmatrix} 1 \\ -0.25 \end{bmatrix}.$$

and we take the matrices $A(\omega)$ and $B(\omega)$ and disturbance vector $w(\omega)$ as

$$A(\omega) = A_0 + \Delta_A^{(1)}\omega_1 + \Delta_A^{(2)}\omega_2 + \Delta_A^{(3)}\omega_3 \tag{9.15}$$

$$B(\omega) = B_0 + \Delta_B^{(1)}\omega_4 + \Delta_B^{(2)}\omega_5 \tag{9.16}$$

$$w(\omega) = w^{(1)}\omega_6 + w^{(2)}\omega_7 \tag{9.17}$$

where $\omega_1, \ldots, \omega_7$ are independent random scalars uniformly distributed on the interval $[0, 1]$, and:

$$\Delta_A^{(1)} = \begin{bmatrix} 0.01 & 0.05 \\ -0.05 & -0.01 \end{bmatrix}, \quad \Delta_A^{(2)} = \begin{bmatrix} -0.01 & -0.05 \\ 0 & -0.01 \end{bmatrix}$$

$$\Delta_A^{(3)} = \begin{bmatrix} 0 & 0 \\ 0.05 & 0.02 \end{bmatrix}, \quad \Delta_B^{(1)} = \begin{bmatrix} 0.03 \\ -0.02 \end{bmatrix}$$

$$\Delta_B^{(2)} = \begin{bmatrix} -0.03 \\ 0.02 \end{bmatrix}, \quad w_1 = \begin{bmatrix} 0.2 \\ -0.2 \end{bmatrix}, \quad w_2 = \begin{bmatrix} -0.2 \\ 0.2 \end{bmatrix}$$

We once again use the cost of Proposition 8.5 with $\gamma = 1$, $Q = \mathrm{diag}(1, 1)$ and $R = 1$. For implementation of Algorithm 9.1 we consider the constraint given as

$$[\ -0.5 \quad 1\ ]x(t+1) \leq 1$$

and a 'relaxed' version

$$[\ -0.5 \quad 1\ ]x(t+1) \leq 1 + 0.8$$

so we are taking $r = 1.8$. We choose $N_c = 3$ and $\bar{M} = 2$ so that the constraint

$$\sum_{k=t}^{t+2} M_k \leq 2 \tag{9.18}$$

will be satisfied at all times t in closed-loop under Algorithm 9.1.

For comparison purposes, we also consider running the sampled MPC algorithm of the previous chapter for the chance constraint:

$$\mathbb{P}_t\{[\ -0.5 \quad 1\ ]x(t+1) \leq 1\} \geq 0.67$$

The results of the two approaches can be seen in Figures 9.1 and 9.2. The behaviour of the two controllers clearly differs in closed-loop operation due to the robust bound (9.18) that must be satisfied in the latter case. This is shown in a little more detail in Figures 9.3 and 9.4, where it is clear that every closed-loop trajectory violating the constraint at $t = 0$ and $t = 1$ is forced to satisfy it at $t = 2$.

Table 9.1: Comparison of MPC optimisations

	Sampled constraints	Time-average constraints
Optimisation type	Repeated QP	QP
Solver	Algorithm 8.3 / Gurobi	Algorithm 9.1 / Gurobi
Algorithm	Dual simplex	Dual simplex
% Constraints satisfied	68	66
Sparsity factor	0.005	0.005
Average solver time /ms	132	69
Closed-loop cost	198.3	195.2

Some quantities of interest are listed in Table 9.1 for the two MPC algorithms. The most interesting result is that the average cost for the two algorithms is comparable, despite Algorithm 9.1 only requiring a single robust optimisation and hence being solvable in significantly less time. This suggests that such algorithms might be a favourable alternative to MPC controllers involving chance constraints.

9.A Proofs

Lemma 9.2

Note that at any time t we have $M_t \leq b_t^t$, as (since we are assuming the optimisation is feasible at time t) the constraint may only be violated in closed-loop if $b_t^t = 1$. We will firstly show that the value $\underline{v}^{t+1}$ from step 1 of the algorithm run at time $t+1$ satisfies $\underline{v}_{t+1}^{t+1} \leq \underline{v}_t^{t+1}$ and hence that $b_{t+1}^{t+1} \geq b_t^{t+1}$. We have

$$\underline{v}_{t+1}^{t+1} = \sum_{i=t+1-N_c}^{t} M_i \leq \sum_{i=t+1-N_c}^{t-1} M_i + b_t^t = \underline{v}_t^{t+1}$$

where the inequality follows directly from $M_t \leq b_t^t$. Hence we can write $\underline{v}_{t+1}^{t+1} \leq \underline{v}_t^{t+1}$ and as an obvious consequence $b_{t+1}^{t+1} \geq b_t^{t+1}$ since b_{t+1}^{t+1} will only be zero if b_t^{t+1} was zero and $\underline{v}_t^{t+1} = \underline{v}_{t+1}^{t+1} = \bar{M}$. Now suppose that we have $\underline{v}_{t+1}^k \leq \underline{v}_t^k$ and $b_{t+1}^k \geq b_t^k$ for some k, then for $k+1$ we must verify the relationship:

$$\underline{v}_{t+1}^{k+1} = \underline{v}_{t+1}^k - M_{k-N_c} + b_{t+1}^k \leq \underline{v}_t^k - M_{k-N_c} + b_t^k = \underline{v}_t^{k+1} \tag{9.19}$$

As we have $\underline{v}_{t+1}^k \leq \underline{v}_t^k$, if this relationship did not hold it would imply $b_{t+1}^k > b_t^k$, which means that $b_t^k = 0$ and $b_{t+1}^k = 1$ since this is a binary variable. But in turn, considering step 2 of the algorithm, this implies that $\underline{v}_t^k = \bar{M}$ and $\underline{v}_{t+1}^k < \bar{M}$ so that we conclude $\underline{v}_t^k + b_t^k = \underline{v}_{t+1}^k + b_{t+1}^k$ and hence that (9.19) actually holds - a contradiction. Hence (9.19) and $b_{t+1}^{k+1} \geq b_t^{k+1}$ must both be true if $\underline{v}_{t+1}^k \leq \underline{v}_t^k$ and $b_{t+1}^k \geq b_t^k$ for some k. As we have shown that these conditions hold for $k = t$ and for $k+1$ if they hold for k, we have $\underline{v}_{t+1}^k \leq \underline{v}_t^k$ and $b_{t+1}^k \geq b_t^k$ for all $k = t, \ldots, t+N-1$ by induction.

Theorem 9.3

It is clear from the form of the optimisation in step 3 of Algorithm 9.1 that the claimed bound holds for any time t at which that optimisation is feasible. From Lemma 9.2 we have recursive feasibility of step 3 of Algorithm 9.1, and hence the bound holds for all t if the optimisation is feasible initially.

Theorem 9.4

The recursive feasibility result follows from the usual method of showing that there exists a feasible 'tail' sequence of each sequence of predicted variables at the next time. In this case, if we have $(\underline{c}_t, \underline{\alpha}_t, \underline{b}_t) \in \mathcal{T}_b(x_t)$, then the tail sequences feasible at $t+1$ can be constructed by removing the first element of each of the vectors $\underline{c}_t$, $\underline{\alpha}_t$, and $\underline{b}_t$ and appending them with 0, α^N, and 0 respectively. It is straightforward to verify that this gives sequences satisfying (9.14) after considering the definition of L_s, but we do not do so explicitly here for brevity as the method is essentially identical to the recursive feasibility proofs from previous chapters.

Satisfaction of the bound

$$\frac{1}{T}\sum_{k=t}^{t+T} L(F(\omega_t)x + v(\omega_t) - 1) \leq 1 - \epsilon$$

follows from the fact that the constraints (9.14b) and (9.14f) give a sufficient condition for this bound for any given t. As we have recursive feasibility, it then holds for all t if the optimisation is feasible initially.

Theorem 9.5

Identical to the proof of Theorem 8.15, for either algorithm.

10 Conclusion

This thesis has presented a general method of constraint handling, using a prediction tube, for the model predictive control of systems with parameter uncertainty. We showed how to formulate the MPC of constrained linear difference inclusion, linear parameter-varying, and linear stochastic difference inclusion systems in this framework, giving recursive feasibility and stability results. We also considered some related problems such as applying subset conditions involving sets defined by robust constraints, finding output admissible sets for chance-constrained systems, and applying time-average constraints in stochastic MPC.

10.1 Contributions of the thesis

To close, we outline in more detail the specific contributions of the work, then suggest some extensions that could be investigated in the future.

Chapter 4: Robust inclusions

The analysis of robust set inclusions in Chapter 4 is a useful theoretical contribution, allowing for the definition of prediction tubes for parameter-uncertain systems for arbitrary uncertainty sets, and a wide-variety of tube parameterisations. Previously, this was only possible when the uncertainty was polytopic and the tube parameterisation g an identity function as in [ECK12a, FKC15], or with a homothetic parameterisation as in [RC13] . Additionally, the analysis of conservatism arising from the form of the $H(\theta)$ matrix functions used to apply the inclusions is novel, and leads to more systematic methods of designing the tube parameterisation to minimise this conservatism.

Chapter 5: Tube MPC for LDI systems

The tube MPC for LDI systems presented in Chapter 5 is a significant generalisation of the work in both [ECK12a] and [RC13], in that it allows for the uncertainty to be contained in an arbitrary compact set, the tube parameterisation can be given by any affine function $g(\alpha)$ and can vary with prediction time k, and cost functions giving both l_2-norm bounds and convergence to a target set are considered. We additionally showed that if the methods of Chapter 4 are used to build the tube parameterisation then the resulting convex optimisation problems are highly sparse with a block-banded structure, implying that they are quickly solved using standard solvers that can exploit such sparsity.

Chapter 6: Tube MPC for LPV systems

Chapter 6 extends the tube MPC to LPV-A systems, and shows that it is possible to optimise over controls that are affine in the current value of the uncertain parameter. This reduces conservatism in both constraint handling and in the quadratic function used to upper bound the minimax cost. To the best of the author's knowledge, this parameterisation is novel, and is possible because of the form of the tube MPC constraints that need only consider predictions 'one step at a time'. Other methods appear in the literature for optimising over gain-scheduled feedbacks for constrained systems such as in [LA00], but these tend to handle constraints poorly due to the use of ellipsoidal bounding, and are outperformed by the tube MPC presented here as a result.

Chapter 7: Sample-admissible sets

Chapter 7 considers the theoretical problem of characterising the maximum constraint-admissible set for an LDI under a chance constraint. Although it is possible to express this maximum admissible set in closed-form, we see that in the general case it has two major deficiencies: it is very difficult to compute the set of chance constraints defining it, and it may not retain its invariance property when considering sample approximations. We then introduce a new, more restrictive class of constraint admissible sets for such problems designed to ameliorate these difficulties. We show that the maximum and minimum sets in this new class are well-defined and present an algorithm for computing the chance constraints defining this maximum set. Finally, we suggest a simple approach to finding polytopic approximations to this chance-constrained set using sampling. The results of this chapter are novel.

Chapter 8: Tube MPC for stochastic LDIs

In Chapter 8, the tube MPC given in Chapter 5 is extended to consider chance-constrained systems, where chance constraints are tackled using sampling methods. Sampling methods have been presented in the literature before for handling uncertain systems, for instance in [SFFM14], but the major difference here is that the formulation is recursively feasible and hence comes with a guarantee of satisfaction of the sampled constraints in closed-loop. In a special case where the sampling is only done one time step in advance, this also allows for coloured noise and provides a bound on the probability of satisfying the chance constraint. The online computation is performed using a specialised algorithm that uses a heuristic to remove sampled constraints and improve the cost function. Such methods are not new in the context of chance-constrained optimisation [CG11], but we show here that it is possible to ensure a cost decrease in the algorithm at every iteration when using such a heuristic, hence ensuring closed-loop stability of the MPC controller even if the algorithm is terminated before convergence.

Chapter 9: Time-average constraints

Chapter 9 presents two simple possible alternatives to using chance constraints in stochastic MPC, motivated by the observation that in many stochastic constrained control problems it is desired to bound the time-average of some quantity. This time-average can be strengthened to a robust constraint on the sum of constraint violations in a finite period, rather than to a chance constraint. By allowing the controller to respond to the effects of past violations, two algorithms are presented that solve the problem, both requiring a single convex optimisation after a preprocessing step. A stochastic MPC formulation that 'remembers' previous violations and reacts accordingly was given previously in [KGOJ14], but there the focus was an asymptotic guarantee on the average number of violations. In contrast we give stronger robust bounds on the violation permissible in any time period of a specified length. The method is extended to the case that it is required to bound the sum of convex loss functions of the amount of constraint violation, allowing for controllers to to place greater importance on avoiding larger violations.

10.2 Suggested future work

Of course, it is inevitable in research work that some questions will remain. In the author's view, the most important open question relating to this thesis is whether the confidence result of Chapter 8, Proposition 8.16, generalises to the case where $M > 1$, so that samples are retained and reused at more than one time t. This is not a question that is unique to the tube MPC presented here, as a positive answer for this MPC formulation would likely imply similar results when considering linear systems with additive uncertainties, for instance. The importance of this is that if it is possible to give a confidence bound when samples are reused between time steps, then it is possible to construct recursively feasible MPC controllers that guarantee these bounds in closed-loop, and that have the advantage of sampling over the whole prediction horizon.

Here are some other questions to motivate extensions of the current work:

- Is it possible to use a nonlinear terminal control law (such as a piecewise linear feedback) for the tube MPC of LDI systems?
- Can the tube MPC of LPV systems be extended to consider bounds on rates of change of the uncertain parameter?
- For polytopic uncertainty, can the tube MPC be computed efficiently using multiparametric techniques?
- Does there exist an algorithm for computing polyhedral approximations to a chance-constrained set that does not require vertex enumeration?
- Is it possible to find maximal control admissible sets for systems with robust constraints on the number of constraint violations over time?

For now, though, we leave them unanswered.

11 Bibliography

[AG95] Pierre Apkarian and Pascal Gahinet. A convex characterization of gain-scheduled h-infinity controllers. *Automatic Control, IEEE Transactions on*, 40(5):853–864, 1995.

[AGB95] Pierre Apkarian, Pascal Gahinet, and Greg Becker. Self-scheduled H-infinity control of linear parameter-varying systems: a design example. *Automatica*, 31(9):1251–1261, 1995.

[Bat04] Ivo Batina. *Model predictive control for stochastic systems by randomized algorithms*. Technische Universiteit Eindhoven, 2004.

[BBBM05] Francesco Borrelli, Mato Baotić, Alberto Bemporad, and Manfred Morari. Dynamic programming for constrained optimal control of discrete-time linear hybrid systems. *Automatica*, 41(10):1709–1721, 2005.

[BBC11] Dimitris Bertsimas, David B Brown, and Constantine Caramanis. Theory and applications of robust optimization. *SIAM review*, 53(3):464–501, 2011.

[BBM00] Alberto Bemporad, Francesco Borrelli, and Manfred Morari. Piecewise linear optimal controllers for hybrid systems. In *American Control Conference, 2000. Proceedings of the 2000*, volume 2, pages 1190–1194. IEEE, 2000.

[BBM+02] Alberto Bemporad, Francesco Borrelli, Manfred Morari, et al. Model predictive control based on linear programming˜ the explicit solution. *IEEE Transactions on Automatic Control*, 47(12):1974–1985, 2002.

[BBM03] Alberto Bemporad, Francesco Borrelli, and Manfred Morari. Min-max control of constrained uncertain discrete-time linear systems. *Automatic Control, IEEE Transactions on*, 48(9):1600–1606, 2003.

[BCH98] Stephen Boyd, César Crusius, and Anders Hansson. Control applications of nonlinear convex programming. *Journal of Process Control*, 8(5):313–324, 1998.

[BCK14] Johannes Buerger, Mark Cannon, and Basil Kouvaritakis. An active set solver for input-constrained robust receding horizon control. *Automatica*, 50(1):155–161, 2014.

[BCK16] Johannes Buerger, Mark Cannon, and Basil Kouvaritakis. Active set solver for min-max robust control with state and input constraints. *International Journal of Robust and Nonlinear Control*, 2016.

[BEGFB94] Stephen P Boyd, Laurent El Ghaoui, Eric Feron, and Venkataramanan Balakrishnan. *Linear matrix inequalities in system and control theory*, volume 15. SIAM, 1994.

[Bel57] Richard Bellman. Dynamic Programming. *Princeton, NJ*, 1957.

[Ber05] Dimitri P Bertsekas. Dynamic programming and suboptimal control: A survey from ADP to MPC. *European Journal of Control*, 11(4):310–334, 2005.

[Bes10] Thomas Besselmann. *Constrained Optimal Control - Piecewise Affine and Linear Parameter-Varying Systems*. PhD thesis, ETH Zurich, 2010.

[Bit88] Georges Bitsoris. On the positive invariance of polyhedral sets for discrete-time systems. *Systems & control letters*, 11(3):243–248, 1988.

[Bla91] F Blanchini. Constrained control for uncertain linear systems. *Journal of optimization theory and applications*, 71(3):465–484, 1991.

[Bla99] Franco Blanchini. Survey paper: Set invariance in control. *Automatica (Journal of IFAC)*, 35(11):1747–1767, 1999.

[BLM12] Thomas Besselmann, Johan Löfberg, and Manfred Morari. Explicit MPC for LPV systems: Stability and optimality. *Automatic Control, IEEE Transactions on*, 57(9):2322–2332, 2012.

[BM07] Franco Blanchini and Stefano Miani. *Set-theoretic methods in control*. Springer, 2007.

[BMDP00] A. Bemporad, M. Morari, V. Dua, and E. N. Pistikopoulos. The explicit linear quadratic regulator for constrained systems. In *Proceedings of the American Contr. Conferenc*, pages 872–876. 2000.

[BMDP02] Alberto Bemporad, Manfred Morari, Vivek Dua, and Efstratios N Pistikopoulos. The explicit linear quadratic regulator for constrained systems. *Automatica*, 38(1):3–20, 2002.

[BO09] Lars Blackmore and Masahiro Ono. Convex chance constrained predictive control without sampling. In *Proceedings of the AIAA Guidance, Navigation and Control Conference*, pages 7–21, 2009.

[BSW+02] Ivo Batina, Anton Stoorvogel, Siep Weiland, et al. Optimal control of linear, stochastic systems with state and input constraints. In *Decision and Control, 2002, Proceedings of the 41st IEEE Conference on*, volume 2, pages 1564–1569. IEEE, 2002.

[BTN98] Aharon Ben-Tal and Arkadi Nemirovski. Robust convex optimization. *Mathematics of Operations Research*, 23(4):769–805, 1998.

[BV04] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[Cal10] Giuseppe Carlo Calafiore. Random convex programs. *SIAM Journal on Optimization*, 20(6):3427–3464, 2010.

[CCK06] Paul D Couchman, Mark Cannon, and Basil Kouvaritakis. Stochastic MPC with inequality stability constraints. *Automatica*, 42(12):2169–2174, 2006.

[CCKE14] Qifeng Cheng, Mark Cannon, Basil Kouvaritakis, and Martin Evans. Stochastic MPC for systems with both multiplicative and additive disturbances. In *Proceedings of the 19th IFAC World Congress*, pages 2291–2296, 2014.

[CDT11] Giuseppe C Calafiore, Fabrizio Dabbene, and Roberto Tempo. Research on probabilistic methods for control system design. *Automatica*, 47(7):1279–1293, 2011.

[CF13] Giuseppe C Calafiore and Lorenzo Fagiano. Stochastic model predictive control of LPV systems via scenario optimization. *Automatica*, 49(6):1861–1866, 2013.

[CFF02] Alessandro Casavola, Domenico Famularo, and Giuseppe Franzè. A feedback min-max MPC algorithm for LPV systems subject to bounded rates of change of parameters. *Automatic Control, IEEE Transactions on*, 47(7):1147–1153, 2002.

[CFF04] Alessandro Casavola, Domenico Famularo, and Giuseppe Franzé. Robust constrained predictive control of uncertain norm-bounded linear systems. *Automatica*, 40(11):1865–1876, 2004.

[CG08] Marco C Campi and Simone Garatti. The exact feasibility of randomized solutions of uncertain convex programs. *SIAM Journal on Optimization*, 19(3):1211–1230, 2008.

[CG11] Marco C Campi and Simone Garatti. A sampling-and-discarding approach to chance-constrained optimization: feasibility and optimality. *Journal of Optimization Theory and Applications*, 148(2):257–280, 2011.

[CGM02] Francesco A Cuzzola, Jose C Geromel, and Manfred Morari. An improved approach for constrained robust model predictive control. *Automatica*, 38(7):1183–1189, 2002.

[CK04] Marek Capinski and Peter E Kopp. Measure, integral and probability. 2004.

[CK05] Mark Cannon and Basil Kouvaritakis. Optimizing prediction dynamics for robust MPC. *Automatic Control, IEEE Transactions on*, 50(11):1892–1897, 2005.

[CKRC11] Mark Cannon, Basil Kouvaritakis, Saša V Raković, and Qifeng Cheng. Stochastic tubes in model predictive control with probabilistic constraints. *Automatic Control, IEEE Transactions on*, 56(1):194–200, 2011.

[CKW09a] Mark Cannon, Basil Kouvaritakis, and Xingjian Wu. Model predictive control for systems with stochastic multiplicative uncertainty and probabilistic constraints. *Automatica*, 45(1):167–172, 2009.

[CKW09b] Mark Cannon, Basil Kouvaritakis, and Xingjian Wu. Probabilistic constrained MPC for multiplicative and additive stochastic uncertainty. *Automatic Control, IEEE Transactions on*, 54(7):1626–1632, 2009.

[CM87] Peter J Campo and Manfred Morari. Robust model predictive control. In *American Control Conference, 1987*, pages 1021–1026. IEEE, 1987.

[D+84] John C Doyle et al. Lecture notes in advances in multivariable control. In *ONR/Honeywell workshop*, 1984.

[DCB14] Steven Diamond, Eric Chu, and Stephen Boyd. Cvxpy: A python-embedded modeling language for convex optimization, version 0.2, 2014.

[Die04] Moritz Diehl. Robust dynamic programming for min-max model predictive control of constrained uncertain systems. *Automatic Control, IEEE Transactions on*, 49(12):2253–2257, 2004.

[ECK12a] Martin Evans, Mark Cannon, and Basil Kouvaritakis. Linear stochastic MPC under finitely supported multiplicative uncertainty. In *American Control Conference (ACC), 2012*, pages 442–447. IEEE, 2012.

[ECK12b] Martin Evans, Mark Cannon, and Basil Kouvaritakis. Robust and stochastic linear MPC for systems subject to multiplicative uncertainty. In *4th IFAC Nonlinear Model Predictive Control Conference*, volume 4, pages 335–341, 2012.

[ECK15] Martin A Evans, Mark Cannon, and Basil Kouvaritakis. Robust MPC tower damping for variable speed wind turbines. *Control Systems Technology, IEEE Transactions on*, 23(1):290–296, 2015.

[EGOL98] Laurent El Ghaoui, Francois Oustry, and Hervé Lebret. Robust solutions to uncertain semidefinite programs. *SIAM Journal on Optimization*, 9:33–52, 1998.

[FC16] James Fleming and Mark Cannon. Admissible sets for chance-constrained difference inclusions. In *2016 American Control Conference, Boston, USA*, 2016.

[FCK14] James Fleming, Mark Cannon, and Basil Kouvaritakis. Stochastic tube MPC for LPV systems with probabilistic set inclusion conditions. In *53rd IEEE Conference on Decision and Control, Los Angeles, USA*, pages 4783–4788. IEEE, 2014.

[FCK16] James Fleming, Mark Cannon, and Basil Kouvaritakis. Stochastic MPC using probabilistic tubes and sample approximations of chance constraints. *To appear*, 2016.

[FKC13] James Fleming, Basil Kouvaritakis, and Mark Cannon. Regions of attraction and recursive feasibility in robust MPC. In *Control & Automation (MED), 2013 21st Mediterranean Conference on*, pages 801–806. IEEE, 2013.

[FKC15] James Fleming, Basil Kouvaritakis, and Mark Cannon. Robust tube MPC for linear systems with multiplicative uncertainty. *Automatic Control, IEEE Transactions on*, 60(4):1087–1092, 2015.

[GBY08] Michael Grant, Stephen Boyd, and Yinyu Ye. Cvx: Matlab software for disciplined convex programming, 2008.

[GKM06] Paul J Goulart, Eric C Kerrigan, and Jan M Maciejowski. Optimization over state feedback policies for robust control with constraints. *Automatica*, 42(4):523–533, 2006.

[Gou07] Paul James Goulart. *Affine feedback policies for robust control with constraints*. PhD thesis, University of Cambridge, 2007.

[GT91] Elmer G Gilbert and Kok Tin Tan. Linear systems with state and control constraints: The theory and application of maximal output admissible sets. *Automatic Control, IEEE Transactions on*, 36(9):1008–1020, 1991.

[Gur95] Leonid Gurvits. Stability of discrete linear inclusion. *Linear algebra and its applications*, 231:47–85, 1995.

[GZM+16] Sergio Grammatico, Xiaojing Zhang, Kostas Margellos, Paul Goulart, and John Lygeros. A scenario approach for non-convex control design. *IEEE Transactions on Automatic Control*, 61(2):334–345, 2016.

[HBP07] Keld Hammerum, Per Brath, and Niels Kjølstad Poulsen. A fatigue approach to wind turbine control. In *Journal of Physics: Conference Series*, volume 75, page 012081. IOP Publishing, 2007.

[K+60] Rudolf Emil Kalman et al. Contributions to the theory of optimal control. *Bol. Soc. Mat. Mexicana*, 5(2):102–119, 1960.

[Kal60] Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. *Journal of Fluids Engineering*, 82(1):35–45, 1960.

[KBM94] Mayuresli V Kothare, V Balakrishnan, and M Morari. Robust constrained model predictive control using linear matrix inequalities. In *American Control Conference, 1994*, volume 1, pages 440–444. IEEE, 1994.

[KBM96] Mayuresh V Kothare, Venkataramanan Balakrishnan, and Manfred Morari. Robust constrained model predictive control using linear matrix inequalities. *Automatica*, 32(10):1361–1379, 1996.

[KC15] Basil Kouvaritakis and Mark Cannon. *Model Predictive Control: Classical, Robust and Stochastic*. Springer, 2015.

[KCRC10] Basil Kouvaritakis, Mark Cannon, Saša V Raković, and Qifeng Cheng. Explicit use of probabilistic distributions in linear predictive control. *Automatica*, 46(10):1719–1724, 2010.

[KCT04] Basil Kouvaritakis, Mark Cannon, and V Tsachouridis. Recent developments in stochastic MPC and sustainable development. *Annual Reviews in Control*, 28(1):23–35, 2004.

[Ker01] Eric Colin Kerrigan. *Robust constraint satisfaction: Invariant sets and predictive control*. PhD thesis, University of Cambridge, 2001.

[KG96] Hassan K Khalil and JW Grizzle. *Nonlinear systems*, volume 3. Prentice Hall, New Jersey, 1996.

[KG98] Ilya Kolmanovsky and Elmer G Gilbert. Theory and computation of disturbance invariant sets for discrete-time linear systems. *Mathematical problems in engineering*, 4(4):317–367, 1998.

[KGBM04] Michal Kvasnica, Pascal Grieder, Mato Baotić, and Manfred Morari. Multi-parametric toolbox (MPT). In *Hybrid systems: computation and control*, pages 448–462. Springer, 2004.

[KGCO11] Milan Korda, Ravi Gondhalekar, Jiri Cigler, and Frauke Oldewurtel. Strongly feasible stochastic model predictive control. In *Decision and Control and European Control Conference (CDC-ECC), 2011*, pages 1245–1251. IEEE, 2011.

[KGOJ14] Milan Korda, Ravi Gondhalekar, Frauke Oldewurtel, and Colin N Jones. Stochastic MPC framework for controlling the average constraint violation. *Automatic Control, IEEE Transactions on*, 59(7):1706–1721, 2014.

[KM00] Eric C Kerrigan and Jan M Maciejowski. Invariant sets for constrained nonlinear discrete-time systems with application to feasibility in model predictive control. In *Decision and Control, 2000. Proceedings of the 39th IEEE Conference on*, volume 5, pages 4951–4956. IEEE, 2000.

[KMV13] William D Kalies, Konstantin Mischaikow, and Robert CAM Vandervorst. Lattice structures for attractors I. *arXiv preprint arXiv:1307.1737*, 2013.

[KRK+05] Koustas I Kouramas, SV Raković, Eric C Kerrigan, JC Allwright, and David Q Mayne. On the minimal robust positively invariant set for linear difference inclusions. In *Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC'05. 44th IEEE Conference on*, pages 2296–2301. IEEE, 2005.

[KRS00] Basil Kouvaritakis, J Anthony Rossiter, and Jan Schuurmans. Efficient robust predictive control. *Automatic Control, IEEE Transactions on*, 45(8):1545–1549, 2000.

[KSFD13] Michal Kvasnica, Alexander Szucs, Miroslav Fikar, and Jan Drgona. Explicit MPC of LPV systems in the controllable canonical form. In *Control Conference (ECC), 2013 European*, pages 1035–1040. IEEE, 2013.

[LA00] Yaohui Lu and Yaman Arkun. Quasi-min-max MPC algorithms for LPV systems. *Automatica*, 36(4):527–540, 2000.

[LADT15] Matthias Lorenzen, Frank Allgower, Fabrizio Dabbene, and Roberto Tempo. An improved constraint-tightening approach for stochastic MPC. In *American Control Conference (ACC), 2015*, pages 944–949. IEEE, 2015.

[Löf03a] Johan Löfberg. Approximations of closed-loop minimax MPC. In *Decision and Control, 2003. Proceedings. 42nd IEEE Conference on*, volume 2, pages 1438–1442. IEEE, 2003.

[Löf03b] Johan Löfberg. *Minimax approaches to robust model predictive control*, volume 812. Linköping University Electronic Press, 2003.

[lPMBA05] De la Pena, D Munoz, A Bemporad, and T Alamo. Stochastic programming applied to model predictive control. In *Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC'05. 44th IEEE Conference on*, pages 1361–1366. IEEE, 2005.

[lPMBF04] De la Peña, David Muñoz, A Bemporad, and Carlo Filippi. Robust explicit MPC based on approximate multi-parametric convex programming. In *Decision and Control, 2004. CDC. 43rd IEEE Conference on*, volume 3, pages 2491–2496. IEEE, 2004.

[Lö04] J. Löfberg. YALMIP : A toolbox for modeling and optimization in MATLAB. In *Proceedings of the CACSD Conference*, Taipei, Taiwan, 2004.

[Lö12] J. Löfberg. Automatic robust convex programming. *Optimization methods and software*, 27(1):115–129, 2012.

[MK87] Katta G Murty and Santosh N Kabadi. Some NP-complete problems in quadratic and nonlinear programming. *Mathematical programming*, 39(2):117–129, 1987.

[MKVWF11] David Q Mayne, Eric C Kerrigan, EJ Van Wyk, and P Falugi. Tube-based robust nonlinear model predictive control. *International Journal of Robust and Nonlinear Control*, 21(11):1341–1353, 2011.

[MR09] DQ Mayne and JB Rawlings. *Model Predictive Control: Theory and Design*. 2009.

[MRRS00] David Q Mayne, James B Rawlings, Christopher V Rao, and Pierre OM Scokaert. Constrained model predictive control: Stability and optimality. *Automatica*, 36(6):789–814, 2000.

[MSR05] David Q Mayne, María M Seron, and SV Raković. Robust model predictive control of constrained linear systems with bounded disturbances. *Automatica*, 41(2):219–224, 2005.

[NNY94] Yurii Nesterov, Arkadii Nemirovskii, and Yinyu Ye. *Interior-point polynomial algorithms in convex programming*, volume 13. SIAM, 1994.

[NS06] Arkadi Nemirovski and Alexander Shapiro. Convex approximations of chance constrained programs. *SIAM Journal on Optimization*, 17(4):969–996, 2006.

[O+12] Gurobi Optimization et al. Gurobi optimizer reference manual. *URL: http://www. gurobi. com*, 2012.

[OJM08] Frauke Oldewurtel, Colin N Jones, and Manfred Morari. A tractable approximation of chance constrained stochastic MPC based on affine disturbance feedback. In *Decision and Control, 2008. CDC 2008. 47th IEEE Conference on*, pages 4731–4736. IEEE, 2008.

[OPJ+10] Frauke Oldewurtel, Alessandra Parisio, Colin Jones, Manfred Morari, Dimitrios Gyalistras, Markus Gwerder, Vanessa Stauch, Beat Lehmann, and Katharina Wirth. Energy efficient building climate control using stochastic model predictive control and weather predictions. In *Proceedings of the 2010 American control conference*, pages 5100–5105, 2010.

[PBG62] LS Pontryagin, VG Boltyanskii, and RV Gamkrelidze. The mathematical theory of optimal processes. *New York*, 1962.

[PRSDM05a] B Pluymers, JA Rossiter, J Suykens, and B De Moor. Reduced complexity polyhedral invariant sets for LPV systems. *Katholieke Universiteit Leuven, Leuven, B, Internal Report*, pages 04–230, 2005.

[PRSDM05b] B Pluymers, JA Rossiter, JAK Suykens, and Bart De Moor. The efficient computation of polyhedral invariant sets for linear systems with polytopic uncertainty. In *American Control Conference (ACC), 2005*, pages 804–809. IEEE, 2005.

[PRSDM05c] B Pluymers, JA Rossiter, JAK Suykens, and Bart De Moor. A simple algorithm for robust MPC. In *Proceedings of the IFAC World Congress*, 2005.

[PRSDM05d] Bert Pluymers, JA Rossiter, JAK Suykens, and Bart De Moor. Interpolation based MPC for LPV systems using polyhedral invariant sets. In *American Control Conference, 2005. Proceedings of the 2005*, pages 810–815. IEEE, 2005.

[RC13] Saša V Rakovic and Qifeng Cheng. Homothetic tube MPC for constrained linear difference inclusions. In *Control and Decision Conference (CCDC), 2013 25th Chinese*, pages 754–761. IEEE, 2013.

[RKC+12] Saša V Raković, Basil Kouvaritakis, Mark Cannon, Christos Panos, and Rolf Findeisen. Parameterized tube model predictive control. *Automatic Control, IEEE Transactions on*, 57(11):2746–2761, 2012.

[RKFC12] Saša V Raković, Basil Kouvaritakis, Rolf Findeisen, and Mark Cannon. Homothetic tube model predictive control. *Automatica*, 48(8):1631–1638, 2012.

[RKKM05] SV Raković, Eric C Kerrigan, Konstantinos I Kouramas, and David Q Mayne. Invariant approximations of the minimal robust positively invariant set. *Automatic Control, IEEE Transactions on*, 50(3):406–410, 2005.

[Roc70] R Tyrrell Rockafellar. Convex analysis. 1970.

[SFFM14] Georg Schildbach, Lorenzo Fagiano, Christoph Frei, and Manfred Morari. The scenario approach for stochastic model predictive control with bounds on closed-loop constraint violations. *Automatica*, 50(12):3009–3018, 2014.

[SFM13] Georg Schildbach, Lorenzo Fagiano, and Manfred Morari. Randomized solutions to convex programs with multiple chance constraints. *SIAM Journal of Optimization*, 23(4):2479–2501, 2013.

[SGC97] Carsten Scherer, Pascal Gahinet, and Mahmoud Chilali. Multiobjective output-feedback control via LMI optimization. *Automatic Control, IEEE Transactions on*, 42(7):896–911, 1997.

[SN99] Alexander T Schwarm and Michael Nikolaou. Chance-constrained model predictive control. *AIChE Journal*, 45(8):1743–1752, 1999.

[SR98] Pierre OM Scokaert and James B Rawlings. Constrained linear quadratic regulation. *Automatic Control, IEEE Transactions on*, 43(8):1163–1169, 1998.

[Stu99] Jos F Sturm. Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. *Optimization methods and software*, 11(1-4):625–653, 1999.

[TCD12] Roberto Tempo, Giuseppe Calafiore, and Fabrizio Dabbene. *Randomized algorithms for analysis and control of uncertain systems: with applications*. Springer Science & Business Media, 2012.

[TJB03] Petter Tøndel, Tor Arne Johansen, and Alberto Bemporad. Evaluation of piecewise affine control via binary search tree. *Automatica*, 39(5):945–950, 2003.

[VHSB01] DH Van Hessem, CW Scherer, and OH Bosgra. LMI-based closed-loop economic optimization of stochastic process operation under state and input constraints. In *Decision and Control, 2001. Proceedings of the 40th IEEE Conference on*, volume 5, pages 4228–4233. IEEE, 2001.

[VPKGM13] Bart Van Parys, Daniel Kuhn, Paul Goulart, and Manfred Morari. Distributionally robust control of constrained stochastic systems. 2013.

[WSS04] Nobutaka Wada, Koji Saito, and Masami Saeki. Model predictive control for linear parameter varying systems using parameter dependent Lyapunov function. In *Circuits and Systems, 2004. MWSCAS'04. The 2004 47th Midwest Symposium on*, volume 3, pages iii–133. IEEE, 2004.

[WYPB95] Fen Wu, Xin Hua Yang, Andy Packard, and Greg Becker. Induced L2-norm control for LPV system with bounded parameter variation rates. In *American Control Conference, Proceedings of the 1995*, volume 3, pages 2379–2383. IEEE, 1995.

[YJBJ+76] Dante C Youla, Hamid Jabr, Joseph J Bongiorno Jr, et al. Modern Wiener-Hopf design of optimal controllers–part II: The multivariable case. *Automatic Control, IEEE Transactions on*, 21(3):319–338, 1976.

[ZGS+15] Xiaojing Zhang, Sergio Grammatico, Georg Schildbach, Paul Goulart, and John Lygeros. On the sample size of random convex programs with structured dependence on the uncertainty. *Automatica*, 60:182–188, 2015.

[ZM93] Zhi Q Zheng and Manfred Morari. Robust stability of constrained model predictive control. In *American Control Conference, 1993*, pages 379–383. IEEE, 1993.