Explicit robust constrained control for linear systems : analysis, implementation and design based on optimization

Ngoc Anh Nguyen

NNT : 2015SACLC012

THESE DE DOCTORAT
DE
L'UNIVERSITE PARIS-SACLAY
PREPAREE A
"CENTRALESUPELEC"

ECOLE DOCTORALE N° 580 STIC
Sciences et technologies de l'information et de la communication

Spécialité de doctorat Automatique

Par

Mr. Ngoc Anh NGUYEN

Explicit robust constrained control for linear systems : analysis, implementation and design based on optimization

Thèse présentée et soutenue à Gif-sur-Yvette le 26th Novembre 2015:

Composition du Jury :

Professor, Miroslav Fikar	Slovak University of Technology in Bratislava	Rapporteur
Doctor, Maria Seron	University of Newcastle	Rapportrice
Professor, Tor Arne Johansen	Norwegian University of Science and Technology	Examinateur
Senior Researcher, Luca Zaccarian	LAAS--CNRS, Toulouse	Examinateur
Associate Professor, Vivek Dua	University College London	Examinateur
Senior Researcher, Antoine Girard	Laboratoire de Signaux et Systèmes	Président
Professor, Sorin Olaru	Laboratoire de Signaux et Systèmes, CentraleSupelec	Directeur de thèse
Professor, Pedro Rodriguez-Ayerbe	Laboratoire de Signaux et Systèmes, CentraleSupelec	Co-encadrant de thèse

To my family

Nothing is nothing

Abstract

Piecewise affine (PWA) feedback control laws have received significant attention due to their relevance for the control of constrained systems, hybrid systems; equally for the approximation of nonlinear control. However, they are often associated with serious implementation issues. Motivated by the challenges on this class of particular controllers, this thesis is mostly related to their analysis and design.

The first part of this thesis aims to compute the robustness and fragility margins for a given PWA control law and a linear discrete-time system. More precisely, the robustness margin is defined as the set of linear time-varying systems such that the given PWA control law keeps the trajectories inside a given feasible set. On a different perspective, the fragility margin contains all the admissible variations of the control law coefficients such that the positive invariance of the given feasible set is still guaranteed. It will be shown that if the given feasible set is a polytope, then so are these robustness/fragility margins.

The second part of this thesis focuses on inverse optimality problem for the class of PWA controllers. Namely, the goal is to construct an optimization problem whose optimal solution is equivalent to the given PWA function. The methodology is based on *convex lifting* : an auxiliary $1-$dimensional variable which enhances the convexity characterization into recovered optimization problem. Accordingly, if the given PWA function is continuous, the optimal solution to this reconstructed optimization problem will be shown to be unique. Otherwise, if the continuity of this given PWA function is not fulfilled, this function will be shown to be one optimal solution to the recovered problem.

In view of applications in linear model predictive control (MPC), it will be shown that *any continuous PWA control law can be obtained by a linear MPC problem with the prediction horizon at most equal to 2 prediction steps*. Aside from the theoretical meaning, this result can also be of help to facilitate implementation of PWA control laws by avoiding storing state space partition.

Another utility of *convex liftings* will be shown in the last part of this thesis to be a control Lyapunov function. Accordingly, this convex lifting will be deployed in the so-called *robust control design based on convex liftings* for linear system affected by bounded additive disturbances and polytopic uncertainties. Both implicit and explicit controllers can be obtained. This method can also guarantee the recursive feasibility and robust stability. However, this control Lyapunov function is only defined over the maximal $\lambda-$contractive set for a given $0 \leq \lambda < 1$ which is known to be smaller than the maximal controllable set. Therefore, an extension of the above method to the $N-$steps controllable set will be presented. This method is based on *a cascade of convex liftings* where an auxiliary variable will be used to emulate a Lyapunov function. Namely, this variable will be shown to be non-negative, to strictly decrease for N first steps and to stay at 0 afterwards. Accordingly, robust stability is sought.

Les lois de commande affines par morceaux ont attiré une grande attention de la communauté Automatique grâce à leur pertinence pour la commande de systèmes sous contraints ou hybrides ; mais également pour l'approximation de lois de commande nonlinéaires. Pourtant, leur mise en oeuvre est soumise à quelques difficultés. Motivé par les défis liés à cette classe de commandes, cette thèse porte sur leur analyse, mise en oeuvre et synthèse.

La première partie de cette thèse a pour but le calcul de la marge de robustesse et de la marge de fragilité pour une loi de commande affine par morceaux prédéfinie et un système linéaire discret. Plus précisément, la marge de robustesse est définie comme l'ensemble des systèmes linéaires à paramètres variants que la loi PWA donnée peut commander en boucle fermée pour maintenir les trajectoires dans la région faisable. D'ailleurs, la marge de fragilité comprend toutes les variations des coefficients de la commande donnée telle que l'invariance de la région faisable soit encore garantie. Il est montré que si la région faisable donnée est un polytope, ces marges sont aussi des polytopes.

La deuxième partie de ce manuscrit est consacrée au problème de l'optimalité inverse pour la classe des fonctions affines par morceaux. C'est-à-dire, l'objective est de définir un problème d'optimisation pour lequel la solution optimale est équivalente à la fonction affine par morceaux donnée. La méthodologie est fondée sur le *convex lifting*, i.e., un variable auxiliaire, scalaire, qui permet de définir un ensemble convex à partir de la partition de paramètres de la fonction affine par morceaux donnée. Il est montré que si la fonction affine par morceaux donnée est continue, la solution optimale de ce problème redéfini sera unique. Par contre, si la continuité n'est pas satisfaite, cette fonction affine par morceaux sera une solution optimale parmi les autres du problème redéfini.

En ce qui concerne l'application dans la commande prédictive, il sera montré que n'importe quelle loi de commande affine par morceaux continue peut être obtenue par un autre problème de commande prédictive avec l'horizon de prédiction au plus égal à 2. A côté de cet aspect théorique, ce résultat sera utile pour faciliter la mise en oeuvre des lois de commandes affines par morceaux en évitant l'enregistrement de la partition de l'espace d'état.

Dans la dernière partie de ce manuscrit, une famille de *convex liftings* servira comme des fonctions de Lyapunov. En conséquence, ces "convex liftings" seront déployés pour synthétiser des lois de commande robustes pour des systèmes linéaires incertains, également en présence de perturbations additives bornées. Des lois implicites et explicites seront obtenues en même temps. Cette méthode permet de guarantir la faisabilité récursive et la stabilité robuste. Cependant, cette fonction de Lyapunov est limitée à l'ensemble $\lambda-$contractive maximal avec une constante scalaire $0 \leq \lambda < 1$, qui est plus petit que l'ensemble controllable maximal. Pour cette raison, une extension de cette méthode pour l'ensemble controllable de $N-$pas, sera présentée. Cette méthode est fondée sur des convex liftings en cascade où une variable auxiliaire sera utilisée pour servir comme une fonction de Lyapunov. Plus précisément, cette variable est non-négative, strictement décroissante pour les N premiers pas et égale toujours à 0 après. Par conséquent, la stabilité robuste est guarantie.

Résumé

Cette thèse s'intéresse à l'analyse, à la mise en oeuvre et à la synthèse des lois de commandes affines par morceaux. Tout d'abord, la marge de robustesse et la marge de fragilité de la loi de commande donnée sont définies. La marge de robustesse permet de définir un ensemble de modèles autour du modèle nominal guarantissant l'invariance de la région faisable et la marge de fragilité permet de définir un ensemble d'erreurs de quantification de la loi de commande lors de la mise en oeuvre permettant de guarantir également l'invariance de la région faisable.

Afin de réduire la mémoire nécessaire pour la mise en oeuvre des lois de commande explicites (affines par morceaux), la deuxième partie de cette thèse s'intéresse à la construction d'un problème d'optimisation équivalent permettant d'obtenir la même loi de commande donnée mais avec une representation plus compacte. Ce problème d'optimalité inverse est résolu grâce à l'utilisation des convex liftings.

Finalement, il est connu que la commande prédictive est conservative et que la solution optimale devient plus complexe quand l'horizon de prédiction croît. Par conséquent, c'est nécessaire de proposer une méthode simple de synthèse des commandes robustes capable de faire face à des incertitudes polytopiques et à perturbations additives bornées.

Tous les problèmes mentionnés au-dessus seront adressés dans cette thèse. La contribution de ces travaux est esquissé au dessous :

— une analyse de robustesse et de fragilité pour une loi de commande affine par morceaux et un système linéaire ;
— solution du problème d'optimalité inverse fondée sur le *convex lifting* ;
— synthèse de commandes robustes pour des systèmes linéaires affectés par des incertitudes polytopiques et des perturbations additives bornées.

Chacune est présentée en détail par la suite.

Marges de robustesse et de fragilité explicites pour commandes affines par morceaux

A partir de l'analyse, ce chapitre considère le calcul des marges de robustesse et de fragilité pour une commande affine par morceaux donnée et un système linéaire discret. L'idée est fondée sur le principe de l'invariance positive. Autrement dit, étant donnée une commande affine par morceaux, définie sur une région feasable $\mathcal{X}$:

$$f_{pwa}(x) = H_i x + G_i \quad \text{pour} \quad x \in \mathcal{X}_i, \tag{1}$$

et un système linéaire variant

$$x_{k+1} = A(k)x_k + B(k)u_k$$

où

$$[A(k)\ B(k)] \in \Psi = \operatorname{conv}\left\{[A_1\ B_1], \ldots, [A_L\ B_L]\right\}, \tag{2}$$

le problème de robustesse a pour but de trouver l'ensemble des systèmes linéaires $\Psi_{\text{rob}} \subseteq \Psi$ tel que :

$$(A(k) + B(k)H_i)x + B(k)G_i \in \mathcal{X} \quad \forall x \in \mathcal{X}_i \ \text{ et } \ \forall\,[A(k)\ B(k)] \in \Psi_{\text{rob}}.$$

Il est montré que Ψ_{rob} est un polytope. Son calcul peut être effectué par deux approches diférentes ; i.e. la représentation par les sommets et la représentation par les demi-espaces.

De même, le problème de fragilité vise à calculer l'ensemble des variations des coefficients de la commande affine par morceaux donnée tel que la positive invariance de $\mathcal{X}$ soit encore assurée. A différence du problème de robustesse, le problème de fragilité considère un système linéaire invariant :

$$x_{k+1} = Ax_k + Bu_k \tag{3}$$

et une commande affine par morceaux donnée comme (1). La commande mise en oeuvre est écrite dans la forme ci-dessous ;

$$f_{pwa}(x) = (H_i + \delta_{H_i})x + (G_i + \delta_{G_i}) \quad \text{pour} \quad x \in \mathcal{X}_i. \tag{4}$$

D'après la définition, le problème de fragilité est équivalent à trouver l'ensemble de $(\delta_{H_i}, \delta_{G_i})$, décrit par Δ_i^u tel que :

$$(A + B(H_i + \delta_{H_i}))x + B(G_i + \delta_{G_i}) \in \mathcal{X}, \ \forall x \in \mathcal{X}_i \ \text{ et } \ \forall\ (\delta_{H_i}, \delta_{G_i}) \in \Delta_i^u.$$

Il est montré que Δ_i^u décrit un polyèdre. Son calcul explicite est utile dans la mise en oeuvre des commandes explicites sous une représentation des coefficients avec précision finie.

Fondée sur la même philosophie, ce problème peut s'étendre calcul de l'ensemble de perturbations additives pour un système linéaire invariant :

$$x_{k+1} = Ax_k + Bu_k + w_k, \tag{5}$$

en boucle fermée avec la commande affine par morceaux (1) lors que l'invariance positive de $\mathcal{X}$ est conservée.

Finalement, les erreurs admissibles pour la description des régions dans la partition de l'espace d'état seront adressées. Ce problème est considéré indépendamment des problèmes de fragilité des coefficients de la loi explicite ; c'est-à-dire, en considérant que les coefficients de la loi de commande donnée n'ont pas d'erreurs numériques. Considérer à la fois les erreurs sur la description des régions et des coefficients de la commande, fait perdre la linéarité de la formulation et le calcul des marges.

Optimalité inverse fondée sur le convex lifting

Il est connu que un problème linéaire/quadratique paramétré est écrit sous la forme suivante :

$$\mathbf{u}^*(x) = \arg\min_{\mathbf{u}} J(\mathbf{u}, x) \quad \text{s.t.} \quad G\mathbf{u} \leq Ex + W,$$

où $\mathbf{u} \in \mathbb{R}^{d_\mathbf{u}}$ représente le variable de décision et $x \in \mathbb{R}^{d_x}$ signifie le paramètre. De plus, la fonction de coût prend la forme suivante :

$$J(\mathbf{u}, x) = \mathbf{u}^T H \mathbf{u} + (x^T D + C)\mathbf{u},$$

où $H = H^T$ est semidéfinie positive. Si $H = 0$ et $D = 0$, ce problème d'optimisation devient un problème de programmation linéaire paramétré.

La solution optimale du problème ci-dessus est une fonction affine par morceaux définie sur une partition polyhédrale $\{\mathcal{X}_i\}_{i \in \mathcal{I}_N}$ i.e.

$$\mathbf{u}^*(x) = F_i x + G_i \quad \text{pour} \quad x \in \mathcal{X}_i.$$

Le problème d'optimalité inverse pour la famille des fonctions affines par morceaux, appelé différemment par le problème de programmation linéaire ou quadratique paramétré inverse, a pour but de construire un problème d'optimisation dont la solution optimale est équivalente à la fonction affine par morceaux donnée ; c'est-à-dire chercher des matrices $H_x, H_\mathbf{u}, H_z, K$ et une fonction de coût $J(x, z, \mathbf{u})$ tel que $\mathbf{u}^*(x)$ soit un sous-composant de la solution optimale du problème d'optimisation suivant :

$$\min_{[z^T\ \mathbf{u}^T]^T} J(x, z, \mathbf{u}) \quad \text{s.t.} \quad H_z z + H_\mathbf{u}\mathbf{u} + H_x x \leq K,$$

où z décrit une variable auxiliaire. Il est montré que la solution de tel problème d'optimalité inverse est fondée sur le concept de *convex lifting*. En conséquence, il sera montré que $z \in \mathbb{R}$ est suffisant pour définir ce problème inverse.

Par ailleurs, pour les fonctions affines par morceaux continues, il sera montré que la solution optimale de ce problème d'optimalité inverse est unique. Par contre, pour la famille des fonctions affines par morceaux discontinues, l'unicité de la solution optimale du problèm d'optimisation reconstruit sera perdue. Plus précisément, il sera montré que la fonction affine par morceaux donnée est une parmi ses autre solutions optimales.

Synthèse de commande prédictive

Ce chapitre commence par une application de l'optimalité inverse dans la commande prédictive linéaire. Un résultat important pour la commande prédictive linéaire sera exposé : *n'importe quelle commande affine par morceaux peut être obtenue par un problème de commande prédictive linéaire avec l'horizon au plus égale à 2.*

A côté de l'aspect théorique, ce résultat est aussi utile pour faciliter la mise en oeuvre des lois de commande affines par morceaux, en particulier obtenues par la programmation quadratique paramétrée et permet d'éviter le stockage des partitions de l'espace d'état. Cela permet la mise en oeuvre des lois de commande affines par morceaux dans des calculateurs moins chers.

La deuxième partie de ce chapitre présente une méthode de synthèse des commandes robustes fondée sur des *convex liftings* qui sont définie sur l'ensemble maximal $\lambda-$contractive comme une estimation du domain d'attraction pour un $\lambda \in [0, 1)$ donné. Tels *convex liftings* servent comme des fonctions de Lyapunov et seront utilisés plus tard pour synthétiser des lois de commande robustes pour des systèmes linéaires affectés par des incertitudes polytopiques et perturbations additives bornées. Ces lois de commande peuvent être obtenues sous les formes implicite et explicite. Il sera montré que la méthode proposée peut guarantir la feasabilité récursive et la stabilité robuste. Pour la synthèse implicite, la formulation est simple et facile à mettre en oeuvre ; plus précisément, à chaque temp d'échantillonage, elle demande la résolution d'un problème de programmation linéaire. Une limitation de cette méthode est de calculer tout d'abord l'ensemble maximal $\lambda-$contractive, puisqu'il exige une procédure répétitive et peut demander du temps de calcul considérable. Heureusement, ce calcul est fait hors ligne. Cependant, cet ensemble maximal $\lambda-$contractive est plus petit que l'ensemble maximal controllable.

Comme une généralisation de la synthèse ci-dessus, la troisième partie de ce chapitre présente une nouvelle synthèse des commandes sur l'ensemble de $N-$pas

controllable. Cette méthode est fondée sur des convex liftings en cascade. Elle utilise une nouvelle variable auxiliaire qui sert comme une fonction de Lyapunov ; i.e. elle est non-négative, décroît strictement jusqu'au pas $N+1$ et reste à 0 par la suite.

Acknowledgments

This thesis's completion has been supported from many people to whom I would like to express my gratitude and my warmest thanks.

My first thanks go to my supervisors Prof. Sorin OLARU and Prof. Pedro RODRIGUEZ-AYERBE for their constant supports, ideas and precious discussions. Throughout three years, their ideas have channeled me in right direction and supported my first steps in the research world. Also, they were always beside me when I met difficulties to encourage and to overcome together these obstacles. Their comments and discussions on my first ideas were really helpful to correct and improve them. I'm indebted to them for all these supports. Only some words are few to say all these.

My thanks are due to Prof. Miroslav FIKAR and Dr. Maria SERON for their time spent on the evaluation of this thesis. Also, my thanks are given to Prof. Tor Arne JOHANSEN, Prof. Luca ZACCARIAN, Associate Prof. Vivek DUA, Prof. Antoine GIRARD for their acceptance to be examiners of my thesis.

I would like to thank Dr. Michal KVASNICA for allowing me to visit his laboratory where I learned many things from the works by Juraj, Balint, Martin... My thanks also go to Dr. Martin GULAN for fruitful discussions and for welcoming me during my stay in Bratislava. I still remember cabbage soup from Prof. Boris. Rohal'-Ilkiv.

Also, I would like to thank Prof. Morten HOVD for hosting me in NTNU and for his valuable comments on our joint works.

This thesis is built in a pleasant environment of work where many colleagues e.g. Djawad, Mohamad, Sofiane, Miassa, Kate, Nikola, Vasso, Rajesh, Seif, Mohammed, Abid, Iris, Marjorie... have together shared experiences and knowledges in works and life. They made my life in CentraleSupélec easier.

Also, my thanks go to the former and current secretaries of this department: Josiane and Pascale for helping me a lot in administration in order that I can go for conferences and get refund.

I would like to thank also Prof. Eric ZAMAÏ and Prof. Laurent LEFÈVRE for giving me an opportunity to study in Grenoble INP.

I will never forget the happy moments in Valence where my second family stays. I would use this occasion to express my sincerest thanks to "hai Bac, chi

Trang" for everything they did for me. My thanks also go to my Vietnamese friends: chi Vy, Trang, anh Son, chi Ha, Tri, Tuan Anh, Hung, anh Lai, Hoang Anh, em Linh, em Vuong. They have helped me a lot particularly for my two first years in France. Also, lunches become interesting with the other Vietnamese friends in CentraleSupélec: anh Nhat, anh Dung, anh Thanh, anh Binh, Quang, anh Hieu, anh Bien.

My thanks also go to my classmates at Hanoi University of Science and Technology who live or lived in Paris: Minh, Duong, Cuong, Khai.

A special thank goes to Tra Mi for being always by my side in any situation.

Overall, i would express my gratitude to my parents and my sister for everything. I'm sorry for being so far from you for more than 5 years. I don't know how long this situation still continues. Life is really hard.

Finally, i would like to thank the life for letting me see kind people who have helped to change my mind about everything of this life.

Contents

Notation

For a conventional use, throughout this manuscript,

— $\mathbb{R}$ denotes the set of real numbers,
— $\mathbb{R}_+$ denotes the set of non-negative real numbers,
— $\mathbb{R}_{>0}$ denotes the set of strictly positive real numbers,
— $\mathbb{N}$ denotes the set of non-negative integers,
— $\mathbb{N}_{>0}$ denotes the set of positive integers.

Given $N \in \mathbb{N}_{>0}$, by $\mathcal{I}_N$, we denote the set of indices as follows:

$$\mathcal{I}_N = \{i \in \mathbb{N}_{>0} \mid i \leq N\}.$$

Given two points $x, y \in \mathbb{R}^d$, we use $\langle x, y\rangle$ to denote the inner product of these two points i.e. $\langle x, y\rangle = x^T y = y^T x$. By $\rho(x, y)$, we denote the Euclidean distance between these two points. Also we write $\rho_X(x)$ to denote the distance between a point $x \in \mathbb{R}^d$ and a set $X \subset \mathbb{R}^d$. It can be mathematically defined as follows:

$$\rho_X(x) = \inf_{y \in X} \rho(x, y).$$

Given a function $f : X \subset \mathbb{R}^m \to \mathbb{R}^n$, by $f(X)$ we denote the set defined as follows:

$$f(X) := \{y \in \mathbb{R}^n \mid \exists x \in X \text{ s.t. } y = f(x)\}.$$

Also, if $f(x) = Ax$ for $x \in X$ and a matrix of appropriate dimension A, then we write $AX = f(X)$.

Given two sets $S_1, S_2 \subseteq \mathbb{R}^d$, we write $S_1 \backslash S_2$ to denote the set of points which belong to S_1 but don't belong to S_2; i.e.

$$S_1 \backslash S_2 := \left\{x \in \mathbb{R}^d \mid x \in S_1, x \notin S_2\right\}.$$

Given two points $x, y \in \mathbb{R}^d$, and $\lambda \in \mathbb{R}$, then $\lambda x + (1 - \lambda)y$ is called *an affine combination* of two points x, y. Also, the set of all affine combinations of x, y is the line:

$$L(x, y) := \{\lambda x + (1 - \lambda)y \mid \lambda \in \mathbb{R}\}.$$

Similarly, $\mathrm{aff}(\mathcal{X})$ called the *affine hull* of $\mathcal{X}$, denotes the set of all affine combinations of elements in $\mathcal{X}$, i.e.,

$$\mathrm{aff}(\mathcal{X}) = \left\{ \sum_{i=1}^{k} \lambda_i x_i \mid k \in \mathbb{N}_{>0}, x_i \in \mathcal{X}, \lambda_i \in \mathbb{R}, \sum_{i=1}^{k} \lambda_i = 1 \right\}.$$

$\dim(\mathcal{X})$ denotes the dimension of $\mathrm{aff}(\mathcal{X})$.

Particularly, a *convex combination* of two points $x, y \in \mathbb{R}^d$ is described by

$$\lambda x + (1-\lambda)y, \text{ for a given } \lambda \in [0,\ 1]\,.$$

Moreover, the *convex hull* of a set $\mathcal{X} \subset \mathbb{R}^d$ denoted by $\mathrm{conv}(\mathcal{X})$, represents the set of all convex combinations of all finite subsets of $\mathcal{X}$. In other words,

$$\mathrm{conv}(\mathcal{X}) := \left\{ \sum_{i=1}^{n} \lambda_i x_i \mid \lambda_i \in \mathbb{R}_+,\ x_i \in \mathcal{X},\ \forall i \in \mathcal{I}_n,\ \sum_{i=1}^{n} \lambda_i = 1,\ n \in \mathbb{N}_{>0} \right\},$$

meaning the smallest convex set which contains $\mathcal{X}$.

— $\mathrm{int}(\mathcal{X})$ denotes the interior of $\mathcal{X}$.
— $\mathrm{bd}(\mathcal{X})$ denotes the set of points which lie on the boundary of a closed set $\mathcal{X}$.
— $\mathrm{cl}(\mathcal{X})$ denotes the *closure* of $\mathcal{X}$.
— $\mathrm{Card}(\mathcal{X})$ denotes the *cardinal number* of $\mathcal{X}$.
— $\mathrm{ext}(\mathcal{X})$ denotes the set of *extreme points* of a convex set $\mathcal{X}$.

Given $P_1, P_2 \subset \mathbb{R}^d$, $P_1 \oplus P_2$ denotes the Minkowski sum of P_1, P_2 and is defined as follows:

$$P_1 \oplus P_2 := \left\{ y_1 + y_2 \in \mathbb{R}^d \mid y_1 \in P_1,\ y_2 \in P_2 \right\}.$$

1 (0) denotes a vector of appropriate dimension with the elements equal to 1 (0). Also, 1_n (0_n) implies that $1\,(0) \in \mathbb{R}^n$. I denotes an identity matrix. Further, I_n means that $I \in \mathbb{R}^{n\times n}$.

Given two vectors $x = [x_i] \in \mathbb{R}^d$, $y = [y_i] \in \mathbb{R}^d$, the relation $x \leq y$ implies that $x_i \leq y_i$, $\forall i \in \mathcal{I}_d$. Otherwise a positive (semi)definite matrix $A \in \mathbb{R}^{n\times n}$ is denoted by $A(\geq) > 0$.

Given $x = [x_i] \in \mathbb{R}^d$, by $\|x\|_p$, we denote the $p-$norm of vector x. Namely, some special norms usually employed in this thesis are defined below

— $p = 1$, $\|x\|_1 = \sum_{i=1}^{d} |x_i|$,
— $p = 2$, $\|x\|_2 = \sqrt{x^T x}$,
— $p = \infty$, $\|x\|_\infty = \max\limits_{i \in \mathcal{I}_d} |x_i|$.

For a given matrix $A \in \mathbb{R}^{m\times n}$, then $\mathrm{vec}(A)$ denotes the vector composed of the columns of matrix A, i.e.

$$\mathrm{vec}(A) = \left[A^T(\cdot,1)\dots A^T(\cdot,n)\right]^T \in \mathbb{R}^{mn},$$

where $A(\cdot,i)$ denotes the i^{th} column of matrix A. Also, $A(j,\cdot)$ denotes the j^{th} row of matrix A.

Also, given two matrices $A = [a_{ij}] \in \mathbb{R}^{m\times n}$, $B \in \mathbb{R}^{p\times q}$, their Kronecker tensor product is denoted by $A \otimes B$ and is defined as follows:

$$A \otimes B := \begin{bmatrix} a_{11}B & \dots & a_{1n}B \\ \vdots & \ddots & \vdots \\ a_{m1}B & \dots & a_{mn}B \end{bmatrix}.$$

Given a set of finite points $S = \{s_1,\dots,s_n\} \subset \mathbb{R}^d$, then $[S]$ is used to denote a matrix whose columns correspond to the elements of S in an arbitrary order; i.e.

$$[S] = [s_1 \dots s_n] \in \mathbb{R}^{d\times n}.$$

The unit simplex denoted by $S_L \subset \mathbb{R}^L$, is defined as follows:

$$\mathcal{S}_L := \left\{x \in \mathbb{R}^L \mid x \geq 0,\ 1_L^T x = 1\right\}. \tag{6}$$

If P denotes a full-dimensional polyhedron in $\mathbb{R}^d$, then throughout this manuscript $\mathcal{V}(P)$ denotes the set of its vertices, $\mathcal{R}(P)$ denotes the set of its extreme rays, $\mathcal{F}(P)$ denotes the set of its facets.

If $\mathcal{S} \subset \mathbb{R}^d$ denotes a full-dimensional set and $\mathbb{S}$ denotes a subspace of $\mathbb{R}^d$, we use $\mathrm{Proj}_{\mathbb{S}}\,\mathcal{S}$ to denote the orthogonal projection of $\mathcal{S}$ onto the subspace $\mathbb{S}$.

Chapter 1

Introduction

Mathematical optimization has a longstanding history and becomes nowadays an important branch of mathematics. Its objective is to find a best element with respect to some criteria by minimizing or maximizing them subject to a set of constraints. The development of mathematical optimization has created new vectors of development for related researches, particularly in applied mathematical domains: e.g. computer science, economy, management science... In control theory, the first seeds of mathematical optimization are known to be planted by Pontryagin and Richard Bellman. Over several decades, the so-called *optimal control* field has bloomed with many sub-areas and becomes a very active research branch.

The most popular form of optimal control is called *linear quadratic control* i.e. minimizing a quadratic cost function. In the continuous time domain, it can be written in the following form:

$$\min_{u(t)} \frac{1}{2}x^T(T_f)P(T_f)x(T_f) + \frac{1}{2}\int_{T_0}^{T_f}(x^T(t)Qx(t) + u^T(t)Ru(t))dt \tag{1.1}$$

subject to a linear time invariant system

$$\dot{x}(t) = Ax(t) + Bu(t), \tag{1.2}$$

and the initial condition $x(T_0) = x_0$. Note that (A, B) is assumed to be controllable and Q, R are positive semidefinite and positive definite matrices, respectively. These matrices are also called weighting matrices. T_f denotes the final time. Also, $\frac{1}{2}x^T(T_f)Px(T_f)$ denotes the terminal cost function.

Particularly, the infinite horizon problem minimizes an infinite time cost function i.e.

$$J = \frac{1}{2}\int_{T_0}^{\infty}(x^T(t)Qx(t) + u^T(t)Ru(t))dt. \tag{1.3}$$

This infinite horizon problem has the solution in the following form

$$u(t) = -Kx(t),$$

where K is the solution to the algebraic Ricatti equation

$$\begin{aligned} K &= R^{-1}B^TP \\ 0 &= -PA - A^TP + PBR^{-1}B^TP - Q. \end{aligned}$$

Similarly, in the discrete-time domain, a linear quadratic control problem can be written in the following form:

$$\min_{u_k} \sum_{k=0}^{\infty} (x_k^T Q x_k + u_k^T R u_k)$$

with respect to a linear discrete-time invariant system

$$x_{k+1} = Ax_k + Bu_k.$$

Accordingly, its solution is also computed via the discrete version of the Ricatti equation:

$$P = P^TAP - A^TPB(R + B^TPB)^{-1}B^TPA + Q. \tag{1.4}$$

It is shown in Chmielewski and Manousiouthakis [1996], Sznaier and Damborg [1987] that a (constrained) infinite horizon quadratic optimal control problem is equivalent to a (constrained) finite horizon one. More precisely, consider the constrained infinite-time quadratic optimal control problem:

$$\begin{aligned} &\min_{\mathbf{x},\mathbf{u}} \sum_{k=0}^{\infty} (x_k^T Q x_k + u_k^T R u_k) \\ \text{s.t.} \quad & x_{k+1} = Ax_k + Bu_k, \; x_0 = x^0 \in \mathbb{X}, \\ & \mathbf{x} = \{x_k\}_{k=0}^{\infty}, \; \mathbf{u} = \{u_k\}_{k=0}^{\infty}, \\ & x_k \in \mathbb{X} \subset \mathbb{R}^{d_x}, \; u_k \in \mathbb{U} \subset \mathbb{R}^{d_u}, \; \forall k \in \mathbb{N}, \end{aligned} \tag{1.5}$$

where $\mathbb{X}, \mathbb{U}$ are assumed to be compact and convex sets; and to contain the origin in their interior. Then, there exists an $N \in \mathbb{N}_{>0}$ such that the optimal solution to (1.5) solves the following problem:

$$\begin{aligned} &\min_{\mathbf{x}^N,\mathbf{u}^N} \sum_{k=0}^{N-1} (x_k^T Q x_k + u_k^T R u_k) + x_N^T P x_N \\ \text{s.t.} \quad & x_{k+1} = Ax_k + Bu_k, \; x_0 = x^0 \in \mathbb{X}, \\ & \mathbf{x}^N = \{x_k\}_{k=1}^{N}, \; \mathbf{u}^N = \{u_k\}_{k=0}^{N-1}, \\ & x_k \in \mathbb{X} \subset \mathbb{R}^{d_x}, \; u_k \in \mathbb{U} \subset \mathbb{R}^{d_u}, \; \forall\, 0 \leq k \leq N, \end{aligned} \tag{1.6}$$

where P is the solution to the Ricatti equation (1.4).

Further, if $\mathbb{X}, \mathbb{U}$ are polytopes, then it is shown in Bemporad et al. [2002], Feller et al. [2013], Gutman and Cwikel [1987], Olaru and Dumur [2004], Pistikopoulos et al. [2007], Seron et al. [2003], Tøndel et al. [2003] that optimal solution to problem (1.6) possesses the *piecewise affine structure*, leading to a piecewise affine closed loop system. It is worth emphasizing that this rising class of controllers has received significant attention from the control community due to its relevance for small dimensional systems. However, they still exist major problems in terms of implementation:

— the lack of a theoretical framework for implementation of PWA controllers under finite precision arithmetic to guarantee closed-loop stability,

— the limitation of their implementation into low-cost platforms.

Motivated by the above problems, this thesis focuses in the first part on the so-called *robustness and fragility analysis* of given PWA control laws and linear systems. Subsequently, to overcome the second problem, the so-called *inverse optimality problem* for the class of these controllers will also be discussed.

Inverse optimality was raised and solved by Kalman in Kalman [1964] for linear continuous systems. Originally, it aims to find an optimization formulation which recovers the given state feedback control law. More precisely, consider a stabilizing state feedback $u^* = Kx$ and a continuous time-invariant (LTI) system

$$\dot{x} = Ax + Bu, \tag{1.7}$$

where x, u stand for the state and control variables, respectively and $[A\ B]$ is assumed to be controllable. Inverse optimality aims to find a pair of weighting matrices Q, R characterizing the cost function:

$$J_\infty(x,u) = \frac{1}{2}\int_0^{+\infty} (x^TQx + u^TRu), \tag{1.8}$$

such that the given stabilizing control law $u^* = Kx$ is the solution to minimizing $J_\infty(x,u)$ problem, namely

$$u^* = Kx = \underset{u}{\text{argmin}}\, J_\infty(x,u). \tag{1.9}$$

Interested in this idea, many studies revisited inverse optimality for different systems and exploited it as a new approach for stabilizing and robust control designs e.g. in Freeman and Kokotovic [1996], Kong et al. [2012], Krstic and Li [1998], Krstic and Tsiotras [1999], Larin [2003], Løvaas et al. [2009], Nguyen et al. [2014b], Ostertag [2011], Rowe and Maciejowski [2000].

Over several decades of development, inverse optimality may still be valuable for control design in relationship with recently emerging classes of systems as *hybrid systems*, particularly *piecewise affine systems*. It is known that PWA systems have received significant studies due to the fact that this class of systems is relevant to emulate nonlinear systems as well as to facilitate stabilizing control design and stability analysis Angelis [2001]. However, our interest is limited to inverse optimality problem for the class of PWA control laws and linear time-invariant system. Obviously, inverse optimality problem for the class of PWA control laws becomes more difficult in comparison to the classical one as a piecewise affine function can be considered as a collection of several affine functions, usually known not to be component-wise convex or concave. Aside from the main goal to facilitate implementation of PWA controllers, solving this problem presents a new concept called *convex liftings* which will be later of use to construct control Lyapunov function, leading to new procedures to design stabilizing/robust control laws for linear constrained systems.

Accordingly, our main contributions can be outlined as follows:

— robustness and fragility analysis for piecewise affine control laws and a linear system;

— solving inverse parametric linear/quadratic programming problem via convex liftings;

— robust control design for constrained linear system affected by bounded additive disturbances and polytopic uncertainties.

Each of these parts will be detailed in the sequel.

1.1 Explicit robustness and fragility margins for PWA controllers

Starting from the analysis, this chapter considers the computation of robustness and fragility margins for piecewise affine controllers and a linear discrete-time system. The idea is based on the positive invariance principle. More precisely, given a PWA controller defined over a feasible set $\mathcal{X}$

$$f_{pwa}(x) = H_i x + G_i \quad \text{for} \quad x \in \mathcal{X}_i, \tag{1.10}$$

and a linear time-varying system

$$x_{k+1} = A(k)x_k + B(k)u_k$$

where

$$[A(k)\ B(k)] \in \Psi = \text{conv}\left\{[A_1\ B_1], \ldots, [A_L\ B_L]\right\}, \tag{1.11}$$

the robustness problem aims to find the set of linear systems $\Psi_{\text{rob}} \subseteq \Psi$ such that:

$$(A(k) + B(k)H_i)x + B(k)G_i \in \mathcal{X} \quad \forall x \in \mathcal{X}_i \text{ and } \forall [A(k)\ B(k)] \in \Psi_{\text{rob}}.$$

It will be shown that Ψ_{rob} is a polytope. Its computation can be carried out via two different approaches, i.e. vertex representation and halfspace representation.

Similarly, the fragility problem aims to find the set of admissible variations for given piecewise affine controller gains in order that the positive invariance of $\mathcal{X}$ is still guaranteed. Unlike the robustness problem, the fragility problem considers a linear time-invariant system:

$$x_{k+1} = Ax_k + Bu_k \tag{1.12}$$

and a given piecewise affine controller as in (1.10). The implemented controller can be written in the following form:

$$f_{pwa}(x) = (H_i + \delta_{H_i})x + (G_i + \delta_{G_i}) \quad \text{for} \quad x \in \mathcal{X}_i. \tag{1.13}$$

According to the definition, the fragility problem amounts to finding the set of $(\delta_{H_i}, \delta_{G_i})$ denoted by Δ_i^u such that

$$(A + B(H_i + \delta_{H_i}))x + B(G_i + \delta_{G_i}) \in \mathcal{X}, \;\; \forall x \in \mathcal{X}_i \;\; \text{and} \;\; \forall \; (\delta_{H_i}, \delta_{G_i}) \in \Delta_i^u.$$

It will be also shown that Δ_i^u represents a polyhedron. Its explicit computation can be of help in the implementation of explicit control laws under finite precision arithmetic.

Based on the same methodology, the problem can also be extended to finding the set of admissible additive disturbances for the linear time-invariant system:

$$x_{k+1} = Ax_k + Bu_k + w_k, \tag{1.14}$$

controlled by a piecewise affine controller (1.10) while preserving the positive invariance of $\mathcal{X}$.

Finally, the admissible errors for the description of the regions in the state space partition will also be tackled. This problem is independently considered without assuming that the control law gains admit numerical round-offs. Considering simultaneously errors on the description of regions and errors on their associated control law gains leads to the loss of linearity in the formulation, as well as the computation of these errors.

1.2 IPL/QP problems via convex liftings

It is well known that a parametric linear/quadratic programming (PL/QP) problem can be written in the following form:

$$\mathbf{u}^*(x) = \arg\min_{\mathbf{u}} \; J(\mathbf{u}, x) \quad \text{s.t.} \quad G\mathbf{u} \leq Ex + W,$$

where $\mathbf{u} \in \mathbb{R}^{d_\mathbf{u}}$ denotes the decision variable and $x \in \mathbb{R}^{d_x}$ stands for the parameter. Also, the cost function has the following form:

$$J(\mathbf{u}, x) = \mathbf{u}^T H \mathbf{u} + (x^T D + C)\mathbf{u},$$

where $H = H^T$ is positive semidefinite. If $H = 0$ and $D = 0$, then this optimization problem becomes parametric linear programming (PLP).

Optimal solution to the above problem is known to be a piecewise affine function defined over a polyhedral partition $\{\mathcal{X}_i\}_{i \in \mathcal{I}_N}$ i.e.

$$\mathbf{u}^*(x) = F_i x + G_i \quad \text{for} \quad x \in \mathcal{X}_i.$$

Conversely, inverse optimality problem for the class of PWA functions, alternatively called inverse parametric linear/quadratic programming (IPL/QP) problem, aims at constructing an optimization problem provided its optimal solution is equivalent to the given PWA function, namely finding matrices $H_x, H_\mathbf{u}, H_z, K$ and cost function $J(x, z, \mathbf{u})$ such that $\mathbf{u}^*(x)$ is a sub-component of the optimal solution to the following optimization problem:

$$\min_{[z^T \ \mathbf{u}^T]^T} J(x, z, \mathbf{u}) \quad \text{s.t.} \quad H_z z + H_\mathbf{u} \mathbf{u} + H_x x \le K,$$

where z represents an auxiliary variable. The solution to such an inverse optimality problem will be shown to rely on a so-called *convex lifting* concept. Accordingly, it will be shown that $z \in \mathbb{R}$ is sufficient for this inverse problem.

Also, for the continuous PWA functions, it will be shown that the optimal solution to this inverse optimality problem is unique. Otherwise, for the class of discontinuous PWA functions, the uniqueness of optimal solution to this recovered optimization problem will be lost. Accordingly, the given PWA function will be shown to be one among the optimal solutions to this problem.

1.3 Model predictive control redesign

This chapter starts from application of inverse optimality problem in linear model predictive control (MPC). An important result for linear MPC will be stated, namely *any continuous piecewise affine control law can be obtained by a linear MPC with the prediction horizon at most equal to 2 prediction steps.* Apart from the theoretical aspect, this result is also meaningful to facilitate implementation of PWA control laws, particularly induced from parametric quadratic programming problems and enables one to avoid storing the state space partition. This allows piecewise affine control laws to be implemented into low-cost (less demanding memory) platforms.

The second part of this chapter presents a so-called *robust control design based on convex liftings* which is defined over the maximal $\lambda-$contractive set as a domain of attraction for a given $0 \leq \lambda < 1$. Such a so-called *convex lifting* serves as a control Lyapunov function and is later of use to synthesize robust controller for linear system affected by bounded additive disturbances and polytopic uncertainties. Such a robust controller can be obtained in both implicit and explicit forms. It will be shown that the proposed method can guarantee the recursive feasibility and robust stability. For the implicit control design, the formulation is simple and easy to implement i.e. it requires solving a linear programming problem at each sampling time. The drawback of this formulation is to compute at first glance the maximal $\lambda-$contractive set, since it requires a repetitive procedure and takes some time to achieve. Fortunately, this computation is carried out offline. However, this maximal $\lambda-$contractive set is known to be smaller than the maximal controllable set.

As a generalization of the above design procedure, the third part of this chapter puts forward a new design procedure over the $N-$steps controllable set. This procedure relies on a so-called *cascade of convex liftings*. This method presents a new auxiliary variable which can serve as a Lyapunov function, i.e. is non-negative, strictly decreases until step $N + 1$ and stays later at 0.

Chapter 2

Background

Contents

This chapter aims to recall fundamental definitions and results in constrained control topics, which will be of use later for presenting the contribution of this thesis. More details about convex analysis and convex optimization can be found in prominent references Boyd and Vandenberghe [2004], Rockafellar [1970].

The first section will be dedicated to set-theoretic notions, mostly related to convexity in finite dimensional spaces. These elements will be essential for the description of sets in relationship with state space dynamics. This part is complemented in Section 2.2 with notions of polyhedra and polytopes which are particular cases of most interest in linear dynamics and linear constraints.

In the second part of the chapter, we move towards classical tools and notions in control theory with Lyapunov stability, specific notions for linear systems and basics for Model Predictive Control.

2.1 Convex sets and convex functions

2.1.1 Compact sets

A set $\mathbb{X}$ is called a *metric space* if it is assigned a *distance* $\rho : \mathbb{X} \times \mathbb{X} \to \mathbb{R}$ satisfying:

1. $\rho(x, y) \geq 0$ for any $x, y \in \mathbb{X}$; $\rho(x, y) = 0$ if and only if $x = y$,
2. $\rho(x, y) = \rho(y, x)$,
3. $\rho(x, y) + \rho(y, z) \geq \rho(x, z)$ for any $x, y, z \in \mathbb{X}$.

For a given $x \in \mathbb{X}$ and $\delta > 0$, define the *open ball* $\overset{o}{B}(x, \delta)$ and the *closed ball* $B(x, \delta)$ as follows:

$$\begin{aligned}\overset{o}{B}(x, \delta) &= \{y \in \mathbb{X} \mid \rho(x, y) < \delta\}\\ B(x, \delta) &= \{y \in \mathbb{X} \mid \rho(x, y) \leq \delta\}.\end{aligned}$$

A set $\mathcal{X} \subset \mathbb{X}$ is called *open* if for any $x \in \mathcal{X}$, there exists $\delta > 0$ such that the open ball $\overset{o}{B}(x, \delta)$ is a subset of $\mathcal{X}$; i.e. $\overset{o}{B}(x, \delta) \subset \mathcal{X}$.

A sequence $\{x_n\}_0^\infty$ of points of a set $\mathcal{X}$, is called *convergent* to $x \in \mathcal{X}$ if

$$\lim_{n \to \infty} \rho(x_n, x) = 0.$$

A set $\mathcal{X}$ is called *closed* if the limit of every convergent sequence of points in $\mathcal{X}$ belongs to $\mathcal{X}$.

A set $\mathcal{X}$ is called *compact* if for any infinite sequence of points in $\mathcal{X}$, there exists a subsequence that converges to a point $x \in \mathcal{X}$.

In the Euclidean space, the *distance* implies the Euclidean distance between two points, defined as follows:

$$\rho(x, y) = \sqrt{\langle x - y, x - y \rangle}.$$

Also, a *compact set* in the Euclidean space can be understood as a *bounded* and *closed* set.

Let's consider simple examples to illustrate these notions. Interval $(1,\ 2)$ is an *open set* according to its definition. Otherwise, $[1,\ 2]$ is a *compact set* due to its boundedness and closedness. However, interval $(1,\ 2]$ is not a compact set. In fact, consider a sequence $\left\{x_n = 1 + \frac{1}{n}\right\}_1^\infty$. This sequence contains points in $(1,\ 2]$ but converges to $1 \notin (1,\ 2]$. Therefore, $(1,\ 2]$ is not *closed*, thus not *compact*. Similarly, interval $(-\infty, 1]$ is not a compact set, however $(-\infty, 1] \cup \{-\infty\}$ is a compact set.

2.1.2 Support and Separation

Given a vector $a \in \mathbb{R}^d$ and $\alpha \in \mathbb{R}$, a hyperplane $\mathcal{H}$ can be described as follows:

$$\mathcal{H} := \left\{x \in \mathbb{R}^d \mid \langle a, x\rangle = \alpha\right\}.$$

Define also the following sets

$$\begin{aligned} \mathcal{H}^- &:= \left\{x \in \mathbb{R}^d \mid \langle a, x\rangle < \alpha\right\} \\ \mathcal{H}^+ &:= \left\{x \in \mathbb{R}^d \mid \langle a, x\rangle > \alpha\right\}. \end{aligned}$$

Such $\mathcal{H}^-, \mathcal{H}^+$ are called *open halfspaces*. Their closures $\mathrm{cl}(\mathcal{H}^-), \mathrm{cl}(\mathcal{H}^+)$, mathematically described by:

$$\begin{aligned} \mathrm{cl}(\mathcal{H}^-) &:= \left\{x \in \mathbb{R}^d \mid \langle a, x\rangle \le \alpha\right\} \\ \mathrm{cl}(\mathcal{H}^+) &:= \left\{x \in \mathbb{R}^d \mid \langle a, x\rangle \ge \alpha\right\}, \end{aligned}$$

are called *closed halfspaces*.

Moreover, given a set $S \subset \mathbb{R}^d$, such a hyperplane $\mathcal{H}$ *cuts* S if there exist two points $x_1, x_2 \in S$ such that $\langle a, x_1\rangle > \alpha$ and $\langle a, x_2\rangle < \alpha$. Clearly, a hyperplane cuts the whole space $\mathbb{R}^d$ into two halfspaces.

Consider now a closed set $S \subset \mathbb{R}^d$, such a hyperplane $\mathcal{H}$ is called a *supporting hyperplane* of S at some point $x \in S$ if $x \in \mathcal{H} \cap S$ and either $S \subset \mathrm{cl}(\mathcal{H}^-)$ or $S \subset \mathrm{cl}(\mathcal{H}^+)$. Suppose the former case happens, then $\mathrm{cl}(\mathcal{H}^-)$ is also called a *supporting halfspace* of S at x.

Now, given two sets $S_1, S_2 \subset \mathbb{R}^d$, we say that such a hyperplane $\mathcal{H}$ *separates* S_1, S_2 if $\mathcal{H}$ cuts the whole space $\mathbb{R}^d$ into two closed halfspaces, one contains S_1 and the other contains S_2. Also, S_1, S_2 are called *strictly separated* by $\mathcal{H}$ if they are separated and $S_1 \cap \mathcal{H} = S_2 \cap \mathcal{H} = \emptyset$.

Note that the closed halfspace $\mathrm{cl}(\mathcal{H}^+)$ has an equivalent description:

$$\mathrm{cl}(\mathcal{H}^+) = \left\{x \in \mathbb{R}^d \mid \langle -a, x\rangle \le -\alpha\right\}.$$

Therefore, the description of a closed (open) halfspace $\mathcal{H}_{a,\alpha}$ can be unified as follows:

$$\mathcal{H}_{a,\alpha} = \left\{ x \in \mathbb{R}^d \mid \langle a, x \rangle \leq \alpha \right\},$$

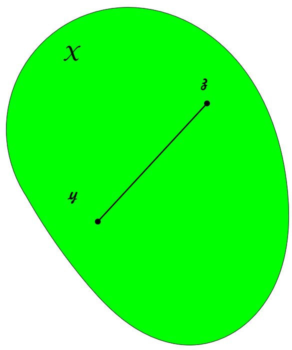
where a is called *outer normal vector*. Henceforth, this description will be of use when talking about a general closed halfspace.

2.1.3 Convex sets

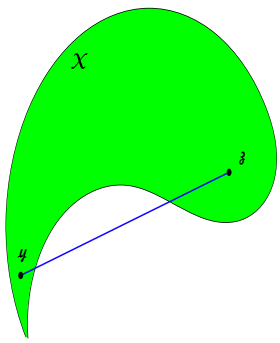
A set $\mathcal{X} \subseteq \mathbb{R}^d$ is called *convex* if for any pair of two points in $\mathcal{X}$, the line connecting these two points is a subset of $\mathcal{X}$. This condition can be mathematically expressed as follows:

$$\forall y, z \in \mathcal{X},\ \ \lambda y + (1 - \lambda) z \in \mathcal{X} \ \text{ for all } \lambda \in [0,\ 1]\,. \tag{2.1}$$

An example of convex and nonconvex sets can be found in Figure 2.1.



(a) Convex set



(b) Nonconvex set

Figure 2.1: An illustrative example of convex and nonconvex sets.

Let $\mathcal{X} \subset \mathbb{R}^d$ be a compact, convex set, a point $x \in \mathcal{X}$ is called *extreme point* if it does not belong to the relative interior of any segment connecting two points in $\mathcal{X}$. In other words, if there exist $y, z \in \mathcal{X}$ such that $x = \lambda y + (1 - \lambda) z$ for some $0 < \lambda < 1$, then $x = y = z$. To illustrate this point, consider a ball $\mathcal{B} \subset \mathbb{R}^d$, centered at x_0 with the radius $r \in \mathbb{R}_{>0}$. This ball can be described as follows:

$$\mathcal{B} := \left\{ x \in \mathbb{R}^d \mid \langle x - x_0, x - x_0 \rangle \leq r^2 \right\},$$

and known to be a convex set. Recall that $\mathrm{bd}(\mathcal{B})$ denotes the set of points on the boundary of $\mathcal{B}$, i.e.

$$\mathrm{bd}(\mathcal{B}) := \left\{ x \in \mathbb{R}^d \mid \langle x - x_0, x - x_0 \rangle = r^2 \right\}.$$

It can be observed that any point belongs to $\mathrm{bd}(\mathcal{B})$ is an extreme point of this ball.

The following properties are fundamental for convex sets.

— If $\mathcal{X}$ is a convex set, then $\mathrm{cl}(\mathcal{X})$ and $\mathrm{int}(\mathcal{X})$ are convex sets.

— The intersection of convex sets is a convex set. However, the union of convex sets may not be a convex set.

— If $A, B \subset \mathbb{R}^d$ are convex, then $A \oplus B$, λA for any $\lambda \in \mathbb{R}$ are convex.

— Any affine transformation of a convex set A is convex.

Carathéodory's theorem has important applications in many fields. It is recalled below from Section 2.3 in Grünbaum [1967] or Theorem 1.1.4 in Schneider [2013].

Theorem 2.1.1 *If $\mathcal{X} \subset \mathbb{R}^d$ and $x \in \mathrm{conv}(\mathcal{X})$, then x is expressible in the form*

$$x = \sum_{i=0}^{d} \alpha_i x_i, \quad for \;\; \alpha_i \geq 0, x_i \in \mathcal{X}, \forall i \in \mathcal{I}_d \cup \{0\}, \sum_{i=0}^{d} \alpha_i = 1.$$

This theorem means that any point in $\mathrm{conv}(\mathcal{X})$ can be expressed via a convex combination of $d+1$ or fewer points in $\mathcal{X}$. The following result is of fundamental importance. This is stated in Section 2.2 of Grünbaum [1967].

Theorem 2.1.2 *Each closed, convex subset of $\mathbb{R}^d$ is the intersection of all closed halfspaces of $\mathbb{R}^d$ which contain this set. Each open convex subset of $\mathbb{R}^d$ is the intersection of all open halfspaces of $\mathbb{R}^d$ which contain it.*

Based on this result, the following consequence introduces a closer viewpoint in the representation of a convex set. It is presented via Corollary 1.3.5 in Schneider [2013].

Corollary 2.1.3 *Every nonempty, closed, convex set is the intersection of its supporting halfspaces.*

Note that for a closed, convex set, there always exists at least one supporting hyperplane at each point on its boundary.

Recall that $\mathrm{ext}(\mathcal{X})$ is used to denote the set of extreme points of a given convex set $\mathcal{X}$. The following result extracted from Section 2.4 in Grünbaum [1967], provides another important insight for a convex set.

Theorem 2.1.4 *Let K be a compact, convex subset of $\mathbb{R}^d$, then $K = \mathrm{conv}\,(\mathrm{ext}(K))$. Moreover, if $K = \mathrm{conv}(A)$ then $\mathrm{ext}(K) \subseteq A$.*

Otherwise, a subset $\mathcal{X} \subset \mathbb{R}^d$ is called a *cone* if it is closed under positive scalar multiplication; i.e. if $x \in \mathcal{X}$, then $\lambda x \in \mathcal{X}$ for any $\lambda > 0$. From this definition, such a set is the union of rays emanating from the origin. Also, the origin may not be included. Further, a *convex cone* is a cone and convex set. The following result is recalled from Theorem 2.5 in Rockafellar [1970]. It is a direct consequence of the second property for convex sets, presented above.

Theorem 2.1.5 *The intersection of an arbitrary collection of convex cones is a convex cone.*

Based on its definition and Corollary 2.1.3, a convex cone can be described as follows:

Corollary 2.1.6 *Let $a_i \in \mathbb{R}^d$, $i \in \mathcal{I}$ for an arbitrary index set I. Then a closed convex cone $K \subset \mathbb{R}^d$ can be represented by:*

$$K := \left\{ x \in \mathbb{R}^d \mid \langle a_i, x \rangle \leq 0, \;\; \textit{for all } i \in I \right\}.$$

The convex cone in this corollary is a *closed* set. If the inequalities are replaced with strict inequalites then this convex cone represents an *open* set. Also, the right-hand side of each closed halfspace is 0, due to the fact that any supporting hyperplane of a closed convex cone includes the origin.

For unbounded convex sets, it is necessary to recall the notion of a *ray*. Consider a convex set $K \subseteq \mathbb{R}^d$, $r \in \mathbb{R}^d$ is called a ray of K if there exists $x \in K$ such that $x + \lambda r \in K$ for every $\lambda \geq 0$. Also, r is called an *extreme ray* of K if r is not expressible via a convex combination of two distinct rays of K i.e. r_1, r_2 are two rays of K s.t.

$$r = \alpha r_1 + (1 - \alpha) r_2, \text{ for some } \alpha \in (0, \; 1) \;\; \Rightarrow r = r_1 = r_2.$$

The following result provides an important insight for the representation of polyhedra. This result is extracted from Section 2.5 in Grünbaum [1967].

Theorem 2.1.7 *A convex set $K \subset \mathbb{R}^d$ is unbounded if and only if K contains a ray.*

This inclusion is a premise of the so-called *Minkowski-Weyl theorem* for polyhedra. This theorem will be recalled in Section 2.2.3.

Let $K \subset \mathbb{R}^d$ be a closed convex set. $F \subset K$ is called a *face of K* if either $F = \emptyset$, or $F = K$, or there exists a supporting hyperplane H of K such that $F = H \cap K$. Also $\emptyset, K$ are called *improper* faces of K. The following property of faces is recalled from Section 2.4 in Grünbaum [1967].

Theorem 2.1.8 *If F is a face of a closed convex set K, then* $\mathrm{ext}(F) = F \cap \mathrm{ext}(K)$.

This theorem means that the extreme points of a face of a closed convex set are also among extreme points of this closed convex set itself.

2.1.4 Convex functions

Given a function denoted by $f : \mathcal{X} \subseteq \mathbb{R}^d \to \mathbb{R}$, this function is called *convex* if for any $x, y \in \mathcal{X}$,

$$f(\lambda x + (1-\lambda)y) \leq \lambda f(x) + (1-\lambda) f(y),\ \forall \lambda \in [0,\, 1]\,.$$

Also, such a function f is called *strictly convex* if for any $x, y \in \mathcal{X}, x \neq y$,

$$f(\lambda x + (1-\lambda)y) < \lambda f(x) + (1-\lambda) f(y),\ \forall \lambda \in (0,\, 1)\,.$$

Convex functions have important properties. Some of them are recalled below.

— Let $f : \mathcal{X} \subseteq \mathbb{R}^n \to \mathbb{R}$, denote a real-valued function defined over a compact set $\mathcal{X}$, $\mathrm{epi}(f)$ denotes the *epigraph* of the function f, and is defined as follows:

$$\mathrm{epi}(f) = \left\{ \left[x^T\, \mu\right]^T \in \mathbb{R}^{n+1} \mid x \in \mathcal{X},\, \mu \in \mathbb{R},\, \mu \geq f(x) \right\}. \tag{2.2}$$

Then the function $f(x)$ is convex if and only if $\mathrm{epi}(f) \subset \mathbb{R}^{n+1}$ is a convex set.

— Given two convex functions $f_1 : \mathcal{X} \subseteq \mathbb{R}^n \to \mathbb{R}$ and $f_2 : \mathcal{X} \subseteq \mathbb{R}^n \to \mathbb{R}$, it can be observed that $f_1 + f_2$ also represents a convex function defined over $\mathcal{X}$.

— If $f_1, \ldots, f_n$ represent real-valued, convex functions, defined over the same domain, then

$$f = \max\{f_1, \ldots, f_n\}$$

is also a convex function. Also, the epigraph of f is the intersection of the epigraphs of $f_1, \ldots, f_n$.

The famous Jensen's inequality is recalled below.

Theorem 2.1.9 *If $f : \mathbb{R}^d \to \mathbb{R}$ is a convex function, then*

$$f\left(\sum_{i=1}^{n} \lambda_i x_i\right) \leq \sum_{i=1}^{n} \lambda_i f(x_i),$$

for $\lambda_i \geq 0, x_i \in \mathbb{R}^d$ for all $i \in \mathcal{I}_n$ and $\sum_{i=1}^{n} \lambda_i = 1$.

2.2 Polyhedra and Polytopes

2.2.1 Polytopes

A compact, convex set $S \subset \mathbb{R}^d$ is called *polytope* if its set of extreme points is finite. Accordingly, the set $\mathrm{ext}(S)$ will alternatively denote the *vertices* of S. Henceforth, $\mathcal{V}(S)$ will be of use to denote the set of vertices of S. According to this definition, a polytope naturally inherits the properties of a convex set.

Theorem 2.2.1 *If $P \subset \mathbb{R}^d$ is a polytope, any $x \in P$ can be described via a convex combination of $\mathcal{V}(P)$; i.e.*

$$x = \sum_{v \in \mathcal{V}(P)} \alpha(v) v,$$

where for each $v \in \mathcal{V}(P)$, $\alpha(v) \geq 0$, and $\sum_{v \in \mathcal{V}(P)} \alpha(v) = 1$.

The above result is a direct consequence of Theorem 2.1.4. Note also that $\mathrm{Card}\, \mathcal{V}(P) < +\infty$ due to its definition. This expression is alternatively said to be the *vertex representation* or sometimes $\mathcal{V}-$representation.

The computation of the set of vertices for a given polytope is called *vertex enumeration*. This computation can be carried out by many different algorithms. Most of numerical examples involving vertex representation in this thesis are obtained with the algorithm in Avis and Fukuda [1992].

As resulted from Corollary 2.1.3, the so-called *halfspace representation* or briefly $\mathcal{H}-$representation of a polytope is recalled via the following theorem. This is quoted from Theorem 2.4.3 in Schneider [2013].

Theorem 2.2.2 *Every polytope is the intersection of finitely many closed halfspaces.*

This result implies that describing a polytope requires a finite number of closed halfspaces. Particularly, these halfspaces are supporting halfspaces at the facets [1] of this polytope. In other words, a polytope $P \subset \mathbb{R}^d$ can be described as follows

$$P = \left\{ x \in \mathbb{R}^d \mid Ax \leq b \right\},$$

for some $A \in \mathbb{R}^{m \times d}$, $b \in \mathbb{R}^m$. Further, each row of matrix A represents an outer normal vector of polytope P. Also, the transformation from the vertex representation ($\mathcal{V}-$representation) to the halfspace representation ($\mathcal{H}-$representation) is called *facet enumeration*. The vertex and facet enumeration problems are dual and both studied in Avis and Fukuda [1992]. These algorithms are available in CDD by Komei Fukuda in Fukuda [1997] and are also integrated in MPT (Multi-parametric Toolbox) Herceg et al. [2013]. Note that the vertex enumeration problem via these proposed algorithms can be carried out in time $O(ndv)$, where n denotes the number of halfspaces, d denotes the dimension of this polytope, v is its number of vertices. Dually, the facet enumeration problem proposed therein can be carried out in time $O(ndv)$ where v denotes the number of facets, n denotes the number of given points in the Euclidean space of dimension d.

1. The definition of a *facet* of a polytope/polyhedron is presented in Section 2.2.4.

An illustration is presented in Figure 2.2. This polytope is described via the vertex and halfspace representations as follows:

$$P = \operatorname{conv}\left\{ \begin{bmatrix} -1 \\ -1 \end{bmatrix}, \begin{bmatrix} -1 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ -1 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \end{bmatrix} \right\},$$

$$P = \left\{ x \in \mathbb{R}^2 \mid \begin{bmatrix} 1 & 0 \\ -1 & 0 \\ 0 & 1 \\ 0 & -1 \end{bmatrix} x \leq \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \right\}.$$

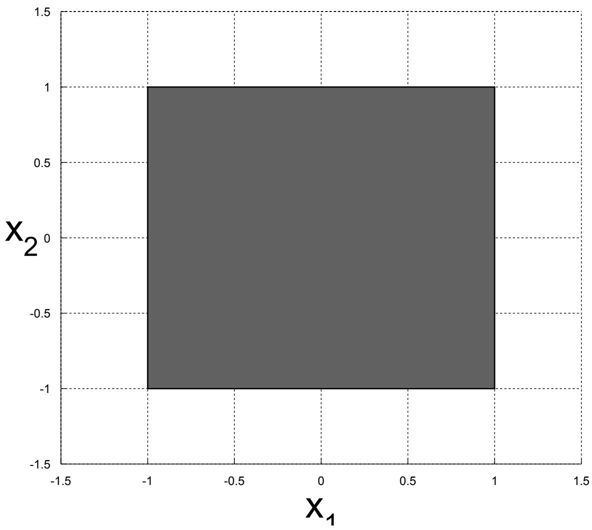


Figure 2.2: A polytope in $\mathbb{R}^2$.

2.2.2 Polyhedral cones

A convex cone has been defined in Section 2.1.3. This section focuses on a particular class of cones called *polyhedral cones*. Hereafter, closed, convex, polyhedral cones are exclusively considered. For simplicity, in the remainder of this manuscript, a polyhedral cone implies a closed, convex, polyhedral cone. Accordingly, a polyhedral cone $C \subset \mathbb{R}^d$ contains the origin, and can be described by the intersection of a finite number of closed halfspaces as follows:

$$C := \left\{ x \in \mathbb{R}^d \mid Ax \leq 0 \right\},$$

where $A \in \mathbb{R}^{m \times d}$ and $0 \in \mathbb{R}^m$.

Dually, a polyhedral cone is also expressible via its finite set of extreme rays (vectors). More clearly, there exists a finite set $Y = \{y_1, \ldots, y_n\} \subset \mathbb{R}^d$ of rays such that

$$C = \text{cone}(Y), \ \ \text{cone}(Y) = \left\{ \sum_{i=1}^{n} \lambda_i y_i \mid y_i \in Y, \ \lambda_i \geq 0, \ \forall i = 1, \ldots, n \right\},$$

where $\text{cone}(Y)$ is said to be the *conical hull* of Y. Also, $\sum_{i=1}^{n} \lambda_i y_i$ for $\lambda_i \geq 0, \forall i = 1, \ldots, n$ is called a *conical combination*.

A particularly important polyhedral cone is the positive quadrant of $\mathbb{R}^n$. This quadrant denoted by C, can be dually represented by a collection of n rays. In $\mathbb{R}^2$, this is represented by

$$C = \text{cone}\left\{ \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \end{bmatrix} \right\}, \ C = \left\{ x \in \mathbb{R}^2 \mid \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} x \leq \begin{bmatrix} 0 \\ 0 \end{bmatrix} \right\}.$$

2.2.3 Polyhedra

A polyhedron usually implies an unbounded, polyhedral and convex set. Similar to a polytope, a polyhedron is also defined as the intersection of finitely many closed halfspaces. Dually, according to Theorem 2.1.7, a polyhedron as an unbounded convex set, should include a ray. A famous theorem related to the structure of polyhedra, known as the Minkowski-Weyl theorem, is recalled as follows.

Theorem 2.2.3 *A polyhedron $P \subseteq \mathbb{R}^d$ can be represented as the Minkowski sum of the convex hull of a finite set of points and conical hull of finite vectors; i.e.*

$$P = \text{conv}(\mathcal{V}(P)) \oplus \text{cone}(\mathcal{R}(P)),$$

where $\mathcal{V}(P), \mathcal{R}(P)$ stand for the sets of vertices and extreme rays of polyhedron P, respectively.

For illustration, consider a polyhedron $P \subset \mathbb{R}^2$, whose halfspace representation is shown below:

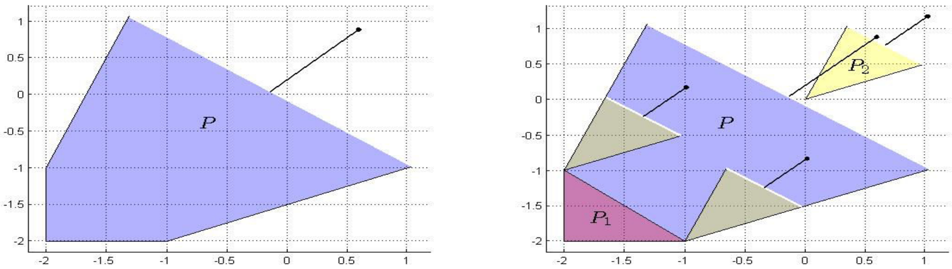
$$P = \left\{ x \in \mathbb{R}^2 \mid \begin{bmatrix} -1 & 0 \\ -3 & 1 \\ 1 & -2 \\ 0 & -1 \end{bmatrix} x \leq \begin{bmatrix} 2 \\ 5 \\ 3 \\ 2 \end{bmatrix} \right\}.$$

This polyhedron is presented in Figure 2.3a. Also, it can be decomposed into the Minkowski sum as follows:

$$P = P_1 \oplus P_2,$$

$$P_1 = \text{conv}\left\{ \begin{bmatrix} -1 \\ -2 \end{bmatrix}, \begin{bmatrix} -2 \\ -2 \end{bmatrix}, \begin{bmatrix} -2 \\ -1 \end{bmatrix} \right\}, \ P_2 = \text{cone}\left\{ \begin{bmatrix} 2 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 3 \end{bmatrix} \right\}.$$

These decomposed cone and polytope, presented in Figure 2.3b, are also called *summands* of P.



(a) Polyhedron P.

(b) Decomposition of P into the Minkowski sum of P_1, P_2.

Figure 2.3: An illustrative example for the Minkowski-Weyl theorem.

2.2.4 Additional notes

For simplicity, given a full-dimensional polyhedron $P \subseteq \mathbb{R}^d$, $k-$face implies a face of dimension k for $k = 0, \ldots, d$. More specially, $0-$face is also called *vertex*, $1-$face is denoted as *edge*. Also, $(d-1)-$face is alternatively called *facet*.

If P is a polytope, then its faces are also lower-dimensional polytopes.

Recall that throughout this manuscript, $\mathcal{V}(P), \mathcal{R}(P)$ are used to denote the set of vertices and extreme rays of P, respectively. Also, $\mathcal{F}(P)$ denotes the set of facets of the given polyhedron P.

The following properties of polyhedron/polytope are also important:

— Any affine transformation of a polyhedron is also a polyhedron.

— Given two polyhedra $P_1, P_2 \subseteq \mathbb{R}^d$, then $P_1 \oplus P_2$, λP_1, $\lambda > 0$ are also polyhedra.

— The orthogonal projection of a polyhedron is also a polyhedron.

Recall that the orthogonal projection is denoted by $\mathrm{Proj}()$ throughout this manuscript. The definition of the orthogonal projection is recalled as follows.

Given a polyhedron $P \subseteq \mathbb{R}^d$, the orthogonal projection onto the space of $(d-1)$ first coordinates i.e. $\mathbb{R}^{d-1}$, is defined by

$$\mathrm{Proj}_{\mathbb{R}^{d-1}} P = \left\{ x \in \mathbb{R}^{d-1} \mid \exists y \in \mathbb{R}, \text{ s.t. } \begin{bmatrix} x \\ 0 \end{bmatrix} + \begin{bmatrix} 0_{d-1} \\ y \end{bmatrix} \in P \right\}.$$

The orthogonal projection is known to be computationally demanding. A famous method for this computation is the Fourier-Motzkin elimination George and Eaves [1973], Ziegler [1995]. More specially, this projection for a polytope can also be alternatively carried out by computing its vertices and projecting these vertices onto a given subspace. The latter operation is easily computed by removing the last coordinate of each vertex. Finally, to obtain the resulting polytope, one needs to compute the convex hull of the projected vertices.

A polyhedron in the halfspace representation can be considered as linear constraints which are closely related to linear programming. Another important result named *Farkas' lemma* is recalled as follows. It is quoted from Theorem 7.1 in Schrijver [1998].

Theorem 2.2.4 *Let $a_1, \ldots, a_m, b$ be vectors in $n-$dimensional space. Then*

— *Either, b is a nonnegative linear combination of linearly independent vectors from $a_1, \ldots, a_m$;*

— *or, there exists a hyperplane $\{x \mid c^T x = 0\}$ containing $t-1$ linearly independent vectors from $a_1, \ldots, a_m$ such that $c^T b < 0$ and $c^T a_1, \ldots, c^T a_m \geq 0$ where $t = \operatorname{rank}\,[a_1 \ldots a_m\ b]$.*

2.3 Stability and Robust stability

2.3.1 Stability

This section aims to recall some definitions of stability useful for later developments. Consider the following discrete-time, autonomous system:

$$x_{k+1} = f(x_k), \tag{2.3}$$

where $x_k \in \mathbb{R}^d$ stands for the state at time k and $f : \mathbb{R}^d \to \mathbb{R}^d$, $f(0) = 0$. The stability and asymptotic stability definitions for this system are recalled in the sequel.

Definition 2.3.1 Consider the autonomous system (2.3). The origin is said to be *stable* in the sense of Lyapunov if for any $\epsilon > 0$, there exists a $\delta > 0$ such that

$$|x_0| \leq \delta, \;\Rightarrow\; |x_k| \leq \epsilon, \;\forall k \geq 0.$$

Definition 2.3.2 The origin is called *asymptotically stable* if it is stable and

$$\lim_{k\to\infty} |x_k| = 0.$$

Recall that a function $V : \mathbb{R}^d \to \mathbb{R}$ is called *positive definite* if $V(0) = 0$ and $V(x) > 0$ for all $x \neq 0$. Accordingly, the celebrated result of Lyapunov's stability theory is expressed via the following statement.

Theorem 2.3.3 *(Lyapunov) Consider system* (2.3)*. Suppose there exists a positive definite function* $V : \mathbb{R}^d \to \mathbb{R}$ *satisfying:*

$$V(f(x)) - V(x) \leq 0, \ \forall x,$$

then system (2.3) *is stable. Also, if this function satisfies*

$$V(f(x)) - V(x) < 0, \ \forall x \neq 0,$$

then system (2.3) *is asymptotically stable.*

2.3.2 Positively invariant sets

Set-theoretic approach in control theory has been extensively studied due to its relevance in control theory, particularly robust control designs for constrained systems Bitsoris [1988a,b], Bitsoris and Vassilaki [1995], Blanchini [1999], Blanchini and Miani [2008], Gilbert and Tan [1991], Kerrigan [2001], Kolmanovsky and Gilbert [1998], Rakovic et al. [2004, 2005, 2006], and recently in fault-tolerant control Olaru et al. [2010], Stoican and Olaru [2013]. This has important connections with the control design, the notation of *positively invariant* set being used for stability study. Moreover, in the presence of bounded persistent disturbances, *asymptotic stability* of the origin cannot be sought and the robust positively invariant set and the ultimate bounds have to be employed. These notions are recalled below for completeness.

Consider a discrete-time invariant system

$$x_{k+1} = f(x_k, u_k), \tag{2.4}$$

where x_k, u_k stand for the state and control variables at sampling time k. These are subject to constraints

$$x_k \in \mathbb{X} \subseteq \mathbb{R}^{d_x}, \ u_k \in \mathbb{U} \subseteq \mathbb{R}^{d_u}, \tag{2.5}$$

such that $\mathbb{X}, \mathbb{U}$ contain the origin in their interior. Let $u = \kappa(x) \in \mathbb{U}$ be a given controller. The definition of a *positively invariant set* is recalled as follows:

Definition 2.3.4 Given system (2.4), a set $\Omega \subseteq \mathbb{X}$ is called *positively invariant* with respect to a given control law $u = \kappa(x) \in \mathbb{U}$ if

$$\forall x \in \Omega, \ f(x, \kappa(x)) \in \Omega.$$

Definition 2.3.5 Given system (2.4), a set $\Omega \subseteq \mathbb{X}$ is called *control invariant* if

$$\forall x \in \Omega, \ \exists u(x) \in \mathbb{U} \ \text{ s.t. } \ f(x, u(x)) \in \Omega.$$

Consider next a discrete-time, invariant system subject to bounded persistent disturbances

$$x_{k+1} = f(x_k, u_k, w_k). \tag{2.6}$$

The state and control variables are subject to constraints (2.5), while the disturbances satisfy the following constraint:

$$w_k \in \mathbb{W}. \tag{2.7}$$

The definition of a *robust positively invariant set* $\Omega \subseteq \mathbb{X}$, is recalled below with respect to a given control law $u = \kappa(x) \in \mathbb{U}$.

Definition 2.3.6 Let system (2.6) be given. A set $\Omega \subseteq \mathbb{X}$ is called *robust positively invariant* with respect to the dynamics (2.6) and a control law $u = \kappa(x) \in \mathbb{U}$ if

$$\forall x \in \Omega,\ f(x, \kappa(x), w) \in \Omega \ \text{ for all } \ w \in \mathbb{W}.$$

Definition 2.3.7 Let system (2.6) be given. A set $\Omega \subseteq \mathbb{X}$ is called *robust control invariant* with respect to the dynamics (2.6) if

$$\forall x \in \Omega,\ \exists u(x) \in \mathbb{U} \ \text{ s.t. } \ f(x, u(x), w) \in \Omega \ \text{ for all } \ w \in \mathbb{W}.$$

Domain of attraction is defined as a subset of all points which can be steered to a target set (usually known to be the origin or a positively invariant set). Namely, for any point belonging to a domain of attraction, there always exists control law satisfying constraints (2.5) such that the state is driven to a target set despite any disturbance $w_k \in \mathbb{W}$.

As mentioned before, in the presence of bounded persistent disturbances, guaranteeing asymptotic stability of the origin is difficult. However, the positive invariance principle is meaningful to guarantee *robust stability* by steering the state into a robust positively invariant set and keeping it inside this set with an appropriate state feedback satisfying input constraints.

Let system (2.6) be controlled by a given state feedback $u_k = \kappa(x_k) \in \mathbb{U}$. $\mathcal{X} \subseteq \mathbb{X}$ denotes a domain of attraction for system (2.6) and $\Omega \subset \mathcal{X}$ denotes a robust positively invariant set as a target set despite any disturbance $w_k \in \mathbb{W}$. Accordingly, the closed loop is expressed in the following form:

$$x_{k+1} = f(x_k, \kappa(x_k), w_k). \tag{2.8}$$

The definition of *robust stability* for system (2.8) is presented in the sequel.

Definition 2.3.8 System (2.8) is called *robust stable* with respect to robust positively invariant set Ω if for any $\epsilon > 0$ and $x_0 \in \mathcal{X}$, there exists a finite $T \in \mathbb{N}_{>0}$ such that

$$\rho_\Omega(x_k) \leq \epsilon,\ \forall k \geq T,\ \forall w_i \in \mathbb{W} \ \text{ where } \ 0 \leq i \leq k-1.$$

Based on the principle of asymptotic stability and robust positively invariant set, the following definition is of use in the sequel based on a suitable robust control Lyapunov function.

Definition 2.3.9 Let system (2.6) be controlled by a state feedback $u = \kappa(x) \in \mathbb{U}$. $\mathcal{X} \subseteq \mathbb{X}$ denotes a domain of attraction and $\Omega \subset \mathcal{X}$ denotes a robust positively invariant set. System (2.6) is called *robust stable* if there exists a Lyapunov function $V : \mathcal{X} \to \mathbb{R}$ such that

— $V(x) > 0$ for every $x \in \mathcal{X}\backslash\Omega$, $V(x) = 0$ for all $x \in \Omega$

— $V(f(x, \kappa(x), w)) - V(x) < 0, \quad \forall x \in \mathcal{X}\backslash\Omega, \quad \forall w \in \mathbb{W}$.

As the other methods based on a robust control Lyapunov function, finding such a function $V(x)$ is of importance in design procedure, as well as in guaranteeing the feasibility of proposed method. This aspect will be detailed later.

2.4 Linear systems

Throughout this manuscript, a discrete-time, linear system will be of interest. It is expressed as follows:

$$x_{k+1} = A(k)x_k + B(k)u_k + w_k, \tag{2.9}$$

where $[A(k)\ B(k)]$ belongs to a *polytopic uncertainty set* Ψ

$$\Psi = \text{conv}\left\{[A_1\, B_1], \ldots, [A_L\, B_L]\right\}, \tag{2.10}$$

meaning that any $[A\ \ B] \in \Psi$ can be expressed by

$$[A\ \ B] = \sum_{i=1}^{L} \alpha_i\, [A_i\, B_i]\,,$$

for $\alpha_i \geq 0$, $\forall i \in \mathcal{I}_L$ and $\sum_{i=1}^{L} \alpha_i = 1$.

Also, the state, control variables and disturbances are subject to constraints:

$$x_k \in \mathbb{X} \subset \mathbb{R}^{d_x},\ \ u_k \in \mathbb{U} \subset \mathbb{R}^{d_u},\ \ w_k \in \mathbb{W} \subset \mathbb{R}^{d_x}, \tag{2.11}$$

where polytopes $\mathbb{X}, \mathbb{U}, \mathbb{W}$ contain the origin in their interior.

Note that by assuming $\mathbb{X}, \mathbb{U}, \mathbb{W}$ are polytopes, the problem is restricted to linear constraints. Accordingly, the computation of an appropriate robust positively invariant set and a domain of attraction becomes much easier. These tasks will be detailed in the sequel. Also, the definition of robust stability for system (2.9) in the sense of Lyapunov can be in the following form:

Definition 2.4.1 Given a robust positively invariant set Ω and a domain of attraction $\mathcal{X} \subseteq \mathbb{X}$, consider the linear system (2.9) subject to constraints (2.11) and a control law $u = \kappa(x) \in \mathbb{U}$. The closed loop is called *robustly stable* if there exists a Lyapunov function $V(x) : \mathcal{X} \to \mathbb{R}_+$ and an $\alpha \in [0, 1)$ such that:

— $V(x) = 0$ for all $x \in \Omega$, $V(x) > 0$ for all $x \in \mathcal{X} \backslash \Omega$,

— $V(A(k)x_k + B(k)\kappa(x_k) + w_k) - \alpha V(x_k) \leq 0$, $\forall w_k \in \mathbb{W}$, $\forall x_k \in \mathcal{X} \backslash \Omega$ and $\forall\ [A(k)\ B(k)] \in \Psi$.

2.4.1 Robust positively invariant sets

This section aims to recall some existing results for computing particular robust positively invariant sets for linear systems. First, it is important to choose a suitable state feedback $u_k = Kx_k \in \mathbb{U}$. This control law gain needs to satisfy that there exists a Lyapunov function $V(x)$ over a robust positively invariant domain, say Ω, such that

$$V((A(k) + B(k)K)x_k) - V(x_k) < 0 \quad \forall x_k \in \Omega,\ \forall\, [A(k)\ B(k)] \in \Psi.$$

The computation of such a gain is studied e.g. in Kothare et al. [1996]. A simple formulation is presented below, based on the same principle:

$$\min_{Z,Y} \; - logdet(Z)$$

subject to

$$Z = Z^T > 0$$
$$\begin{bmatrix} Z & (A_i Z + B_i Y)^T \\ A_i Z + B_i Y & Z \end{bmatrix} > 0 \;\text{ for all }\; i \in \mathcal{I}_L.$$

Then gain K is determined by

$$K = YZ^{-1}.$$

It is well known that the above formulation is a linear matrix inequality (LMI) problem and solvable by using semidefinite programming. Interested readers can find details in Boyd et al. [1994]. These conditions can be relaxed by using a parameter dependent Lyapunov function Daafouz and Bernussou [2001].

Note also that if system (2.9) does not take model uncertainty into account, such a gain K can be easily computed from the Riccati equation with some positive definite weighting matrices Q, R in the classical linear quadratic optimal control design.

With respect to the state feedback $u_k = Kx_k$, the computation of a robust positively invariant set for system (2.9) has been put forward in Nguyen [2014] and is recalled in Algorithm 2.1. Note however that before this study, many results for computing such positively invariant sets for a nominal system; i.e. system (2.9) without model uncertainties and bounded persistent disturbances, were presented, see Bitsoris [1988a,b], Bitsoris and Vassilaki [1995], Gilbert and Tan [1991], Vassilaki and Bitsoris [1989].

Algorithm 2.1 Computation of the maximal robust positively invariant set denoted by Ω_M, with respect to state feedback $u_k = Kx_k$.

Input: System (2.9), $u_k = Kx_k$, $\mathbb{X}$ given by (2.11).
Output: The maximal robust positively invariant set Ω_M.

1: Initialize $\Omega_M = \mathbb{X}$.
2: $\Omega_M^+ = \{x \in \Omega_M \mid (A_i + B_iK)x \oplus \mathbb{W} \subseteq \Omega_M \ \forall i \in \mathcal{I}_L\}$.
3: **If** $\Omega_M^+ = \Omega_M$ **then** Stop,
4: **Else** $\Omega_M \leftarrow \Omega_M^+$. Return to step 2.
5: **End**

Note also that prominent studies on the computation of the maximal and minimal positively invariant sets for a linear, discrete-time invariant system affected by bounded persistent disturbances are referred to Kolmanovsky and Gilbert [1998], Rakovic et al. [2005]. Still, if in system (2.9), additive disturbances are not taken into consideration, then the minimal robust positively invariant set coincides with the origin due to its asymptotic stability.

2.4.2 Controllable sets

A domain of attraction is of importance to design (robust) control laws for constrained systems, possibly affected by bounded additive disturbances and/or polytopic model uncertainties. Therefore, the computation of such a set is also of interest. In this section, a candidate for these domains of attraction called *controllable sets* [2] is discussed. These sets are defined as follows.

Definition 2.4.2 Consider system (2.9) subject to model uncertainty (2.10) and constraints (2.11). Let a robust positively invariant set Ω and $N \in \mathbb{N}_{>0}$ be given. A set denoted by $\mathcal{K}_N(\Omega) \subseteq \mathbb{X}$ is called the $N-$*steps controllable set* if any point belonging to this set, can reach to Ω in N steps, while staying inside $\mathbb{X}$ despite

---

2. Note that in Kerrigan [2001], these sets are also called *robust controllable sets*.

any disturbances in $\mathbb{W}$ and model uncertainties Ψ, i.e.

$$\begin{aligned}
&\mathcal{K}_0(\Omega) := \Omega, \\
&\mathcal{K}_N(\Omega) := \Big\{ x_0 \in \mathbb{X} \mid \exists u_0, \ldots, u_{N-1} \in \mathbb{U} \;\text{ s.t. }\; x_N \in \Omega, \\
&\qquad\qquad\qquad\qquad \forall w_i \in \mathbb{W}, \forall \left[A(i)\ B(i)\right] \in \Psi \;\text{ with }\; 0 \leq i \leq N-1 \Big\}.
\end{aligned}$$

Based on this definition, many algorithms to compute $\mathcal{K}_N(\Omega)$ are proposed in different studies e.g. Kerrigan [2001]. One of them is recalled in Algorithm 2.2.

Algorithm 2.2 Construct the $N-$steps controllable set $\mathcal{K}_N(\Omega)$

Input: $N \in \mathbb{N}_{>0},\ \ \Omega \subset \mathbb{X}$.

Output: $\mathcal{K}_N(\Omega)$

1: **For** $i = 1 : N$
2: $\quad \widehat{\mathcal{K}}_i := \left\{ (x, u) \in \mathbb{X} \times \mathbb{U} \mid (A_j x + B_j u) \oplus \mathbb{W} \subseteq \mathcal{K}_{i-1}(\Omega),\ \forall j \in \mathcal{I}_L \right\}.$
3: $\quad \mathcal{K}_i(\Omega) = \text{Proj}_{\mathbb{R}^{d_x}} \widehat{\mathcal{K}}_i.$
4: **End**

The *maximal controllable set* $\mathcal{K}_\infty(\Omega)$ can thus be approximated via Algorithm 2.2 when $N \to \infty$. Within an $\epsilon-$approximation, this algorithm is shown to be finitely terminated; i.e. there exists $N \in \mathbb{N}_{>0}$, such that $\mathcal{K}_N(\Omega) \subset \mathcal{K}_{N+1}(\Omega) \subset \mathcal{K}_N(\Omega) \oplus B_{d_x}(\epsilon)$, where $B_{d_x}(\epsilon) := \left\{ x \in \mathbb{R}^{d_x} \mid \|x\|_\infty \leq \epsilon \right\}$.

2.4.3 Contractive sets

Another candidate for the characterization of domain of attraction is a so-called *contractive set*. A necessary and sufficient condition for asymptotic stability of a nominal system is that the domain of attraction is (non necessarily successive) contractive as proved in Gutman and Cwikel [1986, 1987]. Later, this contractivity property is exploited for linear system affected by bounded additive disturbances and polytopic model uncertainties in Blanchini [1994]. Induced from this property, a Lyapunov function is found to guarantee robust stability. The definition of a *contractive set* is recalled as follows:

Definition 2.4.3 Consider system (2.9) subject to model uncertainty (2.10) and constraints (2.11). A set $\mathcal{X} \subseteq \mathbb{X}$ is called $\lambda-$*contractive* for $0 \leq \lambda < 1$ if for any $x_k \in \mathcal{X}$, there exists a control law $u_k = \kappa(x_k) \in \mathbb{U}$ such that

$$(A(k)x_k + B(k)\kappa(x_k)) \oplus \mathbb{W} \subseteq \lambda\mathcal{X},\ \forall\ [A(k)\ \ B(k)] \in \Psi.$$

Note that if $\lambda = 1$, then $\mathcal{X}$ becomes a robust control invariant set, as defined in Definition 2.3.7. The *maximal* $\lambda-$*contractive set* is defined as the set containing all $\lambda-$contractive sets. Such a set can also be chosen as a domain of attraction and will be of use to design robust control for system (2.9) via an appropriate optimization problem and will be presented in Chapter 5. For completeness, we recall a procedure to compute the maximal $\lambda-$contractive set for a given $0 \leq \lambda < 1$ in Algorithm 2.3. Such an algorithm has been put forward in Blanchini [1994].

Algorithm 2.3 Construct the maximal $\lambda-$contractive set

Input: System (2.9), constraints (2.11), model uncertainties (2.10), a given $\lambda \in [0,\ 1)$.

Output: The maximal $\lambda-$contractive set, denoted by P_λ.

1: $S = \mathbb{X}$.
2: $\widehat{S}^+ = \{(x, u) \in S \times \mathbb{U} \mid (A_i x + B_i u) \oplus \mathbb{W} \subseteq \lambda S,\ \forall i \in \mathcal{I}_L\}$.
3: $S^+ = \mathrm{Proj}_{\mathbb{R}^{d_x}} \widehat{S}^+$.
4: **If** $S^+ = S$, **then** Stop,
5: **Else** $S \leftarrow S^+$. Return to step 2.
6: **End**
7: $P_\lambda \leftarrow S$.

2.4.4 Additional notes

Throughout this section, some remarks to facilitate the computation of the sets presented in Sections 2.4.1, 2.4.2, 2.4.3 are discussed in-depth. In particular, given two polyhedra $P_1, P_2 \subseteq \mathbb{R}^d$, the following question will be investigated:

$$\textit{whether or not } P_1 \subseteq P_2.$$

Suppose the halfspace representation of P_1, P_2 are as follows:

$$\begin{aligned} P_1 &:= \left\{x \in \mathbb{R}^d \mid H_1 x \leq K_1\right\},\ H_1 \in \mathbb{R}^{r_1 \times d}, K_1 \in \mathbb{R}^{r_1} \\ P_2 &:= \left\{x \in \mathbb{R}^d \mid H_2 x \leq K_2\right\},\ H_2 \in \mathbb{R}^{r_2 \times d}, K_2 \in \mathbb{R}^{r_2}. \end{aligned} \tag{2.12}$$

This problem can be answered via the Extended Farkas Lemma Bitsoris [1988a,b], Schrijver [1998]. This result is recalled below.

Lemma 2.4.4 *Given two polyhedra* (2.12)*,* $P_1 \subseteq P_2$ *if there exists a matrix* $M = [m_{ij}] \in \mathbb{R}^{r_2 \times r_1}$ *such that:*

$$\begin{aligned} & m_{ij} \geq 0,\ \forall i \in \mathcal{I}_{r_2}, \forall j \in \mathcal{I}_{r_1}, \\ & M H_1 = H_2, \\ & M K_1 \leq K_2. \end{aligned} \tag{2.13}$$

Note also that (2.13) is an LMI problem [3]. Therefore, it can be solved by using semidefinite programming Boyd et al. [1994]. According to this lemma, if (2.13) is feasible, then $P_1 \subseteq P_2$, otherwise it is not the case.

If P_1 is a polytope, the verification becomes easier. Indeed, let the vertex representation of P_1 be given by:

$$P_1 = \operatorname{conv}\{v_1, \ldots, v_n\},$$

then $P_1 \subseteq P_2$ if

$$H_2 v_i \leq K_2, \ \forall i \in \mathcal{I}_n.$$

Dually, if P_1, P_2 are polytopes and also the vertex representation of P_2 is in the following form:

$$P_2 = \operatorname{conv}\{x_1, \ldots, x_m\},$$

then $P_1 \subseteq P_2$ leads to $v_i \in P_2$ for all $i \in \mathcal{I}_n$. For each v_i, there exist $\alpha_{ji} \geq 0$ for $j \in \mathcal{I}_m$ such that $\sum_{j \in \mathcal{I}_m} \alpha_{ji} = 1$ and $v_i = \sum_{j \in \mathcal{I}_m} \alpha_{ji} x_j$. In short, this inclusion test can be written in the form:

$$[v_1 \ldots v_n] = [x_1 \ldots x_m]\,\alpha, \ 1_m^T \alpha = 1_n^T, \ \alpha = [\alpha_{ji}] \in \mathbb{R}_+^{m \times n}.$$

Accordingly, Algorithms 2.1, 2.2, 2.3 can be enforced by the use of such simple optimization based tests of set inclusion. For instance, consider Algorithm 2.1, in step 2

$$(A_i + B_i K)x \oplus \mathbb{W} \subseteq \Omega_M$$

amounts to

$$(A_i + B_i K)x + w \in \Omega_M \ \text{ for all } \ w \in \mathcal{V}(\mathbb{W}).$$

More clearly, if the halfspace representation of Ω_M is given by $\Omega_M = \{x \in \mathbb{R}^{d_x} \mid Fx \leq h\}$, then inclusion $(A_i + B_i K)x + w \in \Omega_M$ is equivalent to:

$$F(A_i + B_i K)x \leq h - Fw.$$

To illustrate the above algorithms, consider the following system:

$$x^+ = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u + w.$$

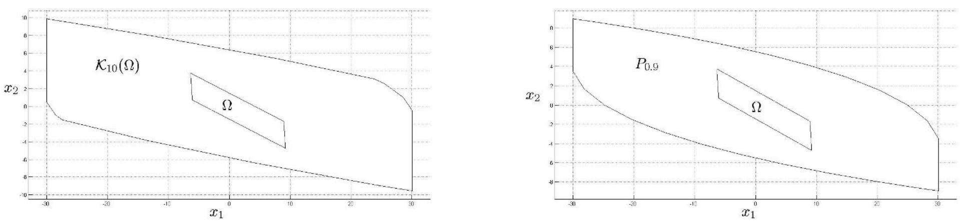
3. Recall that the above vector inequality implies the component-wise inequalities, i.e. given two vectors $a = [a_i], b = [b_i] \in \mathbb{R}^d$,

$$a \leq b \iff a_i \leq b_i, \ \forall i \in \mathcal{I}_d.$$

The system state, input and disturbances are subject to the following constraints:

$$\mathbb{X} = \left\{x \in \mathbb{R}^2 \mid \|x\|_\infty \leq 30\right\}, \mathbb{U} = [-2,\ 2], \mathbb{W} = \left\{w \in \mathbb{R}^2 \mid \|w\|_\infty \leq 0.5\right\}.$$

A linear state feedback $u = [-0.4854 \ \ -1.3676]\, x$ is computed from the Riccati equation with weighting matrices $Q = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, $R = 0.5$. The maximal robust positively invariant set Ω with respect to the above linear controller and the $10-$steps controllable set are shown in Figure 2.4a. Also, the maximal $0.9-$contractive set $P_{0.9}$ is presented in Figure 2.4b.



(a) Ω and $\mathcal{K}_{10}(\Omega)$ (b) Ω and $P_{0.9}$

Figure 2.4: An illustrative example for the maximal robust positively invariant set Ω, the $10-$steps controllable set $\mathcal{K}_{10}(\Omega)$ and the maximal $0.9-$contractive set $P_{0.9}$.

2.5 Model predictive control

2.5.1 Basics of model predictive control

Model predictive control (MPC) has attracted great attention from control community due to its relevance in industrial applications, and its compatibility in related domains e.g. optimization. It aims to solve an optimization problem over a suitable finite horizon scheme subject to constraints, at each sampling time, where the current state is known and the finite sequence of controls stands for the decision variable. In this optimization problem, a model of the given system is of use to predict *future behaviors* of the system. The first control in this sequence is then applied to the given system, see more in Richalet et al. [1978].

A reason for a great interest in MPC may come from the fact that any industrial process has *physical constraints*. Also, optimization problem can easily take constraints into consideration even if they are non-convex. An MPC problem is recalled as follows.

Let system (2.4) and constraints (2.5) be considered. An MPC problem aims to minimize a cost function:

$$J(\mathbf{u}, x_k) = \sum_{i=0}^{N-1} L(x_{k+i|k}, u_{k+i|k}) + V(x_{k+N|k}), \tag{2.14}$$

where $x_{k+i|k}, u_{k+i|k}$ represent the state and control variables at time $k+i$, predicted at time k. $\mathbf{u}$ denotes the sequence of controls over the prediction horizon

$$\mathbf{u} = \left[u_{k|k}^T \ldots u_{k+N-1|k}^T\right]^T.$$

Also $L(x_{k+i|k}, u_{k+i|k})$ stands for the stage cost and $V(x_{k+N|k})$ denotes the terminal cost function. It is assumed that the stage cost $L(x_{k+i|k}, u_{k+i|k})$ represents a continuous, time-invariant, non-negative function defined over $\mathbb{X} \times \mathbb{U}$. Still, the terminal cost $V(x_{k+N|k})$ represents a continuous, time-invariant, non-negative function defined over $\mathbb{X}$.

Appropriate constraints may be added to guarantee the stability:

$$x_{k+N|k} \in \mathcal{X}_f \subset \mathbb{X}. \tag{2.15}$$

In summary, with the measured state, MPC requires solving the following optimization problem at each sampling time:

$$\begin{aligned} \mathbf{u}^* &= \arg\min_{\mathbf{u}} J(\mathbf{u}, x_k) \\ &\text{subject to} \\ & x_{k+i+1|k} = f(x_{k+i|k}, u_{k+i|k}),\ \forall i = 0, \ldots, N-1,\ x_{k|k} = x_k \\ & x_{k+i|k} \in \mathbb{X},\ u_{k+i|k} \in \mathbb{U},\ \forall i = 0, \ldots, N-1 \\ & x_{k+N|k} \in \mathcal{X}_f. \end{aligned} \tag{2.16}$$

The first control $u^*(x_k) = u^*_{k|k} = \mathbf{u}^*(1 : d_u, \cdot)$ is then applied to the dynamical system. This procedure is repeated with the next measured state, predicted by the closed loop $x_{k+1} = f(x_k, u^*(x_k))$.

Remark 2.5.1 Note that the terminal cost function $V(x_{k+N|k})$ is usually chosen such that the optimal cost function $J(\mathbf{u}^*, x_k)$ represents a Lyapunov function.

From the optimization viewpoint, problem (2.16) can be alternatively expressed in the following form:

$$\begin{aligned} \mathbf{u}^* &= \arg\min_{\mathbf{u}} J(\mathbf{u}, x_k) \\ &\text{subject to} \\ & g_i(\mathbf{u}, x_k) \leq 0,\ \forall i = 1, \ldots, m, \\ & h_j(\mathbf{u}, x_k) = 0,\ \forall j = 1, \ldots, n, \end{aligned} \tag{2.17}$$

where $g_i(\mathbf{u}, x_k)$ represents inequality constraints and $h_j(\mathbf{u}, x_k)$ represents equality constraints.

It is well known that optimal solution to problem (2.17) can also be approximated by so-called *piecewise affine functions* with respect to a pre-chosen tolerable error Grancharova and Johansen [2012], Johansen [2002, 2004]. Note that problem (2.17) is in the general case a nonlinear programming problem. Finding explicitly exact optimal solution is thus very difficult, even the exact optimal solution does not possess the piecewise affine structure. However, in certain cases, exact optimal solution can be sought and also represents a piecewise affine function.

2.5.2 Explicit solutions

As mentioned before, in certain cases, exact optimal solutions to particular MPC problems possess the piecewise affine structure. Such particular MPC problems are called throughout this manuscript *linear MPC*. It is shown in Bemporad et al. [2002], Feller et al. [2013], Gutman and Cwikel [1987], Olaru and Dumur [2004], Pistikopoulos et al. [2007], Seron et al. [2003], Tøndel et al. [2003] that linear MPC can be characterized by linear, time-invariant systems:

$$x_{k+1} = Ax_k + Bu_k, \tag{2.18}$$

subject to linear constraints and the cost function as a linear or quadratic function. Namely, the state constraints $\mathbb{X}$, the control input constraints $\mathbb{U}$ as in (2.11) and the terminal constraints $\mathcal{X}_f$ as in (2.15), represent polyhedra. Also, the cost function (2.14) in a linear MPC problem is usually expressed in the following forms:

— the stage cost and the terminal cost take a $2-$norm form; i.e.

$$\begin{aligned} L(x_{k+i|k}, u_{k+i|k}) &= x_{k+i|k}^T Q x_{k+i|k} + u_{k+i|k}^T R u_{k+i|k}, \\ V(x_{k+N|k}) &= x_{k+N|k}^T P x_{k+N|k}, \end{aligned}$$

where $Q = Q^T$ is a positive semi-definite matrix and R, P are positive definite, symmetric matrices.

— the stage cost and the terminal cost take an $1/\infty-$norm form; i.e.

$$\begin{aligned} L(x_{k+i|k}, u_{k+i|k}) &= \|Qx_{k+i|k}\|_p + \|Ru_{k+i|k}\|_p, \\ V(x_{k+N|k}) &= \|Px_{k+N|k}\|_p, \end{aligned}$$

where $p = 1/\infty$ and Q, R, P are matrices of appropriate dimensions.

From the optimization viewpoint, such a linear MPC problem can be presented in the following form:

$$\begin{aligned} \mathbf{u}^*(x_k) = \arg\min_{\mathbf{u}} \ & \mathbf{u}^T H \mathbf{u} + (C^T x_k + D)^T \mathbf{u} \\ \text{subject to} \quad & G\mathbf{u} \leq Ex_k + W \end{aligned} \tag{2.19}$$

where $\mathbf{u}$ stands for the decision variable and the current state x_k represents the parameter. $H = H^T$ is a positive semidefinite matrix, C, D, G, E, W represent matrices of appropriate dimension.

Explicit optimal solution to (2.19) has the following form:

$$\mathbf{u}^*(x_k) = F_i x_k + G_i \quad \text{for} \quad x_k \in \mathcal{X}_i, \tag{2.20}$$

where the state space $\mathcal{X}$ is splitted into finite many polyhedral regions

$$\mathcal{X} = \bigcup_{i \in \mathcal{I}_N} \mathcal{X}_i \subseteq \mathbb{X}. \tag{2.21}$$

This representation of optimal solution to (2.19) is called as introduced at the beginning *piecewise affine function* Bemporad et al. [2002].

Particular structure of such piecewise affine control laws allows for a change in the implementation, compared to the traditional fashion. Namely, instead of solving online an MPC problem, these particular control laws only require a simple function evaluation, since control law gains are embedded into a look-up table, stored at the hardware level Kvasnica [2009].

Remark 2.5.2 To guarantee that the optimal cost function

$$J(\mathbf{u}^*(x_k), x_k) = (\mathbf{u}^*)^T(x_k) H \mathbf{u}^*(x_k) + (C^T x_k + D)^T \mathbf{u}^*(x_k)$$

is a Lyapunov function, P can be computed as the solution to the Riccati equation in case of $2-$norm. Otherwise, in case of $1/\infty-$norm, the computation of P becomes more complicated. The counterpart of the Riccati equation for these cases is named the *Minkowski-Lyapunov equation*. Recent remarkable result for these cases is presented in Raković and Lazar [2014].

Remark 2.5.3 It is shown in Bemporad et al. [2003], Nguyen et al. [2011] that if an MPC problem (2.16) satisfies the following conditions:

— the system is linear, subject to polytopic uncertainties and bounded additive disturbances as in (2.9),

— $\mathbb{X}, \mathbb{U}, \mathcal{X}_f$ are polyhedra,

— the cost function (2.14) represents a linear function of $\mathbf{u}$ and x_k.

then its optimal solution also represents a piecewise affine function.

To illustrate this explicit solution, consider again the double integrator system:

$$\begin{aligned} x_{k+1} &= \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} x_k + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u_k, \\ y_k &= \begin{bmatrix} 1 & 0 \end{bmatrix} x. \end{aligned}$$

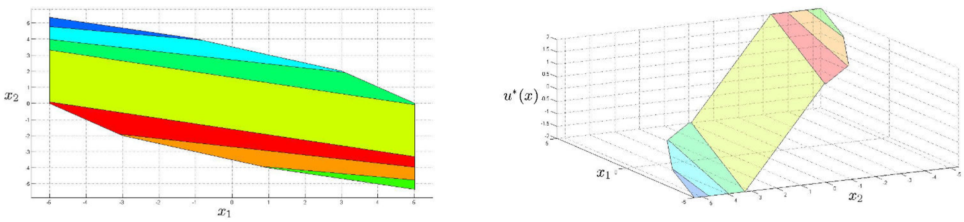
The output and control variables are subject to constraints:

$$y_k \in [-5,\ 5]\,,\ \ u_k \in [-2,\ 2]\,.$$

The weighting matrices are chosen as follows: $Q = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, $R = 1$. The linear unconstrained control and terminal matrix P computed from the Riccati equation are as follows:

$$u_k = \begin{bmatrix} 0.4221 & 1.2439 \end{bmatrix} x_k,\ \ P = \begin{bmatrix} 2.9471 & 2.3692 \\ 2.3692 & 4.6131 \end{bmatrix}.$$

Accordingly, the terminal constraints $\mathcal{X}_f$ are chosen to be the maximal output



(a) State space parition (b) Piecewise affine controller

Figure 2.5: An illustrative example for explicit solution.

admissible set, see Gilbert and Tan [1991]. The cost function is defined over the prediction horizon 10:

$$J(\mathbf{u}, x_k) = \sum_{i=0}^{9} (x_{k+i|k}^T Q x_{k+i|k} + u_{k+i|k}^T R u_{k+i|k}) + x_{k+10|k}^T P x_{k+10|k}. \tag{2.22}$$

Resulted from this MPC problem, the state space partition is presented in Figure 2.5a where the yellow region represents the terminal constraints $\mathcal{X}_f$. Also, the optimal control as the first element in the sequence of optimization argument is shown in Figure 2.5b.

Chapter 3

Explicit robustness and fragility margins for PWA controllers

Contents

Starting from the analysis stage, this chapter focuses on robustness and fragility analysis problems raised from the implementation of PWA control laws. These problems consider capacities of a given PWA controller to cope with different disturbing sources which can affect closed loop stability. Most of the results in this chapter were presented in Nguyen et al. [2014a, 2015a], Olaru et al. [2013].

3.1 Introduction

When analyzing a control law, both practitioner and theoretician take into account the capacity to cope with disturbances and model uncertainties. This characteristic is classically denoted in control theory as *robustness*. The presence of additive disturbances in the control system structure is due to measurement noises and external perturbation sources. Otherwise, the uncertainty stems from model reduction, linearization of nonlinear elements, imperfect mathematical model or partial information on the parameters. These elements are unavoidable in the control design by the essence of their causes and the practical need of complexity reduction in model-based design. As a consequence, the robustness consideration of the closed-loop is necessary.

First, this chapter tackles the robustness problem in the presence of model uncertainty for PWA feedback control laws. This class of control laws is shown to lead in closed-loop to a hybrid system formulation Heemels et al. [2001]. Another reason for interest in PWA controllers and their robustness is motivated from great attention in optimization based control, particularly in model predictive control (MPC) via parametric convex programming problem as introduced in Section 2.5.2.

Different types of uncertainties exist in the robust control literature c.f. Boyd et al. [1994], Kothare et al. [1996], Rugh and Shamma [2000], Zhou et al. [1996]. Two important classes are referred to as:

— the parametric uncertainties understood as variations of coefficients of the model with a pre-imposed structure,

— uncertainty covering all the neglected dynamics and other norm-bounded uncertainties.

Throughout this manuscript, the former one is of interest due to the fact that unstructured, norm-bounded uncertainties in general lead to an augmented state space and the extension of a predefined controller leads to nonuniqueness and related well-posedness problems which go beyond the scope of the current study of PWA dynamics defined over a given state space partition.

Subsequently, from the practical viewpoint, closed-loop stability may be affected in the implementation of control laws due to the fact that numerical round-offs usually happen. The set of admissible variations of control law gains, for which the implemented control law still guarantees closed loop stability, is named as the *fragility margin*. Note that studies on this problem are found in literature in Dorato [1998], Keel and Bhattacharyya [1997]. Unfortunately, these studies neither present a procedure to explicitly compute such a margin, nor cover the present interest in PWA control laws. As far as it concerns the fragility margin for PWA control laws, the set of possible inaccuracies on the local control law gains for each region is referred to without assuming any uncertainty on the description of the associated state space partition. Taking uncertainties on the region description into consideration will lead to overlapping regions and topological changes in state space partition, with implications in nonuniqueness of the trajectories.

Instead, based on the same methodology, the problem of possible inaccuracies on the region description will be independently considered without taking into account numerical errors on control law gains. Still, the study will be subsequently extended to find admissible additive disturbances with free variations on local control law gains.

The methodology presented next will rely on the *robust positive invariance principles* Bitsoris [1988b], Blanchini [1999], Blanchini and Miani [2008], Vassilaki et al. [1988]. Robust positive invariance is known to be associated with *robust stability* by the fact that the trajectories are maintained inside a subset of the state space, namely a positively invariant set. Guaranteeing *robust asymptotic stability* needs further convergence and is not treated in this chapter. Note however that verifying asymptotic stability may incorporate the existence of a suitable Lyapunov function once the positive invariance is fulfilled. This work for a nominal system can be found in Hovd and Olaru [2012], Hovd et al. [2010].

3.2 Preliminaries

This section aims to recall some basic definitions needed for the developments in the sequel to facilitate the presentation and problem formulations.

Definition 3.2.1 A collection of N full-dimensional polyhedra $\mathcal{X}_i \subset \mathbb{R}^{d_x}$, denoted by $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$, is called a *polyhedral partition of a polyhedron* $\mathcal{X} \subseteq \mathbb{R}^{d_x}$ if the following conditions hold:

— $\bigcup_{i\in\mathcal{I}_N} \mathcal{X}_i = \mathcal{X}$,

— $\mathrm{int}(\mathcal{X}_i) \cap \mathrm{int}(\mathcal{X}_j) = \emptyset,\ \forall (i,j) \in \mathcal{I}_N^2,\ i \neq j$.
Two regions $\mathcal{X}_i, \mathcal{X}_j$ are called *neighboring* or *adjacent* if $i \neq j$, $(i,j) \in \mathcal{I}_N^2$, $\dim(\mathcal{X}_i \cap \mathcal{X}_j) = d_x - 1$. Further, if $\mathcal{X}$ is a polytope, then $\{\mathcal{X}_i\}_{i \in \mathcal{I}_N}$ is called a *polytopic partition*.

Based on a polyhedral partition, the definition of a PWA function is defined as follows:

Definition 3.2.2 A function $f_{pwa}(x) : \mathcal{X} \subseteq \mathbb{R}^{d_x} \to \mathbb{R}^{d_u}$, defined over a polyhedral partition $\{\mathcal{X}_i\}_{i \in \mathcal{I}_N}$ of a polyhedron $\mathcal{X}$, is called a *piecewise affine function* if

$$f_{pwa}(x) = H_i x + G_i, \quad \text{for} \ \ x \in \mathcal{X}_i, \tag{3.1}$$

where $H_i \in \mathbb{R}^{d_u \times d_x}, G_i \in \mathbb{R}^{d_u}$.

The *continuity* property of a PWA function is of interest.

Definition 3.2.3 Such a PWA function $f_{pwa}(x) : \mathcal{X} \subseteq \mathbb{R}^{d_x} \to \mathbb{R}^{d_u}$, is *continuous* if for any pair of neighboring regions $(\mathcal{X}_i, \mathcal{X}_j)$, in the polyhedral partition $\{\mathcal{X}_i\}_{i \in \mathcal{I}_N}$:

$$H_i x + G_i = H_j x + G_j, \quad \text{for any} \ \ x \in \mathcal{X}_i \cap \mathcal{X}_j.$$

From the above definition, it can be seen that the discontinuity of a PWA function happens at common frontiers between neighboring regions. Further, at any point on common frontiers, a PWA function can take any value. However, we restrict attention to the discontinuous functions $f_{pwa}(x)$ defined as follows throughout this thesis:

$$f_{pwa}(x) = \begin{cases} H_i x + G_i \text{ for } x \in \mathrm{int}(\mathcal{X}_i), \\ H_i x + G_i \text{ or } H_j x + G_j \text{ for } x \in \mathcal{X}_i \cap \mathcal{X}_j, \end{cases} \tag{3.2}$$

for every pair of neighboring regions $(\mathcal{X}_i, \mathcal{X}_j)$.

Another way to deal with these discontinuities is to take *multi-values* into account at the frontiers. Namely, for a pair of neighboring regions, the given function at each point of their common frontier can take many different values. This extended multi-valued function can be alternatively called a *set-valued map*.

Consider a discrete linear time invariant (LTI) system:

$$x_{k+1} = A x_k + B u_k, \tag{3.3}$$

where $A \in \mathbb{R}^{d_x \times d_x}, B \in \mathbb{R}^{d_x \times d_u}$.

If system (3.3) is controlled by a PWA control law as defined in Definition 3.2.2, then the closed loop is a piecewise affine system, i.e.

$$x_{k+1} = (A + BH_i) x_k + BG_i, \quad \text{for} \ \ x_k \in \mathcal{X}_i.$$

If the region $\mathcal{X}$, over which the given control law (3.1) is defined, is positively invariant (for the definition of a positively invariant set, the reader is referred to Section 2.3.2), the above closed-loop system is stable.

The developments of this chapter rely on the following standing assumptions:

Assumption 3.2.4 $\mathcal{X}$ is a polytope.

Assumption 3.2.5 $\mathcal{X}$ is assumed to be positively invariant with respect to the LTI system (3.3) and the PWA control law (3.1).

Assumption 3.2.6 The PWA controller (3.1) is continuous.

Assumption 3.2.7 $0 \in \operatorname{int}(\mathcal{X})$.

Note that the feasible region $\mathcal{X}$ can be in general not convex. This may happen in MPC problems whose constraints are not convex Grancharova and Johansen [2012]. Assumption 3.2.4 restricts the problem to a convex framework, particularly to polytopes which is a common characterization of MPC problems based on linear prediction models and linear input, state constraints. Also, Assumption 3.2.5 should be understood as a guarantee of proper MPC design, in the sense that the trajectories are kept inside the state space $\mathcal{X}$ by the given PWA controller.

If $\mathcal{X}$ represents a polytope, then $\{\mathcal{X}_i\}_{i \in \mathcal{I}_N}$, associated with the PWA control law (3.1), is a polytopic partition. It follows that each region in this partition is also a polytope. It will be assumed (see details in Section 2.2) that these polytopes are described via both the vertex and halfspace representations.

Assumptions 3.2.6, 3.2.7 are common properties of PWA control laws obtained from MPC design and can be relaxed without any effect on main results of this chapter.

For a unified use of notation, the halfspace representation of the polytopes of interest is presented as follows:

$$
\begin{aligned}
\mathcal{X} &= \left\{x \in \mathbb{R}^{d_x} \mid Fx \leq h\right\}, \quad \text{where } F \in \mathbb{R}^{r \times d_x},\ h \in \mathbb{R}^r \\
\mathcal{X}_i &= \left\{x \in \mathbb{R}^{d_x} \mid F_i x \leq h_i\right\}, \quad \text{where } F_i \in \mathbb{R}^{r_i \times d_x},\ h_i \in \mathbb{R}^{r_i},
\end{aligned}
\tag{3.4}
$$

for every $i \in \mathcal{I}_N$. Also, the vertices of polytopes $\mathcal{X}$, $\mathcal{X}_i$ are denoted by

$$
\begin{aligned}
\mathcal{V}(\mathcal{X}) &= \{v_1, \dots, v_q\} \\
\mathcal{V}(\mathcal{X}_i) &= \{w_{i1}, \dots, w_{iq_i}\}, \ \forall i \in \mathcal{I}_N.
\end{aligned}
$$

The vertex representation of these polytopes as shown in Section 2.2 can be expressed in the following form:

$$
\begin{aligned}
\mathcal{X} &= \operatorname{conv}\{v_1, \dots, v_q\} \\
\mathcal{X}_i &= \operatorname{conv}\{w_{i1}, \dots, w_{iq_i}\}, \ \forall i \in \mathcal{I}_N.
\end{aligned}
\tag{3.5}
$$

Define also the following sets:

$$\begin{aligned} \mathcal{W}_i &= \mathcal{V}(\mathcal{X}_i), \quad \text{for all} \quad i \in \mathcal{I}_N, \\ \mathcal{W} &= \bigcup_{i \in \mathcal{I}_N} \mathcal{V}(\mathcal{X}_i). \end{aligned} \tag{3.6}$$

Based on the above notations, the definition of the following matrices are also of help for the forthcoming sections:

$$\begin{aligned} V &= [\mathcal{V}(\mathcal{X})] \in \mathbb{R}^{d_x \times q}, \ U = [f_{pwa}(\mathcal{W})] \in \mathbb{R}^{d_u \times p} \\ V_i &= [\mathcal{W}_i] \in \mathbb{R}^{d_x \times q_i}, \ U_i = [f_{pwa}(\mathcal{W}_i)] \in \mathbb{R}^{d_u \times q_i} \\ W &= [\mathcal{W}] \in \mathbb{R}^{d_x \times p}. \end{aligned} \tag{3.7}$$

3.3 Explicit robustness margin for PWA control laws

3.3.1 Problem formulation

The robustness problem aims to find the set of model uncertainties for which closed loop stability is still ensured. Accordingly, a discrete-time linear system as (2.9) is considered. However, additive disturbances are not taken into consideration here. More precisely, the system is in the following form:

$$x_{k+1} = A(k)x_k + B(k)u_k, \tag{3.8}$$

where $[A(k) \ B(k)]$ is assumed to belong to a model uncertainty set Ψ as in (2.10). For reading ease, this uncertainty set is recalled below:

$$\Psi := \text{conv}\left\{[A_1 \ B_1], \ldots, [A_L \ B_L]\right\}. \tag{3.9}$$

System (3.8) is controlled by a PWA control law (3.1). Then, the closed loop can be expressed by

$$x_{k+1} = (A(k) + B(k)H_i)x_k + B(k)G_i, \quad \text{for} \quad x_k \in \mathcal{X}_i. \tag{3.10}$$

The main goal is to characterize the *robustness margin* defined as the set of model uncertainties, denoted by $\Psi_{\text{rob}} \subseteq \Psi$ such that $\mathcal{X}$ is positively invariant with respect to the closed loop (3.10) for any $[A(k) \ B(k)] \in \Psi_{\text{rob}}$.

Recall that a polytope can be expressed by the convex hull of its vertices, represented as vectors in an Euclidean space. However, in case these vertices are matrices, each vertex will be understood as a vector composed of the elements of this matrix. In other words, the polytopic set (3.9) can be equivalently written in the following form:

$$\Psi = \text{conv}\left\{\text{vec}([A_1 \ B_1]), \ldots, \text{vec}([A_L \ B_L])\right\}.$$

For ease of presentation, the set Ψ defined as in (3.9) is also denoted as *polytopic model uncertainty set*. Then, any model $[A\ B] \in \Psi$ can be written in the form of a convex combination of the extreme model realizations; i.e. there exists $\alpha = [\alpha_1 \ldots \alpha_L] \in \mathbb{R}_+^L$ such that $\sum_{i=1}^{L} \alpha_i = 1$ and

$$[A\ B] = \sum_{i=1}^{L} \alpha_i \, [A_i\ B_i] \, . \tag{3.11}$$

Based on the above premise, computing the robustness margin Ψ_{rob} amounts to finding its associated set of coefficients α in (3.11), denoted by $\Psi_{\mathrm{rob}}^{\alpha}$ due the fact that Ψ_{rob} and $\Psi_{\mathrm{rob}}^{\alpha}$ are isomorphic. Clearly, $\Psi_{\mathrm{rob}}^{\alpha} \subseteq \mathcal{S}_L$ where $\mathcal{S}_L$ is the unit simplex defined in (6). The definition of $\Psi_{\mathrm{rob}}^{\alpha}$ can be written in the following form:

$$\Psi_{\mathrm{rob}}^{\alpha} := \left\{ \alpha \in \mathcal{S}_L \mid \sum_{i=1}^{L} \alpha_i \, [A_i\ B_i] \in \Psi_{\mathrm{rob}} \right\}.$$

The robustness margin Ψ_{rob} can be characterized, based on the local structure of PWA dynamics. The first fundamental property of such a set Ψ_{rob} is stated via the following theorem. This theorem shows that the robustness margin inherits the convexity despite the non-linearity of the closed loop dynamics.

Theorem 3.3.1 Ψ_{rob} *is a convex set.*

Proof: Consider $[A(k_1)\ B(k_1)]\,,\ [A(k_2)\ B(k_2)] \in \Psi_{\mathrm{rob}}$. It will be proved that $\mu\,[A(k_1)\ B(k_1)] + (1-\mu)\,[A(k_2)\ B(k_2)] \in \Psi_{\mathrm{rob}}$, for any $\mu \in [0,\ 1]\,.$

Indeed, due to the positive invariance of $\mathcal{X}$, the following holds for any $i \in \mathcal{I}_N$:

$$\begin{aligned} (A(k_1) + B(k_1)H_i)\mathcal{X}_i \oplus B(k_1)G_i \subseteq \mathcal{X} \\ (A(k_2) + B(k_2)H_i)\mathcal{X}_i \oplus B(k_2)G_i \subseteq \mathcal{X} \end{aligned}$$

Due to the convexity of $\mathcal{X}$, then for any $\mu \in [0,\ 1]$

$$\begin{aligned} (1-\mu)((A(k_2)+&B(k_2)H_i)\mathcal{X}_i \oplus B(k_2)G_i) \\ &\oplus \mu((A(k_1) + B(k_1)H_i)\mathcal{X}_i \oplus B(k_1)G_i) \subseteq \mathcal{X}. \end{aligned} \tag{3.12}$$

Inclusion (3.12) implies that $\mu\,[A(k_1)\ B(k_1)] + (1-\mu)\,[A(k_2)\ B(k_2)] \in \Psi_{\mathrm{rob}}$, meaning the convexity of Ψ_{rob}. $\square$

Remark 3.3.2 Note that if $L < d_x(d_x + d_u)$ then the representation of the robustness margin via $\Psi_{\mathrm{rob}}^{\alpha}$ is more effective than the one via the elements of state space matrices $\mathrm{vect}([A\ B])$, in case all the elements of the state space matrices $[A(k)\ B(k)]$ in (3.8) are uncertain. Otherwise, the constructive results presented in the sequel, still handle the latter case.

3.3.2 Construction based on the vertex representation

Once the notation is given in (3.4)–(3.7) and the structure of the robustness margin is clarified, its computation starting from the vertex representation, is presented in the following theorem.

Theorem 3.3.3 *Consider system* (3.8) *subject to model uncertainties* (3.9). *For a PWA control law* (3.1) *satisfying Assumptions 3.2.4, 3.2.5, 3.2.6, the robustness margin can be obtained by:*

$$\Psi_{\text{rob}}^{\alpha} = \text{Proj}_{\mathbb{R}^L} \Upsilon_v \tag{3.13}$$

where Υ_v *represents the following set:*

$$\Upsilon_v = \Bigg\{ (\alpha, M) \in \mathcal{S}_L \times \mathbb{R}_+^{q \times p} \mid \\ \sum_{j=1}^{L} \alpha_j (A_j W + B_j U) = VM,\ 1^T M = 1^T \Bigg\}, \tag{3.14}$$

with W, U, V defined in (3.7), $p = \text{Card}(\mathcal{W}), q = \text{Card}(\mathcal{V}(\mathcal{X}))$, $\mathcal{S}_L$ *defined in* (6) *and* M *represents any matrix with the non-negative elements, satisfying* (3.14).

Proof: If Ψ_{rob} denotes the robustness margin, then for any $[A(k)\ B(k)] \in \Psi_{\text{rob}}$, the following holds due to the positive invariance of $\mathcal{X}$:

$$(A(k) + B(k)H_i)x + B(k)G_i \in \mathcal{X},\ \forall x \in \mathcal{X}_i,\ \forall i \in \mathcal{I}_N. \tag{3.15}$$

Any $[A(k)\ B(k)] \in \Psi_{\text{rob}} \subseteq \Psi$ is expressible by a convex combination of the extreme models. It follows that (3.15) can be written in the following form:

$$\sum_{j=1}^{L} \alpha_j (A_j + B_j H_i)x + \alpha_j B_j G_i \in \mathcal{X}, \tag{3.16}$$

for some $\alpha = [\alpha_1 \ldots \alpha_L]^T \in \mathcal{S}_L$. Note that inclusion (3.16) holds true for any $x \in \mathcal{X}_i$, then it holds for all vertices of $\mathcal{X}_i$. Namely,

$$\sum_{j=1}^{L} \alpha_j (A_j + B_j H_i)w_{il} + \alpha_j B_j G_i \in \mathcal{X},\quad \text{for all}\ \ l \in \mathcal{I}_{q_i}, \tag{3.17}$$

where w_{il}, $l \in \mathcal{I}_{q_i}$ are defined in (3.5). For each $w_{il} \in \mathcal{V}(\mathcal{X}_i)$, there exists $y_{il} \in \mathcal{X}$ such that

$$\sum_{j=1}^{L} \alpha_j (A_j + B_j H_i)w_{il} + \alpha_j B_j G_i = y_{il}. \tag{3.18}$$

Note also that $y_{il} \in \mathcal{X}$ can be written in the form of a convex combination of the vertices of $\mathcal{X}$, defined in (3.5), then there exists $\gamma_{il} \in \mathcal{S}_q$ i.e. $1^T \gamma_{il} = 1$ such that

$$y_{il} = [\mathcal{V}(\mathcal{X})]\, \gamma_{il} = V \gamma_{il}.$$

This end leads to another representation of (3.18) as follows:

$$\sum_{j=1}^{L} \alpha_j (A_j + B_j H_i) w_{il} + \alpha_j B_j G_i = V \gamma_{il}. \tag{3.19}$$

Note that based on (3.15), inclusion (3.19) holds for every $i \in \mathcal{I}_N$, $l \in \mathcal{I}_{q_i}$. This fact leads to

$$\sum_{j=1}^{L} \alpha_j (A_j W + B_j U) = V M,$$

where matrix $M \in \mathbb{R}_+^{q \times p}$ is composed of the above vectors γ_{il} as its columns with respect to the notation from (3.4)–(3.7); i.e. $1^T M = 1^T$.

Finally, it can be observed that Υ_v as defined in (3.14) represents a parameterized polyhedron in the space of α and the elements of M. Therefore, the robustness margin can be obtained from the orthogonal projection of Υ_v onto the space of α as presented in (3.13). □

This result can also be of help to certificate whether the given PWA controller is robust with respect to a given polytopic model uncertainty set. Accordingly, if there exist matrices $M_1, \ldots, M_L$ of non-negative elements such that:

$$A_i W + B_i U = V M_i,\ 1_q^T M_i = 1_p^T,\ \forall i \in \mathcal{I}_L,$$

then this PWA controller can be said to be robust.

3.3.3 Construction based on the halfspace representation

Based on the same methodology, but starting from the halfspace representation of a polytope, the next theorem presents a dual approach for computing the robustness margin. Note also that the notation of interest has already been introduced in (3.4)–(3.7).

Theorem 3.3.4 *Consider system* (3.8) *subject to uncertainties* (3.9) *and a PWA control law* (3.1) *satisfying Assumptions 3.2.4, 3.2.5, 3.2.6. The robustness margin can be obtained by:*

$$\Psi_{\text{rob}}^{\alpha} = \text{Proj}_{\mathbb{R}^L} \Upsilon_h \tag{3.20}$$

where Υ_h represents the following set:

$$\begin{aligned}\Upsilon_h = \Big\{ &(\alpha, M_1, \ldots, M_N) \in \mathcal{S}_L \times \mathbb{R}_+^{r\times r_1} \times \ldots \times \mathbb{R}_+^{r\times r_N} \mid \\ &\sum_{j=1}^{L} \alpha_j F(A_j + B_j H_i) = M_i F_i,\ M_i h_i \leq h - F\sum_{j=1}^{L} \alpha_j B_j G_i,\ \forall i \in \mathcal{I}_N \Big\},\end{aligned} \tag{3.21}$$

and M_i, $i \in \mathcal{I}_N$ are matrices of non-negative elements satisfying constraints (3.21).

Proof: For any $[A(k)\ B(k)] \in \Psi_{\text{rob}}$, $i \in \mathcal{I}_N$, the positive invariance of $\mathcal{X}$ leads to $(A(k) + B(k)H_i)x + B(k)G_i \in \mathcal{X}$, for every $x \in \mathcal{X}_i$. It follows that

$$\mathcal{X}_i \subseteq \{x \in \mathcal{X} \mid (A(k) + B(k)H_i)x + B(k)G_i \in \mathcal{X}\}.$$

In other words, this inclusion can be expressed in the following form

$$\{x \in \mathcal{X} \mid F_i x \leq h_i\} \subseteq \{x \in \mathcal{X} \mid F((A(k) + B(k)H_i)x + B(k)G_i) \leq h\}.$$

Using the Extended Farkas Lemma as presented in Section 2.4.4, there exists a matrix M_i with the non-negative elements; i.e. $M_i \in \mathbb{R}_+^{r\times r_i}$ satisfying:

$$\begin{aligned} &F(A(k) + B(k)H_i) = M_i F_i, \\ &M_i h_i \leq h - FB(k)G_i \end{aligned} \tag{3.22}$$

Also, any $[A(k)\ B(k)] \in \Psi$ can be written in the form of a convex combination of the extreme models, then (3.22) can be expressed as follows:

$$\begin{aligned} &\sum_{j=1}^{L} \alpha_j F(A_j + B_j H_i) = M_i F_i, \\ &M_i h_i \leq h - F\sum_{j=1}^{L} \alpha_j B_j G_i. \end{aligned} \tag{3.23}$$

Note also that (3.23) holds true $\forall i \in \mathcal{I}_N$, leading to (3.21). It can be observed that Υ_h as defined in (3.21), represents a polyhedron of α and the elements of M_i, $\forall i \in \mathcal{I}_N$, therefore the robustness margin can be obtained from the orthogonal projection of Υ_h onto the space of α, as presented in (3.20). $\square$

Remark 3.3.5 Note that the sets Υ_v, Υ_h, defined in (3.14), (3.21), respectively are described by linear constraints. However, these are not in the regular forms introduced in Section 2.2. Therefore, the robustness margin computation might not be straightforward from the compact expressions in (3.14) and (3.21). These technical aspects will be detailed in Section 3.7.

3.3.4 Illustrative example

An illustration is carried out on a linear system with uncertainty set described by:

$$[A_1\ B_1] = \begin{bmatrix} 1 & 0 & 0 \\ 0.1 & 1 & 1.5 \end{bmatrix}, [A_2\ B_2] = \begin{bmatrix} 1 & 0 & 1.5 \\ 0.5 & 1.5 & 1 \end{bmatrix}, [A_3\ B_3] = \begin{bmatrix} 1.5 & 0 & 1 \\ 3.8 & 1 & 1 \end{bmatrix},$$

in the presence of constraints on the control variable and the output variable:

$$-5 \leq u_k \leq 5, -5 \leq y_k \leq 5,$$

with the nominal model chosen to synthesize a PWA control law:

$$\begin{aligned} A &= 0.3A_1 + 0.2A_2 + 0.5A_3 = \begin{bmatrix} 1.25 & 0 \\ 2.03 & 1.1 \end{bmatrix}, \\ B &= 0.3B_1 + 0.2B_2 + 0.5B_3 = \begin{bmatrix} 0.8 \\ 1.15 \end{bmatrix}, \ C = [1\ 0]. \end{aligned}$$

A continuous PWA control law is designed with prediction horizon 2, weighting matrices $Q = I_2, R = 1$ and the terminal constraints chosen as the maximal output admissible set Gilbert and Tan [1991]. The state space partition is presented in Figure 3.1, whereas its associated PWA control law is shown in Figure 3.2.

Figure 3.3 shows the image of $\Psi^{\alpha}_{\text{rob}}$ obtained from the vertex representation, via the orthogonal projection on the plane $[\alpha_1\ \alpha_2]^T$. Also, Figure 3.4 depicts $\text{Proj}_{[\alpha_1\ \alpha_2]^T}\Psi^{\alpha}_{\text{rob}}$ obtained from the halfspace representation. Note that the shaded violet region represents the whole region of α_1, α_2. The blue point denotes the considered nominal system, this point coincides with a vertex of this robustness margin set. This numerical example verifies that $\Psi^{\alpha}_{\text{rob}}$ obtained from the vertex and halfspace representations are identical. Further, it is observed that this robustness margin differs from the classical notion, because the given control law cannot guarantee the positive invariance of the feasible region $\mathcal{X}$ if the nominal system is perturbed away from the robustness margin.

3.3.5 Further results on the robustness margin

The convexity of the robustness margin was proved in Theorem 3.3.1. As a consequence of Theorems 3.3.3/3.3.4 which characterize in a dual manner the robustness margin, further structural properties can be formally stated.

Corollary 3.3.6 *The robustness margin Ψ_{rob} is polytopic.*

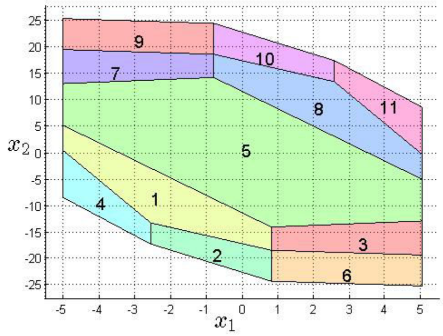


Figure 3.1: State space partition.

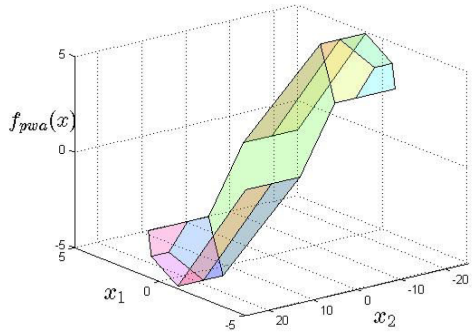


Figure 3.2: Associated PWA controller.

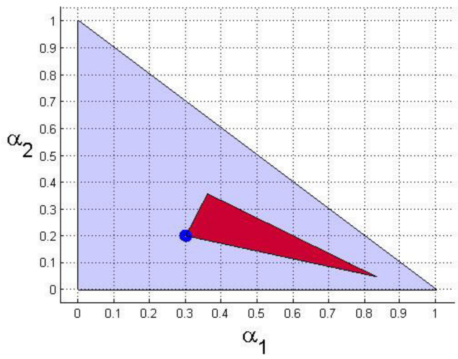


Figure 3.3: Robustness margin in the plane of α_1, α_2 obtained from the vertex representation.

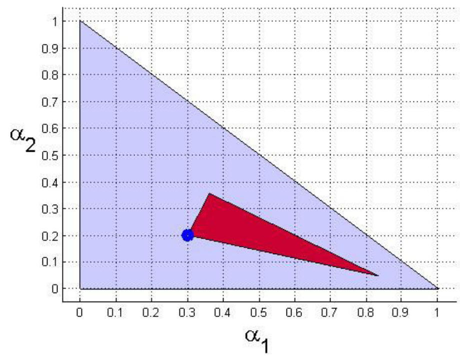


Figure 3.4: Robustness margin in the plane of α_1, α_2 obtained from the halfspace representation.

Proof: Due to the boundedness of $\mathcal{S}_L$, the robustness margin Ψ_{rob} is bounded. Also, Υ_v as defined in (3.14) or Υ_h in (3.21) is described by linear constraints, therefore Υ_v, Υ_h represent polyhedra. The robustness margin, obtained from the orthogonal projection of these polyhedra, also represents a polyhedron. In conclusion, the robustness margin is a bounded polyhedron, meaning a polytope according to the definitions in Section 2.2. The proof is complete. $\square$

Recall that Theorem 3.3.3 is stated under Assumptions 3.2.4, 3.2.5, 3.2.6, but its formulation can be relaxed in case additional assumption regarding the inclusion of the origin in the interior of $\mathcal{X}$, is considered.

Corollary 3.3.7 *Under the hypotheses of Theorem 3.3.3, if in addition Assumption 3.2.7 holds, then the robustness margin can be obtained as follows:*

$$\Psi_{\text{rob}}^{\alpha} = \text{Proj}_{\mathbb{R}^L} \Upsilon_v^*,$$

where Υ_v^* is defined by:

$$\Upsilon_v^* = \left\{ (\alpha, M) \in \mathcal{S}_L \times \mathbb{R}_+^{q \times p} \mid 1^T M \le 1^T, \ \sum_{j=1}^{L} \alpha_j (A_j W + B_j U) = VM \right\}.$$

Proof: For any $[A(k) \ B(k)] \in \Psi_{\text{rob}}$

$$(A(k) + B(k)H_i)x + B(k)G_i \in \mathcal{X}, \ \forall x \in \mathcal{X}_i, \forall i \in \mathcal{I}_N.$$

There exists a $\beta \in [0, \ 1]$ such that

$$(A(k) + B(k)H_i)x + B(k)G_i \in \beta\mathcal{X}, \ \forall x \in \mathcal{X}_i, \ \forall i \in \mathcal{I}_N.$$

Due to Assumption 3.2.7, $\beta\mathcal{X} \subseteq \mathcal{X}$. Following the same line as in the proof of Theorem 3.3.3, there also exists a matrix $M^* \in \mathbb{R}_+^{q \times p}$ such that:

$$\sum_{j=1}^{L} \alpha_j (A_j W + B_j U) = \beta V M^*, \ 1^T M^* = 1^T.$$

Replacing $M = \beta M^*$ leads to the definition of Υ_v^*. This completes the proof. □

Note also that Corollary 3.3.7 may be of help for further developments of robustness margin while guaranteeing asymptotic stability of the origin. Accordingly, a contractivity condition of $\mathcal{X}$ may be required when appropriate constraints are imposed, whereby $1^T M \le 1^T$ is replaced with $1^T M \le \beta 1^T$ for some $\beta \in [0, \ 1)$.

Also, the robustness margins obtained from Theorem 3.3.3 and Corollary 3.3.7 should be identical in spite of different formulations.

Moreover, the continuity of a PWA control law can be relaxed. Accordingly, if Assumption 3.2.6 is dropped, then the robustness margin for a discontinuous PWA controller does not lose any fundamental property. Recall that we are interested in discontinuous PWA functions defined as in (3.2).

Corollary 3.3.8 *Under the hypotheses of Corrollary 3.3.7, if Assumption 3.2.6 is dropped, then the robustness margin can be obtained as follows:*

$$\Psi_{\text{rob}}^{\alpha} = \text{Proj}_{\mathbb{R}^L} \Upsilon_v^{**},$$

where Υ_h^{**} is defined as:

$$\Upsilon_v^{**} = \Bigg\{ (\alpha, M_1, \ldots, M_N) \in \mathcal{S}_L \times \mathbb{R}_+^{q \times q_1} \times \ldots \mathbb{R}_+^{q \times q_N} \mid$$
$$1^T M_i \le 1^T, \ \sum_{j=1}^{L} \alpha_j (A_j V_i + B_j (H_i V_i + 1_{q_i}^T \otimes G_i)) = V M_i, \forall i \in \mathcal{I}_N \Bigg\}.$$

Proof: The argument follows the same line as the one of Corollary 3.3.7 with a particularity that each vertex of the state space partition (elements of $\mathcal{W}$) may correspond to different values of the given control law. All these values have to be taken into account in computation of the robustness margin. More precisely, for any $[A(k)\ B(k)] \in \Psi_{\text{rob}}$, and an $i \in \mathcal{I}_N$

$$(A(k) + B(k)H_i)x + B(k)G_i \in \mathcal{X},\ \forall x \in \mathcal{X}_i.$$

Then there exists $M_i \in \mathbb{R}_+^{q \times q_i}$ such that

$$\sum_{j=1}^{L} \alpha_j (A_j V_i + B_j (H_i V_i + 1_{q_i}^T \otimes G_i)) = V M_i,\ 1^T M_i \leq 1^T. \tag{3.24}$$

Due to the positive invariance of $\mathcal{X}$, (3.24) holds true for every $i \in \mathcal{I}_N$. The proof is complete. □

3.4 Explicit fragility margin for PWA control laws

The fragility problem aims to find the set of admissible variations on the control law gains such that closed loop stability is still ensured. This has been extensively studied for PID controllers and other classical controllers Dorato [1998], Keel and Bhattacharyya [1997]. Such a set is alternatively called *fragility margin*. The problem formulation for the class of PWA control laws is detailed in the sequel.

3.4.1 Problem formulation

Consider the discrete LTI system (3.3) and a PWA control law (3.1). The fragility problem aims to find the set of admissible variations on the local control law gains H_i, G_i such that the positive invariance of $\mathcal{X}$ is preserved. Due to the PWA structure, the fragility margin of control law for each region of the state space partition can be independently studied.

As introduced before, the closed-loop dynamics represent a PWA system. Let $\delta_{H_i}, \delta_{G_i}$ denote the variation variables for the control law gains of region $\mathcal{X}_i$. Also, Δ_i^u is used to denote the set of admissible variations $\delta_{H_i}, \delta_{G_i}$ such that the positive invariance of $\mathcal{X}$ is preserved. More clearly,

$$x_{k+1} = (A + B(H_i + \delta_{H_i,k}))x_k + B(G_i + \delta_{G_i,k}) \in \mathcal{X}, \tag{3.25}$$

for all $x_k \in \mathcal{X}_i$ and any $\left[\text{vec}^T(\delta_{H_i,k})\ \delta_{G_i,k}^T\right]^T \in \Delta_i^u \subset \mathbb{R}^{d_x d_u + d_u}$.

Similar to the robustness margin case, the study of fragility problem relies on the positive invariance principles Athanasopoulos et al. [2014], Benlaoukli and

Olaru [2009], Benlaoukli et al. [2009], Bitsoris [1988b], Blanchini [1999], Hennet [1995], Tahir and Jaimoukha [2012]. The first characteristic of the fragility margin is stated in the following theorem.

Theorem 3.4.1 *The local fragility margins Δ_i^u, $\forall i \in \mathcal{I}_N$ are convex.*

Proof: The proof is similar to the one of Theorem 3.3.1. □

3.4.2 Construction based on the vertex representation

According to the notations introduced in (3.4)–(3.7), the fragility margin can be obtained, based on the following construction.

Theorem 3.4.2 *Consider the discrete LTI system (3.3) and the PWA control law (3.1) satisfying Assumptions 3.2.4, 3.2.5, 3.2.6. The fragility margin of the controller over region $\mathcal{X}_i$ is obtained from:*

$$\Delta_i^u = \mathrm{Proj}_{(\delta_{H_i},\delta_{G_i})}\Lambda_v, \tag{3.26}$$

where Λ_v is defined as follows:

$$\Lambda_v = \Big\{(\delta_{H_i}, \delta_{G_i}, M_i) \in \mathbb{R}^{d_u \times d_x} \times \mathbb{R}^{d_u} \times \mathbb{R}_+^{q \times q_i} \mid 1^T M_i = 1^T \\ [A\ B]\begin{bmatrix} V_i \\ U_i \end{bmatrix} + B\delta_{H_i} V_i + B\delta_{G_i} 1^T = V M_i \Big\}. \tag{3.27}$$

Proof: Due to the positive invariance of $\mathcal{X}$, the following holds:

$$(A + B(H_i + \delta_{H_i}))x + B(G_i + \delta_{G_i}) \in \mathcal{X},\ \forall x \in \mathcal{X}_i.$$

This can be written in the following form:

$$[A\ B]\begin{bmatrix} x \\ f_{pwa}(x) \end{bmatrix} + B\delta_{H_i} x + B\delta_{G_i} \in \mathcal{X},\ \forall x \in \mathcal{X}_i. \tag{3.28}$$

Inclusion (3.28) holds for every $x \in \mathcal{X}_i$, meaning that it holds for every vertex of $\mathcal{X}_i$; i.e.

$$[A\ B]\begin{bmatrix} w_{il} \\ f_{pwa}(w_{il}) \end{bmatrix} + B\delta_{H_i} w_{il} + B\delta_{G_i} \in \mathcal{X},\ \forall w_{il} \in \mathcal{V}(\mathcal{X}_i). \tag{3.29}$$

Then for each $w_{il} \in \mathcal{V}(\mathcal{X}_i)$, there exists $y_{il} \in \mathcal{X}$ such that

$$[A\ B]\begin{bmatrix} w_{il} \\ f_{pwa}(w_{il}) \end{bmatrix} + B\delta_{H_i} w_{il} + B\delta_{G_i} = y_{il}.$$

Such a $y_{il} \in \mathcal{X}$ can be described by a convex combination of the vertices of $\mathcal{X}$; i.e.

$$\exists \gamma_{il} \in \mathcal{S}_q, \quad \text{such that} \;\; y_{il} = V\gamma_{il}, \;\; 1^T\gamma_{il} = 1. \tag{3.30}$$

(3.29) and (3.30) lead to:

$$[A \;\, B] \begin{bmatrix} w_{il} \\ f_{pwa}(w_{il}) \end{bmatrix} + B\delta_{H_i} w_{il} + B\delta_{G_i} = V\gamma_{il}. \tag{3.31}$$

Note also that (3.31) holds true for every $w_{il} \in \mathcal{V}(\mathcal{X}_i)$, leading to the definition of Λ_v as in (3.27), where matrix M_i is composed of the columns as γ_{il} for all $l \in \mathcal{I}_{q_i}$. The proof is complete. □

Remark 3.4.3 It is also important to emphasize that if the description of the state space partition $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ is perturbed, the description of Λ_v will no longer be linear, thus not representing a polyhedron. Instead, the fragility margin of state space partition will be independently studied later in Section 3.6.

As a consequence, the following corollary presents an important property of the fragility margin Δ_i^u for the control law over region $\mathcal{X}_i$.

Corollary 3.4.4 *The local fragility margin Δ_i^u is a polyhedron.*

Proof: The proof follows the same arguments of the one for Corollary 3.3.6. □

Corollary 3.4.5 *Under the hypotheses of Theorem 3.4.2, if in addition Assumption 3.2.7 holds, then the fragility margin of the controller over region $\mathcal{X}_i$ is obtained as:*

$$\Delta_i^u = \text{Proj}_{(\delta_{H_i}, \delta_{G_i})} \Lambda_v^*,$$

where Λ_v^* is defined as follows:

$$\Lambda_v^* = \Bigg\{ (\delta_{H_i}, \delta_{G_i}, M_i) \in \mathbb{R}^{d_u \times d_x} \times \mathbb{R}^{d_u} \times \mathbb{R}_+^{q \times q_i} \mid 1^T M_i \leq 1^T, \\ [A \;\, B] \begin{bmatrix} V_i \\ U_i \end{bmatrix} + B\delta_{H_i} V_i + B\delta_{G_i} 1^T = V M_i \Bigg\}.$$

Proof: The proof follows the same line as Corollary 3.3.7. □

Remark 3.4.6 Similar to the robustness margin, in spite of a relaxation of Corollary 3.4.5 over Theorem 3.4.2, the margins obtained from these two results are equivalent.

Remark 3.4.7 The fragility margin can be of help in the context of explicit MPC design under finite precision arithmetic as discussed in Suardi et al. [2014].

3.4.3 Construction based on the halfspace representation

Based on the halfspace representation of a polyhedron, the fragility margin can also be obtained from the following dual result.

Theorem 3.4.8 *Consider the LTI system* (3.3) *and the PWA control law* (3.1) *satisfying Assumptions 3.2.4, 3.2.5, 3.2.6. The fragility margin of the controller over region $\mathcal{X}_i$ is obtained from:*

$$\Delta_i^u = \mathrm{Proj}_{(\delta_{H_i}, \delta_{G_i})} \Lambda_h$$

where Λ_h is defined as:

$$\begin{aligned} \Lambda_h = \Big\{ &(\delta_{H_i}, \delta_{G_i}, M_i) \in \mathbb{R}^{d_u \times d_x} \times \mathbb{R}^{d_u} \times \mathbb{R}_+^{r \times r_i} \mid \\ & F(A + B(H_i + \delta_{H_i})) = M_i F_i, \; M_i h_i \le h - FB(G_i + \delta_{G_i}) \Big\}. \end{aligned} \tag{3.32}$$

Proof: For an $i \in \mathcal{I}_N$, the positive invariance of $\mathcal{X}$ leads to:

$$(A + B(H_i + \delta_{H_i}))x + B(G_i + \delta_{G_i}) \in \mathcal{X}, \; \forall x \in \mathcal{X}_i.$$

This inclusion can also be written in the following form:

$$\{x \in \mathcal{X} \mid F_i x \le h_i\} \subseteq \{x \in \mathcal{X} \mid F(A + B(H_i + \delta_{H_i}))x \le h - FB(G_i + \delta_{G_i})\}.$$

Using the Extended Farkas Lemma (in Section 2.4.4), the above inclusion holds at the price of the existence of a matrix $M_i \in \mathbb{R}^{r \times r_i}$ with the non-negative elements such that:

$$\begin{aligned} & F(A + B(H_i + \delta_{H_i})) = M_i F_i, \\ & M_i h_i \le h - FB(G_i + \delta_{G_i}). \end{aligned}$$

These complete the proof. □

3.4.4 Illustrative example

Again, consider the numerical example in Subsection 3.3.4. Region 6 has the halfspace representation and its corresponding controller as follows:

$$\begin{aligned} F_6 &= \begin{bmatrix} -1 & 1 & -0.2073 & 0.2073 \\ 0 & 0 & -0.9783 & 0.9783 \end{bmatrix}^T, \\ h_6 &= \begin{bmatrix} -0.8 & 5 & 23.6177 & -17.9116 \end{bmatrix}^T, \\ u(x) &= \begin{bmatrix} -1.5625 & 0 \end{bmatrix} x + 6.25. \end{aligned}$$

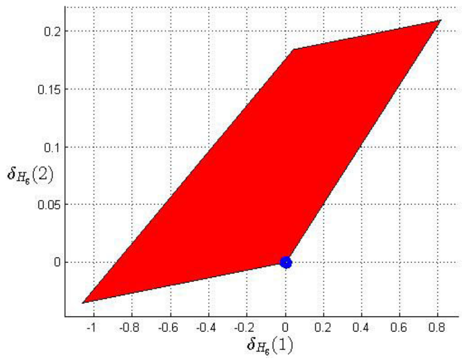


Figure 3.5: Fragility margin of the controller in region $\mathcal{X}_6$ obtained from the halfspace representation.

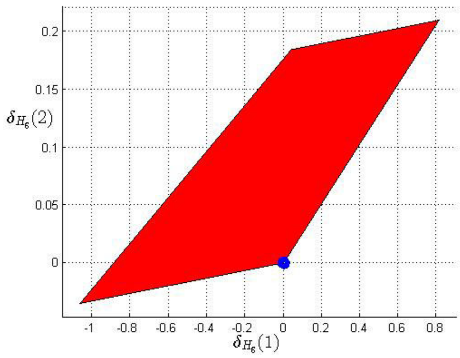


Figure 3.6: Fragility margin of the controller in region $\mathcal{X}_6$ obtained from the vertex representation.

The fragility margin for the control law of region $\mathcal{X}_6$ is illustrated in Figures 3.5 and 3.6. These numerical results prove that the fragility margins obtained from two dual approaches are theoretically identical. It can be seen that the slope gain H_6 without parametric error of the control law associated with this region is pointed out at point $(0, 0)$ in blue which is a vertex of the fragility margin set. It is easy to see that this control law is fragile since if the control law gain H_6 is perturbed away from the fragility set, then closed loop stability may be lost.

3.5 Robustness margin with respect to additive disturbances

3.5.1 Problem formulation

This section focuses on the effect of additive disturbances on the stability of a nominal system, controlled by a PWA control law. Again, consider the discrete LTI system (3.3), and the PWA control law (3.1) satisfying Assumptions 3.2.4, 3.2.5, 3.2.6. Suppose there is no numerical error on the given PWA control law gains. The goal is to find the set of additive disturbances denoted by $\Delta^w \subset \mathbb{R}^{d_x}$ such that the positive invariance of $\mathcal{X}$ is preserved. More precisely, the closed loop dynamics are kept inside $\mathcal{X}$ despite any additive disturbances $w \in \Delta^w$:

$$Ax + B(H_i x + G_i) + w \in \mathcal{X}, \quad \forall x \in \mathcal{X}_i, \ \forall i \in \mathcal{I}_N, \ \forall w \in \Delta^w. \tag{3.33}$$

3.5.2 Construction based on the vertex representation

The robustness margin Δ^w can be computed via the following theorem, using the notations presented in (3.4)–(3.7).

Theorem 3.5.1 *Given an LTI system* (3.3) *and a PWA control law* (3.1) *satisfying Assumptions 3.2.4, 3.2.5, 3.2.6, the robustness margin Δ^w can be obtained as:*

$$\Delta^w = \left\{ w \in \mathbb{R}^n \middle| (1 \otimes F)w \leq (1 \otimes h) - (I \otimes F \begin{bmatrix} A & B \end{bmatrix})\text{vec}\left(\begin{bmatrix} W \\ U \end{bmatrix} \right) \right\}, \tag{3.34}$$

where $1 \in \mathbb{R}^p, I \in \mathbb{R}^{p \times p}$ *and* $p = \text{Card}(\mathcal{W})$.

Proof: Due to the positive invariance of $\mathcal{X}$, inclusion (3.33) holds true for any $w_{il} \in \mathcal{V}(\mathcal{X}_i)$, meaning

$$\begin{bmatrix} A & B \end{bmatrix} \begin{bmatrix} w_{il} \\ f_{pwa}(w_{il}) \end{bmatrix} + w \in \mathcal{X}, \quad \forall w_{il} \in \mathcal{V}(\mathcal{X}_i), \ \forall w \in \Delta^w. \tag{3.35}$$

In other words, with the halfspace representation of $\mathcal{X}$, inclusion (3.35) can be written in the following form:

$$F \begin{bmatrix} A & B \end{bmatrix} \begin{bmatrix} w_{il} \\ f_{pwa}(w_{il}) \end{bmatrix} + Fw \leq h, \quad \forall w_{il} \in \mathcal{V}(\mathcal{X}_i), \ \forall w \in \Delta^w. \tag{3.36}$$

Clearly, inclusion (3.36) holds for every region $\mathcal{X}_i$ in the given state space partition. This end leads to formulation (3.34). □

3.5.3 Construction based on the halfspace representation

Similar to the above sections, the set of additive disturbances satisfying (3.33), can also be determined via the halfspace representation of $\mathcal{X}_i$ by using the Extended Farkas Lemma.

Theorem 3.5.2 *Given an LTI system* (3.3) *and a PWA control law* (3.1) *satisfying Assumptions 3.2.4, 3.2.5, 3.2.6, Δ^w can be obtained as:*

$$\Delta^w = \text{Proj}_{\mathbb{R}^{d_x}} \Theta, \tag{3.37}$$

where Θ is defined as follows:

$$\begin{aligned} \Theta = \Big\{ & (w, M_1, \ldots, M_N) \in \mathbb{R}^{d_x} \times \mathbb{R}_+^{r \times r_1} \times \ldots \times \mathbb{R}_+^{r \times r_N} \Big| \\ & M_i F_i = F(A + BH_i), \ M_i h_i + Fw \leq h - FBG_i, \ \forall i \in \mathcal{I}_N \Big\}. \end{aligned} \tag{3.38}$$

Proof: Relation (3.33) holds for every $x \in \mathcal{X}_i$, leading to:

$$\{x \in \mathcal{X} \mid F_i x \leq h_i\} \subseteq \{x \in \mathcal{X} \mid F(A + BH_i)x + FBG_i + Fw \leq h\}. \quad (3.39)$$

Exploiting the Extended Farkas Lemma, (3.39) holds if there exists a matrix M_i composed of non-negative elements such that:

$$F(A + BH_i) = M_i F_i, \; M_i h_i \leq h - Fw - FBG_i. \quad (3.40)$$

Further, inclusion (3.40) holds true for every $i \in \mathcal{I}_N$, leading to the definition of Θ as in (3.38). Note that Θ represents a parameterized set of w and the elements of M_i, $\forall i \in \mathcal{I}_N$, therefore the set of admissible disturbances satisfying (3.33) can be obtained from the orthogonal projection of Θ onto the space of w. □

Remark 3.5.3 Note that the study of robustness (fragility) margin considering simultaneously model uncertainties (numerical errors on the control law gains) and additive disturbances can also be conducted, following the same constructions presented above. Clearly, in the presence of additive disturbances, the robustness/fragility margin is smaller than the one without such perturbations.

3.5.4 Illustrative example

The nominal system and the control law designed in Subsection 3.3.4 are again considered to illustrate in an *explicit* form the set of admissible additive disturbances. This set is depicted in Figure 3.7 and can be noticed to be lower-dimensional. This is due to the fact that with respect to the positive invariance of $\mathcal{X}$, admissible additive disturbances can only perturb along the boundary of $\mathcal{X}$. Also, due to the constraints on the output $-5 \leq [1 \;\, 0]\, x \leq 5$, there is no freedom on w_1 to ensure the satisfaction of constraints, whereas the feasible set $\mathcal{X}$ is not contractive.

3.6 Fragility margin of state space partition

This section concentrates on the so-called *explicit fragility of the state space partition* problem stemming from implementation of PWA control laws. Namely, it aims to find the set of tolerable errors for the description of the given polytopic partition $\{\mathcal{X}_i\}_{i \in \mathcal{I}_N}$, provided the positive invariance property of $\mathcal{X}$ is not lost.

Note that if the halfspace representation of each region in the given polytopic partition is considered, then the fragility margin is no longer described by linear constraints. This is principally related to the sensitivity of the halfspace representation. Instead, this computation can be carried out via the vertex representation, whereby the errors on the halfspace description can be implicitly deduced.

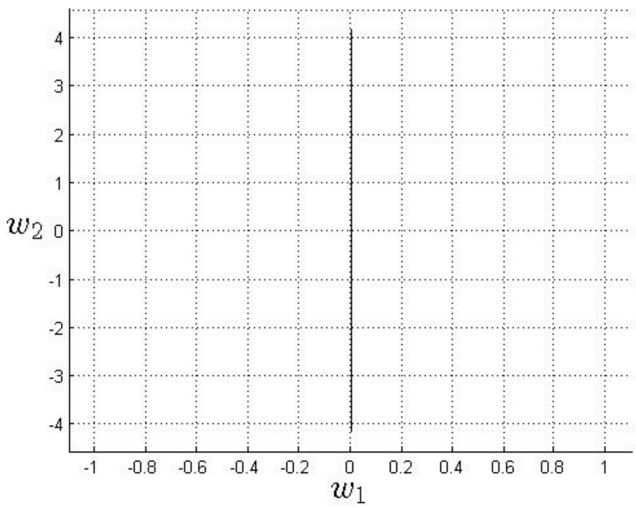


Figure 3.7: The set of admissible disturbances Δ^w.

Consider the LTI system (3.3) and the PWA control law (3.1) satisfying Assumptions 3.2.4, 3.2.5, 3.2.6 associated with the polytopic partition $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$. Suppose the PWA control law gains are fixed. Consider the vertex representation of region $\mathcal{X}_i$ as in (3.5). Its representation in the presence of description errors, introducing a perturbed set $\widetilde{\mathcal{X}}_i$, can be represented as follows:

$$\widetilde{\mathcal{X}}_i = \mathrm{conv}\left\{w_{i1} + \delta_{i1}, \ldots, w_{iq_i} + \delta_{iq_i}\right\}.$$

A solution to the fragility margin of region $\mathcal{X}_i$ will be expressed in terms of δ_{il}, $\forall l \in \mathcal{I}_{q_i}$. The validity of this solution is associated with the following assumption.

Assumption 3.6.1 The polytope $\mathcal{X} = \bigcup_{i\in\mathcal{I}_N} \mathcal{X}_i$ is not subject to uncertainties; i.e. $\mathcal{X} = \bigcup_{i\in\mathcal{I}_N} \widetilde{\mathcal{X}}_i$.

This assumption ensures that the positive invariance can be stated and analyzed in terms of an explicit inclusion:

$$(A + BH_i)x + BG_i \in \mathcal{X},\ \forall x \in \widetilde{\mathcal{X}}_i \subseteq \mathcal{X}, \tag{3.41}$$

with the right-hand side presented by $\mathcal{X}$, free of variations. The set of admissible errors $\delta_i = \left[\delta_{i1}^T \ldots \delta_{iq_i}^T\right]^T \in \mathbb{R}^{d_x q_i}$ denoted by Δ_i^v, can be computed via the following theorem.

Theorem 3.6.2 *Consider the LTI system* (3.3) *and the PWA control law* (3.1) *satisfying Assumptions 3.2.4, 3.2.5, 3.2.6 and 3.6.1. The fragility margin of the vertex* *representation of each region $\mathcal{X}_i$, in the polytopic partition $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ can be obtained as follows:*

$$\Delta_i^v = \left\{ \delta_i \in \mathbb{R}^{d_x q_i} \mid \begin{bmatrix} I \otimes F \\ I \otimes F(A + BH_i) \end{bmatrix} \delta_i \right. \\ \left. \leq \begin{bmatrix} 1 \otimes h - (I \otimes F)\mathrm{vec}(V_i) \\ 1 \otimes h - (I \otimes F \begin{bmatrix} A & B \end{bmatrix})\mathrm{vec}\left(\begin{bmatrix} V_i \\ U_i \end{bmatrix}\right) \end{bmatrix} \right\}, \tag{3.42}$$

where $1 \in \mathbb{R}^{q_i}$ *and* $I \in \mathbb{R}^{q_i \times q_i}$.

Proof: Due to Assumption 3.6.1, any $x \in \widetilde{\mathcal{X}}_i \subseteq \mathcal{X}$ can be expressed in the form of a convex combination of the vertices of $\widetilde{\mathcal{X}}_i$, i.e. there exists $\gamma_i = [\gamma_{i1} \ldots \gamma_{iq_i}]^T \in \mathcal{S}_{q_i}$ such that $x = \sum_{l=1}^{q_i} \gamma_{il}(w_{il} + \delta_{il}) \in \mathcal{X}$. It follows that $F(w_{il} + \delta_{il}) \leq h, \ \forall l \in \mathcal{I}_{q_i}$ leading to:

$$(I \otimes F)\delta_i \leq 1 \otimes h - (I \otimes F)\mathrm{vec}(V_i). \tag{3.43}$$

Furthermore, (3.41) holds true for every $x \in \widetilde{\mathcal{X}}_i$, then so does it for the vertices of $\widetilde{\mathcal{X}}_i$. It follows that

$$(A + BH_i)(w_{il} + \delta_{il}) + BG_i \in \mathcal{X}, \ \forall l \in \mathcal{I}_{q_i}. \tag{3.44}$$

In other words, inclusion (3.44) can be written in the following form:

$$F(A + BH_i)\delta_{il} \leq h - F \begin{bmatrix} A & B \end{bmatrix} \begin{bmatrix} w_{il} \\ f_{pwa}(w_{il}) \end{bmatrix}. \tag{3.45}$$

Recall that (3.45) holds true for every $l \in \mathcal{I}_{q_i}$, leading to:

$$(I \otimes F(A + BH_i))\delta_i \leq 1 \otimes h - (I \otimes F \begin{bmatrix} A & B \end{bmatrix})\mathrm{vec}\left(\begin{bmatrix} V_i \\ U_i \end{bmatrix}\right). \tag{3.46}$$

Inclusions (3.43) and (3.46) complete the proof. □

From the above result, the following set:

$$\widehat{\mathcal{X}}_i = \mathrm{conv}\left\{ \bigcup_{l\in\mathcal{I}_{q_i}} w_{il} \oplus \mathrm{Proj}_{\delta_{il}} \Delta_i^v \right\}, \tag{3.47}$$

represents the maximal erroneous halfspace representation of $\mathcal{X}_i$. More clearly, if $\widetilde{\mathcal{X}}_i$ stands for the implemented halfspace representation of $\mathcal{X}_i$, then any implemented $\widetilde{\mathcal{X}}_i \subseteq \widehat{\mathcal{X}}_i$ can guarantee the positive invariance of $\mathcal{X}$ with respect to the local PWA control law.

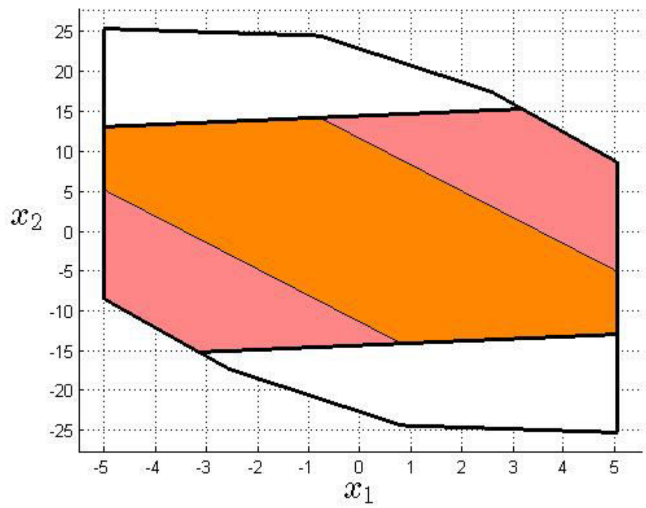


Figure 3.8: The shaded pink region represents $\widehat{\mathcal{X}}_5$, defined in (3.47).

To illustrate this study, the state space partition and the PWA control law designed in Subsection 3.3.4, are reconsidered. The white polytope in Figure 3.8, represents $\mathcal{X}$. For illustration, the unconstrained region $\mathcal{X}_5$, which is the orange polytope, is illustrated. The pink polytope represents $\widehat{\mathcal{X}}_5$, defined in (3.47). It implies that for any implemented representation $\widetilde{\mathcal{X}}_5$ of $\mathcal{X}_5$, satisfying $\widetilde{\mathcal{X}}_5 \subseteq \widehat{\mathcal{X}}_5$, the positive invariance of $\mathcal{X}$ is ensured with respect to the above PWA control law.

3.7 Computational aspects

The above formulations for computation of the robustness and fragility margins are not in the canonical representations (vertex/halfspace representations). Therefore, to explicitly compute these margins, transformations from these matrix equalities/inequalites into canonical representations will be discussed in this section.

3.7.1 Explicit robustness margin of PWA controllers

3.7.1.1 The vertex representation

Consider (3.14) element by element; for $l \in \mathcal{I}_p$:

$$\begin{aligned}\Psi_l^\alpha := \Big\{ \alpha \in \mathcal{S}_L \, | 1^T M(\cdot, l) = 1, \; M(\cdot, l) \in \mathbb{R}_+^q, \\ \sum_{j=1}^{L} \alpha_j (A_j W(\cdot, l) + B_j f_{pwa}(W(\cdot, l))) = V M(\cdot, l) \Big\},\end{aligned} \tag{3.48}$$

then the robustness margin can be computed as:

$$\Psi_{\text{rob}}^\alpha = \bigcap_{l \in \mathcal{I}_p} \Psi_l^\alpha. \tag{3.49}$$

If $\hat{w}_l = \begin{bmatrix} W(\cdot, l) \\ f_{pwa}(W(\cdot, l)) \end{bmatrix}$, then (3.48) can be written in the following form where the variable is $\beta_l = \left[\alpha_1 \ldots \alpha_L \; M^T(\cdot, l)\right]^T \geq 0$

$$\begin{bmatrix} [A_1 \; B_1] \, \hat{w}_l \ldots [A_L \; B_L] \, \hat{w}_l & -V \\ 0_L^T & 1_q^T \\ 1_L^T & 0_q^T \end{bmatrix} \beta_l = \begin{bmatrix} 0_{d_x} \\ 1 \\ 1 \end{bmatrix} \tag{3.50}$$

This system of equations in the form $\mathcal{A}\beta_l = \mathcal{B}$ has a family of solutions:

$$\beta_l = \mathcal{A}_s t + \mathcal{B}_s,$$

where $\mathcal{A}_s$ denotes an orthonormal basis for the null space of matrix $\mathcal{A}$ (i.e. $\mathcal{A}\mathcal{A}_s = 0$), $\mathcal{B}_s$ denotes a feasible solution of equation (3.50) and t stands for a vector of appropriate dimension. Due to the non-negativity of all elements in β_l, the admissible set of t denoted by Φ_t is defined as follows: $\Phi_t = \{t \mid -\mathcal{A}_s t \leq \mathcal{B}_s\}$. Note also that

$$\Phi_{\beta_l} := \{\beta_l \mid (3.50) \text{ holds}\} = \mathcal{A}_s \Phi_t \oplus \mathcal{B}_s$$

represents a polytope. Therefore, due to the above relation, Φ_t also represents a polytope. So one only needs to calculate all vertices of Φ_{β_l} by applying the transformation to the vertices of Φ_t. Thus, the set Ψ_l^α of coefficients α for which (3.48) holds, is obtained via the orthogonal projection of Φ_{β_l} onto the space of α : i.e. $\Psi_l^\alpha = \text{Proj}_{\mathbb{R}^L} \Phi_{\beta_l}$. Note that to avoid computing directly the orthogonal projection, it suffices to compute the vertices of Φ_{β_l} and keep the L first coordinates for each vertex. Subsequently, Ψ_l^α is obtained by computing the convex hull of these reduced vertices. Finally, the robustness margin is obtained by inclusion (3.49).

3.7.1.2 The halfspace representation

From the definition of Υ_h in (3.21), if one defines the following set:

$$\Upsilon_i = \Bigg\{ (\alpha, M_i) \in \mathcal{S}_L \times \mathbb{R}_+^{r\times r_i} \Bigg| \sum_{j=1}^{L} \alpha_j F(A_j + B_j H_i) = M_i F_i,\ M_i h_i \leq h - F \sum_{j=1}^{L} \alpha_j B_j G_i \Bigg\}, \tag{3.51}$$

then the robustness margin can be obtained from the following inclusion

$$\Psi_{\text{rob}}^{\alpha} = \bigcap_{i\in\mathcal{I}_N} \text{Proj}_{\mathbb{R}^L} \Upsilon_i. \tag{3.52}$$

To facilitate the computation, it is necessary to transform Υ_i into a canonical representation. Indeed, the equation in (3.51) can be decoupled row by row as follows:

$$\begin{aligned} M_i(k,\cdot)F_i &= [\alpha_1 \dots \alpha_{L-1}]\, Z_k + F(k,\cdot)(A_L + B_L H_i),\ \forall k \in \mathcal{I}_r, \\ Z_k &= \begin{bmatrix} F(k,\cdot)(A_1 - A_L + B_1 H_i - B_L H_i) \\ \vdots \\ F(k,\cdot)(A_{L-1} - A_L + B_{L-1} H_i - B_L H_i) \end{bmatrix}. \end{aligned} \tag{3.53}$$

Denote the following vector: $z = \left[\text{vec}^T(M_i^T)\ \alpha_1 \dots \alpha_{L-1}\right]^T$, then (3.53) can be written in the following form:

$$\begin{aligned} D_1 z = E_1,\ D_1 &= \begin{bmatrix} F_i & \dots & 0_{r_i\times d_x} \\ \vdots & \ddots & \vdots \\ 0_{r_i\times d_x} & \dots & F_i \\ -Z_1 & \dots & -Z_r \end{bmatrix}^T, \\ E_1 &= (I_r \otimes (A_L + B_L H_i)^T)\text{vec}(F_i^T). \end{aligned} \tag{3.54}$$

Similarly, the inequality in (3.51) can be expressed in the form:

$$D_2 z \leq E_2,\ D_2 = \begin{bmatrix} h_i & \dots & 0_{r_i} \\ \vdots & \ddots & \vdots \\ 0_{r_i} & \dots & h_i \\ Y_1 & \dots & Y_r \end{bmatrix}^T,\ E_2 = h - F B_L G_i, \tag{3.55}$$

where $Y_k = \left[F(k,\cdot)(B_1 - B_L)G_i \dots F(k,\cdot)(B_{L-1} - B_L)G_i\right]^T,\ \forall k \in \mathcal{I}_r.$

It is known that the solution to (3.54) is a set of z which depends on t such that $z = \mathcal{D}_1 t + \mathcal{E}_1$, where $\mathcal{D}_1$ represents an orthonormal basis for the null space of D_1 and $\mathcal{E}_1$ denotes a feasible solution to (3.54). Due to the non-negativity of z, the values of t satisfy $-\mathcal{D}_1 t \leq \mathcal{E}_1$. Accordingly, from (3.55), the set of t denoted by Φ_t, can be described by:

$$\Phi_t = \{t \mid -\mathcal{D}_1 t \leq \mathcal{E}_1,\ D_2\mathcal{D}_1 t \leq E_2 - D_2\mathcal{E}_1\}.$$

Recall that the set of z denoted by Φ_z, is an affine transformation of Φ_t; i.e. $\Phi_z = \mathcal{D}_1\Phi_t \oplus \mathcal{E}_1$. Finally, $\mathrm{Proj}_{\mathbb{R}^L}\Upsilon_i$ can be derived from $\mathrm{Proj}_{\mathbb{R}^{L-1}}\Phi_z$.

3.7.2 Explicit fragility margin of PWA controllers

For simplicity, without loss of generality, variations in H_i are exclusively considered.

3.7.2.1 The vertex representation

Based on the computation of fragility margin in (3.26) and (3.27), it is easier to consider Λ_v column by column. Indeed, define the following set:

$$\begin{aligned}\Lambda_l = \Big\{&\delta_{H_i} \in \mathbb{R}^{d_u \times d_x} \mid 1^T M_i(\cdot,l) = 1,\ M_i(\cdot,l) \in \mathbb{R}^q_+,\\ &\begin{bmatrix} A & B\end{bmatrix}\begin{bmatrix} V_i(\cdot,l)\\ U_i(\cdot,l)\end{bmatrix} + B\delta_{H_i}V_i(\cdot,l) = VM_i(\cdot,l)\Big\},\ l \in \mathcal{I}_{q_i}.\end{aligned} \tag{3.56}$$

Then the fragility margin can be obtained as:

$$\Delta_i^u = \bigcap_{l\in\mathcal{I}_{q_i}} \Lambda_l.$$

Now, it is necessary to transform the constraints of Λ_l into a canonical representation. Denote $\hat{w}_{il} = \left[V_i^T(\cdot,l)\ U_i^T(\cdot,l)\right]^T$, then (3.56) can be equivalently written as a set of linear equations whose variable is denoted by β_{il} and defined as $\beta_{il} = \left[\mathrm{vec}^T(\delta_{H_i})\ M_i^T(1:q-1,l)\right]^T \in \mathbb{R}^{d_u d_x + q - 1}$, i.e.

$$\begin{bmatrix} V_i^T(\cdot,l)(I_{d_x}\otimes B(1,\cdot)) & \\ \vdots & -\widetilde{V} \\ V_i^T(\cdot,l)(I_{d_x}\otimes B(d_x,\cdot)) & \end{bmatrix}\beta_{il} = v_q - [A\ B]\,\hat{w}_{il}, \tag{3.57}$$

where $\widetilde{V} = [v_1 - v_q \ldots v_{q-1} - v_q]$ with respect to the notation in (3.5).

Equation (3.57) in the form $\mathcal{A}\beta_{il} = \mathcal{B}$, has a family of solution $\beta_{il} = \mathcal{A}_s t + \mathcal{B}_s$ where $\mathcal{A}_s$ denotes an orthonormal basis for the null space of $\mathcal{A}$, $\mathcal{B}_s$ denotes a feasible solution of (3.57), t denotes a vector of appropriate dimension. Moreover, due to the non-negativity of $\beta_{il}(1 + d_x d_u : d_x d_u + q - 1) = M_i(1 : q - 1, l)$, the values of t satisfy: $-\mathcal{A}_s^{(2)} t \leq \mathcal{B}_s^{(2)}$, where the matrices of interest are defined as follows:

$$\begin{aligned}\left[\mathcal{A}_s^{(1)}\ \mathcal{B}_s^{(1)}\right] &= \left[\mathcal{A}_s\ \mathcal{B}_s\right](1 : d_x d_u, \cdot),\\ \left[\mathcal{A}_s^{(2)}\ \mathcal{B}_s^{(2)}\right] &= \left[\mathcal{A}_s\ \mathcal{B}_s\right](1 + d_x d_u : d_x d_u + q - 1, \cdot).\end{aligned}$$

Recall that $1^T M_i(1 : q - 1, l) \leq 1$. Thus, the set of admissible t denoted by Φ_t, can be described by: $\Phi_t = \left\{t \mid -\mathcal{A}_s^{(2)} t \leq \mathcal{B}_s^{(2)},\ 1^T \mathcal{A}_s^{(2)} t \leq 1 - 1^T \mathcal{B}_s^{(2)}\right\}$. The set of admissible β_{il} denoted by $\Phi_{\beta_{il}}$ is computed by

$$\Phi_{\beta_{il}} = \{\beta_{il} \mid (3.57) \text{ holds}\} = \mathcal{A}_s \Phi_t \oplus \mathcal{B}_s,$$

representing a polyhedron. Further, due to the boundedness of $\beta_{il}(1 + d_x d_u : d_x d_u + q - 1)$, Φ_t is a polytope, thus so is $\Lambda_l = \mathcal{A}_s^{(1)} \Phi_t \oplus \mathcal{B}_s^{(1)}$. Repeat the same computation for all $l \in \mathcal{I}_{q_i}$ leading to the fragility margin Δ_i^u.

3.7.2.2 The halfspace representation

Recall that variations in H_i are exclusively considered. Therefore, Λ_h in (3.32) represents a polyhedron with the variable denoted by z and defined by $z = \left[z_1^T\ z_2^T\right]^T$ $z_1 = \text{vec}(M_i^T)$, $z_2 = \text{vec}(\delta_{H_i})$, i.e.

$$\begin{aligned}\Lambda_h = \Big\{&(\delta_{H_i}, M_i) \in \mathbb{R}^{d_u \times d_x} \times \mathbb{R}_+^{r \times r_i} \mid \\ &F(A + B(H_i + \delta_{H_i})) = M_i F_i,\ M_i h_i \leq h - FBG_i\Big\}.\end{aligned} \tag{3.58}$$

Considering row by row the equality in (3.58) leads to:

$$M_i(k, \cdot) F_i = F(k, \cdot)(A + B(H_i + \delta_{H_i})),\ \forall k \in \mathcal{I}_r.$$

Then, the equality in (3.58) can be equivalently written in the form of z as follows

$$\begin{aligned}&D_1 z = E_1,\ E_1 = (I_r \otimes (A + BH_i)^T)\text{vec}(F^T),\\ &D_1 = \begin{bmatrix} F_i & \dots & 0_{r_i \times d_x} \\ \vdots & \ddots & \vdots \\ 0_{r_i \times d_x} & \dots & F_i \\ Z_1 & \dots & Z_r \end{bmatrix}^T,\ Z_k = I_{d_x} \otimes (-B^T F^T(k, \cdot)),\ \forall k \in \mathcal{I}_r.\end{aligned} \tag{3.59}$$

Similarly, the inequality in (3.58) can be written as follows:

$$D_2 z_1 \leq E_2,\ D_2 = \begin{bmatrix} h_i & \dots & 0_{r_i} \\ \vdots & \ddots & \vdots \\ 0_{r_i} & \dots & h_i \end{bmatrix}^T,\ E_2 = h - FBG_i. \tag{3.60}$$

The family of solutions to the set of equations in (3.59) is in the form $z = \mathcal{A}_s t + \mathcal{B}_s$, where $\mathcal{A}_s$ denotes an orthonormal basis for the null space of D_1, $\mathcal{B}_s$ denotes a feasible solution to (3.59). For simplicity, the following matrices are defined:

$$\mathcal{A}_s^{(1)} = \mathcal{A}_s(1 : rr_i, \cdot),\ \mathcal{A}_s^{(2)} = \mathcal{A}_s(rr_i + 1 : rr_i + d_x d_u, \cdot),$$
$$\mathcal{B}_s^{(1)} = \mathcal{B}_s(1 : rr_i, \cdot),\ \mathcal{B}_s^{(2)} = \mathcal{B}_s(rr_i + 1 : rr_i + d_x d_u, \cdot).$$

Due to the non-negativity of z_1 and (3.60), the set of t denoted by Φ_t can be computed as:

$$\Phi_t = \left\{ t \mid -\mathcal{A}_s^{(1)} t \leq \mathcal{B}_s^{(1)},\ D_2 \mathcal{A}_s^{(1)} t \leq E_2 - D_2 \mathcal{B}_s^{(1)} \right\}.$$

As a consequence, the fragility margin Δ_i^u can be obtained as $\Delta_i^u = \mathcal{A}_s^{(2)} \Phi_t \oplus \mathcal{B}_s^{(2)}$.

3.8 Complexity analysis

Due to the similarity between the robustness margin and the fragility margin, the complexity analysis of the robustness margin via two different approaches will be exclusively studied.

3.8.1 The vertex representation

From (3.48) and (3.49), the computation of the robustness margin requires solving (3.50) for all elements of $\mathcal{W}$. Therefore, it is necessary to analyze the complexity for solving (3.50).

Indeed, according to Golub and Van Loan [2012], solving a system of linear equations $Ax = b, x \in \mathbb{R}^n$, has a complexity of $\mathcal{O}(n^2)$. Therefore, solving (3.50) requires a complexity of $\mathcal{O}((q + L)^2)$. Further, computing an orthonormal basis for the null space of a matrix $A \in \mathbb{R}^{m \times n}$ using the singular value decomposition (SVD) method, has a complexity of $\mathcal{O}(n^3)$ according to Chan [1987]. Thus computing an orthonormal basis for the null space of $\mathcal{A}$ in (3.50) has a complexity of $\mathcal{O}((q + L)^3)$.

Also, instead of computing the orthogonal projection via the Fourier-Motzkin elimination, it is less demanding to compute the vertices of Ψ_l^α via an affine transformation of those of Φ_t. Recall that the algorithm put forward in Avis and Fukuda

[1992], carries out the vertex enumeration problem for a polytope in $\mathcal{O}(ndv)$, where v denotes the number of its vertices, d denotes its dimension and n denotes the number of its supporting halfspaces. Dually, the facet enumeration problem has a complexity of $\mathcal{O}(ndv)$, where d still denotes the dimension of this polytope, n denotes the number of points, v denotes the number of its facets. Back to the solutions to (3.50), the vertex enumeration problem for $\Phi_t \in \mathbb{R}^{q+L-d_x-2}$ has a complexity of $\mathcal{O}((q+L)(q+L-d_x-2)v)$ where v stands for the number of vertices of Φ_t. Note that the maximal number of vertices of Φ_t is equal to

$$f_0 = \binom{q+L}{q+L-d_x-2},$$

where $\binom{n}{k}$ denotes the number of $k-$combinations from the set of n elements; i.e. $\binom{n}{k} = \frac{n!}{k!(n-k)!}$. Therefore, the vertex enumeration of Φ_t can be achieved at most in

$$\mathcal{O}\left((q+L)(q+L-d_x-2)f_0\right).$$

The vertices of Ψ_l^α can be obtained by an affine transformation of those of Φ_t. Finally, obtaining Ψ_l^α via the convex hull amounts to solving the facet enumeration problem, as mentioned above, having a complexity of $\mathcal{O}(vLf_{L-1}(L,v))$, where v represents the number of vertices of Φ_t and $f_{L-1}(L,v)$ stands for the number of facets of Ψ_l^α. Recall that Ψ_l^α is in fact an affine transformation of Φ_t. Accordingly, the number of facets of Ψ_l^α i.e. $f_{L-1}(L,v)$ is smaller than the number of facets of Φ_t, known to be equal to $q+L$. It follows that the facet enumeration of Ψ_l^α has a complexity at most equal to $\mathcal{O}(f_0L(q+L))$.

Finally, the computation of Ψ_l^α has a complexity at most equal to:

$$\begin{aligned}&\mathcal{O}((q+L)^2) + \mathcal{O}((q+L)^3) + \mathcal{O}\left((q+L)(q+L-d_x-2)f_0\right) + \mathcal{O}(f_0L(q+L))\\ &= \mathcal{O}\left((q+L)(q+L-d_x-2)f_0\right),\end{aligned}$$

since q always satisfies $q \geq d_x + 1$.

In conclusion, the robustness margin is obtained by the intersection of Ψ_l^α, $\forall l \in \mathcal{I}_p$ as presented in (3.49), thus the total time complexity is equal to:

$$p\mathcal{O}\left((q+L)(q+L-d_x-2)f_0\right).$$

3.8.2 The halfspace representation

Similarly, from (3.54), computing an orthonormal basis for the null space of D_1 has a complexity of $\mathcal{O}((rr_i+L-1)^3)$. Also, finding a feasible solution to (3.54) has a complexity of $\mathcal{O}((rr_i+L-1)^2)$. Note also that

$$\dim(\Phi_t) = rr_i + L - 1 - rd_x = r(r_i - d_x) + L - 1.$$

Further, the number of constraints describing Φ_t is equal to $r(r_i + 1) + L - 1$. Then the maximal number of vertices is equal to

$$f_{0,i} = \binom{r(r_i + 1) + L - 1}{r(r_i - d_x) + L - 1}.$$

Therefore, the vertex enumeration problem of Φ_t has a complexity at most equal to $\mathcal{O}((r(r_i - d_x) + L - 1)(r(r_i + 1) + L - 1)f_{0,i})$.

Similar to the vertex representation case, $\mathrm{Proj}_{\mathbb{R}^L}\Upsilon_i$ is an affine transformation of Φ_t, therefore the number of facets of $\mathrm{Proj}_{\mathbb{R}^L}\Upsilon_i$ is smaller than the number of facets of Φ_t. Accordingly, the facet enumeration of $\mathrm{Proj}_{\mathbb{R}^L}\Upsilon_i$ has a complexity at most equal to

$$\mathcal{O}(vL(r(r_i + 1) + L - 1)).$$

Finally, the computation of $\mathrm{Proj}_{\mathbb{R}^L}\Upsilon_i$ has the biggest complexity of

$$\begin{aligned}
&\mathcal{O}((rr_i + L - 1)^2) + \mathcal{O}((r(r_i - d_x) + L - 1)(r(r_i + 1) + L - 1)f_{0,i}) \\
&\qquad\qquad + \mathcal{O}((rr_i + L - 1)^3) + \mathcal{O}(f_{0,i}L(r(r_i + 1) + L - 1)) \\
&= \mathcal{O}((r(r_i - d_x) + L - 1)(r(r_i + 1) + L - 1)f_{0,i}) + \mathcal{O}(f_{0,i}L(r(r_i + 1) + L - 1)), \\
&= \mathcal{O}((r(r_i - d_x) + L - 1)(r(r_i + 1) + L - 1)f_{0,i}),
\end{aligned}$$

since each region $\mathcal{X}_i$ in the given polytopic partition is a polytope, leading to $r_i \geq d_x + 1$.

In conclusion, the robustness margin via the halfspace representation can be obtained at most in

$$\sum_{i=1}^{N} \mathcal{O}((r(r_i - d_x) + L - 1)(r(r_i + 1) + L - 1)f_{0,i}).$$

3.8.3 Complexity comparison

This subsection aims to compare the computational complexity of the robustness margin via two different approaches. It is already known that the $\mathcal{V}-$representation and $\mathcal{H}-$representation are dual, thus it is difficult to clearly state that one is better than the other.

According to Table 3.1, it can be relatively observed that if the number of vertices of the given polytopic partition satisfies:

$$p(q + L - d_x - 2) < \sum_{i=1}^{N}(r(r_i - d_x) + L - 1),$$

then the approach via the $\mathcal{V}-$representation is less complex than the one via the $\mathcal{H}-$representation. Otherwise, the approach via the $\mathcal{H}-$representation is less computationally demanding than the other.

Approaches	Computational complexity
$\mathcal{V}-$representation	$p\mathcal{O}\left((q+L)(q+L-d_x-2)f_0\right)$
$\mathcal{H}-$representation	$\sum_{i=1}^{N}\mathcal{O}((r(r_i-d_x)+L-1)(r(r_i+1)+L-1)f_{0,i})$

Table 3.1: Comparison of computational complexities for the robustness margin via two different approaches.

For illustration, consider again the numerical example in Subsection 3.3.4. In this example $q = 8$, $p = 22$, $d_x = 2$, $L = 3$, $r = 8$, $N = 11$, $r_i \in \{4, 5, 6\}$, for $i \in \mathcal{I}_{11}$. It can be observed that the approach via the vertex representation is more efficient than the one via the halfspace representation, since the latter one has to work with polyhedral operations for high dimensional sets. This is advocated via computational times, presented in Table 3.2.

Approaches	Computational time [s]
$\mathcal{V}-$representation	0.6550
$\mathcal{H}-$representation	> 3600

Table 3.2: Comparison of computational time for the robustness margin via two different approaches. This simulation is carried on a computer with an Intel Core i5, M430, 2.27 Ghz, Ram 4G. This computer was equipped with a 32 bit version of Windows 7.

Remark 3.8.1 To make the computation more efficient, it is better to avoid working with high dimensional sets by considering smaller polyhedra with appropriate variables and projecting them onto the space of interest. Finally, the robustness margin can be obtained as their intersection. For example, consider (3.54). Instead of constructing a high dimensional polyhedron as in (3.54), one can construct smaller polyhedra with variables $\left[M_i^T(k,\cdot)\ \alpha_1 \ldots, \alpha_{L-1}\right]^T$ for all $k \in \mathcal{I}_r$. Subsequently, to obtain the robustness margin, it suffices to project these polyhedra onto the space of $[\alpha_1 \ldots \alpha_{L-1}]^T$ and compute their intersection.

3.9 Conclusions

A measure of the robustness and fragility margins for the class of PWA controllers has been introduced in this chapter. Two points of view have been presented with respect to the closed-loop dynamics of a linear system with a piecewise affine control law: the robustness with respect to parametric model uncertainties and the fragility of the piecewise affine control function. For both cases it has been shown that the margins are represented by convex sets of admissible parameter variations. Following this idea, extensions to the explicit fragility margin of the state space partition and the robustness margin with respect to additive disturbances have also been tackled. These problems also lead to polyhedral set descriptions. The approach allows one to have a generic vision about the margins related to PWA control laws and also provides new insight in the implementation limitations for this class of controllers. We conclude this chapter with an observation that the robustness margin with respect to parametric model uncertainties is denoted as *matched portion* of model in other texts e.g. Barmish and Leitmann [1982]. The characterization of this robustness margin amounts to finding a *matching condition* for parametric uncertainties to guarantee the positive invariance.

Chapter 4

IPL/QP problems via convex liftings

Contents

This chapter focuses on inverse optimality problem for the piecewise affine (PWA) functions. The results herein have been published in Nguyen et al. [a, 2014c,d, 2015b,c,d,e,g], *Inverse parametric linear/quadratic programming* (IPL/QP) aims to construct an appropriate optimization problem composed of a set of linear constraints and a cost function such that the optimal solution to such a problem is equivalent to the given continuous piecewise affine function defined over a polyhedral partition. This chapter introduces a constructive procedure to find this formulation. The approach is based on the so-called *convex liftings* Rybnikov [1999]. As the first geometrical result, an algorithm to construct convex liftings of a given convexly liftable *cell complex* will be put forward. Moreover, it will be shown that a polyhedral partition or a non-convexly liftable cell complex can be subdivided into a convexly liftable cell complex without changing boundaries between the regions of the partition. Following this idea, an important result will be presented: any continuous piecewise affine function defined over a polyhedral partition is the solution of a parametric linear/quadratic programming problem which can be numerically constructed. Furthermore, this convex liftings based method requires at most one scalar auxiliary variable.

Based on a similar methodology, the extension of IPL/QP problem for the class of *discontinuous piecewise affine functions* will also be studied. It will be shown that any possibly discontinuous PWA function is optimal solution to a parametric convex optimization problem. However, the uniqueness of this optimal solution is no longer guaranteed.

4.1 Introduction

Parametric convex programming (PCP) has attracted significant attention from the control community due to its interesting applications in model predictive control. A parametric convex programming problem, characterized by a set of linear constraints and a linear/quadratic cost function, is called a parametric linear/quadratic programming problem (PL/QP). It is well known that optimal solution of such a linear/quadratic programming problem is a piecewise affine (PWA) function defined over a polyhedral partition of the parameter space. In fact, in control theory, this class of control laws emerged in the last decade as an approximation of the classical nonlinear control laws with respect to a predefined approximation error Grancharova and Johansen [2012], Johansen [2002, 2004]. Then, it was shown that this piecewise affine structure is inherited by the exact optimal solution of a linear MPC problem with respect to a linear/quadratic cost function Bemporad et al. [2002], Feller et al. [2013], Olaru and Dumur [2004], Pistikopoulos et al. [2007], Seron et al. [2003], Tøndel et al. [2003]. Note that piecewise affine control laws may also result from other control design approaches than MPC, an approach based on interpolation for robust PWA control law design can be found in Nguyen [2014].

Inverse parametric convex programming (IPCP) aims to build a parametric convex optimization problem characterized by an appropriate constraint set and a cost function such that its optimal solution contains as a subcomponent the given function. In particular, inverse parametric linear/quadratic programming aims to construct a linear constraint set and a linear/quadratic cost function such that a subcomponent of the optimal solution to their associated optimization problem is equivalent to a given PWA function defined over a given polyhedral partition.

This inverse optimality problem has been investigated for some years and has resulted in interesting results in the general nonlinear continuous functions case Baes et al. [2008], and recently with respect to continuous piecewise affine functions Hempel et al. [2015], Nguyen et al. [2014d]. One of the main reasons for this interest in inverse parametric convex programming problems is related to implementation of piecewise affine control laws Nguyen et al. [2015e]. It is already known that if the state space partition of the given PWA control law contains numerous regions, two major limitations are well recognized:

— its implementation requires substantial computer memory to store the partition and its associated control law gains,

— at each sampling instant, the point-location problem determining to which region the current state belongs, becomes more expensive.

Therefore, reducing the implementation complexity for this class of controllers is necessary. Finding a simpler optimization problem, which generates an equivalent piecewise affine control law, is a promising approach for implicit implementation. Note that some efforts for fast MPC study can be found in Wang and Boyd [2010] where the structure of optimization problem is shown to benefit online solvers. Also, results towards the complexity reduction of PWA control laws can also be found in Kvasnica and Fikar [2012], Kvasnica et al. [2013].

This chapter is closely related to the first results of inverse parametric convex programming presented in Baes et al. [2008]. The authors proved that *every continuous feedback law can be obtained by PCP*. This is an insightful mathematical result. However, it remains pure theoretical; neither a *constructive procedure* nor a qualitative interpretation of the dimension of the optimization arguments is provided in Baes et al. [2008]. The present work is motivated by a comment therein: *A natural question that can arise from this note would be to particularize our results to piecewise linear controllers: can any continuous piecewise linear feedback law be obtained by parametric linear programming? Should such a construction be possible, it might offer computational advantages for explicit MPC algorithms.* The answer is positive, and one solution to such an inverse optimality problem is recently found in Hempel et al. [2015] wherein an indirect solution, built upon a decomposition of a continuous PWA function into the difference of two continuous convex functions, is introduced. It is shown therein that the number of auxiliary variables is at most $2d_u$, where d_u represents the dimension of the given PWA function to be recovered.

Here, stronger results are obtained using a different approach i.e. the convex lifting approach. Furthermore, it will be proved that the proposed method can recover the given PWA function with at most one supplementary scalar variable. The major contributions in this direction are:

1. the introduction of the convex lifting concept for use in the inverse optimality problem,
2. a convex liftability related condition for the existence of a solution for the inverse optimality problem,
3. a constructive procedure based on linear/quadratic programming and polyhedral operations for obtaining such a solution,
4. a partition refining algorithm which produces an equivalent PWA function for those that do not fulfill convex liftability conditions.

The theoretical results prove to have a number of implications in control design. The most important is related to linear MPC and can be stated as follows: *every continuous piecewise affine control law can be recovered via a model predictive control problem with a control horizon at most equal to 2 prediction steps.*

The most important concept used in these developments: the *lifting* can be defined in the geometrical sense as an inverse operation of orthogonal projection. It has been applied in different fields: e.g. mechanics, geometry, signal processing, control, etc. As emphasized by its definition, this operation allows lifting of a given partition onto a higher dimensional space. In particular, a so-called *convex lifting* of a given partition in $\mathbb{R}^d$ amounts to a convex PWA surface in $\mathbb{R}^{d+1}$ such that its image via the orthogonal projection onto $\mathbb{R}^d$ coincides with the given partition. It is worth stressing that such a convex lifting of a partition has each pair of adjacent (neighboring) regions in the given partition lifted onto two distinct hyperplanes, as defined in Rybnikov [1999]. This concept differs from the notion of *convex function* employed in Hempel et al. [2015] which allows two adjacent regions to be lifted onto the same hyperplane.

Note that a polyhedron bounded by such a convex lifting in $(d+1)-$dimensional space is called an *affinely equivalent polyhedron* associated with the given partition. It is worth reminding that the lifting notion was introduced for the first time in Maxwell's research publications e.g. Maxwell [1864] some 150 years ago. Later, a plethora of studies were dedicated for the existence conditions of such a *convex lifting* of a given partition Aurenhammer [1987a,b], Crapo and Whiteley [1993, 1994], Nguyen et al. [2014c], Rybnikov [1999], Schulz [2008]. In fact, the *equilibrium stress* notion was presented in Maxwell [1864] to obtain reciprocal diagrams of a given partition on the plane. Afterwards, an isomorphism so-called *Maxwell correspondence* between equilibrium stresses and convex lifting in $\mathbb{R}^2$ was advocated in Crapo and Whiteley [1993, 1994], Schulz [2008]. Subsequently, these equivalences were generalized and extended to partitions in general dimensional space $\mathbb{R}^d$ in studies by Rybnikov and Aurenhammer Aurenhammer [1987a,b], Rybnikov [1999]. However, most of these results are difficult to apply in numerical methods such as those usually employed in linear control design. One of the goals of this work is to revisit these concepts and extract the most important elements needed in the inverse optimal control design. Indeed, in regard to applications in linear MPC, a difficulty arises: the explicit solution of a linear MPC problem with respect to a quadratic cost function is not, in many cases, convexly liftable. As a consequence, control theory needs a systematic approach for the use of a lifting procedure in the inverse optimality problem. This aspect will be discussed in details to provide a complete solution for *any* PWA function.

4.2 Preliminaries

The definition of a polyhedral partition $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ has been presented in Definition 3.2.1. Note however that a polyhedral partition is not necessarily a subdivision of a polyhedron. Instead, the union of its components i.e. $\mathcal{X} = \bigcup_{i\in\mathcal{I}_N} \mathcal{X}_i$, can be a non-convex set. A simple example to illustrate a non-convex polyhedral set can be found in Figure 4.1. In many texts, such a particular set is still called *a polyhedral set* even if it is not convex. It is mainly due to the fact that its boundaries are described by linear constraints. In some places of this chapter, the term *non-convex polyhedral/polytopic set* is of use to denote such particular sets. Also, a subdivision of a non-convex polyhedral set into several polyhedra is still called a *polyhedral partition* and *polytopic partition* if this non-convex polyhedral set is compact. Apart from this particularity, a polyhedron/polytope is always understood as a convex polyhedron/polytope.

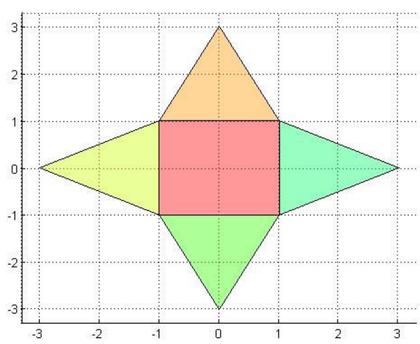


Figure 4.1: A non-convex polytopic set.

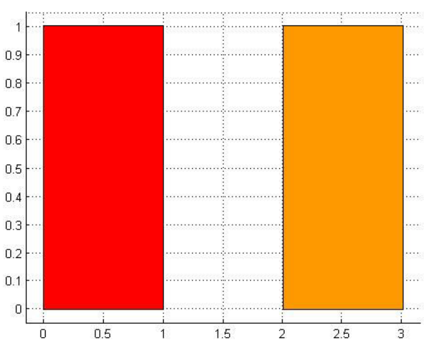


Figure 4.2: A cell complex.

Another particular class of polyhedral partitions is called *cell complexes* which will be usually employed in this chapter. Its definition is presented in Grünbaum [1967] and is recalled below for completeness.

Definition 4.2.1 A finite family $\mathcal{C}$ of polyhedra in $\mathbb{R}^d$ will be called a *cell complex* provided:

— every face of a member of $\mathcal{C}$ is itself a member of $\mathcal{C}$,
— the intersection of any two members of $\mathcal{C}$ is a face of these two members.

From the above definition, a *cell complex* may not need to build a full-dimensional set. An example in this sense is the boundary of a polytope in $\mathbb{R}^3$. This boundary is a cell complex, consisting of the facets, the edges, the vertices of this polytope and the empty set, known to be an *improper face* of a polyhedron. However, this cell complex does not build a full-dimensional set in $\mathbb{R}^3$, which is the first condition of a *polyhedral partition*.

For simplicity, a *cell complex* should be understood, in this manuscript, as a *polyhedral partition* satisfying the face-to-face property i.e. any pair of regions share a common face. Recall that the *empty set* is considered as an *improper face* of a polyhedron. Accordingly, the polytopic partition in Figure 4.2 is a cell complex since the intersection of two regions is empty. However, to keep the convexity, we mostly restrict our attention in this manuscript to a cell complex of a polyhedron. More intuitively, a cell complex of a polyhedron is a polyhedral partition whose facet-to-facet property is fulfilled i.e. any pair of neighboring regions share a common facet.

Another illustration is presented in Figure 4.3. The partition therein is a polytopic partition but not a cell complex in $\mathbb{R}^2$, since the intersection of the red polytope and the yellow one is a facet of the red polytope, but not a facet of the yellow one. Otherwise, the partition in Figure 4.4 is a cell complex due to the fact that the intersection any pair of neighboring regions is their common facet.

Definition 4.2.2 Given a polyhedral partition $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ of a polyhedron $\mathcal{X} \subseteq \mathbb{R}^d$, a *piecewise affine lifting* is described by function $z : \mathcal{X} \to \mathbb{R}$ with:

$$z(x) = a_i^T x + b_i \text{ for any } x \in \mathcal{X}_i, \tag{4.1}$$

and $a_i \in \mathbb{R}^d$, $b_i \in \mathbb{R}$, $\forall i \in \mathcal{I}_N$.

Definition 4.2.3 Given a polyhedral partition $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ of a polyhedron $\mathcal{X} \subseteq \mathbb{R}^d$, a piecewise affine lifting $z(x) = a_i^T x + b_i$ for $x \in \mathcal{X}_i$, is called *convex piecewise affine lifting* if the following conditions hold true:

— $z(x)$ is continuous over $\mathcal{X}$,
— for each $i \in \mathcal{I}_N$, $z(x) > a_j^T x + b_j$ for all $x \in \mathcal{X}_i \backslash \mathcal{X}_j$ and all $j \neq i$, $j \in \mathcal{I}_N$.

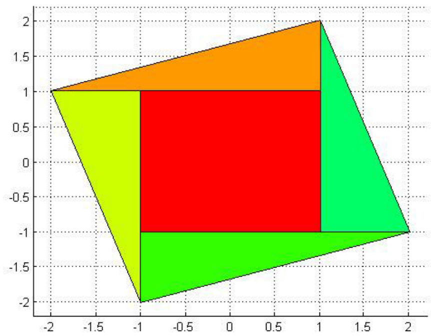


Figure 4.3: A polytopic partition in $\mathbb{R}^2$.

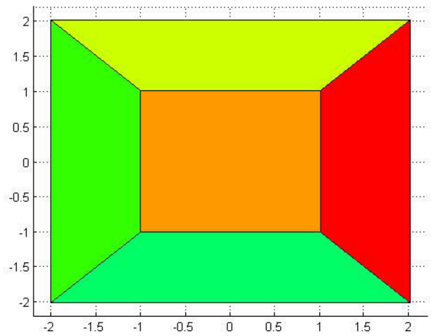


Figure 4.4: A cell complex in $\mathbb{R}^2$.

Note that the second condition in this definition implies that any pair of neighboring regions are lifted onto two distinct hyperplanes. Also, it implies the convexity of this piecewise affine lifting. Again, we note that this definition differs from the term *convex function* which allows two neighboring regions to be lifted onto the same hyperplane.

For ease of presentation, a slight abuse of notation is used hereafter: a *convex lifting* will be understood as a convex piecewise affine lifting.

From the above definition, if $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ is a polyhedral partition of a polyhedron whose convex liftings exist, then $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ should be a cell complex. This observation is formally stated by the following proposition.

Proposition 4.2.4 *A polyhedral partition $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ of a polyhedron $\mathcal{X} \subseteq \mathbb{R}^d$ which admits a convex lifting, is a cell complex.*

Proof: Suppose the given polyhedral partition $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ of a polyhedron $\mathcal{X} \subseteq \mathbb{R}^d$ which admits a convex lifting, is not a cell complex. Let

$$z(x) = a_i^T x + b_i \text{ for } x \in \mathcal{X}_i$$

denote a convex lifting of $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$. As $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ is not a cell complex, there exists a pair of neighboring regions, denoted by $\mathcal{X}_i, \mathcal{X}_j$, whose facet-to-facet property is not fulfilled. Due to the definition of convex liftings, the hyperplane denoted by $\mathcal{H}_0$, containing $\mathcal{X}_i \cap \mathcal{X}_j$, can be described by

$$\mathcal{H}_0 = \left\{ x \in \mathbb{R}^d \mid a_i^T x + b_i = a_j^T x + b_j \right\}.$$

Also, due to the non-satisfaction of the facet-to-facet property, there exists a point, denoted by x_0, such that either $x_0 \in \mathcal{H}_0 \cap \mathcal{X}_i$, $x_0 \notin \mathcal{X}_j$ or $x_0 \in \mathcal{H}_0 \cap \mathcal{X}_j$, $x_0 \notin \mathcal{X}_i$

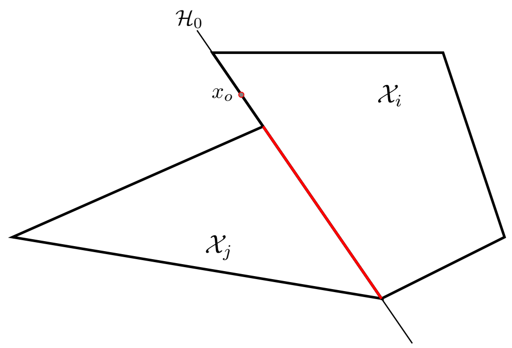


Figure 4.5: An illustration for Proposition 4.2.4.

(an illustration can be found in Figure 4.5). Without loss of generality, suppose the former case happens, then $x_0 \in \mathcal{H}_0$ implies

$$a_i^T x_0 + b_i = a_j^T x_0 + b_j. \tag{4.2}$$

On the other hand, from the definition of convex lifting, $x_0 \in \mathcal{X}_i, x_0 \notin \mathcal{X}_j$ lead to

$$a_i^T x_0 + b_i > a_j^T x_0 + b_j. \tag{4.3}$$

Inclusions (4.2) and (4.3) are clearly contradictory. Therefore, partition $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ has to be a cell complex. The proof is complete. $\square$

Remark 4.2.5 Note that Proposition 4.2.4 holds true not only for polyhedral partitions of a polyhedron, but also for polyhedral partitions of a non-convex polyhedral set.

It is worth stressing that the cell complex characterization of $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ is a necessary condition for the existence of a convex lifting, but not a sufficient condition. Namely, a cell complex still has to satisfy additional conditions for the existence of a convex lifting.

Definition 4.2.6 A given cell complex $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ of a possibly non-convex polyhedral set $\mathcal{X} \subseteq \mathbb{R}^d$ has *an affinely equivalent polyhedron* if there exists a polyhedron $\widetilde{\mathcal{X}} \subset \mathbb{R}^{d+1}$ such that for each $i \in \mathcal{I}_N$:

1. $\exists F_i \in \mathcal{F}(\widetilde{\mathcal{X}})$ satisfying: $\text{Proj}_{\mathbb{R}^d} F_i = \mathcal{X}_i$,
2. if $\underline{z}(x) = \min\limits_{z} z$ s.t. $\begin{bmatrix} x^T & z \end{bmatrix}^T \in \widetilde{\mathcal{X}}$, then $\begin{bmatrix} x \\ \underline{z}(x) \end{bmatrix} \in F_i$ for $x \in \mathcal{X}_i$.

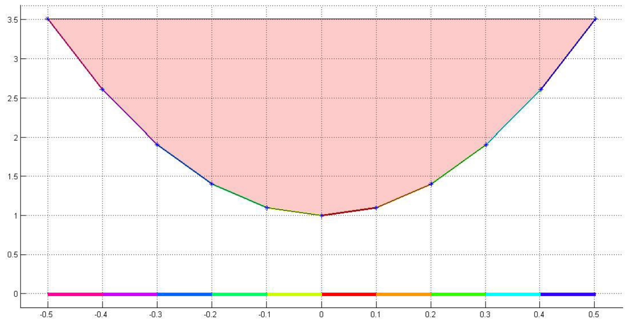


Figure 4.6: An illustration of the operation Xproj and an affinely equivalent polyhedron.

An illustration can be found in Figure 4.6 where the cell complex in $\mathbb{R}$ consists of the multicolored segments along the horizontal axis. One of its affinely equivalent polyhedra in $\mathbb{R}^2$ is the pink shaded region. Moreover, the lower facets of this polytope are an illustration of the facets F_i appearing in the definition.

Remark 4.2.7 This definition does not imply that $\mathcal{X} = \bigcup_{i \in \mathcal{I}_N} \mathcal{X}_i$ has to be a polytope (polyhedron). The second condition in the definition of an affinely equivalent polyhedron ensures that some of the facets of $\widetilde{\mathcal{X}}$ at the lower values of z are exclusively considered. The image of these facets via the orthogonal projection coincides with cell complex $\{\mathcal{X}_i\}_{i \in \mathcal{I}_N}$. An illustration can be found in Figure 4.7 where the given polytopic partition is presented in Figure 4.1, one of its affinely equivalent polyhedron is the polytope above whose lower facets interested in the orthogonal projection are solid-colored. In case $\mathcal{X}$ is a polytope (polyhedron), all the lower facets of $\widetilde{\mathcal{X}}$ are of interest in the orthogonal projection. These facets build a convex surface called a convex lifting as defined previously. Some necessary and sufficient conditions for a cell complex to be convexly liftable are recalled in Section 4.4.1. Note that none of them requires $\{\mathcal{X}_i\}_{i \in \mathcal{I}_N}$ to be a partition of a polytope (polyhedron). However, in applications to inverse parametric linear/quadratic programming problem, this assumption will be of help to guarantee the convexity of recovered optimization problem.

Remark 4.2.8 Note also that in some texts e.g. De Loera et al. [2010] a polyhedral partition admitting a convex lifting, is alternatively called *a regular partition*. Moreover, in Aurenhammer [1987a,b], these partitions are alternatively called *polytopical cell complexes*. In our works, we choose the terminology *convexly liftable cell complex* for a more intuitive reason.

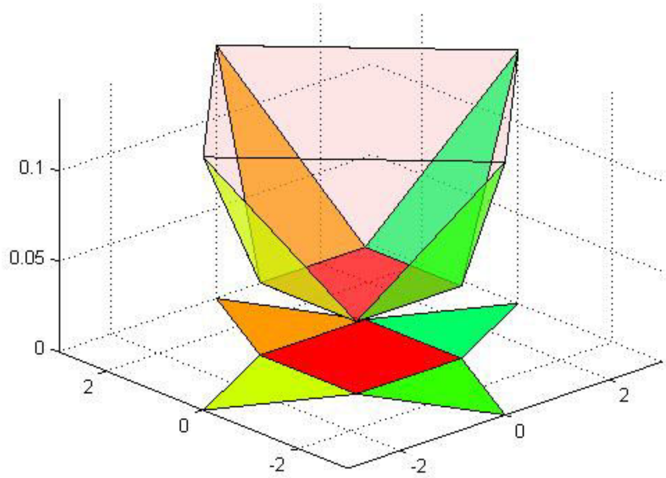


Figure 4.7: An illustration of the lower facets F_i interested in the orthogonal projection in Definition 4.2.6.

In order to link a given cell complex and one of its affinely equivalent polyhedra, it is necessary to introduce a new operator called *partitioned orthogonal projection*.

Definition 4.2.9 Given a polyhedron $\widetilde{\mathcal{X}} \subset \mathbb{R}^{d+1}$, the *partitioned orthogonal projection* of $\widetilde{\mathcal{X}}$ on the first d coordinates is denoted as $\mathrm{Xproj}_{\mathbb{R}^d}\, \widetilde{\mathcal{X}}$, and is defined as:

$$\begin{aligned} \mathrm{Xproj}_{\mathbb{R}^d}\, \widetilde{\mathcal{X}} := \Bigg\{ & \{\mathcal{X}_i\}_{i\in\mathcal{I}_N} \mid \mathcal{X}_i = \mathrm{Proj}_{\mathbb{R}^d} F_i \text{ s.t. } F_i \in \mathcal{F}(\widetilde{\mathcal{X}}), \\ & \text{any } \begin{bmatrix} x \\ \underline{z}(x) \end{bmatrix} \in F_i \text{ satisfies } \underline{z}(x) = \min_z z \text{ subject to } \begin{bmatrix} x^T & z \end{bmatrix}^T \in \widetilde{\mathcal{X}} \Bigg\}. \end{aligned} \tag{4.4}$$

The partitioned orthogonal projection is also illustrated in Figure 4.6. The $x-$coordinates are described by the horizontal axis, and the $z-$coordinates can be described by the vertical axis. The given polytope $\widetilde{\mathcal{X}}$ is the pink one whose lower facets are multicolored. These multicolored facets of $\widetilde{\mathcal{X}}$ are the facets F_i which are the objects of the Xproj operation. The result of this operation is represented by the multicolored regions along the horizontal axis.

It can be observed that if $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N} = \mathrm{Xproj}_{\mathbb{R}^d}\widetilde{\mathcal{X}}$, then the set $\mathcal{X} = \bigcup_{i\in\mathcal{I}_N} \mathcal{X}_i \subseteq \mathbb{R}^d$ is a polyhedron.

Remark 4.2.10 $\mathrm{Xproj}_{\mathbb{R}^d}\widetilde{\mathcal{X}}$ is a *finite* collection of polyhedra, since $\mathcal{F}(\widetilde{\mathcal{X}})$ is a finite collection of polyhedra in $\mathbb{R}^{d+1}$. The uniqueness of the partitioned orthogonal projection is due to the uniqueness of the set of lower facets of $\widetilde{\mathcal{X}}$. $\mathrm{Xproj}_{\mathbb{R}^d}\widetilde{\mathcal{X}}$ returns a cell complex.

Remark 4.2.11 From the two last definitions, if partition $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ of a possibly non-convex polyhedral set $\mathcal{X} \subseteq \mathbb{R}^d$ is convexly liftable and $\widetilde{\mathcal{X}}$ denotes one of its affinely equivalent polyhedra, then $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N} \subseteq \mathrm{Xproj}_{\mathbb{R}^d}\widetilde{\mathcal{X}}$. More precisely, if $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ is a cell complex of a polyhedron then $\mathrm{Xproj}_{\mathbb{R}^d}\widetilde{\mathcal{X}} = \{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$, otherwise $\mathrm{Xproj}_{\mathbb{R}^d}\widetilde{\mathcal{X}} = \bigcup_{i\in\mathcal{I}}\mathcal{X}_i$ with $\mathcal{I}_N \subset \mathcal{I}$.

Remark 4.2.12 Notice also that given a polyhedron $\widetilde{\mathcal{X}} \subset \mathbb{R}^{d+1}$, if z denotes the last coordinate of $\widetilde{\mathcal{X}}$ such that $\begin{bmatrix} x^T & z \end{bmatrix}^T \in \widetilde{\mathcal{X}}$, then $\mathrm{Xproj}_{\mathbb{R}^d}\,\widetilde{\mathcal{X}}$ is nothing other than the cell complex, associated with the optimal solution to the following parametric linear programming problem:

$$\min_z z \text{ subject to } \begin{bmatrix} x^T & z \end{bmatrix}^T \in \widetilde{\mathcal{X}}.$$

4.3 Problem statement

4.3.1 Parametric linear/quadratic programming problems

It is shown in Bemporad et al. [2002], Olaru and Dumur [2004], Pistikopoulos et al. [2007], Seron et al. [2003], Tøndel et al. [2003] that a parametric linear/quadratic programming problem is defined with respect to $d_x, d_{\mathbf{u}} \in \mathbb{N}_{>0}$ as follows:

$$\begin{aligned} &\min_{\mathbf{u}} f(\mathbf{u}, x), \\ \text{subject to:} \quad &G\mathbf{u} \leq W + Ex, \end{aligned} \tag{4.5}$$

where $x \in \mathbb{R}^{d_x}$ represents the parameter vector, $\mathbf{u} \in \mathbb{R}^{d_{\mathbf{u}}}$ represents the decision variable, and $f(\mathbf{u}, x)$ represents a linear/quadratic cost function in $\mathbf{u}$ and x. The above problem has a continuous solution denoted as $\mathbf{u}^*(x)$ (see Olaru and Dumur [2006] and Theorem 4 in Bemporad et al. [2002]), known to be a piecewise affine function defined over a polyhedral partition of the parameter space denoted as $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$:

$$\begin{aligned} \mathcal{X} = \bigcup_{i\in\mathcal{I}_N} \mathcal{X}_i, &\quad \text{is a polyhedron,} \\ \mathbf{u}^*(x) = f_{pwa}(x) = H_i x + G_i, &\quad \forall x \in \mathcal{X}_i. \end{aligned} \tag{4.6}$$

Notice that the optimal solution to a parametric quadratic programming problem is unique Bemporad et al. [2002] when $f(\mathbf{u}, x)$ is a strictly convex function of $\mathbf{u}$.

It is already known that this uniqueness may no longer be preserved in case of a parametric linear programming problem. However, a continuous selection among the optimal solutions to such a linear problem is shown in Olaru and Dumur [2006] to exist.

Conversely, given a continuous PWA function defined over a polyhedral partition, the question is whether there exists an optimization problem such that its optimal solution is equivalent to the given PWA function. The answer is shown in Baes et al. [2008] to be affirmative, although the numerical construction of such an optimization problem is still open. A possible candidate optimization problem may be characterized by a linear/quadratic cost function and a set of linear constraints. Such a set of constraints must be constructed from the given polyhedral partition and its associated PWA function. An approach for such a construction can be based on convex lifting which makes a link with the polyhedral partitions. This convex lifting construction will be detailed in the sequel. For the moment, the definition of an inverse parametric linear/quadratic programming problem is introduced.

4.3.2 Inverse parametric linear/quadratic programming problems

From the mathematical point of view, an inverse parametric linear/quadratic programming problem intends to reconstruct an appropriate optimization problem with respect to a given continuous piecewise affine function $u(x) = f_{pwa}(x)$, defined over a given polyhedral partition $\{\mathcal{X}_i\}_{i \in \mathcal{I}_N}$ of the parameter space $\mathcal{X} \subseteq \mathbb{R}^{d_x}$, such that the optimal solution of this reconstructed problem is equivalent to the given piecewise affine function $f_{pwa}(x)$. It is worth stressing that the equivalence here means that the boundary between two different regions of the parameter space partition corresponding to two different affine functions is preserved, and a subdivision or refinement of the regions corresponding to the same affine function is acceptable. This problem can be briefly stated as follows:

Problem statement: For a given polyhedral partition $\{\mathcal{X}_i\}_{i \in \mathcal{I}_N}$ of the state space $\mathcal{X} \subseteq \mathbb{R}^{d_x}$ associated with a continuous piecewise affine function $f_{pwa}(x) : \mathcal{X} \to \mathbb{R}^{d_u}$, find a linear/quadratic cost function $J(x, z, u)$ and matrices H_x, H_u, H_z, K such that:

$$\begin{cases} f_{pwa}(x) = \mathrm{Proj}_{\mathbb{R}^{d_u}} \arg \min\limits_{[z\ u^T]^T} J(x, z, u), \\ s.t. \quad H_x x + H_z z + H_u u \leq K. \end{cases} \tag{4.7}$$

As mentioned before, the convex-lifting based approach to such an inverse optimality problem is presented next. A definition of *invertibility* needs to be introduced in order to establish the working assumption of this convex lifting based method.

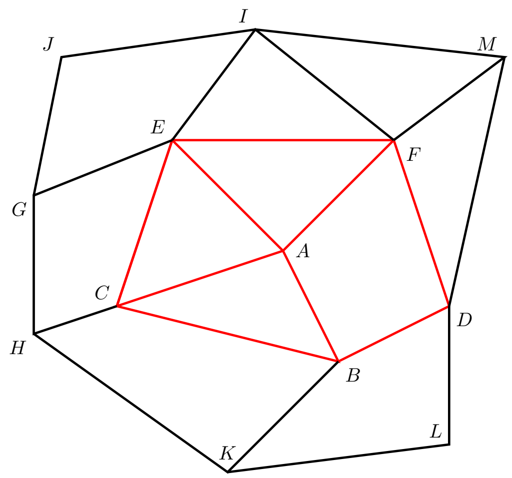


Figure 4.8: An illustration of the star of the $0-$face A in $\mathbb{R}^2$.

Definition 4.3.1 A continuous PWA function defined over a polyhedral partition is called *invertible* if there exists an appropriate constraint set and a cost function such that their associated parametric convex programming problem admits the given continuous PWA function as its optimal solution.

4.4 A constructive convex lifting based approach for IPL/QP

4.4.1 Existing results on convex liftings

The definition of a convex lifting has previously been presented. To clarify some existing results, we now recall the definition of *equilibrium stresses*, briefly denoted as *stresses* c.f. Lee [1996], Rybnikov [1999]. For ease of presentation, let $n(F, C)$ denote the inward unit normal vector to the polyhedron C at its facet F, meaning the unit vector $n(F, C)$ normal to F at F and inward to C.

Further, the *star* of a face in a cell complex $\mathcal{C}$ is its smallest sub-complex containing all faces of $\mathcal{C}$ which contain this face. An illustration can be found in Figure 4.8 where a cell complex consists of several triangles and quadrilaterals, the star of the $0-$face (point) A composes of the red faces:

— $0-$face A,
— $1-$faces AB, AC, AE, AF,
— $2-$faces ABC, ACE, AEF, $AFDB$.

Definition 4.4.1 (Rybnikov [1999]) A real-valued function $s(\cdot)$ defined on the $(d-1)-$faces of a cell complex $K \subset \mathbb{R}^d$ is called a $d-$stress if at each internal

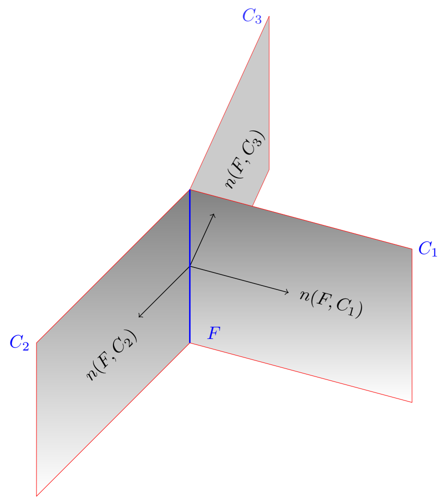


Figure 4.9: An illustration of 3−stress.

$(d-2)$−face F of K:

$$\sum_{C|F\subset C} s(C)n(F,C) = 0, \tag{4.8}$$

where this sum ranges over all $(d-1)$−faces in the star of F (the $(d-1)$−faces such that F is their common facet). The quantities $s(C)$ are the coefficients of this d−stress, are called a *tension* if the sign is strictly positive, and a *compression* if the sign is strictly negative.

An illustration of the above definition is depicted in Figure 4.8. This cell complex contains the following 0−faces in its interior: E, F, A, B, C. Suppose there exists a 2−stress defined over the set of 1−faces of this cell complex, let $s(\cdot)$ denote the coefficients of this 2−stress. Then, considering 0−face A, AE is an 1−face in the star of A, $\overrightarrow{AE}$ is a vector normal to A and inward to 1−face AE, but it is not a unit vector, therefore, the inward unit normal vector at A to AE is $n(A, AE) = \frac{\overrightarrow{AE}}{|AE|}$, where $|AE|$ denotes the Euclidean distance between A and E. Similarly, the inward unit normal vectors at A to the 1−faces AF, AB, AC are $n(A, AF) = \frac{\overrightarrow{AF}}{|AF|}, n(A, AB) = \frac{\overrightarrow{AB}}{|AB|}, n(A, AC) = \frac{\overrightarrow{AC}}{|AC|}$, respectively. Following the definition of stresses at the 0−face A, one obtains:

$$s(AE)n(A, AE) + s(AF)n(A, AF) + s(AB)n(A, AB) + s(AC)n(A, AC) = 0,$$

then the 0−face A is called *in equilibrium*. This cell complex is *convexly liftable* if the 0−faces E, F, A, B, C are in equilibrium with strictly positive function $s(\cdot)$ (this necessary and sufficient condition will be recalled next.)

Another illustration for the definition of 3−stress is shown in Figure 4.9. More clearly, consider an internal 1−face F of an arbitrary cell complex in $\mathbb{R}^3$. Suppose the $2-$faces of this cell complex which share a common facet F, are C_1, C_2, C_3, respectively. The inward unit normal vectors of F to C_1, C_2, C_3, i.e. $n(F, C_1)$, $n(F, C_2)$, $n(F, C_3)$, are shown therein. According to Definition 4.4.1, the condition for the $1-$face F to be in equilibrium is the existence of a $3-$stress: $s(C_1)$, $s(C_2)$, $s(C_3) \in \mathbb{R}$ such that:

$$s(C_1)n(F, C_1) + s(C_2)n(F, C_2) + s(C_3)n(F, C_3) = 0.$$

The equivalence between the existence of stresses and convex liftings for cell complexes in $\mathbb{R}^2$ was already stated through studies by Maxwell in Maxwell [1864], Cremona, Crapo, Whiteley in Crapo and Whiteley [1993, 1994], Schulz in Schulz [2008]. These results were then generalized to the cell complexes in the general dimensional space $\mathbb{R}^d$ through different studies e.g. in Rybnikov [1999]. It was shown therein that there exists a convex lifting for a cell complex in $\mathbb{R}^d$ if and only if one of the followings holds:

— it admits a strictly positive $d-$stress,
— it is an additively weighted Dirichlet-Voronoi diagram
— it is an additively weighted Delaunay decomposition,
— it is the section of a $(d + 1)$-dimensional Dirichlet-Voronoi partition [1].

The above results cover the general class of cell complexes in $\mathbb{R}^d$. Unfortunately, despite the mathematical completeness of the existing results, the verification of these conditions are computationally expensive. Furthermore, they do not provide any hint for the construction of a convex lifting. From applications in control design, due to the dimension of the state space, there are several obstacles to make use of the above results difficult:

— The computation of $d-$stresses for all $(d-1)-$faces of a given cell complex is based on the determination of inward unit normal vectors. With respect to the halfspace/vertex representation of a polytope/polyhedron, this determination is not trivial. As a consequence, the construction of a convex lifting, based on the $d-$stresses notion, may not be appropriate.
— To our best knowledge, recognizing an additively weighted Voronoi diagram or an additively weighted Delaunay decomposition requires solving an optimization problem with bi-linear constraints (c.f. Nguyen et al. [2015d] or related materials in Section 7.3). Thus, this task becomes computationally demanding, once the number of regions of a given cell complex is large.
— The solution to a parametric quadratic programming problem is in some cases a piecewise affine function defined over a polyhedral partition instead of a cell complex.

1. Other related results can be found in Konstantin Rybnikov's thesis Rybnikov [1999], equally in Aurenhammer [1987a,c, 1991]. Note that an additively weighted Dirichlet-Voronoi diagram is in fact a generalization of a power diagram.

Due to the above reasons, applications in control theory need specific algorithms to verify the convex liftability of the cell complexes inherited from parametric linear/quadratic programming problems, and construct their convex liftings if they exist. These discussions and related problems are detailed in the next section. Note however that the construction of convex liftings for some special cases e.g. Voronoi diagrams and Delaunay triangulations and their recognition were already investigated in Aurenhammer [1991], Edelsbrunner and Seidel [1986], Hartvigsen [1992]. The aim of the next subsection is to present such a construction in the general case of cell complexes and shows how to transform a generic polyhedral partition into a convexly liftable cell complex.

4.4.2 A construction of convex liftings

In this subsection, the main objective is to present an algorithm for the construction of an affinely equivalent polyhedron (or a convex lifting) for a given cell complex via linear/quadratic programming. This algorithm exploits the continuity and the convexity of two neighboring regions.

4.4.2.1 Construction for polytopic partitions

Given a cell complex $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ of a polytope $\mathcal{X} \subset \mathbb{R}^d$, $\widetilde{\mathcal{X}} \subset \mathbb{R}^{d+1}$ denotes one of the affinely equivalent polyhedra of $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$. For each region $\mathcal{X}_i$, $i \in \mathcal{I}_N$, the hyperplane, containing the lower facet of $\widetilde{\mathcal{X}}$ whose orthogonal projection onto $\mathbb{R}^d$ coincides with $\mathcal{X}_i$, has the following form:

$$\mathcal{H}_i = \left\{ \begin{bmatrix} x \\ z_i(x) \end{bmatrix} \in \mathbb{R}^{d+1} \mid z_i(x) = a_i^T x + b_i \right\}, \tag{4.9}$$

for suitable $a_i \in \mathbb{R}^d$, $b_i \in \mathbb{R}$.

Let $(i, j) \in \mathcal{I}_N^2$ be an index pair such that $(\mathcal{X}_i, \mathcal{X}_j)$ are neighbors. The *continuity conditions* between them are described as follows:

$$\forall x \in \mathcal{X}_i \cap \mathcal{X}_j,\ i \neq j,\ z_i(x) = z_j(x). \tag{4.10}$$

Moreover, the *convexity conditions* between them can be handled as:

$$\forall x \in \mathcal{X}_i \backslash (\mathcal{X}_i \cap \mathcal{X}_j),\ z_i(x) > z_j(x). \tag{4.11}$$

The conditions (4.10) and (4.11) represent fundamental properties of a convex lifting, therefore they can be used for a construction by considering (a_i, b_i) as variables. Algorithm 4.1 summarizes such a constructive procedure which allows for the computation of the gains $(a_i,\ b_i)$, $\forall i \in \mathcal{I}_N$ of a convex lifting.

The following theorem serves as an explanation of this algorithm.

Algorithm 4.1 An algorithm to construct a convex lifting for a given cell complex $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ of a polytope $\mathcal{X} \subset \mathbb{R}^d$.

Input: $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ and a given constant $c > 0$.

Output: $(a_i,\, b_i)$, $\forall i \in \mathcal{I}_N$, a polyhedron $\widetilde{\mathcal{X}} \subset \mathbb{R}^{d+1}$ such that $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N} = \mathrm{Xproj}_{\mathbb{R}^d}\widetilde{\mathcal{X}}$.

1: Register all pairs of neighboring regions in $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$.

2: For each pair of neighboring regions $(\mathcal{X}_i,\, \mathcal{X}_j)$, $(i,\, j) \in \mathcal{I}_N^2$:

— Add continuity conditions $\forall v \in \mathcal{V}(\mathcal{X}_i \cap \mathcal{X}_j)$:

$$a_i^T v + b_i = a_j^T v + b_j. \tag{4.12}$$

— Add convexity conditions $\forall u \in \mathcal{V}(\mathcal{X}_i), u \notin \mathcal{V}(\mathcal{X}_i \cap \mathcal{X}_j)$:

$$a_i^T u + b_i \geq a_j^T u + b_j + c. \tag{4.13}$$

3: Solve the following convex optimization problem by minimizing a chosen cost function e.g.

$$\min_{a_i,\, b_i} \sum_{i=1}^{N} (a_i^T a_i + b_i^T b_i) \text{ subject to } (4.12), (4.13). \tag{4.14}$$

4: Construct an affinely equivalent polyhedron

$$\widetilde{\mathcal{X}} = \mathrm{conv}\left\{ \begin{bmatrix} v \\ z(v) \end{bmatrix} \in \mathbb{R}^{d+1} \mid v \in \bigcup_{i\in\mathcal{I}_N} \mathcal{V}(\mathcal{X}_i), z(v) = a_i^T v + b_i \text{ if } v \in \mathcal{X}_i \right\}.$$

Theorem 4.4.2 *If problem (4.14) is feasible, then function $z(x) = a_i^T x + b_i$ for $x \in \mathcal{X}_i$ is a convex lifting over the given cell complex $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$.*

Proof: If the optimization problem (4.14) is feasible, then the continuity conditions of function $z(x)$ and the convexity conditions of its epigraph are all fulfilled. Accordingly, for two neighboring regions $(\mathcal{X}_i, \mathcal{X}_j)$, it follows that:

$$\begin{aligned} a_i^T x + b_i &= a_j^T x + b_j, \quad \text{for all } x \in \mathcal{X}_i \cap \mathcal{X}_j, \\ a_i^T x + b_i &> a_j^T x + b_j, \quad \text{for all } x \in \mathcal{X}_i \backslash \mathcal{X}_j. \end{aligned} \tag{4.15}$$

The same inclusion holds for the other pairs of neighboring regions. This leads to the continuity of $z(x)$ and for each $i \in \mathcal{I}_N$:

$$a_i^T x + b_i > a_j^T x + b_j \text{ for all } x \in \mathcal{X}_i \backslash \mathcal{X}_j, \ \forall j \neq i, \, j \in \mathcal{I}_N. \tag{4.16}$$

Therefore, function $z(x) = a_i^T x + b_i$ for $x \in \mathcal{X}_i$ is a convex lifting defined over the cell complex $\{\mathcal{X}_i\}_{i \in \mathcal{I}_N}$, as defined in Definition 4.2.3. $\square$

Remark 4.4.3 Note that due to (4.16), (a_i, b_i) and (a_j, b_j) are different for any $i \neq j$, $(i, j) \in \mathcal{I}_N^2$. This implies that any pair of regions in the given cell complex are lifted onto two distinct hyperplanes. More specially, function $z(x)$ can be written in the form $z(x) = \max\limits_{i \in \mathcal{I}_N} (a_i^T x + b_i)$, known to be a convex function over $\mathcal{X}$.

Also, the hyperplanes $\mathcal{H}_i, \forall i \in \mathcal{I}_N$ defined in (4.9) are supporting hyperplanes of polytope $\tilde{\mathcal{X}}$ at its lower facets.

It is worth emphasizing that the first step in Algorithm 4.1 considers all non identical pairs of different regions without taking the order into account. For each of these pairs, if the intersection of these two regions is a polytope of dimension $d - 1$, then they are neighboring. This task can be performed through the halfspace representation. More clearly, if these two polytopes have the halfspace representation $H_1 x \leq K_1$, $H_2 x \leq K_2$, respectively, then their intersection is a polytope, described by: $\begin{bmatrix} H_1 \\ H_2 \end{bmatrix} x \leq \begin{bmatrix} K_1 \\ K_2 \end{bmatrix}$. Determining the real dimension of a polytope/polyhedron can be carried out using existing routines (see MPT Herceg et al. [2013]).

As for the complexity of this step, if the given cell complex consists of N regions, then the number of cases considered is $\frac{1}{2}N(N-1)$. The second step imposes constraints via the vertices of related polytopes, thus it requires the vertex representation of these regions. Moreover, the third step of this algorithm finds a_i, b_i by solving an optimization problem with respect to a pre-chosen quadratic cost function. This choice of cost function aims to reduce the slope of the facets of this convex lifting corresponding to the regions of the given cell complex. Other choices may be possible, however, if these slopes are large, the computation of an affinely equivalent polyhedron via the convex hull in the next step may lead to numerical sensitivity. In fact, algorithms to compute the convex hull of a set of discrete points are known to be quite expensive and their precision is limited. Therefore, if a coordinate of these points is greater with several orders of magnitude than the other coordinates, this computation may no longer be reasonable. Otherwise, the feasibility of an optimization problem depends simultaneously on the feasibility of the constraint set and the boundedness of the optimal cost function. If the constraint set is feasible, but the optimal cost function is infinite, one cannot conclude that this optimization problem is feasible.

As seen in (4.13), the strict convexity condition (4.11) can be easily transformed into inequality constraints in an optimization problem by adding a positive constant c on the right-hand side of (4.13), thus $>$ can be replaced with $\geq$. Theoreti-

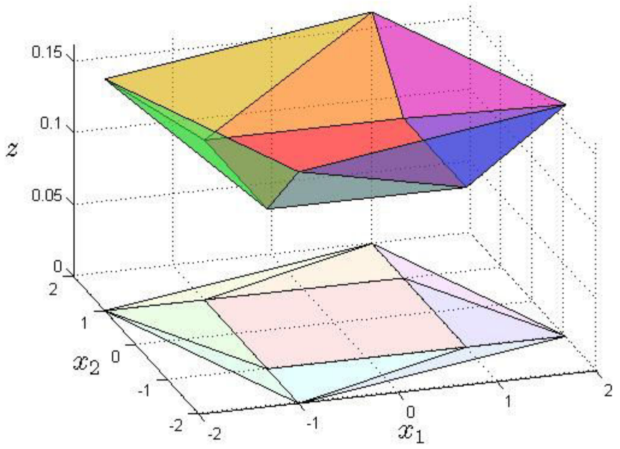


Figure 4.10: A cell complex in $\mathbb{R}^2$ and one of its affinely equivalent polyhedra.

cally, if the given cell complex is convexly liftable, then any choice of this positive constant does not have any effect on the feasibility of the optimization problem (4.14). In fact, if a_i, b_i for $i \in \mathcal{I}_N$ are coefficients of a convex lifting for the given cell complex $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$, then so are $(\alpha a_i, \alpha b_i)$ for any $\alpha > 0$, since (4.12), (4.13) amount to

$$(\alpha a_i)^T v + (\alpha b_i) = (\alpha a_j)^T v + (\alpha b_j) \;\text{ for }\; v \in \mathcal{V}(\mathcal{X}_i \cap \mathcal{X}_j)$$
$$(\alpha a_i)^T u + (\alpha b_i) \geq (\alpha a_j)^T u + (\alpha b_j) + \alpha c \;\text{ for }\; u \in \mathcal{V}(\mathcal{X}_i), u \notin \mathcal{V}(\mathcal{X}_j).$$

In other words, $\widetilde{\ell}(x) = (\alpha a_i)^T x + (\alpha b_i)$ for $x \in \mathcal{X}_i$ also represents a convex lifting of cell complex $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ which may be resulted from Algorithm 4.1 with a given constant αc. Therefore, the optimization problem (4.14) is still feasible with the constant $\alpha c > 0$.

A numerical example of a cell complex of a polytope in $\mathbb{R}^2$ to illustrate Algorithm 4.1 is presented in Figure 4.10. One of its affinely equivalent polyhedra is the shaded polytope with the lower facets multicolored.

Note that Algorithm 4.1 is applicable for the general case of cell complexes of polytopes and polytopic partitions in $\mathbb{R}^d$. It leads to a verification condition for the convex liftability of a given polytopic partition via the feasibility of the associated optimization problem. More precisely, a given polytopic partition in $\mathbb{R}^d$ is convexly liftable if the optimization problem (4.14) is feasible. According to Proposition 4.2.4, if a polyhedral partition is convexly liftable, then it should be a cell complex. Therefore, the optimization problem (4.14) is infeasible for the polytopic partitions whose facet-to-facet property is not fulfilled.

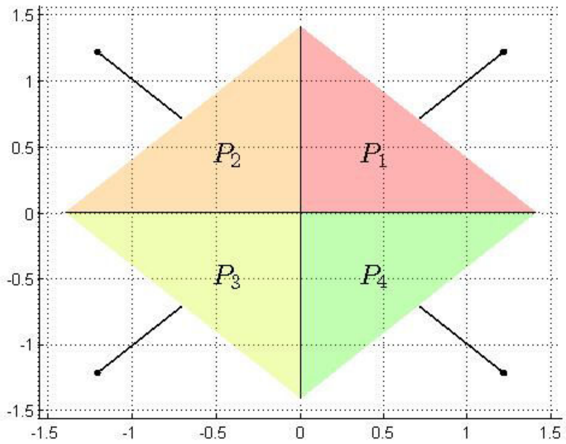


Figure 4.11: A polyhedral partition for which Algorithm 4.1 does not hold.

4.4.2.2 Construction for polyhedral partitions

Based on the vertex representation, Algorithm 4.1 can construct a convex lifting for a polytopic partition whenever it exits. However, for a polyhedral partition including several unbounded polyhedra, this algorithm is not directly applicable. This limitation can be explained via a simple example. Consider a polyhedral partition of the whole space $\mathbb{R}^2$ which contains four quadrants, as shown in Figure 4.11. It can be seen that these four regions share the unique vertex of this partition, known to be the origin $(0, 0)$. It can be observed that the regions P_1, P_2 are neighbors. According to Algorithm 4.1, the continuity constraint is applied at the origin, however no convexity constraint is enforced here. Therefore, the optimization problem (4.14) is only subject to continuity constraints between (P_1, P_2), (P_2, P_3), (P_3, P_4), (P_4, P_1). This leads to the fact that the obtained result from (4.14) is likely $(a_i, b_i) = 0$ for all $i \in \mathcal{I}_4$. Clearly, with these trivial values, Algorithm 4.1 does not result in a convex lifting.

To cover these particular partitions, two intuitive approaches can be proposed:

— either modify this algorithm subject to cell complexes of polyhedra,

— or adjust the given cell complex of polyhedra in such a manner that this algorithm can be directly of use.

The latter approach will be of interest based on an intermediate operation. More precisely, the given polyhedral partition will be restricted to an appropriate bounded region such that the convex liftability of this new partition and the initial one is treated in a similar manner. Accordingly, the use of a hyperbox large enough to contain in its interior all vertices of the initial partition, will be proposed. The existence of such a hyperbox is guaranteed by the fact that a polyhedral partition is a collection of finitely many polyhedra, as a consequence the number of vertices is finite.

Let $B_d(\epsilon)$ denote a hyperbox for a given $\epsilon > 0$ defined as follows:

$$B_d(\epsilon) := \left\{ x \in \mathbb{R}^d \mid \|x\|_\infty \leq \epsilon \right\}.$$

The hyperbox used for the aforementioned goal needs to meet the following assumption:

Assumption 4.4.4 For all $x \in \bigcup_{i \in \mathcal{I}_N} \mathcal{V}(\mathcal{X}_i)$, $x \in \operatorname{int}(B_d(\epsilon))$ for some suitable $\epsilon > 0$.

The following observation shows the convex liftability equivalence between a polyhedral partition and a suitable polytopic partition.

Proposition 4.4.5 *Given a cell complex $\{\mathcal{X}_i\}_{i \in \mathcal{I}_N}$ of an unbounded polyhedron $\mathcal{X} \subseteq \mathbb{R}^d$, then this cell complex is convexly liftable, if and only if there exists a hyperbox $B_d(\epsilon)$ satisfying Assumption 4.4.4 such that the polytopic partition $\{\mathcal{X}_i \cap B_d(\epsilon)\}_{i \in \mathcal{I}_N}$ of bounded set $\mathcal{X} \cap B_d(\epsilon)$ is convexly liftable.*

Proof: $\longrightarrow$ As recalled in Section 4.4.1, the convex liftability of cell complex $\{\mathcal{X}_i\}_{i \in \mathcal{I}_N}$ leads to the existence of a strictly positive function $s(\cdot)$ defined over its $(d-1)-$faces such that every internal $(d-2)-$face is in equilibrium. Consider an internal $(d-2)-$face of $\{\mathcal{X}_i\}_{i \in \mathcal{I}_N}$ denoted by F. Let $\mathcal{F}^{(d-1)}(F)$ denote the set of $(d-1)-$faces of $\{\mathcal{X}_i\}_{i \in \mathcal{I}_N}$ in the star of F i.e. the $(d-1)-$faces sharing a common facet F, then the following inclusion holds true:

$$\sum_{C \in \mathcal{F}^{(d-1)}(F)} s(C)\, n(F, C) = 0.$$

Since $B_d(\epsilon)$ satisfies Assumption 4.4.4, thus if F is an internal $(d-2)-$face of $\{\mathcal{X}_i\}_{i \in \mathcal{I}_N}$, then the face $F \cap B_d(\epsilon)$ is also an internal $(d-2)-$face of the cell complex $\{\mathcal{X}_i \cap B_d(\epsilon)\}_{i \in \mathcal{I}_N}$. Furthermore, $F \cap B_d(\epsilon)$ also satisfies the following property:

$$\sum_{C \in \mathcal{F}^{(d-1)}(F)} s(C) n(F \cap B_d(\epsilon), C \cap B_d(\epsilon)) = 0,$$

due to the fact that $n(F \cap B_d(\epsilon), C \cap B_d(\epsilon)) = n(F, C)$. Therefore any internal face $F \cap B_d(\epsilon)$ is in equilibrium with strictly positive $d-$stress: $s(C \cap B_d(\epsilon)) = s(C) > 0$, meaning the cell complex $\{\mathcal{X}_i \cap B_d(\epsilon)\}_{i \in \mathcal{I}_N}$ is convexly liftable.
$\longleftarrow$ The sufficient condition can be similarly proved. $\square$

With respect to Assumption 4.4.4, the following proposition is of help to construct convex liftings for cell complexes of unbounded polyhedra.

Proposition 4.4.6 *Given a convexly liftable cell complex $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ of a polyhedron $\mathcal{X} \subseteq \mathbb{R}^d$ and a hyperbox $B_d(\epsilon)$ satisfying Assumption 4.4.4, then function $f : \mathcal{X} \cap B_d(\epsilon) \to \mathbb{R}$*

$$f(x) = a_i^T x + b_i \quad \textit{for} \quad x \in \mathcal{X}_i \cap B_d(\epsilon),$$

is a convex lifting of the cell complex $\{\mathcal{X}_i \cap B_d(\epsilon)\}_{i\in\mathcal{I}_N}$, if and only if the function $g : \mathcal{X} \to \mathbb{R}$ defined as follows:

$$g(x) = a_i^T x + b_i \quad \textit{for} \quad x \in \mathcal{X}_i,$$

is also a convex lifting of $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$.

Proof: $\longrightarrow$ First, due to Assumption 4.4.4, the intersection $\mathcal{X} \cap B_d(\epsilon)$ does not have any effect on the internal subdivision of $\mathcal{X}$, since all vertices of the partition $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ lie in the interior of $B_d(\epsilon)$.

Consider now two neighboring regions in the partition $\{\mathcal{X}_i \cap B_d(\epsilon)\}_{i\in\mathcal{I}_N}$, denoted as $\mathcal{X}_i \cap B_d(\epsilon), \mathcal{X}_j \cap B_d(\epsilon)$. As assumed, $f(x)$ is a convex lifting of this polytopic partition, then it can be deduced from its definition that:

$$\begin{aligned} a_i^T x + b_i &= a_j^T x + b_j \ \forall x \in (\mathcal{X}_i \cap B_d(\epsilon)) \cap (\mathcal{X}_j \cap B_d(\epsilon)) \\ a_i^T x + b_i &> a_j^T x + b_j \ \forall x \in (\mathcal{X}_i \cap B_d(\epsilon)) \backslash (\mathcal{X}_j \cap B_d(\epsilon)). \end{aligned}$$

Note also that constraint $a_i^T x + b_i = a_j^T x + b_j$ describes the hyperplane, separating $\mathcal{X}_i \cap B_d(\epsilon)$ and $\mathcal{X}_j \cap B_d(\epsilon)$, then it also separates $\mathcal{X}_i$ and $\mathcal{X}_j$. This end leads to:

$$\begin{aligned} a_i^T x + b_i &= a_j^T x + b_j \ \forall x \in \mathcal{X}_i \cap \mathcal{X}_j, \\ a_i^T x + b_i &> a_j^T x + b_j \ \forall x \in \mathcal{X}_i \backslash \mathcal{X}_j. \end{aligned}$$

Applying this argument to all pairs of neighboring regions, the following inclusion can be obtained:

$$a_i^T x + b_i > a_j^T x + b_j, \ \forall x \in \mathcal{X}_i \backslash \mathcal{X}_j, \forall j \neq i, j \in \mathcal{I}_N,$$

meaning $g(x)$ is a convex lifting of $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$.
$\longleftarrow$ The sufficient condition can be similarly proved. □

The above result is meaningful to construct a convex lifting for the cell complex $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ of an un bounded polyhedron $\mathcal{X} \subseteq \mathbb{R}^d$, from a convex lifting of the polytopic partition $\{\mathcal{X}_i \cap B_d(\epsilon)\}_{i\in\mathcal{I}_N}$ with appropriate $\epsilon > 0$. The remaining step is to determine such an ϵ satisfying Assumption 4.4.4. This determination can be carried out via Algorithm 4.2.

Algorithm 4.2 Computation of a hyperbox satisfying Assumption 4.4.4.

Input: Cell complex $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ and scalar $a > 0$,
Output: ϵ satisfying Assumption 4.4.4.

1: $V = \bigcup_{i\in\mathcal{I}_N} \mathcal{V}(\mathcal{X}_i)$.
2: Solve the following problem

$$\min_{\epsilon} \epsilon \;\text{ s.t. }\; \epsilon \geq 0,\; -(\epsilon - a)1_d \leq x \leq (\epsilon - a)1_d,\;\; \forall x \in V.$$

According to Algorithm 4.2, the insertion of a scalar $a > 0$ in the set of constraints aims to guarantee Assumption 4.4.4. In fact, constraints

$$-(\epsilon - a)1_d \leq x \leq (\epsilon - a)1_d,\; \forall x \in \bigcup_{i\in\mathcal{I}_N} \mathcal{V}(\mathcal{X}_i),$$

imply that $\|x\|_\infty \leq \epsilon - a < \epsilon$, leading to $x \in \operatorname{int}(B_d(\epsilon))$ for all $x \in \bigcup_{i\in\mathcal{I}_N} \mathcal{V}(\mathcal{X}_i)$. Note also that this constant can be freely chosen as long as it is strictly positive.

From this result, to construct a convex lifting for cell complex $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ of an unbounded polyhedron, one can run Algorithm 4.2 to compute $B_d(\epsilon)$, then use Algorithm 4.1 to construct a convex lifting for cell complex $\{\mathcal{X}_i \cap B_d(\epsilon)\}_{i\in\mathcal{I}_N}$. Subsequently, a convex lifting for cell complex $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ can be built as in Proposition 4.4.6. Accordingly, the construction of an affinely equivalent polyhedron is presented in the sequel. Let $\ell(x)$ denote a convex lifting for cell complex $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ i.e.

$$\ell(x) = a_i^T x + b_i \;\text{ for }\; x \in \mathcal{X}_i.$$

Also, define the following sets:

$$\begin{aligned}
&V = \bigcup_{i\in\mathcal{I}_N} \mathcal{V}(\mathcal{X}_i),\; R = \bigcup_{i\in\mathcal{I}_N} \mathcal{R}(\mathcal{X}_i),\\
&\widehat{V} = \left\{ \begin{bmatrix} x^T & \ell(x) \end{bmatrix}^T \mid x \in V \right\},\\
&\widehat{R} = \left\{ \begin{bmatrix} r^T & \widehat{\ell}(r) \end{bmatrix}^T \mid r \in R,\, \widehat{\ell}(r) = a_i^T r \;\text{ if }\; r \in \mathcal{R}(\mathcal{X}_i) \right\}.
\end{aligned}$$

Then an affinely equivalent polyhedron for $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ denoted by Π, can be defined as follows:

$$\Pi_{[x^T\; z]^T} = \operatorname{conv}(\widehat{V}) \oplus \operatorname{cone}(\widehat{R}). \tag{4.17}$$

This construction directly follows the Minkowski-Weyl theorem as presented in Section 2.2.3.

4.4.3 Related results of polyhedral partitions and non-convexly liftable cell complexes

This subsection focuses on non-convexly liftable cell complexes which do not admit convex liftings. From the generality of inverse optimality, a particular treatment needs to be introduced to cover these singular partitions. First, the property related to the dimension of auxiliary variables in construction of convex lifting for such partitions will be clarified. It will be proved that a non-convexly liftable cell complex/polyhedral partition in $\mathbb{R}^d$ remains non-convexly liftable onto any higher dimensional space $\mathbb{R}^n$ with $n > d+1$. One cannot build a convex lifting associated with such a cell complex/polyhedral partition despite an increase of degrees of freedom. To prove it, an intermediate observation also needs to be proved.

Proposition 4.4.7 *Consider a cell complex $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ of a polyhedron $\mathcal{X} \subseteq \mathbb{R}^d$ and a polyhedron $\mathcal{Y}^{(n)} \subset \mathbb{R}^n$, $n > d+1$. Let $x \in \mathbb{R}^d$ and $z \in \mathbb{R}^{n-d}$ denote the first d and the last $n-d$ coordinates of $\mathbb{R}^n$ respectively, i.e. $z = \begin{bmatrix} z^{(d+1)} \dots z^{(n)} \end{bmatrix}^T$. If there exists an index i, with $d+1 \leq i \leq n$ such that the optimal cost function of the following parametric linear programming problem:*

$$(z^{(i)})^*(x) = \min_z z^{(i)} \;\; \textit{subject to} \;\; \begin{bmatrix} x^T \, z^T \end{bmatrix}^T \in \mathcal{Y}^{(n)}, \; x \in \mathcal{X},$$

is a convex lifting for the given cell complex $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$, then there also exists a polyhedron $\mathcal{Y}^{(d+1)} \subset \mathbb{R}^{d+1}$ such that $\mathrm{Xproj}_{\mathbb{R}^d}\mathcal{Y}^{(d+1)} = \{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$.

Proof: If $(z^{(i)})^*(x)$ is a convex lifting of a given cell complex $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$, then the set defined as follows:

$$\mathcal{Y}^{(d+1)} = \left\{ \begin{bmatrix} x \\ z^{(i)} \end{bmatrix} \mid (z^{(i)})^*(x) \leq z^{(i)}, x \in \mathcal{X} \right\} \subset \mathbb{R}^{d+1},$$

represents the epigraph of the function $(z^{(i)})^*(x)$, and is an affinely equivalent polyhedron of cell complex $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$. Following the definition of an affinely equivalent polyhedron, $\mathrm{Xproj}_{\mathbb{R}^d} \mathcal{Y}^{(d+1)} = \{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$. □

The above proposition shows that one supplementary dimension is sufficient for the construction of convex liftings if they exist.

Remark 4.4.8 Note that if a function $z = f(x)$ describes the upper boundary of a polyhedron $\mathcal{Y}^{(d+1)}$ (known to be a concave function), associated with the cell complex $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$, then $z = -f(x)$ is a convex function defined over the same cell complex. Therefore, the above proposition still holds true if the optimal cost function of the following parametric linear programming problem:

$$(z^{(i)})^*(x) = \max_z z^{(i)} \text{ subject to } \begin{bmatrix} x^T \;\; z^T \end{bmatrix}^T \in \mathcal{Y}^{(n)}, \; x \in \mathcal{X},$$

satisfies $-(z^{(i)})^*(x)$ is a convex lifting for the given cell complex $\{\mathcal{X}_i\}_{i \in \mathcal{I}_N}$. Accordingly, such an affinely equivalent polyhedron can be chosen as follows:

$$\mathcal{Y}^{(d+1)} = \left\{ \begin{bmatrix} x \\ z^{(i)} \end{bmatrix} \mid (z^{(i)})^*(x) \geq -z^{(i)}, x \in \mathcal{X} \right\}.$$

Proposition 4.4.7 has an important implication on non-convexly liftable cell complexes, this implication is described in details through the next result.

Corollary 4.4.9 *A cell complex in $\mathbb{R}^d$ is non-convexly liftable in higher dimensional space $\mathbb{R}^n$ for an $n \geq d+1$ if and only if it does not have any affinely equivalent polyhedron in $\mathbb{R}^{d+1}$.*

Proof: The implication $\rightarrow$ is straightforward as long as $n = d+1$ is a particular choice of n.
For the implication $\leftarrow$, suppose the given cell complex is not convexly liftable in $\mathbb{R}^{d+1}$ but convexly liftable in $\mathbb{R}^n$, $n > d+1$. Through Proposition 4.4.7, it is easy to find the contradiction by showing that an appropriate polyhedron in particular subspace $\mathbb{R}^{d+1} \subset \mathbb{R}^n$ can be selected as an affinely equivalent polyhedron. □

A natural question arises whether the partition associated with optimal solution to a parametric linear/quadratic programming problem is convexly liftable. Practically, such non-convexly liftable configurations often appear, particularly in constrained control for linear systems. The non-liftability limitations emphasized in the previous results raise a question about the utility of the convex lifting operation. Fortunately, in control theory, splitting a given polyhedral partition is admissible. However, any modification of the initial boundaries [2] of the given polyhedral partition is not permitted due to the fact that such a modification will destroy the original structure of PWA controller. It leads to the case where two different affine control laws are defined over the same region of state space. Therefore, by preserving the internal boundaries, are there possible refinements for a given polyhedral partition in order to recover the convex liftability property?

Figure 4.12 represents a cell complex composed of several triangles in $\mathbb{R}^2$. This cell complex is not convexly liftable. This fact can be confirmed by the infeasibility of the optimization problem (4.14) in Algorithm 4.1. However, it will be shown next that there exists a subdivision which can retrieve the convex liftability for a given polyhedral partition. This class of subdivision is referred to as a *hyperplane arrangement* (see Aurenhammer [1987a]), defined as the decomposition of a space by a set of hyperplanes. It will be shown that the partitions, induced by such decompositions, are convexly liftable. As a consequence, if a partition is

2. In fact, the modification of boundaries between regions whose associated control laws are similar may be possible, and this trick can help to reduce the complexity of PWA controllers.

not convexly liftable, one can try to refine it such that the original partitioning is maintained and the new cell complex is convexly liftable. The above discussion is summarized in the following theorem.

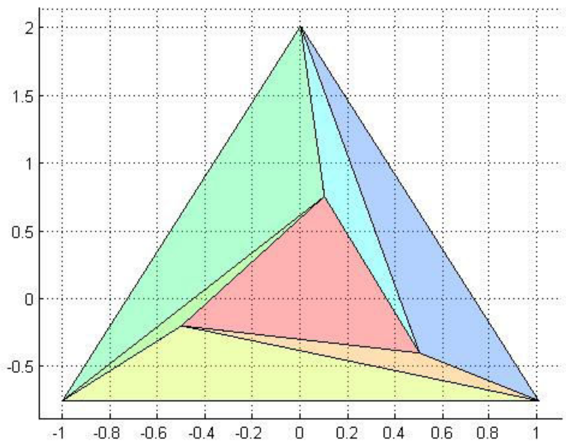


Figure 4.12: A non-convexly liftable cell complex in $\mathbb{R}^2$ extracted from Aichholzer et al. [2003].

Theorem 4.4.10 *Given a non-convexly liftable polyhedral partition $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ of a polyhedron $\mathcal{X} \subseteq \mathbb{R}^d$, there exists at least one subdivision, preserving the internal boundaries of this partition, such that the new cell complex is convexly liftable.*

Proof: Let $\mathcal{H}(\mathcal{X}_i)$ be the set of supporting hyperplanes of $\mathcal{X}_i$ at its facets, $\mathcal{H}(\mathcal{X}) = \bigcup_{i\in\mathcal{I}_N} \mathcal{H}(\mathcal{X}_i)$. We will show that the decomposition of $\mathcal{X}$ by $\mathcal{H}(\mathcal{X})$ makes the new cell complex convexly liftable. As presented above, such a decomposition is denoted as *hyperplane arrangement*. The convex liftability of such a decomposition can be proved by returning to the concept of stresses presented in Subsection 4.4.1.

In fact, considering any $(d-2)-$face F_0 lying in the interior of $\mathcal{X}$, this $(d-2)-$face F_0 is the intersection of finitely many hyperplanes in $\mathcal{H}(\mathcal{X})$. If $\mathcal{F}^{(d-1)}(F_0)$ denotes the set of all $(d-1)-$faces in the star of F_0, then for each $F_i^{(d-1)} \in \mathcal{F}^{(d-1)}(F_0)$, there exists a unique $F_j^{(d-1)} \neq F_i^{(d-1)}$ and $F_j^{(d-1)} \in \mathcal{F}^{(d-1)}(F_0)$ such that $F_i^{(d-1)}, F_j^{(d-1)}$ lie in a common hyperplane of $\mathcal{H}(\mathcal{X})$ and they have a common facet F_0. Accordingly, it can be seen that the inward unit normal vectors to the faces $F_i^{(d-1)}, F_j^{(d-1)}$ at their common facet F_0, denoted by $n(F_0, F_i^{(d-1)})$, $n(F_0, F_j^{(d-1)})$, respectively, satisfy:

$$n(F_0, F_i^{(d-1)}) = -n(F_0, F_j^{(d-1)}).$$

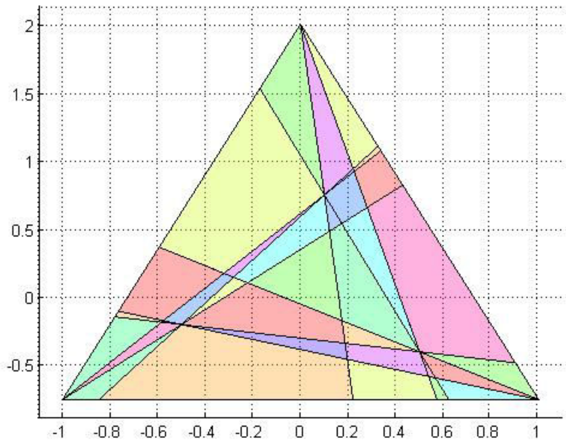


Figure 4.13: A subdivision of the given non convexly liftable cell complex in Figure 4.12 into a convexly liftable cell complex.

Thus, a pair of coefficients of strictly positive stresses $s(F_i^{(d-1)}), s(F_j^{(d-1)})$ exists (e.g. $s(F_i^{(d-1)}) = s(F_j^{(d-1)}) = 1$) such that:

$$s(F_i^{(d-1)})n(F_0, F_i^{(d-1)}) + s(F_j^{(d-1)})n(F_0, F_j^{(d-1)}) = 0.$$

Applying the same argument for every element of $\mathcal{F}^{(d-1)}(F_0)$, one can obtain a strictly positive $d-$stress such that F_0 is in equilibrium. □

Remark 4.4.11 Theorem 4.4.10 is related to Theorem 3 in Aurenhammer [1987a]. It ensures that the optimization problem (4.14) is feasible for any cell complex of a polyhedron, obtained by the hyperplane arrangement technique.

An illustration of this result is provided in Figure 4.13. This hyperplane arrangement technique is applied for the non-convexly liftable cell complex in Figure 4.12.

Remark 4.4.12 Practically, hyperplane arrangement is only one way to show the existence of modifications for the given non-convexly liftable cell complex/polyhedral partition into a convexly liftable cell complex. In control theory, such a modification can increase the complexity of PWA control laws in the implementation. Therefore, such a *complete* refinement may not be necessary in practical applications. Many different refinement techniques exist. We refer to Gulan et al. [2015] for an alternative technique for a class of particular cases in control theory.

4.5 Solution to IPL/QP problems

This section provides a complete view of inverse parametric linear/quadratic programming problems for the given continuous/discontinuous PWA function defined over a polyhedral partition.

4.5.1 IPL/QP for continuous PWA functions over polytopic partitions

The definition of an inverse parametric convex programming problem has been introduced in Subsection 4.3.2. The solution to such inverse optimality problems is built upon the convex lifting approach. For the moment, some regularity assumptions need to be stated to make the present approach reasonable from the construction point of view.

Assumption 4.5.1 The parametric linear/quadratic programming problems are exclusively considered as possible candidates for solutions to inverse optimality problem. As a consequence, the cost function has the following form:

$$J(x, z, u) = \left[x^T \; z \; u^T\right] Q \begin{bmatrix} x \\ z \\ u \end{bmatrix} + C^T \begin{bmatrix} x \\ z \\ u \end{bmatrix}, \tag{4.18}$$

with positive semidefinite matrix $Q^T = Q \geq 0$.

Assumption 4.5.2 The given parameter space partition $\{\mathcal{X}_i\}_{i \in \mathcal{I}_N}$, associated with a given PWA function, is convexly liftable.

Assumption 4.5.3 The parameter space $\mathcal{X} = \bigcup_{i \in \mathcal{I}_N} \mathcal{X}_i$ is a polytope.

Assumption 4.5.1 provides a manageable framework for the constructive inverse optimality procedures. Larger classes of objective functions can provide more degrees of freedom, but the linearity of such parametric convex programming problems is lost. Meanwhile, Assumption 4.5.2 underlines the convex liftability of the given cell complex and can be enforced via the refinement procedure described in the proof of Theorem 4.4.10. This condition is sufficient for the existence of a solution and will be detailed next. Finally, Assumption 4.5.3 restricts the inverse optimality to bounded feasible region in the parameter space with the bound given by a polytope. This assumption is not restrictive in MPC context since the feasible region of the state space is usually a polytope. Moreover, this assumption can be relaxed to *polyhedron*. It is mainly due to the fact that linear constraints are exclusively considered. This relaxation will be discussed in Section 4.5.2.

The following intermediate result is necessary for the development of a constructive solution to the inverse optimality problem.

Proposition 4.5.4 *Let $\Gamma_s \subset \mathbb{R}^{d_s}$ be a polytope with the set of vertices $\mathcal{V}(\Gamma_s) = \{s^{(1)}, \dots, s^{(q)}\}$. For any finite set of points $\{t^{(1)}, \dots, t^{(q)}\} \subset \mathbb{R}^{d_t}$ defining a non-degenerate polytope in $\mathbb{R}^{d_t}$, an extension of the family $\mathcal{V}(\Gamma_s)$ can be obtained in higher dimensional space $\mathbb{R}^{d_s+d_t}$ for the concatenated vectors $\left[s^T \; t^T\right]^T$ defining the set:*

$$V_{[s^T \; t^T]^T} := \left\{ \begin{bmatrix} s^{(1)} \\ t^{(1)} \end{bmatrix}, \dots, \begin{bmatrix} s^{(q)} \\ t^{(q)} \end{bmatrix} \right\}. \tag{4.19}$$

The polytope $\Gamma_{[s^T \; t^T]^T} = \operatorname{conv}(V_{[s^T \; t^T]^T})$ satisfies:

$$V_{[s^T \; t^T]^T} = \mathcal{V}(\Gamma_{[s^T \; t^T]^T}). \tag{4.20}$$

Proof: Geometrically, this proposition shows that if $s^{(i)}$ is a vertex of $\Gamma_s \subset \mathbb{R}^{d_s}$, then with any complementary vector $t^{(i)} \in \mathbb{R}^{d_t}$ leading to an extended vector $\left[s^{(i)T} \; t^{(i)T}\right]^T \in \mathbb{R}^{d_s+d_t}$, this vector represents a vertex of the new polytope $\Gamma_{[s^T \; t^T]^T}$ in $\mathbb{R}^{d_s+d_t}$ defined as the convex hull of the extended set of points $V_{[s^T \; t^T]^T}$. By construction $\mathcal{V}(\Gamma_{[s^T \; t^T]^T}) \subseteq V_{[s^T \; t^T]^T}$. Therefore, in order to prove this claim, we will prove that $\mathcal{V}(\Gamma_{[s^T \; t^T]^T}) \subset V_{[s^T \; t^T]^T}$ leads to a contradiction.

In fact, suppose $\mathcal{V}(\Gamma_{[s^T \; t^T]^T}) \subset V_{[s^T \; t^T]^T}$. According to this assumption, there exists a point in $V_{[s^T \; t^T]^T}$ which lies in the interior of the polytope $\Gamma_{[s^T \; t^T]^T}$ or can be described by a convex combination of the other points. Without loss of generality, let $\begin{bmatrix} s^{(q)} \\ t^{(q)} \end{bmatrix}$ denote this point, then there exists a vector $\alpha \in \mathbb{R}_+^{q-1}$ such that:

$$\begin{bmatrix} s^{(q)} \\ t^{(q)} \end{bmatrix} = \sum_{i=1}^{q-1} \alpha_i \begin{bmatrix} s^{(i)} \\ t^{(i)} \end{bmatrix}, \quad \sum_{i=1}^{q-1} \alpha_i = 1. \tag{4.21}$$

One can easily see from (4.21) that $s^{(q)}$, as a vertex of Γ_s, is described by a convex combination of the other vertices of Γ_s. This inclusion is contradictory to the definition of a vertex of a convex set (see Section 2.2.1 or 2.4 in Grünbaum [1967]). In other words, all elements of $V_{[s^T \; t^T]^T}$ are the vertices of $\Gamma_{[s^T \; t^T]^T}$. □

Remark 4.5.5 Note also that this proposition remains valid for the degenerate case where all points $\left\{t^{(1)}, \dots, t^{(q)}\right\}$ are placed on a hyperplane in $\mathbb{R}^{d_t}$. However, in this case, the new polytope lies practically in a strict subspace of $\mathbb{R}^{d_s+d_t}$. This particular case of values $t^{(i)}, \forall i \in \mathcal{I}_q$, is excluded in the previous result as not relevant for the scope of this chapter, even though the mathematical result holds.

Consider a given cell complex $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ of a polytope $\mathcal{X} \subset \mathbb{R}^{d_x}$ satisfying Assumption 4.5.2 and a continuous PWA function $f_{pwa}(x) : \mathcal{X} \to \mathbb{R}^{d_u}$ defined over this cell complex. For ease of presentation, let $\ell(x)$ denote a convex lifting defined over $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$. Define also the following sets:

$$\begin{aligned}
\Pi_{[x^T\ z]^T} &= \text{conv}\left\{ \left[v^T\ \ell(v)\right]^T \mid v \in \bigcup_{i\in\mathcal{I}_N} \mathcal{V}(\mathcal{X}_i) \right\}, \\
V_{[x^T\ z\ u^T]^T} &= \left\{ \left[v^T\ \ell(v)\ f_{pwa}^T(v)\right]^T \mid v \in \bigcup_{i\in\mathcal{I}_N} \mathcal{V}(\mathcal{X}_i) \right\}, \\
\Pi_{[x^T\ z\ u^T]^T} &= \text{conv}\left(V_{[x^T\ z\ u^T]^T} \right).
\end{aligned} \tag{4.22}$$

Note that $\Pi_{[x^T\ z]^T}$ defined above, represents an affinely equivalent polyhedron of cell complex $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ by appending to the vertices of $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ the corresponding coordinates of convex lifting $\ell(x)$ at these vertices. Subsequently, $\Pi_{[x^T\ z\ u^T]^T}$ is constructed by adding to the vertices of $\Pi_{[x^T\ z]^T}$ the corresponding coordinates of the given continuous PWA function $f_{pwa}(x)$ at the vertices of $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$. The set $\Pi_{[x^T\ z\ u^T]^T}$ will be used as a constraint set in our recovered optimization problem. With respect to the above notation, the solution to an inverse parametric linear/quadratic programming problem can be stated as follows.

Theorem 4.5.6 *Given a continuous PWA function $f_{pwa}(x)$ defined over a polytopic partition $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ satisfying Assumptions 4.5.2, 4.5.3 and the sets defined in (4.22), the followings hold true:*

1. $V_{[x^T\ z\ u^T]^T} = \mathcal{V}(\Pi_{[x^T\ z\ u^T]^T})$ *and* $\Pi_{[x^T\ z]^T} = \text{Proj}_{[x^T\ z]^T}\Pi_{[x^T\ z\ u^T]^T}$,
2. *The given piecewise affine function $f_{pwa}(x)$ is the image via the orthogonal projection onto $\mathbb{R}^{d_u}$ of the optimal solution to the optimization problem below:*

$$\min_{[z\ u^T]^T} z \quad \text{s.t.}\ \left[x^T\ z\ u^T\right]^T \in \Pi_{[x^T\ z\ u^T]^T}. \tag{4.23}$$

Proof: 1. The first claim: $V_{[x^T\ z\ u^T]^T} = \mathcal{V}(\Pi_{[x^T\ z\ u^T]^T})$, is directly deduced from Proposition 4.5.4. The second claim follows from the construction of $\Pi_{[x^T\ z\ u^T]^T}$ having all its vertices as non-degenerate extended vectors of the vertices of $\Pi_{[x^T\ z]^T}$.

2. It is known that $\Pi_{[x^T\ z]^T} \subset \mathbb{R}^{d_x+1}$ represents an affinely equivalent polyhedron of the partition $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$. Let $F^{(i)}_{[x^T\ z]^T}$ for $i \in \mathcal{I}_N$ denote the lower facet of $\Pi_{[x^T\ z]^T}$ such that:

— $\text{Proj}_x F^{(i)}_{[x^T\ z]^T} = \mathcal{X}_i$,

— for any $x \in \mathcal{X}_i$, $\left[x^T\, z^*(x)\right]^T \in F^{(i)}_{[x^T\, z]^T}$ satisfies

$$z^*(x) = \min_z z \quad \text{s.t.} \quad \left[x^T\, z\right]^T \in \Pi_{[x^T\, z]^T}.$$

Also, there exists in higher dimensional space $\mathbb{R}^{d_x+d_u+1}$, a d_x-face denoted as $F^{(i)}_{[x^T\, z\, u^T]^T}$ of $\Pi_{[x^T\, z\, u^T]^T}$ such that:

$$\text{Proj}_{[x^T\, z]^T} F^{(i)}_{[x^T\, z\, u^T]^T} = F^{(i)}_{[x^T\, z]^T}.$$

Thus, a point $\left[x^T\, z\, u^T\right]^T \in \Pi_{[x^T\, z\, u^T]^T}$ satisfying $x \in \mathcal{X}_i$ has the minimal value of z if and only if this point locates on $F^{(i)}_{[x^T\, z\, u^T]^T}$. It is worth stressing that the face $F^{(i)}_{[x^T\, z\, u^T]^T}$ is defined as follows:

$$F^{(i)}_{[x^T\, z\, u^T]^T} = \text{conv}\left\{\left[v^T \quad \ell(v) \quad f^T_{pwa}(v)\right]^T \mid v \in \mathcal{V}(\mathcal{X}_i)\right\}.$$

From the above argument, it follows that there exist non-negative scalars $\alpha(v) \in \mathbb{R}_+$, for $v \in \mathcal{V}(\mathcal{X}_i)$ such that:

$$\sum_{v \in \mathcal{V}(\mathcal{X}_i)} \alpha(v) = 1,$$
$$\left[x^T\ z^*(x)\ (u^*)^T(x)\right]^T = \sum_{v \in \mathcal{V}(\mathcal{X}_i)} \alpha(v) \left[v^T \quad \ell(v) \quad f^T_{pwa}(v)\right]^T.$$

Since this property holds for all elements of $\mathcal{X}_i$, it can be deduced that:

$$\begin{bmatrix} z^*(x) \\ u^*(x) \end{bmatrix} = \sum_{v \in \mathcal{V}(\mathcal{X}_i)} \alpha(v) \begin{bmatrix} \ell(v) \\ f_{pwa}(v) \end{bmatrix} = \begin{bmatrix} \ell(x) \\ f_{pwa}(x) \end{bmatrix}, \quad \forall x \in \mathcal{X}_i.$$

Clearly, $f_{pwa}(x)$ is a sub-component of this optimal solution.

To complete the proof, the uniqueness of such an optimal solution needs to be clarified. Suppose there exist two different optimal solutions to (4.23):

$$\left[z_1^*(x)\ (u_1^*)^T(x)\right]^T = \arg \min_{[z\ u^T]^T} z, \quad \text{s.t.} \quad \left[x^T\, z\, u^T\right]^T \in \Pi_{[x^T\, z\, u^T]^T},$$
$$\left[z_2^*(x)\ (u_2^*)^T(x)\right]^T = \arg \min_{[z\ u^T]^T} z, \quad \text{s.t.} \quad \left[x^T\, z\, u^T\right]^T \in \Pi_{[x^T\, z\, u^T]^T},$$

then it is clear that $z_1^*(x) = z_2^*(x) = \ell(x)$. Accordingly, if $u_1^*(x) \neq u_2^*(x)$ for $x \in \mathcal{X}_i$, there exists a $(d_x+1)-$face denoted as F of $\Pi_{[x^T\, z\, u^T]^T}$ (illustrated in Figure

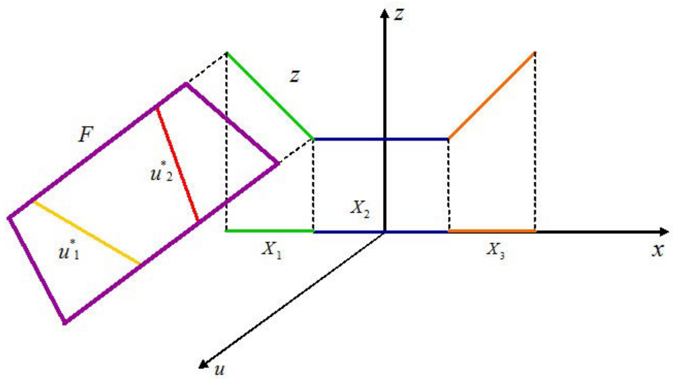


Figure 4.14: An illustration of two different optimal solutions.

4.14) to which two optimal solutions $\left[z_1^*(x)\ (u_1^*)^T(x)\right]^T$ and $\left[z_2^*(x)\ (u_2^*)^T(x)\right]^T$ belong such that F is perpendicular to the space $\left[x^T\ z\right]^T$. This implies that the value of $f_{pwa}(v)$ is not uniquely defined for vertices $v \in \mathcal{V}(\mathcal{X}_i)$. This consequence contradicts the construction of the constraint set $\Pi_{[x^T\ z\ u^T]^T}$ presented in (4.22). Therefore, such two optimal solutions have to be identical leading to the uniqueness. □

Theorem 4.5.6 shows that the construction of constraint set characterized by the polytope $\Pi_{[x^T\ z\ u^T]^T} \subset \mathbb{R}^{d_x+d_u+1}$ depends on the construction of a convex lifting (an affinely equivalent polyhedron $\Pi_{[x^T\ z]^T}$) for the given cell complex $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$. Thus, from the algorithmic point of view it is necessary to focus on the construction of such a convex lifting of the parameter space cell complex as detailed in the previous section. This construction will lead with a simple extension to a complete solution of the inverse optimality problem.

The constructive procedure towards recovering a continuous PWA function defined over a convexly liftable polytopic partition is summarized through Algorithm 4.3:

Theorem 4.5.6 proves the existence of an optimization problem with respect to a linear cost function which has as a sub-component of the optimal solution, a given PWA function defined over a convexly liftable polytopic partition. The following theorem shows the existence of equivalent optimization problem with respect to a *quadratic cost function*.

Theorem 4.5.7 *Consider a continuous PWA function $f_{pwa}(x)$ defined over a polytopic partition $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ satisfying Assumptions 4.5.2, 4.5.3 and the sets defined in (4.22). Function $f_{pwa}(x)$ is the image via the orthogonal projection onto $\mathbb{R}^{d_u}$*

Algorithm 4.3 Linear equivalent optimization problem

Input: A continuous PWA function $f_{pwa}(x)$ defined over a convexly liftable polytopic partition $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ of a polytope $\mathcal{X} \subset \mathbb{R}^{d_x}$.

Output: $\Pi_{[x^T\ z\ u^T]^T}$ and $J(x,z,u)$.

1: Construct a convex lifting $\ell(x)$ for $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ via Algorithm 4.1.

2: Compute $\Pi_{[x^T\ z\ u^T]^T}$ as in (4.22).

3: Define $J(x,z,u) = z$.

4: Solve the following parametric linear programming problem:

$$\left[z^*(x)\ (u^*)^T(x)\right]^T = \arg \min_{[z\ u^T]^T} z \quad \text{subject to} \quad \left[x^T\ z\ u^T\right]^T \in \Pi_{[x^T\ z\ u^T]^T}.$$

5: Obtain the given PWA function: $\text{Proj}_u \begin{bmatrix} z^*(x) \\ u^*(x) \end{bmatrix} = f_{pwa}(x)$.

of the optimal solution to the following optimization problem:

$$\min_{[z\ u^T]^T} (z-\sigma(x))^2 \quad \textit{s.t.} \quad \left[x^T\ z\ u^T\right]^T \in \Pi_{[x^T\ z\ u^T]^T}, \tag{4.24}$$

where $\sigma(x) : \mathcal{X} \to \mathbb{R}$ *denotes any function satisfying:* $\sigma(x) \leq \ell(x)$.

Proof: Consider an affinely equivalent polyhedron $\Pi_{[x^T\ z]^T}$ defined as in (4.22). According to its definition, we obtain:

$$\ell(x) = \min_z z \quad \text{subject to} \quad \left[x^T\ z\right]^T \in \Pi_{[x^T\ z]^T}.$$

Therefore, for any function $\sigma(x) : \mathcal{X} \to \mathbb{R}$ satisfying $\sigma(x) \leq \ell(x)$, the minimization of $(z-\sigma(x))^2$ amounts to the minimization of z subject to the same set of constraints $\Pi_{[x^T\ z\ u^T]^T}$. According to Theorem 4.5.6, the given continuous PWA function $f_{pwa}(x)$ is a sub-component of the optimal solution to (4.23), as well as (4.24). $\square$

Theorem 4.5.7 proposes a generic quadratic cost function of $\left[z\ u^T\right]^T$. If the goal is to obtain a quadratic cost function of $\left[x^T\ z\ u^T\right]^T$, then function $\sigma(x)$ should be chosen as an affine function of x. Algorithm 4.4 summarizes the constructive procedure of an equivalent optimization problem with respect to a quadratic cost function.

Remark 4.5.8 Theorems 4.5.6, 4.5.7 prove that there exists an equivalence between linear programming problems and a class of quadratic programming problems. This characterization may be meaningful in reducing computational complexity of control laws for such a class of optimization based controls.

Algorithm 4.4 Quadratic equivalent optimization problem

Input: A continuous PWA function $f_{pwa}(x)$ defined over a convexly liftable polytopic partition $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ of a polytope $\mathcal{X}\subset\mathbb{R}^{d_x}$.

Output: $\Pi_{[x^T\ z\ u^T]^T}$ and $J(x,z,u)$.

1: Construct a convex lifting $\ell(x)$ for $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ via Algorithm 4.1.

2: Compute $\Pi_{[x^T\ z\ u^T]^T}$ as in (4.22).

3: Choose a function [3] $\sigma(x):\mathcal{X}\to\mathbb{R}$ such that $\sigma(x)\le\ell(x)$.

4: Define $J(x,z,u)=(z-\sigma(x))^2$.

5: Solve the following parametric quadratic programming problem:

$$\left[z^*(x)\ (u^*)^T(x)\right]^T = \arg\min_{[z\ u^T]^T}(z-\sigma(x))^2 \text{ s.t. } \left[x^T\ z\ u^T\right]^T\in\Pi_{[x^T\ z\ u^T]^T}.$$

6: Project the optimal solution onto $\mathbb{R}^{d_u}$: $\text{Proj}_u\begin{bmatrix}z^*(x)\\u^*(x)\end{bmatrix}=f_{pwa}(x)$.

Example 1: To illustrate the above theoretical results, the following continuous PWA function is considered to be recovered:

$$f(x)=\begin{cases}-1.7160x+0.2287 & \text{for } x\in\mathcal{X}_1=\{x\mid -0.1\le x\le 0\}\\ -1.2962x+0.2287 & \text{for } x\in\mathcal{X}_2=\{x\mid 0\le x\le 0.1\}\\ 0.3223x+0.4325 & \text{for } x\in\mathcal{X}_3=\{x\mid -0.2\le x\le -0.1\}\\ 1.5175x-0.0527 & \text{for } x\in\mathcal{X}_4=\{x\mid 0.1\le x\le 0.2\}\\ -1.0224x+0.1636 & \text{for } x\in\mathcal{X}_5=\{x\mid -0.3\le x\le -0.2\}\\ 4.5011x-0.6494 & \text{for } x\in\mathcal{X}_6=\{x\mid 0.2\le x\le 0.3\}\\ -1.9741x-0.1219 & \text{for } x\in\mathcal{X}_7=\{x\mid -0.4\le x\le -0.3\}\\ -0.4211x+0.8273 & \text{for } x\in\mathcal{X}_8=\{x\mid 0.3\le x\le 0.4\}\\ 6.0457x+3.0860 & \text{for } x\in\mathcal{X}_9=\{x\mid -0.5\le x\le -0.4\}\\ 0.7345x+0.3650 & \text{for } x\in\mathcal{X}_{10}=\{x\mid 0.4\le x\le 0.5\}\,.\end{cases}\tag{4.25}$$

This function is visualized in Figure 4.15. Also, a convex lifting of the cell complex $\{\mathcal{X}_i\}_{i\in\mathcal{I}_{10}}$ is shown in Figure 4.16.

The set of constraints for a recovered optimization problem is presented in Figure 4.17 as the shaded pink polytope. Note also that in this figure, the multi-colored line along the $x-$axis represents the given cell complex $\{\mathcal{X}_i\}_{i\in\mathcal{I}_{10}}$, i.e. each region corresponds to a color, whereas the green line over this cell complex represents the continuous PWA function defined in (4.25). The convex lifting for $\{\mathcal{X}_i\}_{i\in\mathcal{I}_{10}}$ shown in Figure 4.16, is also embedded in the space $\mathbb{R}^3$ in Figure

3. One can choose $\sigma(x)$ to be an affine function composing $\ell(x)$.

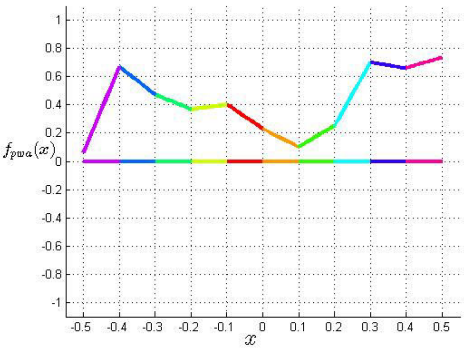


Figure 4.15: A parameter space partition and an associated continuous PWA function.

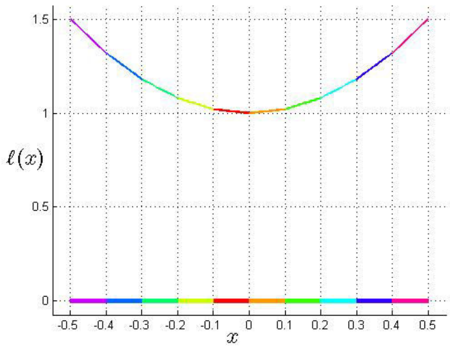


Figure 4.16: A convex lifting for the cell complex $\{\mathcal{X}_i\}_{i\in\mathcal{I}_{10}}$.

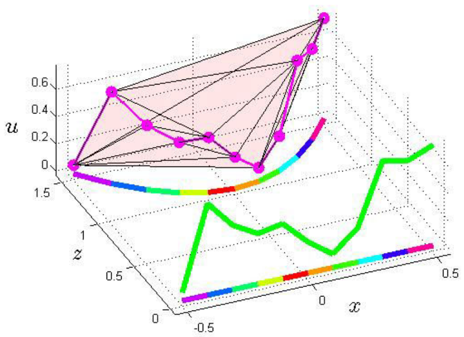


Figure 4.17: A set of constraint $\Pi_{[x^T\ z\ u^T]^T}$.

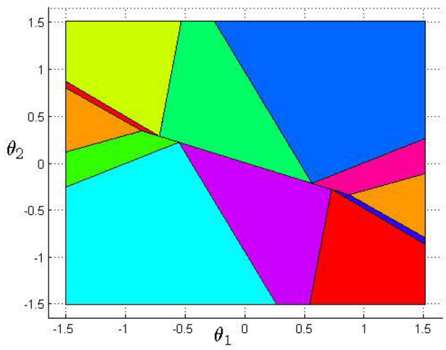


Figure 4.18: A polytopic partition of the parameter space.

4.17 as the multi-colored curve above the cell complex in the plane $\begin{bmatrix} x^T & z \end{bmatrix}^T$. It is necessary to emphasize that the pink curve along several $1-$faces of $\Pi_{[x^T\ z\ u^T]^T}$ represents the optimal solution to the optimization problem (4.23). It can be observed that the orthogonal projection of this optimal solution onto the space of $\begin{bmatrix} x^T & u^T \end{bmatrix}^T$ coincides with the PWA function (4.25).

Example 2: To illustrate a case in which the given polyhedral partition is not convexly liftable, consider example 1 presented in Spjøtvold et al. [2006]:

$$\min_x x^T x \quad \text{s.t. } x \in \mathcal{P}(\theta),\ \theta \in \Theta,$$

$$\Theta = \left\{ \theta \in \mathbb{R}^2 \mid -\frac{3}{2} \leq \theta_i \leq \frac{3}{2},\ i = 1, 2 \right\}.$$

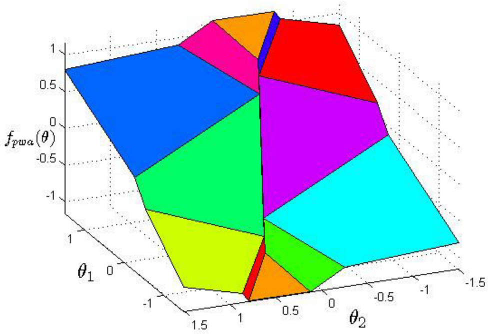


Figure 4.19: A continuous PWA function associated with the polytopic partition in Figure 4.18

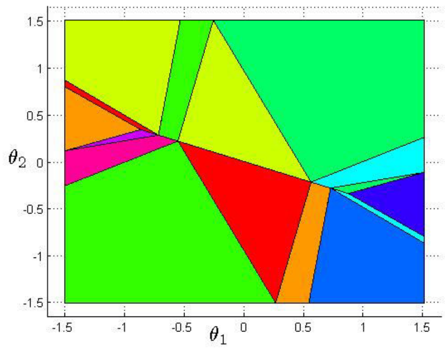


Figure 4.20: A subdivision of the polytopic partition in Figure 4.18 into a convex liftable cell complex.

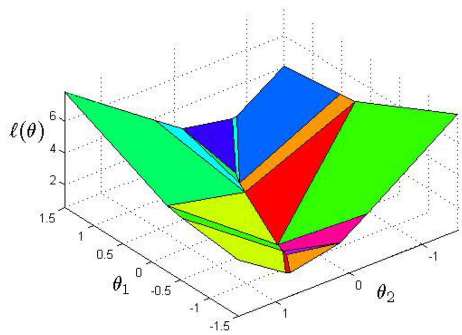


Figure 4.21: A convex lifting for the cell complex in Figure 4.20.

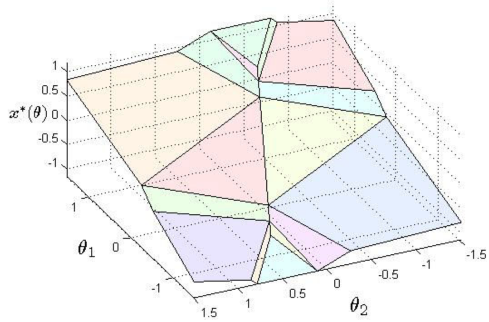


Figure 4.22: A continuous PWA function equivalent to the one shown in Figure 4.19.

$$\mathcal{P}(\theta) = \left\{ x \in \mathbb{R}^3 \left| \begin{array}{c} x_1 - x_3 \leq -1 + \theta_1, \; -x_1 - x_3 \leq -1 - \theta_1 \\ x_2 - x_3 \leq -1 - \theta_2, \; -x_2 - x_3 \leq -1 + \theta_2 \\ \frac{3}{4}x_1 + \frac{16}{25}x_2 - x_3 \leq -1 + \theta_1 \\ -\frac{3}{4}x_1 - \frac{16}{25}x_2 - x_3 \leq -1 - \theta_1 \end{array} \right. \right\}.$$

The parameter space partition is shown in Figure 4.18 to be a polytopic partition but not a cell complex. Its associated continuous PWA function is presented in Figure 4.19. It is clear that this polytopic partition is not convexly liftable. One can also confirm this observation via the infeasibility of the optimization problem (4.14). This requires a subdivision into a convexly liftable cell complex. As shown in the proof of Theorem 4.4.10, a hyperplane arrangement offers a possible refinement, however this complete subdivision is not necessary in this case. A partial subdivision into a convexly liftable cell complex can be found in Figure 4.20. One of its convex liftings is shown in Figure 4.21. Further, a continuous PWA function equivalent to the one shown in Figure 4.19, is presented in Figure 4.22, as a sub-component of the optimal solution to the recovered optimization problem.

4.5.2 IPL/QP for continuous PWA functions over polyhedral partitions

Note that Subsection 4.5.1 proposes two algorithms for recovering a given continuous PWA function defined over convexly liftable polytopic partition of a polytope. Accordingly, the constraint set $\Pi_{[x^T\ z\ u^T]^T}$ as shown in (4.22) is defined by the convex hull of the augmented vertices. This construction becomes no longer valid in case of convexly liftable cell complexes of polyhedra, where each component of such a cell complex may possess not only vertices, but also some rays. Therefore, a more general construction of constraint set needs to be introduced. This section focuses on overcoming this limitation.

Recall that the construction of a convex lifting for a cell complex $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ of a polyhedron $\mathcal{X} \subseteq \mathbb{R}^{d_x}$ satisfying Assumption 4.5.2, has been presented in Subsection 4.4.2.2. For ease of presentation, let $f_{pwa}(x) : \mathcal{X} \to \mathbb{R}^{d_u}$ and $\ell(x) : \mathcal{X} \to \mathbb{R}$ denote the given continuous PWA function to be recovered and a convex lifting for cell complex $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$. These functions are defined as follows:

$$\begin{aligned} f_{pwa}(x) &= H_i x + G_i \ \text{ for } \ x \in \mathcal{X}_i \\ \ell(x) &= a_i^T x + b_i \ \text{ for } \ x \in \mathcal{X}_i. \end{aligned} \tag{4.26}$$

Also, the following sets are defined:

$$\begin{aligned} &V_x = \bigcup_{i\in\mathcal{I}_N} \mathcal{V}(\mathcal{X}_i), R_x = \bigcup_{i\in\mathcal{I}_N} \mathcal{R}(\mathcal{X}_i), \\ &V_{[x^T\ z\ u^T]^T} = \left\{ \left[x^T\ \ell(x)\ f_{pwa}^T(x)\right]^T \mid x \in V_x \right\}, \\ &R_{[x^T\ z\ u^T]^T} = \left\{ \begin{bmatrix} r \\ \widehat{\ell}(r) \\ \widehat{f}(r) \end{bmatrix} \mid r \in R_x, \begin{array}{l} \widehat{\ell}(r) = a_i^T r \\ \widehat{f}(r) = H_i r \end{array} \text{ if } \ r \in \mathcal{R}(\mathcal{X}_i) \right\}, \\ &\Pi_v = \text{conv}(V_{[x^T\ z\ u^T]^T}), \Pi_r = \text{cone}(R_{[x^T\ z\ u^T]^T}), \\ &\Pi_{[x^T\ z\ u^T]^T} = \Pi_v \oplus \Pi_r. \end{aligned} \tag{4.27}$$

Note that V_x, R_x represent the set of vertices and extreme rays of the given cell complex $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$, respectively. Similar to the polytopic partition case, $V_{[x^T\ z\ u^T]^T}$

defines the set of augmented vertices induced from the vertices of $\{\mathcal{X}_i\}_{i \in \mathcal{I}_N}$. Also, $R_{[x^T\ z\ u^T]^T}$ defines the set of extreme rays for the constraint set in our recovered optimization problem. According to the Minkowski-Wiley theorem, the set of constraints $\Pi_{[x^T\ z\ u^T]^T}$ is defined as above. With these notations, the following theorem presents a solution to IPL/QP problems via convex liftings for a convexly liftable cell complex of a polyhedron. This result generalizes the one presented in Section 4.5.1.

Theorem 4.5.9 *Given a continuous PWA function $f_{pwa}(x)$, defined as in (4.26) over a polyhedral partition $\{\mathcal{X}_i\}_{i \in \mathcal{I}_N}$ of a polyhedron $\mathcal{X}$ satisfying Assumption 4.5.2 and the sets defined in (4.27), then $f_{pwa}(x)$ is the image via the orthogonal projection onto $\mathbb{R}^{d_u}$ of the optimal solution to the following parametric linear programming problem:*

$$\min_{[z\ u^T]^T} z \quad \textit{subject to} \quad \left[x^T\ z\ u^T\right]^T \in \Pi_{[x^T\ z\ u^T]^T}. \tag{4.28}$$

Proof: Consider $x \in \mathcal{X}_i$, due to the Minkowski-Weyl theorem for polyhedra (c.f. Subsection 2.2.3), x can be described as follows:

$$x = \sum_{v \in \mathcal{V}(\mathcal{X}_i)} \alpha(v)v + \sum_{r \in \mathcal{R}(\mathcal{X}_i)} \beta(r)r,$$

where $\alpha(v), \beta(r) \in \mathbb{R}_+$ and $\sum_{v \in \mathcal{V}(\mathcal{X}_i)} \alpha(v) = 1$. As a consequence, the convex lifting at x, i.e. $\ell(x)$ can be described in the form:

$$\begin{aligned} \ell(x) = a_i^T x + b_i &= a_i^T \Big(\sum_{v \in \mathcal{V}(\mathcal{X}_i)} \alpha(v)v + \sum_{r \in \mathcal{R}(\mathcal{X}_i)} \beta(r)r\Big) + b_i, \\ &= \sum_{v \in \mathcal{V}(\mathcal{X}_i)} \alpha(v)(a_i^T v + b_i) + \sum_{r \in \mathcal{R}(\mathcal{X}_i)} \beta(r)(a_i^T r). \end{aligned}$$

Similarly,

$$f_{pwa}(x) = \sum_{v \in \mathcal{V}(\mathcal{X}_i)} \alpha(v)(H_i v + G_i) + \sum_{r \in \mathcal{R}(\mathcal{X}_i)} \beta(r)(H_i r).$$

It can be observed that if r is an extreme ray of $\mathcal{X}_i$, then $\left[r^T\ a_i^T r\right]^T$ is an extreme ray of the affinely equivalent polyhedron $\Pi_{[x^T\ z]^T}$ of $\{\mathcal{X}_i\}_{i \in \mathcal{I}_N}$, defined as follows:

$$\Pi_{[x^T\ z]^T} = \operatorname{conv}(V_{[x^T\ z]^T}) \oplus \operatorname{cone}(R_{[x^T\ z]^T}),$$

where

$$
\begin{aligned}
V_{[x^T\ z]^T} &= \left\{ \left[x^T\ \ell(x)\right]^T \mid x \in V_x \right\}, \\
R_{[x^T\ z]^T} &= \left\{ \left[r^T\ \widehat{\ell}(r)\right]^T \mid r \in R_x,\ \widehat{\ell}(r) = a_i^T r \ \text{ if } \ r \in \mathcal{R}(\mathcal{X}_i) \right\}.
\end{aligned}
$$

Therefore, for each region $\mathcal{X}_i$, there exists a facet of $\Pi_{[x^T\ z]^T}$, denoted by $F^{(i)}_{[x^T\ z]^T}$, such that:

$$
\begin{aligned}
&\text{Proj}_{\mathbb{R}^{d_x}} F^{(i)}_{[x^T\ z]^T} = \mathcal{X}_i, \\
&\forall \left[x^T\ z(x)\right]^T \in F^{(i)}_{[x^T\ z]^T},\ \ell(x) = z(x) = \min_z z \ \text{ s.t. } \left[x^T\ z\right]^T \in \Pi_{[x^T\ z]^T}.
\end{aligned}
\tag{4.29}
$$

According to Proposition 4.5.4, all augmented points in $V_{[x^T\ z\ u^T]^T}$ are vertices of Π_v. Thus, lifting onto $\mathbb{R}^{d_x+d_u+1}$ leads to the existence of a d_x-face of $\Pi_{[x^T\ z\ u^T]^T}$, denoted by $F^{(i)}_{[x^T\ z\ u^T]^T}$ such that:

$$
\text{Proj}_{[x^T\ z]^T} F^{(i)}_{[x^T\ z\ u^T]^T} = F^{(i)}_{[x^T\ z]^T}. \tag{4.30}
$$

Note that such a d_x-face $F^{(i)}_{[x^T\ z\ u^T]^T}$ is defined as follows:

$$
\begin{aligned}
F^{(i)}_{[x^T\ z\ u^T]^T} &= F_1^{(i)} \oplus F_2^{(i)} \\
F_1^{(i)} &= \text{conv}\left\{ \left[v^T\ \ell(v)\ f^T_{pwa}(v)\right]^T \mid v \in \mathcal{V}(\mathcal{X}_i) \right\} \\
F_2^{(i)} &= \text{cone}\left\{ \left[r^T\ a_i^T r\ (H_i r)^T\right]^T \mid r \in \mathcal{R}(\mathcal{X}_i) \right\}.
\end{aligned}
$$

Due to (4.29) and (4.30), the minimal value of z at a point $x \in \mathcal{X}_i$, is reached if $\left[x^T\ z\ u^T\right]^T$ lies in $F^{(i)}_{[x^T\ z\ u^T]^T}$. Therefore, optimal solution to (4.28) at x can be described by:

$$
\begin{bmatrix} x \\ z^*(x) \\ u^*(x) \end{bmatrix} = \sum_{v \in \mathcal{V}(\mathcal{X}_i)} \alpha(v) \begin{bmatrix} v \\ a_i^T v + b_i \\ H_i v + G_i \end{bmatrix} + \sum_{r \in \mathcal{R}(\mathcal{X}_i)} \beta(r) \begin{bmatrix} r \\ a_i^T r \\ H_i r \end{bmatrix},
$$

where $\alpha(v), \beta(r) \in \mathbb{R}_+$ and $\sum_{v \in \mathcal{V}(\mathcal{X}_i)} \alpha(v) = 1$. It follows that

$$
\begin{bmatrix} z^*(x) \\ u^*(x) \end{bmatrix} = \begin{bmatrix} a_i^T x + b_i \\ H_i x + G_i \end{bmatrix} = \begin{bmatrix} \ell(x) \\ f_{pwa}(x) \end{bmatrix}, \quad \text{for } x \in \mathcal{X}_i.
$$

To complete the proof, it is necessary to show that the optimal solution to (4.28) is unique. In fact, at a point $x \in \mathcal{X}_i$, suppose there exist two different optimal solutions to (4.28) i.e.

$$\begin{aligned} \left[z_1(x)\ u_1^T(x)\right]^T &= \arg \min_{[z\ u^T]^T} z \quad \text{subject to} \quad \left[x^T\ z\ u^T\right]^T \in \Pi_{[x^T\ z\ u^T]^T} \\ \left[z_2(x)\ u_2^T(x)\right]^T &= \arg \min_{[z\ u^T]^T} z \quad \text{subject to} \quad \left[x^T\ z\ u^T\right]^T \in \Pi_{[x^T\ z\ u^T]^T} \end{aligned}$$

Accordingly, there exist two different d_x-faces, denoted by F_1, F_2, such that $\left[x^T\ z_1(x)\ \ u_1^T(x)\right]^T \in F_1$ and $\left[x^T\ z_2(x)\ \ u_2^T(x)\right]^T \in F_2$. It can be observed that $z_1(x) = z_2(x) = \ell(x)$, leading to

$$\text{Proj}_{[x^T\ z]^T} F_1 = \text{Proj}_{[x^T\ z]^T} F_2 = F^{(i)}_{[x^T\ z]^T}.$$

If $u_1(x) \neq u_2(x)$, then F_1, F_2 lie in a hyperplane of dimension $d_x + 1$ which is orthogonal to the space of $\left[x^T\ z\right]^T$ (again, an illustration can be found in Figure 4.14). Accordingly, $f_{pwa}(v)$ or $\widehat{f}(r)$ in (4.27) is not uniquely defined for some vertices $v \in \mathcal{V}(\mathcal{X}_i)$ or some extreme rays $r \in \mathcal{R}(\mathcal{X}_i)$. This end contradicts the construction of $\Pi_{[x^T\ z\ u^T]^T}$ in (4.27). Therefore, $F_1 = F_2$ leading to the uniqueness of the optimal solution to (4.28). □

Similarly, an equivalent optimization problem with respect to a quadratic cost function is stated in the following theorem.

Theorem 4.5.10 *Given a continuous PWA function $f_{pwa}(x)$, defined as in (4.26) over a polyhedral partition $\{\mathcal{X}_i\}_{i \in \mathcal{I}_N}$ of a polyhedron $\mathcal{X}$ satisfying Assumption 4.5.2 and the sets defined in (4.27), then $f_{pwa}(x)$ is the image via the orthogonal projection of the optimal solution to the following parametric quadratic programming problem:*

$$\min_{[z\ u^T]^T} (z - \sigma(x))^2 \quad \textit{subject to} \quad \left[x^T\ z\ u^T\right]^T \in \Pi_{[x^T\ z\ u^T]^T}, \tag{4.31}$$

where $\sigma(x) : \mathcal{X} \rightarrow \mathbb{R}$ denotes any function satisfying $\sigma(x) \leq \ell(x)$ and $\ell(x)$ is defined in (4.26).

Proof: The proof follows the same line of the proof for Theorem 4.5.7. □

For illustration, consider a continuous PWA function defined in (4.32). This function is defined over a cell complex covering the whole space $\mathbb{R}^2$ and is shown in Figure 4.24. A box $B_2(2.2987)$ satisfying Assumption 4.4.4, is resulted from Algorithm 4.2 with $a = 0.5$. The cell complex $\{\mathcal{X}_i \cap B_2(2.2987)\}_{i \in \mathcal{I}_{11}}$, is shown in Figure 4.23. A convex lifting for the cell complex $\{\mathcal{X}_i\}_{i \in \mathcal{I}_{11}}$ is defined in (4.34) and is illustrated in Figure 4.25. Finally, the set of constraints for a recovered optimization problem, is presented in (4.33).

$$f(x) = \begin{cases} \begin{bmatrix}-0.8359 & -0.1682\end{bmatrix} x - 1.0041 \text{ for } \begin{bmatrix}0.9426 & -0.3339\\ -0.9732 & 0.2298\\ -0.9757 & -0.2192\end{bmatrix} x \leq \begin{bmatrix}-0.6087\\ 0.7435\\ 0.5049\end{bmatrix} \\ \begin{bmatrix}-1.3107 & 0\end{bmatrix} x - 1.3107 \text{ for } \begin{bmatrix}-0.8740 & -0.4859\\ -0.9426 & 0.3339\end{bmatrix} x \leq \begin{bmatrix}0.3881\\ 0.6087\end{bmatrix} \\ \begin{bmatrix}-0.5172 & -0.2434\end{bmatrix} x - 0.7606 \text{ for } \begin{bmatrix}0.9732 & -0.2298\\ -0.9537 & -0.3006\\ -0.9827 & 0.1854\end{bmatrix} x \leq \begin{bmatrix}-0.7435\\ 0.4473\\ 0.7973\end{bmatrix} \\ \begin{bmatrix}-0.0873 & 0\end{bmatrix} x - 0.6167 \text{ for } \begin{bmatrix}-0.9156 & -0.4022\\ -1 & 0\\ 0.9757 & 0.2192\end{bmatrix} x \leq \begin{bmatrix}0.4302\\ 0.6376\\ -0.5049\end{bmatrix} \\ \begin{bmatrix}-0.6440 & 0.3706\end{bmatrix} x - 1.0147 \text{ for } \begin{bmatrix}-0.9983 & 0.0587\\ -0.9324 & -0.3615\\ 0.8740 & 0.4859\end{bmatrix} x \leq \begin{bmatrix}0.5793\\ 0.5709\\ -0.3881\end{bmatrix} \\ \begin{bmatrix}0.3793 & 0.0392\end{bmatrix} x - 0.3402 \text{ for } \begin{bmatrix}-0.6839 & -0.7296\\ 0.9537 & 0.3006\\ 0.7063 & 0.7079\end{bmatrix} x \leq \begin{bmatrix}0.0457\\ -0.4473\\ -0.0016\end{bmatrix} \\ \begin{bmatrix}0 & -0.3410\end{bmatrix} x - 0.3410 \text{ for } \begin{bmatrix}-0.7063 & -0.7079\\ 0.9827 & -0.1854\end{bmatrix} x \leq \begin{bmatrix}0.0016\\ -0.7973\end{bmatrix} \\ \begin{bmatrix}0.2815 & 0.1620\end{bmatrix} x - 0.4435 \text{ for } \begin{bmatrix}0.4448 & -0.8956\\ -0.3531 & 0.9356\\ 0.9156 & 0.4022\end{bmatrix} x \leq \begin{bmatrix}-0.4508\\ 0.5824\\ -0.4302\end{bmatrix} \\ \begin{bmatrix}0.3426 & 0\end{bmatrix} x - 0.3426 \text{ for } \begin{bmatrix}0.3531 & -0.9356\\ 0.6839 & 0.7296\\ 1 & 0\end{bmatrix} x \leq \begin{bmatrix}-0.5824\\ -0.0457\\ -0.6376\end{bmatrix} \\ \begin{bmatrix}0.2026 & 0.3208\end{bmatrix} x - 0.5234 \text{ for } \begin{bmatrix}0.9983 & -0.0587\\ 0.5603 & -0.8283\\ -0.4448 & 0.8956\end{bmatrix} x \leq \begin{bmatrix}-0.5793\\ -0.2680\\ 0.4508\end{bmatrix} \\ \begin{bmatrix}0 & 0.6203\end{bmatrix} x - 0.6203 \text{ for } \begin{bmatrix}0.9324 & 0.3615\\ -0.5603 & 0.8283\end{bmatrix} x \leq \begin{bmatrix}-0.5709\\ 0.2680\end{bmatrix} \end{cases} \tag{4.32}$$

$$\Pi = \left\{ \begin{bmatrix}x\\ z\\ u\end{bmatrix} \in \mathbb{R}^4 \;\middle|\; \begin{bmatrix} -6.9688 & -1 & -7.9494 & -23.5314\\ 12.8738 & 1 & 7.9494 & 91.2072\\ 5.7402 & 1 & 1.4676 & 13.7934\\ 4.2240 & 1 & -1.4199 & -1.6482\\ 4.3714 & -1 & 1.4199 & 8.2062\\ -1 & 2.2658 & -2.3596 & -6.2654\\ -87.0107 & -1 & -7.9494 & 199.3996\\ 2.8453 & -1 & 3.3575 & 16.6271\\ -3.6256 & -1 & -2.8916 & -8.5186\\ 7.2247 & -1 & 11.9250 & 63.5133\\ 5.8210 & 1 & 13.7351 & 78.0817\\ 4.1145 & -1 & 5.6669 & 29.0205\\ 1.0039 & -1 & 3.6478 & 23.7135\\ -4.3377 & -1 & -3.4595 & -10.4611\\ -9.8719 & -1 & -7.9494 & -25.7464\\ 1.9672 & -1 & 1.5689 & 7.1088 \end{bmatrix} \begin{bmatrix}x\\ z\\ u\end{bmatrix} \leq \begin{bmatrix} 15.7214\\ -57.4584\\ -10.6078\\ -1.4820\\ -7.1134\\ 4.9525\\ -58.3234\\ -10.6934\\ 6.1731\\ -37.6509\\ -43.0178\\ -18.0489\\ -13.6189\\ 7.5819\\ 18.8153\\ -4.8920 \end{bmatrix} \right\} \tag{4.33}$$

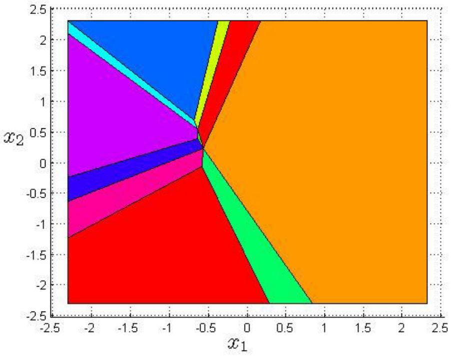


Figure 4.23: The partition obtained via the intersection $\{\mathcal{X}_i \cap B_2(2.2987)\}_{i \in \mathcal{I}_{11}}$.

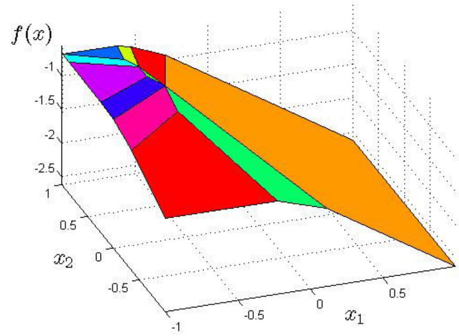


Figure 4.24: The continuous PWA function defined in (4.32) to be recovered.

$$\ell(x) = \begin{cases} \begin{bmatrix} 3.9451 & 0.8995 \end{bmatrix} x + 2.2092 & \text{for } \begin{bmatrix} 0.9426 & -0.3339 \\ -0.9732 & 0.2298 \\ -0.9757 & -0.2192 \end{bmatrix} x \leq \begin{bmatrix} -0.6087 \\ 0.7435 \\ 0.5049 \end{bmatrix} \\ \begin{bmatrix} 4.4962 & 0.7043 \end{bmatrix} x + 2.5651 & \text{for } \begin{bmatrix} -0.8740 & -0.4859 \\ -0.9426 & 0.3339 \end{bmatrix} x \leq \begin{bmatrix} 0.3881 \\ 0.6087 \end{bmatrix} \\ \begin{bmatrix} 0.9494 & 1.6067 \end{bmatrix} x - 0.0792 & \text{for } \begin{bmatrix} 0.9732 & -0.2298 \\ -0.9537 & -0.3006 \\ -0.9827 & 0.1854 \end{bmatrix} x \leq \begin{bmatrix} -0.7435 \\ 0.4473 \\ 0.7973 \end{bmatrix} \\ \begin{bmatrix} -0.6184 & -0.1258 \end{bmatrix} x - 0.1521 & \text{for } \begin{bmatrix} -0.9156 & -0.4022 \\ -1 & 0 \\ 0.9757 & 0.2192 \end{bmatrix} x \leq \begin{bmatrix} 0.4302 \\ 0.6376 \\ -0.5049 \end{bmatrix} \\ \begin{bmatrix} 0.6435 & -1.4377 \end{bmatrix} x + 0.8543 & \text{for } \begin{bmatrix} -0.9983 & 0.0587 \\ -0.9324 & -0.3615 \\ 0.8740 & 0.4859 \end{bmatrix} x \leq \begin{bmatrix} 0.5793 \\ 0.5709 \\ -0.3881 \end{bmatrix} \\ \begin{bmatrix} -1.4310 & 0.8563 \end{bmatrix} x - 1.1957 & \text{for } \begin{bmatrix} -0.6839 & -0.7296 \\ 0.9537 & 0.3006 \\ 0.7063 & 0.7079 \end{bmatrix} x \leq \begin{bmatrix} 0.0457 \\ -0.4473 \\ -0.0016 \end{bmatrix} \\ \begin{bmatrix} -0.4238 & 1.8657 \end{bmatrix} x - 1.1934 & \text{for } \begin{bmatrix} -0.7063 & -0.7079 \\ 0.9827 & -0.1854 \end{bmatrix} x \leq \begin{bmatrix} 0.0016 \\ -0.7973 \end{bmatrix} \\ \begin{bmatrix} -2.1050 & -0.7789 \end{bmatrix} x - 0.8507 & \text{for } \begin{bmatrix} 0.4448 & -0.8956 \\ -0.3531 & 0.9356 \\ 0.9156 & 0.4022 \end{bmatrix} x \leq \begin{bmatrix} -0.4508 \\ 0.5824 \\ -0.4302 \end{bmatrix} \\ \begin{bmatrix} -2.3515 & -0.1258 \end{bmatrix} x - 1.2572 & \text{for } \begin{bmatrix} 0.3531 & -0.9356 \\ 0.6839 & 0.7296 \\ 1 & 0 \end{bmatrix} x \leq \begin{bmatrix} -0.5824 \\ -0.0457 \\ -0.6376 \end{bmatrix} \\ \begin{bmatrix} -1.8506 & -1.2910 \end{bmatrix} x - 0.5929 & \text{for } \begin{bmatrix} 0.9983 & -0.0587 \\ 0.5603 & -0.8283 \\ -0.4448 & 0.8956 \end{bmatrix} x \leq \begin{bmatrix} -0.5793 \\ -0.2680 \\ 0.4508 \end{bmatrix} \\ \begin{bmatrix} -1.2539 & -2.1732 \end{bmatrix} x - 0.3075 & \text{for } \begin{bmatrix} 0.9324 & 0.3615 \\ -0.5603 & 0.8283 \end{bmatrix} x \leq \begin{bmatrix} -0.5709 \\ 0.2680 \end{bmatrix} \end{cases} \tag{4.34}$$

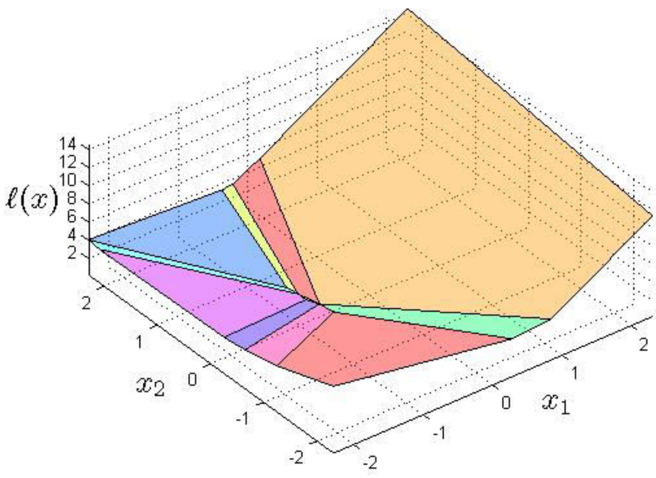


Figure 4.25: A convex lifting for the given cell complex $\{\mathcal{X}_i\}_{i\in\mathcal{I}_{11}}$.

4.5.3 Invertibility and complexity of IPL/QP via convex liftings

This subsection focuses on the important properties of the solution to IPL/QP problems via convex liftings, i.e. the *invertibility* and the *complexity* of the above constructive inverse optimality procedures.

Theorem 4.5.11 *(Invertibility) Given a polyhedral partition $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ of a polyhedron $\mathcal{X} \subseteq \mathbb{R}^{d_x}$, then any continuous piecewise affine function $f_{pwa}(x) : \mathcal{X} \to \mathbb{R}^{d_u}$, defined over $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$, is invertible.*

Proof: If $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ admits an affinely equivalent polyhedron, then Theorems 4.5.9 and 4.5.10 show a formulation of such an inverse parametric linear/quadratic programming problem. In case the convex liftability of $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ is not fulfilled, according to Theorem 4.4.10, $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ can be subdivided into a convexly liftable cell complex such that the internal boundaries are maintained. $f_{pwa}(x)$ is replaced with an equivalent PWA function corresponding to this new convexly liftable cell complex. With these pre-conditionings, the problem is recast to recover a continuous PWA function defined over a convexly liftable cell complex. □

The complexity of an inverse parametric linear/quadratic programming problem based on convex liftings is also stated as follows:

Theorem 4.5.12 *(Complexity) Any continuous PWA function defined over a polyhedral partition of a polyhedron can be equivalently obtained by a parametric* *linear/quadratic programming problem with at most one auxiliary $1-$dimensional variable.*

Proof: Let $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ denote this given polyhedral partition of a polyhedron $\mathcal{X}$. If $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ is convexly liftable, this $1-$dimensional variable describes the convex lifting in the recovered optimization problem. Theorems 4.5.9, 4.5.10 show that this PWA function is invertible through the convex lifting based approach.

Otherwise, in case the given partition is not convexly liftable, Theorem 4.4.10 shows that there exists at least one way to subdivide the given non-convexly liftable polyhedral partition into a convexly liftable cell complex, denoted by $\left\{\widetilde{\mathcal{X}}_i\right\}_{i\in\mathcal{I}_{\widetilde{N}}}$, meanwhile the internal boundaries are maintained. According to this subdivision, the given PWA function $f_{pwa}(x)$ is also subdivided. This new PWA function, say $\widetilde{f}_{pwa}(x)$, is equivalent to $f_{pwa}(x)$ and defined over a convexly liftable cell complex $\left\{\widetilde{\mathcal{X}}_i\right\}_{i\in\mathcal{I}_{\widetilde{N}}}$. Therefore, similar to the first case, a convex lifting of $\left\{\widetilde{\mathcal{X}}_i\right\}_{i\in\mathcal{I}_{\widetilde{N}}}$, represents the $1-$dimensional auxiliary variable. Also, as proved in Theorem 4.5.9/4.5.10, $\widetilde{f}_{pwa}(x)$, associated with $\left\{\widetilde{\mathcal{X}}_i\right\}_{i\in\mathcal{I}_{\widetilde{N}}}$, is invertible via the convex lifting based method. $\square$

Remark 4.5.13 Theorem 4.5.12 clarifies that if there exists an $1-$dimensional component of the given continuous piecewise affine function able to serve as a convex lifting, it is not necessary to construct an auxiliary variable. An affinely equivalent polyhedron $\Pi_{[x^T\ z]^T}$ of $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ is established as in (4.17), where z represents this 1-dimensional component of the given continuous piecewise affine function. Recall that in case this component denoted by $f(x)$, is concave and any pair of neighboring regions are associated with different affine functions, then $-f(x)$ is a convex lifting defined over the same cell complex.

Remark 4.5.14 It can be observed that the proposed method can be theoretically applied for PWA functions, defined on any finite dimensional space. However, numerical limitations can make the construction impractical. It is known that explicit solution turns out to be efficient for small dimensional problems e.g. $d_x = 2, 3, 4$. Therefore, the tractability of this method lies in the tractability of the parametric solvers. Moreover, as shown before, the presented approach relies on the construction of a convex lifting. This construction is the essential step in the proposed method. Hence, the tractability of this construction decides the tractability of the inverse optimality problem via convex liftings. As for the complexity of this construction (Algorithm 4.1), by considering two neighboring regions, the number of constraints (including equality and inequality constraints) is equal to the number of vertices of the first region. If N denotes the number of regions in the given cell complex $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ and v_{max} denotes the maximal number of vertices among the regions in this cell complex, then an upper bound for the number of constraints is equal to $N(N-1)v_{max}/2$. Thus this number of constraints scales quadratically with the number of regions in the given partition.

4.5.4 IPL/QP for discontinuous PWA functions

Note that the solution to IPL/QP problem via convex liftings presented in Subsections 4.5.1, 4.5.2 only focuses on the *continuous PWA functions*. As emphasized, the class of *discontinuous PWA functions* is omitted in those subsections. This subsection aims to compensate the lack of interest for the *discontinuous PWA functions* by investigating inverse optimality. Note also that due to the continuity of the given PWA function, the optimal solution to inverse optimality formulation is unique. However, the uniqueness may no longer be preserved for discontinuous PWA functions. Therefore, finding an optimization problem, whose optimal solution is unique and equivalent to the given discontinuous PWA function, may render this optimization problem nonlinear (non-convex) from the point of view of the ingredients i.e. the cost function and the constraint set.

Motivated by the above discussion, efforts will be made to show how to construct a convex optimization problem that has a given discontinuous PWA function as optimal solution at the price of non-unicity. More precisely, a PWA function equivalent to the given discontinuous PWA function, will be shown to be an optimal solution to an optimization problem. This optimization problem will be shown to be of parametric linear/quadratic programming type.

For ease of presentation, the definition of a discontinuous PWA function $f_{pwa}(x)$ to be recovered is recalled below:

$$f_{pwa}(x) = \begin{cases} H_i x + G_i \text{ for } x \in \operatorname{int}(\mathcal{X}_i), \\ H_i x + G_i \text{ or } H_j x + G_j \text{ for } x \in \mathcal{X}_i \cap \mathcal{X}_j, \end{cases} \tag{4.35}$$

for any pair of neighboring regions $(\mathcal{X}_i, \mathcal{X}_j)$.

Taking the discontinuity into account, the problem formulation should be adjusted. Namely, consider a given cell complex $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ of a polyhedron $\mathcal{X} \subseteq \mathbb{R}^{d_x}$ satisfying Assumption 4.5.2 and a possibly discontinuous PWA function $f_{pwa}(x)$: $\mathcal{X} \to \mathbb{R}^{d_u}$ defined over this cell complex as in (4.35). The goal is to determine $J(x,z,u), H_x, H_u, H_z, K$ such that:

$$\begin{cases} f_{pwa}(x) \in \operatorname{Proj}_{\mathbb{R}^{d_u}} \arg \min\limits_{[z\ u^T]^T} J(x,z,u), \\ \qquad \text{s.t:} \quad H_x x + H_z z + H_u u \leq K. \end{cases} \tag{4.36}$$

For ease of presentation, a convex lifting denoted by $\ell(x)$ for the cell complex $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ is employed here:

$$\ell(x) = a_i^T x + b_i \ \text{ for } \ x \in \mathcal{X}_i. \tag{4.37}$$

Also, $f_{pwa}^{(i)}$ is of use to denote the i^{th} component of the given PWA function at x. According to this notation, define also the following sets and values:

$$\begin{aligned} V_x &= \bigcup_{i\in\mathcal{I}_N} \mathcal{V}(\mathcal{X}_i), \ R_x = \bigcup_{i\in\mathcal{I}_N} \mathcal{R}(\mathcal{X}_i), \\ \overline{f}_{pwa}^{(j)}(x) &= \max_{i\in\mathcal{I}_N|x\in\mathcal{X}_i} H_i(j,\cdot)x + G_i(j), \\ \underline{f}_{pwa}^{(j)}(x) &= \min_{i\in\mathcal{I}_N|x\in\mathcal{X}_i} H_i(j,\cdot)x + G_i(j). \end{aligned} \tag{4.38}$$

Note that $\overline{f}_{pwa}^{(j)}(x)$ $\left(\underline{f}_{pwa}^{(j)}(x)\right)$ is defined as the maximal (minimal) value among the values of the j^{th} component of the affine functions composing $f_{pwa}(x)$ at x, defined over the regions which contain x.

For ease of presentation, define also the following sets

$$\begin{aligned} \mathbf{U}(x) &= \left\{ \begin{bmatrix} u^{(1)}(x) \\ \vdots \\ u^{(d_u)}(x) \end{bmatrix} \middle| \ u^{(j)}(x) \in \left\{ \overline{f}_{pwa}^{(j)}(x), \underline{f}_{pwa}^{(j)}(x) \right\}, j \in \mathcal{I}_{d_u} \right\}, \\ V_{[x^T\ z\ u^T]^T} &= \left\{ \begin{bmatrix} x \\ \ell(x) \\ u(x) \end{bmatrix} \middle| \ x \in V_x, \ u(x) \in \mathbf{U}(x) \right\}, \\ R_{[x^T\ z\ u^T]^T} &= \left\{ \begin{bmatrix} r \\ \widehat{\ell}(r) \\ \widehat{h}(r) \end{bmatrix} \middle| \ r \in R_x, \ \begin{matrix} \widehat{\ell}(r) = a_i^T r \\ \widehat{h}(r) = H_i r \end{matrix} \ \text{ if } \ r \in \mathcal{R}(\mathcal{X}_i) \right\}, \\ \Pi_v &= \operatorname{conv}(V_{[x^T\ z\ u^T]^T}), \ \Pi_r = \operatorname{cone}(R_{[x^T\ z\ u^T]^T}), \\ \Pi_{[x^T\ z\ u^T]^T} &= \Pi_v \oplus \Pi_r. \end{aligned} \tag{4.39}$$

Note that $V_{[x^T\ z\ u^T]^T}$ $\left(R_{[x^T\ z\ u^T]^T}\right)$ represents the set of extended vertices (extreme rays) of the partition $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ in the augmented space. Note that each vertex x of this cell complex may correspond to different augmented points in $V_{[x^T\ z\ u^T]^T}$, since all the values of $\mathbf{U}(x)$ are appended. Also, if for a given $x \in \mathcal{X}$, we have $\underline{f}_{pwa}^{(j)}(x) = \overline{f}_{pwa}^{(j)}(x)$ for every $j \in \mathcal{I}_{d_u}$, then the function $f_{pwa}(x)$ is continuous at x. In this case, $\operatorname{Card}(\mathbf{U}(x)) = 1$. The following observation is useful for the next development.

Proposition 4.5.15 *For any $x \in \mathcal{X}$ and $\mathbf{U}(x)$ defined in (4.39), the following holds:* $\mathbf{U}(x) = \mathcal{V}(\mathrm{conv}(\mathbf{U}(x)))$.

Proof: According to the construction of $\mathbf{U}(x)$, one can see that $\mathcal{V}(\mathrm{conv}(\mathbf{U}(x))) \subseteq \mathbf{U}(x)$. We will prove that $\mathcal{V}(\mathrm{conv}(\mathbf{U}(x))) \subset \mathbf{U}(x)$ leads to a contradiction. Indeed, if $\mathcal{V}(\mathrm{conv}(\mathbf{U}(x))) \subset \mathbf{U}(x)$, then there exists a point $u_0(x) \in \mathbf{U}(x)$ for a given $x \in \mathcal{X}$ such that $u_0(x)$ can be described via a convex combination of the other points $u(x) \in \mathbf{U}(x), u(x) \neq u_0(x)$, i.e.

$$\begin{aligned} & u_0(x) = \sum_{u(x)\neq u_0(x),\, u(x)\in\mathbf{U}(x)} \alpha(u(x))u(x), \\ & \alpha(u(x)) \geq 0, \sum_{u(x)\neq u_0(x),\, u(x)\in\mathbf{U}(x)} \alpha(u(x)) = 1, \end{aligned} \tag{4.40}$$

Note however that $u_0^{(1)}(x)$ takes value in the discrete set $\left\{\overline{f}_{pwa}^{(1)}(x), \underline{f}_{pwa}^{(1)}(x)\right\}$, and any of these two values cannot be described by a convex combination of the other one if these two values are different. Thus, (4.40) holds true if $u^{(1)}(x) = u_0^{(1)}(x)$, meaning:

$$\begin{aligned} & \sum_{u(x)\in\mathbb{U}(x), u(x)\neq u_0(x), u^{(1)}(x)=u_0^{(1)}(x)} \alpha(u(x)) = 1, \\ & \alpha(u(x)) = 0 \text{ for } u^{(1)}(x) \neq u_0^{(1)}(x). \end{aligned}$$

The same argument, applied for the other components $u_0^{(j)}(x),\ j \in \mathcal{I}_{d_u}$, leads to:

$$\begin{aligned} \alpha(u(x)) &= 1, \text{ for } u(x) = u_0(x), \\ \alpha(u(x)) &= 0, \text{ for } u(x) \neq u_0(x). \end{aligned}$$

This inclusion is clearly contradictory with (4.40). $\square$

Proposition 4.5.16 $V_{[x^T\ z\ u^T]^T} = \mathcal{V}(\Pi_v)$, *where* $V_{[x^T\ z\ u^T]^T}, \Pi_v$ *are defined in* (4.39).

Proof: Suppose there exists a point of $V_{[x^T\ z\ u^T]^T}$, denoted by $\left[x^T\ \ell(x)\ u_0^T(x)\right]^T$, which can be described via a convex combination of the other points in this set. Formally, this leads to the following relationship:

$$\begin{bmatrix} x \\ \ell(x) \\ u_0(x) \end{bmatrix} = \sum_{\substack{\left[v^T\ \ell(v)\ u^T(v)\right] \neq \left[x^T\ \ell(x)\ u_0^T(x)\right] \\ \left[v^T\ \ell(v)\ u^T(v)\right]^T \in V_{[x^T\ z\ u^T]^T}}} \alpha\left(\begin{bmatrix} v \\ \ell(v) \\ u(v) \end{bmatrix}\right) \begin{bmatrix} v \\ \ell(v) \\ u(v) \end{bmatrix}, \tag{4.41}$$

where

$$\alpha\left(\begin{bmatrix} v \\ \ell(v) \\ u(v) \end{bmatrix}\right) \geq 0, \quad \sum_{\substack{\left[v^T\,\ell(v)\,u^T(v)\right] \neq \left[x^T\,\ell(x)\,u_0^T(x)\right] \\ \left[v^T\,\ell(v)\,u^T(v)\right]^T \in V_{[x^T\,z\,u^T]^T}}} \alpha\left(\begin{bmatrix} v \\ \ell(v) \\ u(v) \end{bmatrix}\right) = 1.$$

Due to the fact that any $\left[x^T\,\ell(x)\right]^T, x \in V_x$, cannot be described via a convex combination of the other $\left[v^T\,\ell(v)\right]^T, v \neq x, v \in V_x$, then relation (4.41) holds true only for $v = x$, leading to $\alpha\left(\begin{bmatrix} v \\ \ell(v) \\ u(v) \end{bmatrix}\right) = 0$ for $v \neq x$. The remaining coefficients only depend on $u(x)$, therefore for simplicity, the remaining coefficients are denoted by $\alpha(u(x))$ instead of $\alpha\left(\begin{bmatrix} x \\ \ell(x) \\ u(x) \end{bmatrix}\right)$.

To complete the proof, one needs to show that

$$\begin{aligned} u_0(x) &= \sum_{u(x)\neq u_0(x),\, u(x)\in\mathbf{U}(x)} \alpha(u(x))u(x), \\ \alpha(u(x)) &\geq 0, \sum_{u(x)\neq u_0(x),\, u(x)\in\mathbf{U}(x)} \alpha(u(x)) = 1, \end{aligned} \tag{4.42}$$

leads to a contradiction. This end is easily deduced from Proposition 4.5.15 and the proof is complete. □

Based on the above inclusions, the main result of this subsection is presented below.

Theorem 4.5.17 *Given a possibly discontinuous PWA function $f_{pwa}(x)$ (4.35) defined over a polyhedral partition of a polyhedron satisfying Assumption 4.5.2 and the sets defined in (4.39), $f_{pwa}(x)$ is the image via the orthogonal projection onto $\mathbb{R}^{d_u}$ of an optimal solution to the following optimization problem:*

$$\min_{[z\ u^T]^T} z \quad \text{s.t.} \quad \left[x^T\ z\ u^T\right]^T \in \Pi_{[x^T\ z\ u^T]^T}. \tag{4.43}$$

Proof: Given a point $x \in \mathcal{X}_i$ in the cell complex $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$, due to the Minkowski-Weyl theorem (c.f. Subsection 2.2.3), x can be written in the following form:

$$x = \sum_{v\in\mathcal{V}(\mathcal{X}_i)} \alpha(v)v + \sum_{r\in\mathcal{R}(\mathcal{X}_i)} \beta(r)r,$$

where $\alpha(v), \beta(r) \geq 0$, $\sum_{v \in \mathcal{V}(\mathcal{X}_i)} \alpha(v) = 1$. Due to Assumption 4.5.2, let $\ell(x)$ denote a convex lifting for the cell complex $\{\mathcal{X}_i\}_{i \in \mathcal{I}_N}$, then $\ell(x) = a_i^T x + b_i$ for $x \in \mathcal{X}_i$. For an $x \in \mathcal{X}_i$, it follows that:

$$\ell(x) = a_i^T x + b_i = \sum_{v \in \mathcal{V}(\mathcal{X}_i)} \alpha(v)(a_i^T v + b_i) + \sum_{r \in \mathcal{R}(\mathcal{X}_i)} \beta(r)\,(a_i^T r).$$

Thus, if one defines the following sets:

$$\begin{aligned}
V_{[x^T\ z]^T} &= \left\{ \left[x^T\ \ell(x)\right]^T \mid x \in V_x \right\}, \\
R_{[x^T\ z]^T} &= \left\{ \left[r^T\ \widehat{\ell}(r)\right]^T \mid r \in R_x,\ \widehat{\ell}(r) = a_i^T r \ \text{ if } \ r \in \mathcal{R}(\mathcal{X}_i) \right\},
\end{aligned}$$

then $\Pi_{[x^T\ z]^T} = \text{conv}(V_{[x^T\ z]^T}) \oplus \text{cone}(R_{[x^T\ z]^T})$ is an affinely equivalent polyhedron of the cell complex $\{\mathcal{X}_i\}_{i \in \mathcal{I}_N}$.

Following the definition of an affinely equivalent polyhedron, for a region $\mathcal{X}_i$ in the given cell complex $\{\mathcal{X}_i\}_{i \in \mathcal{I}_N}$, there exists a lower facet of $\Pi_{[x^T\ z]^T}$, denoted by $F_{[x^T\ z]^T}$ such that $\text{Proj}_{\mathbb{R}^{d_x}} F_{[x^T\ z]^T} = \mathcal{X}_i$. Also, the optimal solution to the following optimization problem

$$\min_z z \ \text{ subject to } \ \left[x^T\ z\right]^T \in \Pi_{[x^T\ z]^T},$$

falls in $F_{[x^T\ z]^T}$ for all $x \in \mathcal{X}_i$.

Note however that, due to the construction in (4.39), every d_x-face, denoted by $F^{\#}$ and defined as follows:

$$\begin{aligned}
F^{\#} &= F_1^{\#} \oplus F_2^{\#}, \\
F_1^{\#} &= \text{conv}\left\{ \left[v^T\ \ell(v)\ u^T(v)\right]^T \mid v \in \mathcal{V}(\mathcal{X}_i) \right\}, \\
F_2^{\#} &= \text{cone}\left\{ \left[r^T\ a_i^T r\ (H_i r)^T\right]^T \mid r \in \mathcal{R}(\mathcal{X}_i) \right\}, \\
u(v) &\in \text{conv}(\mathbf{U}(v)) \ \text{ for each } \ v \in \mathcal{V}(\mathcal{X}_i),
\end{aligned} \tag{4.44}$$

satisfies: $\text{Proj}_{[x^T\ z]^T} F^{\#} = F_{[x^T\ z]^T}$. Accordingly, consider a point $x \in \mathcal{X}_i$, optimizer of the optimization problem (4.43) at x, may be located on such a d_x-face $F^{\#}$. Due to this non-uniqueness, one can choose the following d_x-face, denoted by $\widetilde{F}$, defined as in (4.44) with $u(v) = H_i v + G_i$ for $v \in \mathcal{X}_i$. Then, for any $x \in \mathcal{X}_i$, the optimizer, located on $\widetilde{F}$, satisfies:

$$\begin{bmatrix} x \\ z^*(x) \\ u^*(x) \end{bmatrix} = \sum_{v \in \mathcal{V}(\mathcal{X}_i)} \alpha(v) \begin{bmatrix} v \\ \ell(v) \\ H_i v + G_i \end{bmatrix} + \sum_{r \in \mathcal{R}(\mathcal{X}_i)} \beta(r) \begin{bmatrix} r \\ a_i^T r \\ H_i r \end{bmatrix},$$

meaning $u^*(x) = H_i x + G_i$. □

Remark 4.5.18 Note that one can easily fix a large enough box in $\mathbb{R}^{d_u}$ to bound all $\mathbf{U}(v)$, $\forall v \in V_x$. Such a constraint can avoid the computation of $\mathbf{U}(v)$ at each vertex of the cell complex $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$. However, such a handy choice ignores the structure of the given PWA function and cannot exploit its continuity property whenever this exists.

Similar to the above construction, a simple extension from parametric linear programming to parametric quadratic programming is introduced below.

Theorem 4.5.19 *Given a possibly discontinuous PWA function* (4.35) *defined over a polyhedral partition* $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ *of a polyhedron* $\mathcal{X}$*, satisfying Assumption 4.5.2,* $\ell(x)$ *defined in* (4.37)*, denotes a convex lifting for* $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$*. Let* $\sigma(x)$ *be a function such that* $\sigma(x) \leq \ell(x)$ *for* $x \in \mathcal{X}$*. The given PWA function* (4.35) *is the image via the orthogonal projection onto* $\mathbb{R}^{d_u}$ *of an optimal solution to the following parametric quadratic programming problem:*

$$\min_{[z\ u^T]^T} (z - \sigma(x))^2 \ \ \textit{s.t.} \ \ \left[x^T\ z\ u^T\right]^T \in \Pi_{[x^T\ z\ u^T]^T}, \tag{4.45}$$

where $\Pi_{[x^T\ z\ u^T]^T}$ *is defined as in* (4.39).

Proof: The proof is similar to the one of Theorem 4.5.7. □

The following theorem presents the main result of this subsection and is also of importance in the context of PWA functions analysis.

Theorem 4.5.20 *Any possibly discontinuous PWA function, defined over a polyhedral partition of a polyhedron, can be equivalently obtained as a selection among the optimal solutions of a parametric linear/quadratic programming problem.*

Proof: Let $f_{pwa}(x)$ denote a given PWA function, defined over a polyhedral partition $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$. If this partition is convexly liftable, Theorems 4.5.17, 4.5.19 show in a constructive manner, parametric linear/quadratic programming problems for which $f_{pwa}(x)$ is a sub-component of one of their optimal solutions.

Otherwise, in case this partition is not convexly liftable, according to Theorem 4.4.10, one can subdivide $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ into a convexly liftable cell complex such that the internal boundaries are maintained. According to this subdivision, $f_{pwa}(x)$ is also subdivided into an equivalent PWA function, say $\widetilde{f}_{pwa}(x)$. Note also that function $\widetilde{f}_{pwa}(x)$ is defined over a convexly liftable partition. Again, as per Theorems 4.5.17, 4.5.19, $\widetilde{f}_{pwa}(x)$ can be obtained via parametric linear/quadratic programming problems. □

Consider a simple discontinuous PWA function $f_{pwa}(x)$, defined over a partition in $\mathbb{R}$ as follows:

$$f_{pwa}(x) = \begin{cases} 0.8116x + 0.5328 & \text{for } -0.4 \leq x < -0.3 \\ 0.3507x + 0.9390 & \text{for } -0.3 \leq x < -0.2 \\ 0.8759x + 0.5502 & \text{for } -0.2 \leq x < -0.1 \\ 0.6225x + 0.5870 & \text{for } -0.1 \leq x < 0 \\ 0.2077x + 0.3012 & \text{for } 0 \leq x < 0.1 \\ 0.4709x + 0.2305 & \text{for } 0.1 \leq x < 0.2 \\ 0.8443x + 0.1948 & \text{for } 0.2 \leq x < 0.3 \\ 0.2259x + 0.1707 & \text{for } 0.3 \leq x \leq 0.4 \end{cases}$$

One can easily check the discontinuity of this function via its values at the vertices of the regions in the parameter space partition (see Figure 4.26). A convex lifting of this partition denoted by $\ell(x)$, is presented below:

$$\ell(x) = \begin{cases} -3.5x + 0.4 & \text{for } -0.4 \leq x \leq -0.3 \\ -2.5x + 0.7 & \text{for } -0.3 \leq x \leq -0.2 \\ -1.5x + 0.9 & \text{for } -0.2 \leq x \leq -0.1 \\ -0.5x + 1 & \text{for } -0.1 \leq x \leq 0 \\ 0.5x + 1 & \text{for } 0 \leq x \leq 0.1 \\ 1.5x + 0.9 & \text{for } 0.1 \leq x \leq 0.2 \\ 2.5x + 0.7 & \text{for } 0.2 \leq x \leq 0.3 \\ 3.5x + 0.4 & \text{for } 0.3 \leq x \leq 0.4 \end{cases}$$

This convex lifting is visualized in Figure 4.27. Following the approach presented previously, an optimization problem that admits the given PWA function as an

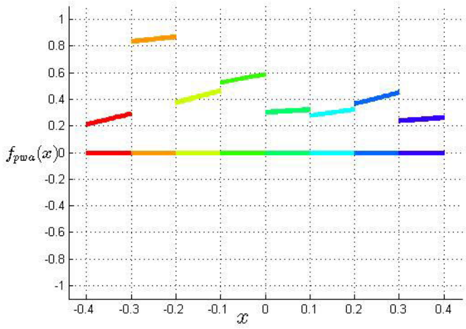


Figure 4.26: A given discontinuous PWA function defined over a partition in $\mathbb{R}$.

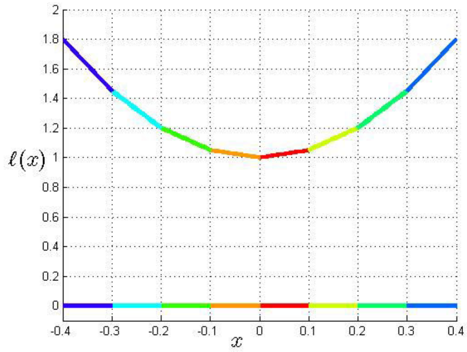


Figure 4.27: A convex lifting for the partition in Figure 4.26.

optimal solution, is presented in (4.46).

$$
\begin{gathered}
\min_{[z\ u^T]^T} z \quad \text{s.t.} \\
\begin{bmatrix}
-0.8944 & 0 \\
-0.5805 & 0.1893 \\
-0.4462 & 0.3412 \\
-0.5547 & 0 \\
-0.2547 & 0.6733 \\
-0.3714 & 0 \\
-0.2747 & 0 \\
-0.4473 & -0.3356 \\
-0.8944 & 0 \\
-0.8200 & 0.2018 \\
0.2738 & 0.7989 \\
0.3139 & 0.8057 \\
-0.6351 & -0.2551 \\
-0.5547 & 0 \\
-0.3714 & 0 \\
-0.4659 & -0.5706 \\
-0.2747 & 0 \\
-0.3217 & -0.8334 \\
-0.1827 & -0.9733 \\
-0.0951 & -0.9955 \\
0.0455 & -0.9968 \\
0.8700 & 0.4919
\end{bmatrix}
\begin{bmatrix} z \\ u \end{bmatrix} \leq
\begin{bmatrix}
-0.4472 \\
-0.7920 \\
-0.8273 \\
-0.8321 \\
-0.6941 \\
-0.9285 \\
-0.9615 \\
-0.8290 \\
0.4472 \\
0.5356 \\
-0.5355 \\
-0.5022 \\
0.7291 \\
0.8321 \\
0.9285 \\
0.6763 \\
0.9615 \\
0.4494 \\
0.1390 \\
0.0045 \\
-0.0659 \\
0.0325
\end{bmatrix} x +
\begin{bmatrix}
-0.8944 \\
-0.4693 \\
-0.2459 \\
-0.4992 \\
0.1406 \\
-0.2600 \\
-0.1099 \\
-0.4799 \\
-0.8944 \\
-0.7015 \\
0.9156 \\
0.9763 \\
-0.7119 \\
-0.4992 \\
-0.2600 \\
-0.6378 \\
-0.1099 \\
-0.5727 \\
-0.4759 \\
-0.3766 \\
-0.1520 \\
1.6815
\end{bmatrix}
\end{gathered}
\tag{4.46}
$$

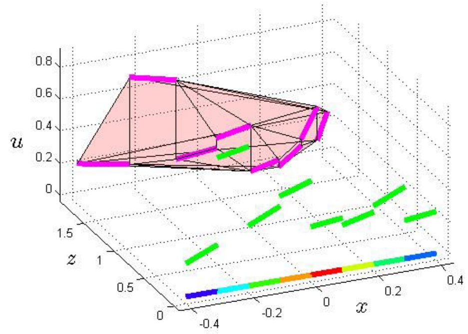


Figure 4.28: Graphical illustration of the recovered optimization problem.

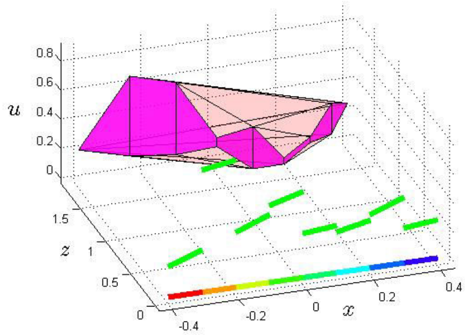


Figure 4.29: Illustration for the set of optimal solutions.

The given discontinuous PWA function is presented in Figure 4.26 wherein the line along the horizontal axis describes the parameter space x and the multicolored curve above describes the given PWA function. Also, in the extended space of parameter x, the original function $f_{pwa}(x)$ and lifting $\ell(x)$, the green curve in Figure 4.28 represents the given PWA function. The shaded pink polytope represents the set of constraints in the inverse optimization problem (4.46). Note also that the dark pink curve describes an optimal solution to this optimization problem. Moreover, the image of this curve onto the space of $\left[x^T\ u^T\right]^T$ coincides with the given PWA function. It is also worth emphasizing that for each segment of state space partition, the set of optimal solutions to the above optimization problem represents a facet of the pink polytope $\Pi_{[x^T\ z\ u^T]^T}$, which is orthogonal to the space $\left[x^T\ z\right]^T$. As illustrated in Figure 4.29, the pink facets represent the set of optimal solutions to the optimization problem (4.46).

Note also that based on the same methodology, inverse parametric linear/quadratic programming problem can also be extended to continuous piecewise affine set-valued maps. These discussions will be presented in Section 7.2.

4.6 Constraint removal of the convex liftings based solution to IPL/QP

The convex liftings-based solution to inverse parametric linear/quadratic programming problem for PWA functions has been presented in Section 4.5. It has been shown that any PWA function can be equivalently obtained by a parametric linear programming problem with at most one auxiliary 1−dimensional variable. This auxiliary variable represents a convex lifting. Accordingly, the main advantage is an important decrease of the dimension of optimization arguments and a relatively compact use of constraints in optimization problems. With respect to the latter aspect, these first results also point out a drawback: the number of constraints may still be relatively large. An important part of these constraints is practically used to bound the feasible set rather than to contribute to the optimal solution. This problem is tackled in this section. The target is the reduction of constraint set towards the minimal number of constraints necessary for the inverse optimality problem. Two algorithms are proposed to eliminate these *redundant constraints* in the IPL/QP formulation. It is worth emphasizing that the class of *continuous PWA functions* are exclusively of interest for the results in this section even though those still hold true for the class of *discontinuous PWA functions*.

For ease of presentation, additional notations are defined as follows. Given a full row rank matrix $M \in \mathbb{R}^{r\times(d+1)}$, $\mathcal{P}(M)$ denotes the polyhedron

$$\mathcal{P}(M) = \left\{x \in \mathbb{R}^d \mid M(\cdot, 1:d)x \leq M(\cdot, d+1)\right\},$$

where $M(\cdot, i)$ denotes the i^{th} column of matrix M and $M(\cdot, i:j)$ represents the matrix formed by the i^{th} to j^{th} columns of M.

Conversely, given a polyhedron P, $\mathcal{P}^{-1}(P)$ denotes the minimal representation (in terms of dimension) of a matrix M satisfying

$$P = \left\{x \in \mathbb{R}^d \mid M(\cdot, 1:d)x \leq M(\cdot, d+1)\right\}.$$

Note that $\mathcal{P}^{-1}(P)$ is not unique for a given polyhedron P, even if its results are in the minimal representation due to the following observation:

Lemma 4.6.1 *Given a matrix M and any diagonal matrix D with the diagonal elements belonging to $\mathbb{R}_{>0}$, then $\mathcal{P}(DM) = \mathcal{P}(M)$.*

Moreover, given two polyhedra $P_M, P_N \subset \mathbb{R}^d$, $M \in \mathcal{P}^{-1}(P_M) \subset \mathbb{R}^{r_M\times(d+1)}$, $N \in \mathcal{P}^{-1}(P_N) \subset \mathbb{R}^{r_N\times(d+1)}$, then $\mathrm{RmSm}(M, N)$ denotes the matrix composed of the rows of M which cannot be described as a scaling of a row of N. Namely, $\mathrm{RmSm}(M, N) = K$ such that

— K is a sub-block of M,

— for any row $K(i, \cdot)$ of K, matrix $\begin{bmatrix} K(i,\cdot) \\ N(j,\cdot) \end{bmatrix}$ has full row rank $\forall j \in \mathcal{I}_{r_N}$.

Also, another operator needs to be introduced for removal of redundant constraints. Given two sets of constraints corresponding to two polyhedra P_M, P_N, $M \in \mathcal{P}^{-1}(P_M) \subset \mathbb{R}^{r_M\times(d+1)}, N \in \mathcal{P}^{-1}(P_N) \subset \mathbb{R}^{r_N\times(d+1)}$, $\mathrm{RmRdd}(M, N)$ denotes the set of the constraints characterizing P_M which are not redundant in the representation of P_N. In fact, there exist different algorithms which carry out removal of redundant constraints, one of them presented in Olaru and Dumur [2005], is recalled here through a mathematical presentation as follows:
$\mathrm{RmRdd}(M, N) = K \in \mathbb{R}^{r_K\times(d+1)}$ such that

— K is a sub-block of M,
— for any $i \in \mathcal{I}_{r_K}$, $\max\limits_{x \,|\, x \in P_N} K(i, 1:d)x > K(i, d+1)$.

This operator has the following property:

Lemma 4.6.2 *Given two polyhedra $P_M \subseteq P_N$, $M \in \mathcal{P}^{-1}(P_M), N \in \mathcal{P}^{-1}(P_N)$, then* $\mathrm{RmRdd}(N, M) = \emptyset$.

It is worth recalling that the convex liftings based solution to IPL/QP problems is usually in the following form:

$$\min_{[z\ u^T]^T} z \ \ \text{s.t.} \ \left[x^T\ z\ u^T\right]^T \in \Pi_{[x^T\ z\ u^T]^T}, \tag{4.47}$$

where $\Pi_{[x^T\ z\ u^T]^T}$ stands for the set of constraints, z represents the auxiliary variable.

Remark 4.6.3 Note that not all constraints describing $\Pi_{[x^T\ z\ u^T]^T}$ in its halfspace representation are meaningful from the optimization point of view, in the sense that they are not active at the optimum. In the geometrical sense, each active constraint corresponds to a supporting hyperplane containing a facet of $\Pi_{[x^T\ z\ u^T]^T}$. It follows that the constraints corresponding to the supporting hyperplanes of $\Pi_{[x^T\ z\ u^T]^T}$ at its facets, which contain the d_x-faces whose orthogonal projections onto $\mathbb{R}^{d_x}$ retrieve the partition $\{\mathcal{X}_i\}_{i \in \mathcal{I}_N}$, are exclusively of use in the IPL/QP formulation.

The relaxation of the constraints which do not become active in the IPL/QP formulation, will modify the feasible set in both parameter and argument space. Thus, the constraint relaxation has to be associated with a restriction in the parameter space which limits the domain of validity for the optimal solution. Note also that these supplementary constraints may give rise to redundancy phenomena. The constraint redundancy is a classical problem in computational geometry investigated in different publications, see for instance Olaru and Dumur [2005].

According to Remark 4.6.3, the first idea for simplification of constraint set, is based on the removal of active constraints of the *maximization* problem characterized by the same cost function and the same constraint set as in (4.47). Note however that not all active constraints from this maximization problem are removable. Some of them contribute to the construction of the solution to (4.47). Therefore, some supplementary constraints are of use to conserve the optimal solution to (4.47). For ease of presentation, the following notations are introduced:

$$\begin{aligned}
&\Pi_{[x^T\ u^T]^T} = \mathrm{Proj}_{[x^T\ u^T]^T}\Pi_{[x^T\ z\ u^T]^T},\ M_{[x^T\ u^T]^T} = \mathcal{P}^{-1}(\Pi_{[x^T\ u^T]^T}),\\
&M_x = M_{[x^T\ u^T]^T}(\cdot, 1:d_x),\ M_u = M_{[x^T\ u^T]^T}(\cdot, d_x+1:d_x+d_u+1),\\
&M_{[x^T\ 0\ u^T]^T} = \left[M_x\, 0\, M_u\right],
\end{aligned}$$

where 0 represents a column vector of appropriate dimension with the elements equal to zero. Π_{max} denotes the polyhedron having the halfspace representation described by the set of constraints, which are active in the following parametric linear programming problem:

$$\max_{[z\ u^T]^T} z \quad \text{s.t.} \quad \left[x^T\ z\ u^T\right]^T \in \Pi_{[x^T\ z\ u^T]^T}.$$

Moreover, define also:

$$\begin{aligned}
&M_{[x^T\ z\ u^T]^T} = \mathcal{P}^{-1}(\Pi_{[x^T\ z\ u^T]^T}),\ M_{max} = \mathcal{P}^{-1}(\Pi_{max}),\\
&\overline{M}_{max} = \operatorname{RmSm}(M_{[x^T\ z\ u^T]^T}, M_{max}),\\
&\widetilde{M} = \begin{bmatrix} \overline{M}_{max} \\ M_{[x^T\ 0\ u^T]^T} \end{bmatrix},\ \widetilde{\Pi} = \mathcal{P}(\widetilde{M}).
\end{aligned}$$

The following result shows an option of these supplementary constraints.

Proposition 4.6.4 *The solutions to two following problems are equivalent:*

$$\begin{aligned}
&\min_{[z\ u^T]^T} z \quad \textit{s.t.} \quad \left[x^T\ z\ u^T\right]^T \in \widetilde{\Pi}\\
&\min_{[z\ u^T]^T} z \quad \textit{s.t.} \quad \left[x^T\ z\ u^T\right]^T \in \Pi_{[x^T\ z\ u^T]^T}.
\end{aligned}$$

Proof: Consider first the case $x_0 \in \operatorname{int}(\mathcal{X}_i)$ for some $\mathcal{X}_i$ which does not have any facet as a facet of $\mathcal{X}$, and the optimal solution to the following problem:

$$\begin{bmatrix} \underline{z}(x_0) \\ \underline{u}(x_0) \end{bmatrix} = \arg \min_{[z\ u^T]^T} z \quad \text{s.t.} \quad \left[x_0^T\ z\ u^T\right]^T \in \Pi_{\left[x_0^T\ z\ u^T\right]^T}. \tag{4.48}$$

Let S_0 denote the polyhedron described by the constraints which are active in the maximization problem:

$$\begin{bmatrix} \overline{z}(x_0) \\ \overline{u}(x_0) \end{bmatrix} = \arg \max_{[z\ u^T]^T} z \quad \text{s.t.} \quad \left[x_0^T\ z\ u^T\right]^T \in \Pi_{\left[x_0^T\ z\ u^T\right]^T}. \tag{4.49}$$

At x_0, $M_0 = \mathcal{P}^{-1}(S_0)$, $\overline{M}_0 = \operatorname{RmSm}(M_{[x^T\ z\ u^T]^T}, M_0)$, then the optimal solution:

$$\begin{bmatrix} \widetilde{z}(x) \\ \widetilde{u}(x) \end{bmatrix} = \arg \min_{[z\ u^T]^T} z \quad \text{s.t.} \quad \left[x^T\ z\ u^T\right]^T \in \mathcal{P}(\overline{M}_0), \tag{4.50}$$

satisfies: $\left[\underline{z}(x_0)\ \underline{u}^T(x_0)\right] = \left[\widetilde{z}(x_0)\ \widetilde{u}^T(x_0)\right]$. Indeed, the optimal solution to (4.48) does not change while removing S_0 as the removed constraints do not contain the

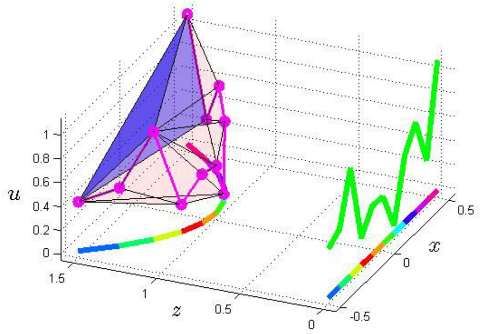


Figure 4.30: Illustration for removal of constraints.

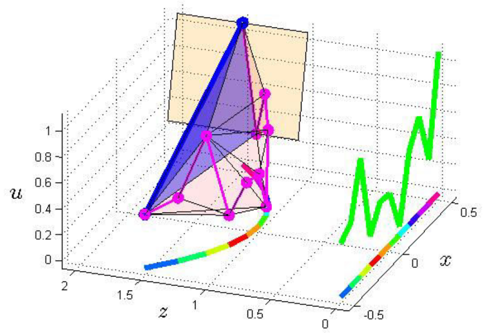


Figure 4.31: Illustration for additional constraints.

optimal solution to (4.48) at x_0. (a graphical illustration is presented in Figure 4.30, where the multicolored line along the $x-$axis represents a partition of the parameter space, and the green curve above stands for its associated continuous PWA function, which needs to be recovered. The shaded polytope represents $\Pi_{[x^T\ z\ u^T]^T}$. The optimal solution to (4.48) is represented by the solid pink line. Also, the blue line represents the optimal solution to (4.49), and it is presented by two constraints described by two hyperplanes containing two facets of $\Pi_{[x^T\ z\ u^T]^T}$, marked with dark color.)

The second case to be considered, is the one of regions $\mathcal{X}_i$ which share common facets with $\mathcal{X}$. Their removal cannot conserve the optimal solution to (4.48) for $x_0 \in \mathcal{X}_i$ (as seen in Figure 4.30, after the removal of two constraints containing two dark facets of $\Pi_{[x^T\ z\ u^T]^T}$, the uniqueness of optimal solution to (4.50) is lost when x_0 belongs to the terminal segment i.e. the pink one containing point $x = 0.5$). Therefore, a restriction of optimal solution needs to be set up via supplementary constraints. These supplementary constraints are chosen here to be related to $\Pi_{[x^T\ 0\ u^T]^T} = \mathcal{P}(M_{[x^T\ 0\ u^T]^T})$. The reason for this choice is to conserve the optimal solution over such a region (as illustrated in Figure 4.31, a constraint of $\Pi_{[x^T\ 0\ u^T]^T}$ corresponding to the yellow hyperplane, is added to conserve the optimal solution over the region $[0.4\ \ 0.5]$.) $\square$

The preceding result leads to Algorithm 4.5 carrying out this removal. The result of this algorithm is a modified set of constraints denoted by $\widetilde{\Pi}^{(1)}$.

Note that steps 3-6 collect all active constraints of $\Pi_{[x^T\ z\ u^T]^T}$ in the maximization problem (4.49). These constraints are eliminated via step 7. After this removal, the remaining constraints may not conserve the original optimal solution to (4.47), therefore, additional constraints described by $\Pi_{[x^T\ 0\ u^T]^T}$ are added. Step 8 carries out the removal of the constraints in $\Pi_{[x^T\ 0\ u^T]^T}$ which are redundant in $\mathcal{P}(\overline{M}_{max})$.

Algorithm 4.5 Constraint removal via maximization problem

Input: $M_{[x^T\ z\ u^T]^T}, M_{[x^T\ 0\ u^T]^T}$.

Output: $\widetilde{\Pi}^{(1)}$.

1: Compute the optimal solution to (4.49): $\begin{bmatrix}\overline{z}(x)\\ \overline{u}(x)\end{bmatrix} : \bigcup_{i\in\mathcal{I}_{\overline{N}}} \overline{\mathcal{X}}_i \rightarrow \mathbb{R}^{d_u+1}$.

2: $M_{max} = []$.

3: **For** $i = 1 : \overline{N}$

4: Find the polyhedron P_0 described by the active constraints at $\left[x^T\ \overline{z}(x)\ \overline{u}^T(x)\right]^T$ for $x \in \overline{\mathcal{X}}_i$.

5: $M_0 = \mathcal{P}^{-1}(P_0)$, $M_{max} = \begin{bmatrix}M_{max}\\ M_0\end{bmatrix}$.

6: **End**

7: $\overline{M}_{max} = \mathrm{RmSm}(M_{[x^T\ z\ u^T]^T}, M_{max})$.

8: $\widetilde{M}^{(0)} = \mathrm{RmRdd}(M_{[x^T\ 0\ u^T]^T}, \overline{M}_{max})$.

9: $\widetilde{M}^{(1)} = \begin{bmatrix}\overline{M}_{max}\\ \widetilde{M}^{(0)}\end{bmatrix}$, $\widetilde{\Pi}^{(1)} = \mathcal{P}(\widetilde{M}^{(1)})$.

Remark 4.6.5 Note that even if Algorithm 4.5 eliminates a number of constraints, it may not be of help in some cases in the overall reduction of constraints. More precisely, through the example shown in Figure 4.30, by Algorithm 4.5 two constraints are removed, but five others are added. In this case, this algorithm does not provide any benefit in terms of constraints reduction, however, it provides an important insight for removal of constraints.

Back to the specific example shown in Figure 4.30, it is observed that the constraints containing the dark facets in Figure 4.32 are not necessary for the minimization problem (4.48). While Algorithm 4.5 can only remove one of them, it leads to an observation that the constraints containing the solid pink line, known as the optimal solution to (4.48), are exclusively of interest. This remark will be the basis of Algorithm 4.6 to carry out the constraint removal. The following proposition summarizes the above comments.

Proposition 4.6.6 *A given continuous PWA function $f_{pwa}(x) : \mathcal{X} \rightarrow \mathbb{R}^{d_u}$ associated with a convexly liftable partition $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ of a polyhedron $\mathcal{X} \subseteq \mathbb{R}^{d_x}$ is the image via the orthogonal projection onto $\mathbb{R}^{d_u}$ of the optimal solution to the problem below:*

$$\min_{[z\ u^T]^T} z \quad s.t. \quad \left[x^T\ z\ u^T\right]^T \in \widetilde{\Pi}^{(2)}, \tag{4.51}$$

4. Matrix 0 has appropriate dimension with the number of columns equal to $d_u + 1$.

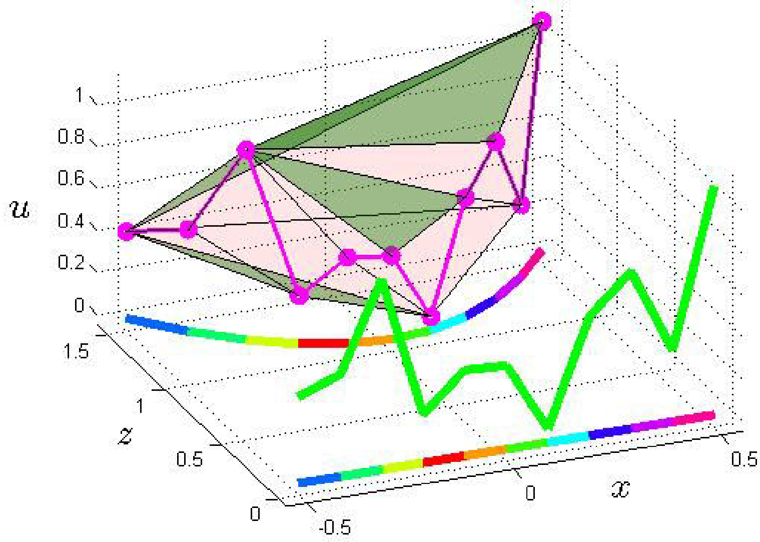


Figure 4.32: Illustration for unnecessary constraints.

with $\widetilde{\Pi}^{(2)}$ *obtained from Algorithm 4.6.*

Algorithm 4.6 Constraint removal via minimization problem

Input: $M_{[x^T\ z\ u^T]^T}$, $\begin{bmatrix} \underline{z}(x) \\ \underline{u}(x) \end{bmatrix}$ the optimal solution to (4.48).

Output: $\widetilde{\Pi}^{(2)}$.

$M_{min} = [\,]$.

For $i = 1 : N$

  Find the polyhedron P_0 described by the constraints which are active at $\left[x^T\ \underline{z}(x)\ \underline{u}^T(x)\right]^T$ for $x \in \mathcal{X}_i$.

  $M_0 = \mathcal{P}^{-1}(P_0),\ M_{min} = \begin{bmatrix} M_0 \\ M_{min} \end{bmatrix}$.

End

$\Pi_{min} = \mathcal{P}(M_{min}),\ \Pi_x = \text{Proj}_{\mathbb{R}^{d_x}} \Pi_{min}$.

$M_x = \mathcal{P}^{-1}(\Pi_x),\ M_f = \mathcal{P}^{-1}(\mathcal{X})$.

$\overline{M}_f = \text{RmRdd}(M_f, M_x)$.

$\overline{M} = \left[\overline{M}_f(\cdot, 1 : d_x)\ \ 0\ \ \overline{M}_f(\cdot, d_x + 1)\right]$.[4]

$M_{min} = \begin{bmatrix} M_{min} \\ \overline{M} \end{bmatrix}$, $\widetilde{\Pi}^{(2)} = \mathcal{P}(M_{min})$.

Proof: The optimality condition conserves the constraints containing the optimal solution to (4.48) as steps 2–5. However, while removing the non-active constraints in $\Pi_{[x^T\ z\ u^T]^T}$, the boundary of the parameter space is lost, therefore, the new set of constraints needs to be restricted to the parameter region $\mathcal{X}$. $\square$

Note that step 8 in Algorithm 4.6 aims to remove redundant constraints of $\mathcal{X}$ in Π_{min}.

Remark 4.6.7 If $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ is a polytopic partition, an alternative for the set of constraints can be in the following form:

$$\Pi_{[x^T\ z\ u^T]^T} = \text{conv}(V_{[x^T\ z\ u^T]^T}) \oplus \text{cone}(\begin{bmatrix}0_{d_x}^T & 1 & 0_{d_u}^T\end{bmatrix}^T),$$

with the vectors $0_{d_x}, 0_{d_u}$ composed of zeros of appropriate dimensions and $V_{[x^T\ z\ u^T]^T}$ is defined as in (4.22). This choice can be of help to avoid constraints describing upper bound of z which are not necessary in the minimization problem (4.23).

Finally, illustrative examples are referred to Section 5.1.

4.7 Conclusions

This chapter presents a method to solve inverse parametric linear/quadratic programming problems. This method relies on convex liftings. It is shown that for any continuous PWA function defined over a polyhedral partition, an appropriate equivalence of this function can be obtained by another parametric linear/quadratic programming problem with an auxiliary variable of dimension equal to 1. This method also covers the class of discontinuous PWA functions. In this case, it is shown that the uniqueness of optimal solution to the recovered optimization problem is no longer preserved. These results will be of help in an alternative implementation of PWA control laws. This aspect will be detailed in the next chapter. Finally, analysis on constraint removal for this method is also considered. It aims at putting forward two algorithms to reduce the number of constraints in the recovered optimization problem.

Chapter 5

Model predictive control redesign

Contents

This chapter aims to present applications of the main results of the above chapters in low complex implementation of PWA control laws as well as in design of stabilizing/robust control for linear systems. This chapter is based on the results presented in Nguyen et al. [a,b, 2015f]

5.1 Efficient alternative implementation of PWA control laws

This section makes use of the results in IPL/QP problems via convex liftings in model predictive control. The basic on linear MPC has been recalled in Section 2.5. It was shown that a linear MPC problem can be easily transformed into a parametric linear/quadratic programming problem, where the optimization argument is defined as the vector composed of the control variables over a finite prediction horizon $N \in \mathbb{N}_{>0}$ i.e.

$$\mathbf{u} = \left[u_{k|k}^T \dots u_{k+N-1|k}^T\right]^T \in \mathbb{R}^{Nd_u},$$

and the current state represents a parameter. In implementation, the first input is of interest $u^*(x_k) = \text{Proj}_{\mathbb{R}^{d_u}} \mathbf{u}^*(x_k)$, as a consequence, it also inherits the piecewise affine structure of optimal solution to this PL/QP problem. The following results are direct consequences of those presented in Section 4.5.

Corollary 5.1.1 *Any continuous PWA control law defined over a polyhedral partition of the state space can be obtained through a parametric linear/quadratic programming problem.*

Proof: See the proof of Theorem 4.5.11. □

Central to the following result is the complexity of a linear MPC problem.

Theorem 5.1.2 *Any continuous explicit solution of a linear MPC problem with respect to a linear/quadratic cost function is equivalently obtained through a linear MPC problem with a linear or quadratic cost function and the control horizon at most equal to 2 prediction steps.*

Proof: Let $u(x) : \mathbb{R}^{d_x} \to \mathbb{R}^{d_u}$ denote a continuous optimal control law to a linear MPC problem, defined over a state space partition $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ of a polyhedron $\mathcal{X} \subseteq \mathbb{R}^{d_x}$. If $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ is not convexly liftable, it needs to be subdivided into a convexly liftable cell complex. A constructive solution for this subdivision was presented in the proof of Theorem 4.4.10. Therefore, one can exclusively focus on the case $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ is convexly liftable.

Now, let $\Pi_{\left[x_k^T\ z\ u_k^T\right]^T}$ denote the set of constraints in the recovered optimization problem i.e.

$$\min_{\left[z\ u_k^T\right]^T} z \quad \text{s.t.} \quad \left[x_k^T\ z\ u_k^T\right]^T \in \Pi_{\left[x_k^T\ z\ u_k^T\right]^T}. \tag{5.1}$$

Define also the following matrices (recall that the operation $\mathcal{P}^{-1}()$ is defined in Section 4.6):

$$H = \mathcal{P}^{-1}(\Pi_{[x_k^T\ z\ u_k^T]^T}),\ H_x = H(\cdot, 1 : d_x),\ H_z = H(\cdot, 1 + d_x),$$
$$H_u = H(\cdot, d_x + 2 : d_x + d_u + 1),\ K = H(\cdot, d_x + d_u + 2).$$

If $d_u = 1$, then it suffices to use z as the second predicted control law i.e. $u_{k+1|k} = z$. Otherwise, the set of constraints $H_x x_k + H_u u_k + H_z z \leq K$ can also be written in the following form:

$$H_x x_k + H_u u_k + [H_z\ 0] \begin{bmatrix} z \\ s \end{bmatrix} \leq K,$$

where 0 denotes a matrix of appropriate dimension, composed of zeros with the number of columns equal to $d_u - 1$. Also, $s \in \mathbb{R}^{d_u - 1}$ denotes auxiliary variable. Again, apply $\begin{bmatrix} z \\ s \end{bmatrix}$ for the next predicted control variable i.e. $u_{k+1|k} = \begin{bmatrix} z \\ s \end{bmatrix}$. Accordingly, (5.1) can be written as follows:

$$\min_{\left[u_k^T\ u_{k+1|k}^T\right]^T} \left[0_{d_u}^T\ 1\ 0_{d_u-1}^T\right] \begin{bmatrix} u_k \\ u_{k+1|k} \end{bmatrix} \ \text{s.t.}\ \ H_x x_k + [H_u\ H_z\ 0] \begin{bmatrix} u_k \\ u_{k+1|k} \end{bmatrix} \leq K,$$

known to be a linear MPC problem with respect to a linear cost function.

On the other hand, according to Theorem 4.5.10, the recovered optimization problem with a quadratic cost function can also be written in the following form:

$$\min_{\left[z\ u_k^T\right]^T} (z - \sigma(x_k))^2 \ \text{ s.t. }\ \left[x_k^T\ z\ u_k^T\right]^T \in \Pi_{\left[x_k^T\ z\ u_k^T\right]^T}, \tag{5.2}$$

where $\sigma(x_k) \leq \ell(x_k)$, $\ell(x_k)$ denotes the convex lifting for the given cell complex $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$, used to compute $\Pi_{[x_k^T\ z\ u_k^T]^T}$. Suppose $\ell(x_k) = a_i^T x_k + b_i$ over a region $\mathcal{X}_i$, it suffices to choose $\sigma(x_k) = a_i^T x_k + b_i$ over $\mathcal{X}$. Accordingly, (5.2) can be easily written in the form of a linear MPC problem with respect to a quadratic function of $\left[u_k^T\ u_{k+1|k}^T\right]^T$. The proof is complete. □

Remark 5.1.3 It is worth stressing that the construction of a convex lifting is of help to facilitate implementation of PWA controllers. More clearly, instead of storing state space partition, this convex lifting and control law gains are stored in the hardware level. Namely, suppose these convex lifting $\ell(x)$ and PWA controller $u(x)$ are denoted as follows:

$$\begin{aligned} \ell(x) &= a_i^T x + b_i \\ u(x) &= H_i x + G_i \end{aligned} \quad \text{for } x \in \mathcal{X}_i.$$

Accordingly, one only needs to store a_i, b_i, H_i, G_i. Also at each sampling time, with the current state x, it only requires determining index i such that

$$a_i^T x + b_i = \max_{j \in \mathcal{I}_N} (a_j^T x + b_j).$$

Then, control law $u(x) = H_i x + G_i$ is applied. This procedure is repeated for the next instant. This implementation allows PWA controllers to be implemented into low-cost platforms since the state space partition is not necessary in this case. Further studies about efficient implementation of PWA control laws is referred to Baotic et al. [2008].

It is worth considering some numerical examples to illustrate the relevance of IPL/QP in implementation of PWA control laws.

5.1.1 Example 1

Consider the double integrator system, mathematically represented as follows:

$$\begin{aligned} x_{k+1} &= \begin{bmatrix} 1 & 0.5 \\ 0 & 1 \end{bmatrix} x_k + \begin{bmatrix} 0.125 \\ 0.5 \end{bmatrix} u_k, \\ y_k &= \begin{bmatrix} 1 & 0 \end{bmatrix} x_k. \end{aligned} \tag{5.3}$$

A cost function is minimized over a prediction horizon $N = 5$ with respect to weighting matrices $Q = \begin{bmatrix} 10 & 0 \\ 0 & 10 \end{bmatrix}$, $R = 0.5$:

$$J = x_{k+5|k}^T P x_{k+5|k} + \sum_{i=0}^{4} (x_{k+i|k}^T Q x_{k+i|k}^T + u_{k+i|k}^T R u_{k+i|k}),$$

with P computed from the Riccati equation. The constraints on the control variable and the output variable are given by:

$$\begin{aligned} & u_{k+i|k} \in [-2,\ 2]\,,\ y_{k+i|k} \in [-5,\ 5]\ \text{ for } 0 \le i \le 4 \\ & x_{k+5|k} \in \mathbb{X}_f, \end{aligned}$$

where $\mathbb{X}_f$ is chosen as the maximal output admissible set (further details in Gilbert and Tan [1991]).

The state space partition known to be convexly liftable, is illustrated in Figure 5.1. The feedback control law computed from the original MPC problem is depicted in Figure 5.2. A convex lifting for the state space cell complex in Figure 5.1 is shown in Figure 5.3. Finally, the result of parametric linear programming problem constructed via Algorithm 4.3 is shown in Figure 5.4. One can see that the recovered PWA control law in Figure 5.4 and the original one in Figure 5.2 are identical.

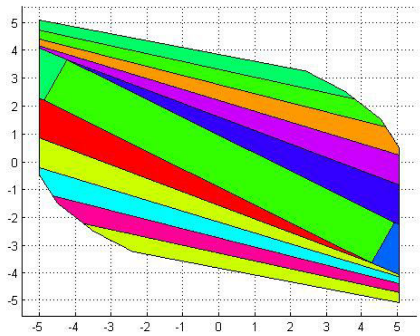


Figure 5.1: State space partition of Example 1.

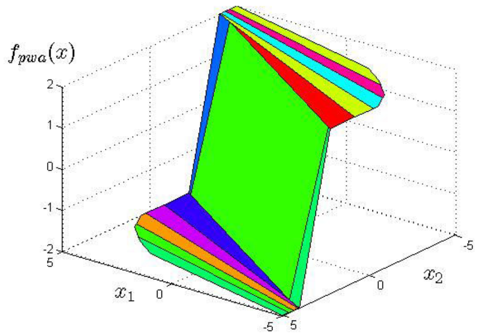


Figure 5.2: The original PWA control law.

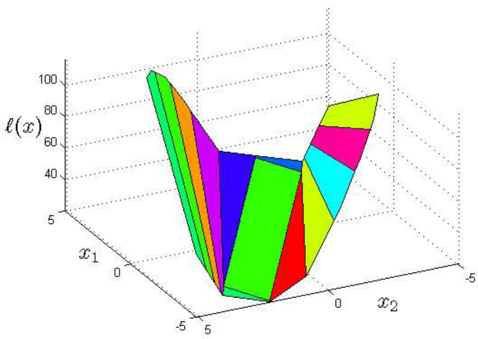


Figure 5.3: A convex lifting for the cell complex in Figure 5.1.

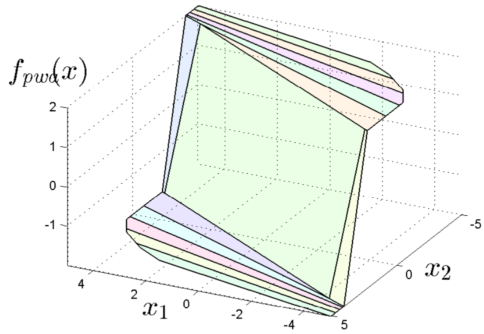


Figure 5.4: The recovered PWA control law.

5.1.2 Example 2

The double integrator system is reconsidered, however, the output constraints are disregarded in order to stress on the input constraints and the particularities of the partition in this case. The constraints on input signal, the prediction horizon and the cost function are preserved. The state space partition is known to be non-convexly liftable, as seen in Figure 5.5. Its associated PWA control law is presented in Figure 5.6. A state space partition equivalent to the original one and satisfying Assumption 4.5.2, is presented in Figure 5.7, according to one of its convex liftings. This partition is obtained by an appropriate refinement of the regions associated with the same saturated control law i.e. $u(x) = 2$ or $u(x) = -2$. Finally, the recovered PWA control law is presented in Figure 5.8, it is easy to see that this PWA control law is equivalent to the original one in Figure 5.6.

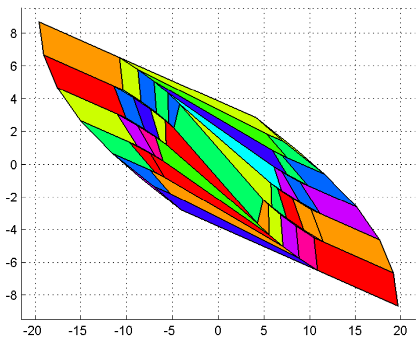


Figure 5.5: State space partition of Example 2.

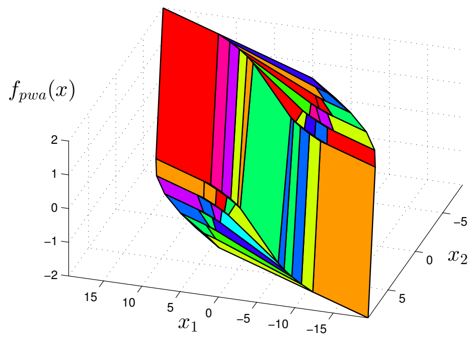


Figure 5.6: The original PWA control law.

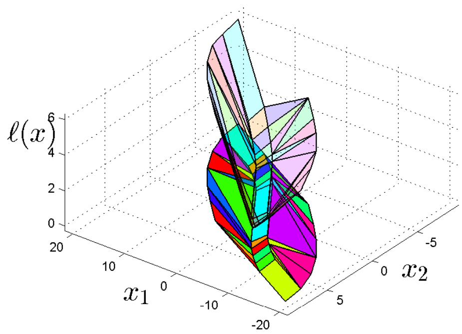


Figure 5.7: Equivalent partition and one of its convex liftings.

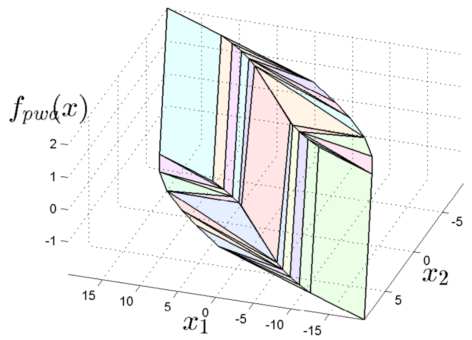


Figure 5.8: A PWA control law equivalent to the original one in Figure 5.6.

5.1.3 Complexity analysis

These two numerical examples consider the double integrator system. The first one needs 24 constraints in the standard formulation of MPC problem with the prediction horizon equal to 5. Meanwhile, the formulation based on inverse optimality i.e. Algorithms 4.3 + 4.6 needs the same number of constraints to obtain the same PWA control law, but the dimension of the optimization argument is equal to 2. This difference leads to a computational time for the explicit solution of the formulation based on inverse optimality smaller than the one by the standard MPC formulation. This complexity reduction is retrieved in the second example. Nothing changes in terms of the dimension of optimization argument with respect to the preceding example. However, the formulation of the standard MPC problem employs 14 constraints, while the formulation based on Algorithms 4.3 + 4.6 finds an equivalent solution by an optimization problem with 43 constraints.

Example	Method	Prediction horizon	Computational time [b] (s)	number of constraints [a] Π	$\mathcal{A}$
1	Standard MPC	5	3.9000	24	
	Algorithm 4.3	2	5.0870	74	
	Algorithms 4.3 + 4.5	2	4.5800	62	0
	Algorithms 4.3 + 4.6	2	0.7410	14	10
2	Standard MPC	5	6.8190	14	
	Algorithm 4.3	2	7.230	81	
	Algorithms 4.3 + 4.5	2	6.399	69	14
	Algorithms 4.3 + 4.6	2	3.073	32	11

a. The number of constraints for Algorithms 4.5, 4.6 consists of two parts: general constraints for the optimization problem after the removal denoted by Π and additional constraints to conserve the optimal solution's structure denoted as $\mathcal{A}$. For Algorithm 4.5, $\Pi = \mathcal{P}(\overline{M}_{max})$ and $\mathcal{A} = \mathcal{P}(\widetilde{M}^{(0)})$. For Algorithm 4.6, $\Pi = \Pi_{min}$ and $\mathcal{A} = \mathcal{P}(\overline{M}_f)$.

b. The numerical examples in this section were carried out in the environment of MPT 3.0 Herceg et al. [2013] on a computer with an Intel Core i5, M430, 2.27 Ghz, Ram 4G. This computer was equipped with a 32 bit version of Windows 7.

Table 5.1: Comparison of different formulations of the IPL/QP problem

Similar to the first example, the computational time for the explicit solution of the formulation based on Algorithms 4.3 + 4.6 is much smaller than the original MPC one. More details can be found in Table 5.1. Accordingly, it can be seen that if the computational time of the explicit solution is gained, the same thing can be said for computing the implicit solution. This leads to the fact that *if one implements the implicit solution to the recovered optimization problem, the online computational time is improved over the implementation of the implicit solution to the original problem.* Also, this implementation does not require storing the control law gains and the regions.

To conclude this section, we refer interested readers to Gulan et al. [2015] for a detailed analysis of a practical application.

5.2 Robust control design based on convex liftings

As presented before, *convex liftings* have been of use to solve inverse parametric linear/quadratic programming problems. It has been shown via some numerical examples that inverse optimality is of help to reduce the implementation complexity of PWA control laws.

In the same line with the studies in Blanchini [1994], Gutman and Cwikel [1987], Nguyen [2014], this section presents an attempt to use *convex liftings* in the design of robust control for linear systems affected by bounded additive disturbances and polytopic uncertainties as defined in (2.9) which can serve as control Lyapunov functions. This method will be proved to guarantee the recursive feasibility and closed loop stability. In terms of implementation, it only requires solving a simple linear programming problem at each sampling instant.

5.2.1 Problem setting

For reading ease, a linear system affected by bounded additive disturbances and polytopic uncertainties is recalled here:

$$x_{k+1} = A(k)x_k + B(k)u_k + w_k, \tag{5.4}$$

where x_k, u_k denote the state, control variables at time k, w_k stands for the disturbance at time k and a polytopic uncertainty set is in the following form:

$$[A(k)\ B(k)] \in \Psi = \text{conv}([A_1\ B_1], \ldots, [A_L\ B_L]). \tag{5.5}$$

The state, control variables and the disturbances are subject to constraints:

$$x_k \in \mathbb{X} \subset \mathbb{R}^{d_x}, u_k \in \mathbb{U} \subset \mathbb{R}^{d_u}, w_k \in \mathbb{W} \subset \mathbb{R}^{d_x}, \tag{5.6}$$

where $d_x, d_u \in \mathbb{N}_{>0}$, $\mathbb{X}, \mathbb{U}, \mathbb{W}$ are polytopes. It is assumed that $\mathbb{X}, \mathbb{U}, \mathbb{W}$ contain the origin in their interior.

The aim is to find a robust control law which can cope with bounded additive disturbances and polytopic model uncertainties such that the closed loop is robustly stable. It is clear that if disturbance w_k is unknown for the computation of control action at instant k, one cannot expect to be able to guarantee asymptotic stability of the origin. The asymptotic stability is replaced with an ultimate boundedness notion Khalil [2002], Kofman et al. [2007].

5.2.2 Positively invariant set and domain of attraction

The first step in this procedure is to find an unconstrained robust control law $u_k = Kx_k \in \mathbb{U}$. A computation of this control law has been proposed in Subsection 2.4.1. A control law is always associated with a (robust) positively invariant set as defined in Subsection 2.3.2. For ease of presentation, let Ω denote such a (robust) positively invariant set. To compute Ω for the linear system (5.4), Algorithm 2.1 may be of help. This algorithm approximates the maximal (robust) positively invariant set associated with this unconstrained control law.

Note however that in the presence of persistent disturbances, Ω is considered as a full-dimensional set. Still, if system (5.4) is only affected by bounded additive disturbances, one can choose Ω as the minimal positively invariant set to reduce the impact of disturbances. Remarkable studies to compute the minimal robust positively invariant set can be found in Kolmanovsky and Gilbert [1998], Rakovic et al. [2005]. Otherwise, if system (5.4) is not affected by additive disturbances and/or is subject to polytopic model uncertainties, $\Omega = \{0\}$ can also be chosen. In this case, asymptotic stability of the origin can be achieved.

On the other hand, a domain of attraction is also of importance. This is defined as a subset of all points in the state space which can be driven to a target set. To guarantee the convergence to a robust positively invariant set Ω, a domain of attraction denoted by $\mathcal{X}$, should ensure that for any point belonging to $\mathcal{X}$, there always exists control law satisfying constraint (5.6), which steers the state to Ω. A candidate for this domain of attraction can be the maximal $\lambda-$contractive set for a $0 \leq \lambda < 1$ as defined in Subsection 2.4.3. With a given $0 \leq \lambda < 1$, the maximal $\lambda-$contractive set for the linear system (5.4) denoted by P_λ can be computed using Algorithm 2.3. Without loss of generality, we assume that $\Omega \subset P_\lambda$. This section will present a control Lyapunov function defined over this contractive set.

5.2.3 Convex lifting construction for control

A definition of convex liftings is presented in Definition 4.2.3. It is known that a polyhedral partition has to fulfill some conditions for the existence of a convex lifting (see Section 4.4.1). In this section, a class of convex liftings will be proved to be control Lyapunov functions. An algorithm for constructing such convex liftings is presented in the sequel. This convex lifting denoted as $\ell(x)$, is defined over a domain of attraction $\mathcal{X}$. Recall that in this section, as discussed in Subsection 5.2.2, the maximal $\lambda-$contractive set P_λ for a given $0 \leq \lambda < 1$, is chosen as a domain of attraction i.e. $\mathcal{X} = P_\lambda$.

Algorithm 5.1 Construct a convex lifting as a control Lyapunov function

Input: A given robust positively invariant set $\Omega \subset \mathbb{R}^{d_x}$, the domain of attraction $\mathcal{X} = P_\lambda \subset \mathbb{R}^{d_x}$ with a given $0 \leq \lambda < 1$ and a scalar $c > 0$.

Output: A convex lifting $\ell(x)$ such that $\ell(x) = 0$ for every $x \in \Omega$.

1: $V_1 = \mathcal{V}(\Omega)$, $\widehat{V}_1 = \left\{ \begin{bmatrix} x \\ 0 \end{bmatrix} \mid x \in V_1 \right\} \subset \mathbb{R}^{d_x+1}$.

2: $V_2 = \mathcal{V}(\mathcal{X})$, $\widehat{V}_2 = \left\{ \begin{bmatrix} x \\ c \end{bmatrix} \mid x \in V_2 \right\} \subset \mathbb{R}^{d_x+1}$.

3: $\Pi = \mathrm{conv}(\widehat{V}_1 \bigcup \widehat{V}_2)$.

4: Solve the parametric linear programming problem:

$$z^*(x) = \min_z z \;\; \text{s.t.} \;\; \begin{bmatrix} x^T & z \end{bmatrix}^T \in \Pi. \tag{5.7}$$

5: $\ell(x) = z^*(x) = a_i^T x + b_i$ for $x \in \mathcal{X}_i$.

Steps 1-2 in Algorithm 5.1 aim to lift the vertices of Ω and $\mathcal{X}$ to $\mathbb{R}^{d_x+1}$ with appropriate heights. Namely, the vertices of Ω are lifted with heights equal to 0, whereas the vertices of $\mathcal{X}$ are lifted with heights equal to the given $c > 0$. Also the convex lifting $\ell(x)$ is generated from the parametric linear programming problem (5.7). Note also that due to the construction, there exists a region in the partition associated with $\ell(x)$ coincident with Ω, since the vertices of Ω are lifted onto a lower facet of Π. The following observation describes the properties of such an $\ell(x)$, generated from Algorithm 5.1.

Lemma 5.2.1 *The function $\ell(x)$ over $\mathcal{X}$, generated from Algorithm 5.1, is continuous, non-negative and convex. Also, $\ell(x) = 0$ for every $x \in \Omega$ and $\ell(x) > 0$ for any $x \in \mathcal{X}\backslash\Omega$.*

Proof: The continuity and convexity of $\ell(x)$ can be easily derived from Theorem IV.3 in Gal [1995] or Theorem 7.1.2.

The second statement is deduced from the construction in step 1. Indeed, consider $x \in \Omega$, then x can be written as a convex combination of the vertices of Ω as: $x = \sum_{v \in V_1} \alpha(v)v$ with $\alpha(v) \geq 0$ and $\sum_{v \in V_1} \alpha(v) = 1$. It is known that $\ell(x)$ over Ω is an affine function, then $\ell(x) = a_i^T x + b_i$ leads to $\ell(x) = 0$ for every $x \in \Omega$.

To complete the proof, it is necessary to show that $\ell(x)$ is a non-negative function. Indeed, as shown above, $\ell(x) = a_i^T x + b_i = 0$ for every $x \in \Omega$, then due to the full dimension of Ω, $a_i = 0$, $b_i = 0$. By the definition of a convex lifting, $\ell(x)$ is a piecewise affine function, thus over a region $\mathcal{X}_j$, one has $\ell(x) = a_j^T x + b_j$ for every $x \in \mathcal{X}_j$. This satisfies the convexity condition for $\mathcal{X}_j \neq \Omega$ ($\mathcal{X}_i = \Omega$):

$$\begin{aligned} a_j^T x + b_j > a_i^T x + b_i = 0, &\quad \text{for every} \quad x \in \mathcal{X}_j\backslash\mathcal{X}_i, \\ a_j^T x + b_j = a_i^T x + b_i = 0, &\quad \text{for every} \quad x \in \mathcal{X}_j \cap \mathcal{X}_i. \end{aligned}$$

The same inclusion for the other affine functions of $\ell(x)$, leads to the non-negativity of $\ell(x)$. Moreover, $\ell(x) > 0$ for every $x \in \mathcal{X}\backslash\Omega$. The proof is complete. □

A simple consequence of Lemma 5.2.1 can be deduced as follows.

Lemma 5.2.2 *For any $x \in \mathcal{X}$ and $0 \leq \beta \leq 1$, $\ell(\beta x) \leq \beta\ell(x)$.*

Proof: Due to the convexity of $\ell(x)$ over $\mathcal{X}$ as proved in Lemma 5.2.1, it leads to

$$\ell(\beta x + (1-\beta)0) \leq \beta\ell(x) + (1-\beta)\ell(0).$$

Due to the assumption that $0 \in \text{int}(\mathbb{W})$, then $0 \in \text{int}(\Omega)$, meaning that $\ell(0) = 0$. This inclusion and the above one imply that $\ell(\beta x) \leq \beta\ell(x)$. □

Algorithm 5.2 Robust control design procedure based on convex liftings

Input: A robust positively invariant set Ω associated with a stabilizing control law $u = Kx$ over Ω. A convex lifting $\ell(x) = a_i^T x + b_i$ for $x \in \mathcal{X}_i$, $i \in \mathcal{I}_N$ as in Algorithm 5.1.

Output: Control law $u^*(x_k)$ at each sampling time.

1: Compute $\ell(x_k)$.

2: **If** $x_k \in \Omega$ then $u^*(x_k) = Kx_k$, jump to Step 6.

3: **Else** Solve the following linear programming problem:

$$\begin{aligned} \left[\alpha^* \ (u_k^*)^T\right]^T &= \arg \min_{\left[\alpha \ u_k^T\right]^T} \alpha \\ \text{s.t. } a_i^T(A(k)x_k + B(k)u_k + w_k) + b_i &\leq \alpha \ell(x_k) \\ \alpha \geq 0, u_k \in \mathbb{U}, \ \forall i \in \mathcal{I}_N, \forall w_k &\in \mathcal{V}(\mathbb{W}), \ \forall [A(k) \ B(k)] \in \mathcal{V}(\Psi). \end{aligned} \tag{5.8}$$

4: Apply $u^*(x_k) = u_k^*$

5: **End**

6: $k \leftarrow k+1$, return to Step 1.

5.2.4 Robust control design procedure

This subsection introduces the procedure for designing robust control laws based on convex liftings. This procedure can guarantee robust stability of the closed loop in the sense of Lyapunov function. A definition of this robust stability is referred to Definition 2.4.1. The design procedure is summarized in Algorithm 5.2.

Remark 5.2.3 Note that the task of verifying whether or not x_k belongs to Ω in Step 2 of Algorithm 5.2, can be easily carried out by checking whether or not $\ell(x_k) = 0$. This property is due to the construction of a convex lifting in Algorithm 5.1. Therefore, it is not necessary to store the constraints describing Ω in the implementation.

Natural questions arise here whether or not the linear programming problem (5.8) is feasible and whether closed loop stability is guaranteed by the proposed procedure. These questions are answered via the following theorem. Accordingly, it will be shown that convex lifting constructed in Algorithm 5.1 can serve as a control Lyapunov function. Thus, the proposed control design can guarantee the robust stability as per Definition 2.4.1.

Theorem 5.2.4 *Given a robust positively invariant set Ω associated with a robust control law gain K and a domain of attraction $\mathcal{X} = P_\lambda$ for a given $0 \leq \lambda < 1$,* *if the initial condition $x_k \in \mathcal{X}$, then the linear programming problem (5.8) is recursively feasible. Furthermore, the closed loop is robustly stable in the sense of Lyapunov.*

Proof: As for the feasibility of (5.8), one can easily see that $0 \leq \ell(x) \leq c$ by the construction in Algorithm 5.1. Therefore, due to the contractivity of $\mathcal{X}$, for any $x_k \in \mathcal{X}$ there always exists $u(x_k) \in \mathbb{U}$ such that:

$$A(k)x_k + B(k)u(x_k) + w_k \in \lambda\mathcal{X} \subset \mathcal{X}$$

for all $w_k \in \mathbb{W}$ and for all $[A(k)\ B(k)] \in \Psi$. Therefore, if $u^*(x_k)$ denotes an optimal solution to (5.8), then one has:

$$\begin{aligned}0 \leq \ell(A(k)x_k + B(k)u^*(x_k) + w_k) \leq \ell(A(k)x_k + B(k)u(x_k) + w_k) \leq c,\\ \forall\, w_k \in \mathbb{W},\ \ \forall\ [A(k)\ B(k)] \in \Psi.\end{aligned}$$

Due to this boundedness, the recursive feasibility of the linear programming problem (5.8) is ensured for a finite, large enough scalar α at each sampling time.

As for robust stability, it will be proved that for all $x_k \in \mathcal{X}\backslash\Omega$:

$$\ell(A(k)x_k + B(k)u^*(x_k) + w_k) < \ell(x_k),\ \forall w_k \in \mathbb{W},\ \forall\,[A(k)\ B(k)] \in \Psi.$$

Indeed, due to the contractivity of $\mathcal{X}$, for any $v \in \mathcal{V}(\mathcal{X})$, there exists a control law, denoted by $u(v) \in \mathbb{U}$ such that $A(k)v + B(k)u(v) + w_k \in \lambda\mathcal{X}$ despite any disturbances $w_k \in \mathbb{W}$ and for all $[A(k)\ B(k)] \in \Psi$. For each $w_k \in \mathbb{W}$ and each $[A(k)\ B(k)] \in \Psi$, there exists $y(k, w_k) \in \mathcal{X}$ such that

$$A(k)v + B(k)u(v) + w_k = \lambda y(k, w_k).$$

Due to Lemma 5.2.2, this inclusion leads to

$$\ell(A(k)v + B(k)u(v) + w_k) = \ell(\lambda y(k, w_k)) \leq \lambda\ell(y(k, w_k)). \tag{5.9}$$

By the construction of $\ell(x)$ in Algorithm 5.1, the following is obtained:

$$\ell(y(k, w_k)) \leq c. \tag{5.10}$$

Also, according to Algorithm 5.1,

$$\ell(v) = c. \tag{5.11}$$

From (5.9), (5.10), (5.11), one can deduce that

$$\ell(A(k)v + B(k)u(v) + w_k) \leq \lambda\ell(v). \tag{5.12}$$

Note that (5.12) holds for all $w_k \in \mathbb{W}$ and for all $[A(k)\ B(k)] \in \Psi$. Moreover, it can be observed that:

$$\begin{aligned} \ell(A(k)v + B(k)u^*(v) + w_k) \leq \ell(A(k)v + B(k)u(v) + w_k), \\ \forall w_k \in \mathbb{W}, \forall [A(k)\ B(k)] \in \Psi, \end{aligned} \tag{5.13}$$

where $u^*(x)$ denotes optimal control to (5.8) at x as used in Algorithm 5.2.

(5.12), (5.13) lead to the following fact:

$$\ell(A(k)v + B(k)u^*(v) + w_k) \leq \lambda\ell(v),\ \forall w_k \in \mathbb{W}, \forall [A(k)\ B(k)] \in \Psi. \tag{5.14}$$

Note that (5.14) holds true for all vertices of $\mathcal{X}$. Now, consider a point $x_k \in \mathcal{X}_i$ in the polytopic partition $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ of $\mathcal{X}$ over which $\ell(x)$ is defined. Without loss of generality, suppose $\mathcal{X}_i \neq \Omega$, then x_k can be described via a convex combination of the vertices of $\mathcal{X}_i$, meaning:

$$x_k = \sum_{v\in\mathcal{V}(\mathcal{X}_i)} \alpha(v)v, \text{ where } \alpha(v) \in \mathbb{R}_+, \sum_{v\in\mathcal{V}(\mathcal{X}_i)} \alpha(v) = 1.$$

Recall that due to the definition of convex lifting, $\ell(x)$ over $\mathcal{X}_i$ is an affine function, then $\ell(x_k)$ can be written in the following form:

$$\ell(x_k) = \sum_{v\in\mathcal{V}(\mathcal{X}_i)} \alpha(v)\ell(v). \tag{5.15}$$

If $v \in \mathcal{V}(\mathcal{X}_i)$ is a vertex of Ω, then due to the robust positive invariance of Ω with respect to a linear feedback $u^*(x) = Kx$, it satisfies

$$\ell(v) = 0 = \ell((A(k) + B(k)K)v + w_k), \forall w_k \in \mathbb{W}, \forall [A(k)\ B(k)] \in \Psi. \tag{5.16}$$

Otherwise, if $v \in \mathcal{V}(\mathcal{X}_i)$ is a vertex of $\mathcal{X}$, then it satisfies (5.14). Therefore, due to the convexity of $\ell(x)$ proved in Lemma 5.2.1 and (5.14), (5.15), (5.16), the following is obtained:

$$\begin{aligned} \lambda\ell(x_k) &= \sum_{v\in\mathcal{V}(\mathcal{X}_i)} \alpha(v)(\lambda\ell(v)) \\ &\geq \sum_{v\in\mathcal{V}(\mathcal{X}_i)} \alpha(v)\ell(A(k)v + B(k)u^*(v) + w_k) \\ &\geq \ell(A(k) \sum_{v\in\mathcal{V}(\mathcal{X}_i)} \alpha(v)v + B(k) \sum_{v\in\mathcal{V}(\mathcal{X}_i)} \alpha(v)u^*(v) + w_k) \\ &= \ell(A(k)x_k + B(k) \sum_{v\in\mathcal{V}(\mathcal{X}_i)} \alpha(v)u^*(v) + w_k). \end{aligned} \tag{5.17}$$

Recall that $u^*(v) \in \mathbb{U}$, $\forall v \in \mathcal{V}(\mathcal{X}_i) \cap \mathcal{V}(\mathcal{X})$ and $u^*(v) = Kv \in \mathbb{U}$, $\forall v \in \mathcal{V}(\mathcal{X}_i) \cap \mathcal{V}(\Omega)$, then it follows that

$$\sum_{v \in \mathcal{V}(\mathcal{X}_i)} \alpha(v) u^*(v) \in \mathbb{U}. \tag{5.18}$$

Therefore, (5.18) leads to:

$$\ell(A(k)x_k + B(k) \sum_{v \in \mathcal{V}(\mathcal{X}_i)} \alpha(v) u^*(v) + w_k) \geq \ell(A(k)x_k + B(k)u^*(x_k) + w_k). \tag{5.19}$$

From (5.17) and (5.19), the following inclusion can be obtained:

$$\lambda \ell(x_k) \geq \ell(A(k)x_k + B(k)u^*(x_k) + w_k), \ \forall w_k \in \mathbb{W}, \ \forall \left[A(k) \ B(k)\right] \in \Psi. \tag{5.20}$$

Recall that $0 \leq \lambda < 1$, therefore

$$\ell(x_k) > \ell(A(k)x_k + B(k)u^*(x_k) + w_k), \ \forall w_k \in \mathbb{W}, \ \forall \left[A(k) \ B(k)\right] \in \Psi, \tag{5.21}$$

meaning $\{\ell(x_k)\}_{k=0}^{\infty}$ is a strictly decreasing sequence outside Ω and bounded in the interval $[0, c]$. Thus, this sequence is convergent to 0. In other words, $\ell(x)$ serves as a Lyapunov function according to Definition 2.4.1. □

Remark 5.2.5 Note that by the construction, the partition associated with a convex lifting in Algorithm 5.1, may not be a Delaunay decomposition as in Scibilia et al. [2009]. This method does not rely on such a decomposition, but relies on a continuous, convex function, defined over this partition. This approach is simple and needs only to solve a linear programming problem at each sampling instant. However, the associated control law is not continuous at the moment the state switches into Ω (see step 2 of Algorithm 5.2). Note also that the checking whether the current state belongs to Ω can be relaxed. Accordingly, one can continue solving the problem (5.8) while trajectories still stay inside Ω. Indeed, if $x_k \in \Omega$, then due to the construction $\ell(x_k) = 0$. Consider the next state, one can see that $Kx_k \in \mathbb{U}$, then it leads to:

$$0 \leq \ell(A(k)x_k + B(k)u^*(x_k) + w_k) \leq \ell(A(k)x_k + B(k)Kx_k + w_k) = 0 = \ell(x_k).$$

This inclusion implies that optimal control law $u^*(x_k) \in \mathbb{U}$ to problem (5.8) also keeps the trajectories inside Ω, if x_k is inside Ω.

Remark 5.2.6 An open problem is to guarantee closed-loop stability of the proposed method for a domain of attraction as the $N-$*steps robust controllable set* denoted by $\mathcal{K}_N(\Omega)$ and defined in Subsection 2.4.2. Note that in this case, proving the strict decrease of $\ell(x)$ becomes more difficult. Also, this strict decrease may not be successive.

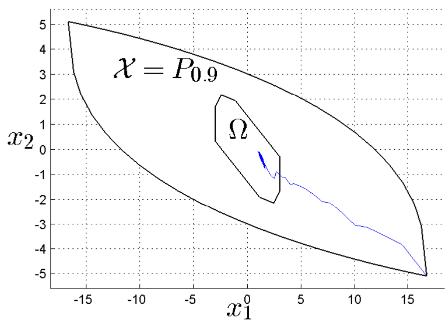


Figure 5.9: The maximal disturbance invariant set Ω and the domain of attraction $\mathcal{X} = P_{0.9}$.

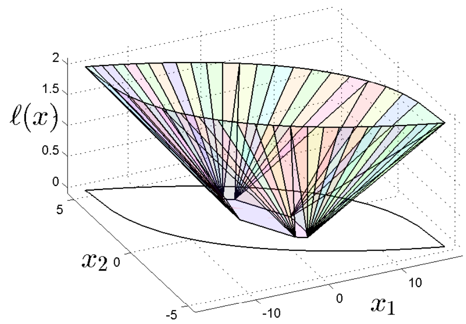


Figure 5.10: A convex lifting $\ell(x)$ constructed by Algorithm 5.1 with $c = 2$.

5.2.5 Illustrative example

For illustration, the double integrator system is considered. For simplicity, bounded additive disturbances are only taken into account:

$$\begin{aligned} x_{k+1} &= \begin{bmatrix} 1 & 0.5 \\ 0 & 1 \end{bmatrix} x_k + \begin{bmatrix} 0.125 \\ 0.5 \end{bmatrix} u_k + w_k \\ y_k &= \begin{bmatrix} 1 & 0 \end{bmatrix} x_k, \end{aligned} \tag{5.22}$$

subject to constraints:

$$-2 \leq u_k \leq 2, \begin{bmatrix} -20 \\ -20 \end{bmatrix} \leq x_k \leq \begin{bmatrix} 20 \\ 20 \end{bmatrix}, \|w_k\|_\infty \leq 0.4. \tag{5.23}$$

A stabilizing control law gain $K = [-0.8246 \ -1.5262]$ is chosen to compute the maximal robust positively (disturbance) invariant set Ω based on the algorithm proposed in Gilbert and Tan [1991]. Also, the maximal $0.9-$contractive set $P_{0.9}$ is computed, based on Algorithm 2.3. These two sets are shown in Figure 5.9. A convex lifting $\ell(x)$ is shown in Figure 5.10 according to Algorithm 5.1 with $c = 2$. Optimal controller which solves the linear programming problem (5.8), is presented in Figure 5.11. Accordingly, the discontinuous change of $u^*(x_k)$ at instant 12 is due to the discontinuity of optimal control to (5.8) while switching into Ω. The closed loop dynamics shown in Figure 5.9 illustrate the fact that this control law ensures robust stability in the sense of Lyapunov. Finally, Figure 5.12 visualizes the strict decrease of convex lifting $\ell(x)$ along the state over $\mathcal{X}\backslash\Omega$.

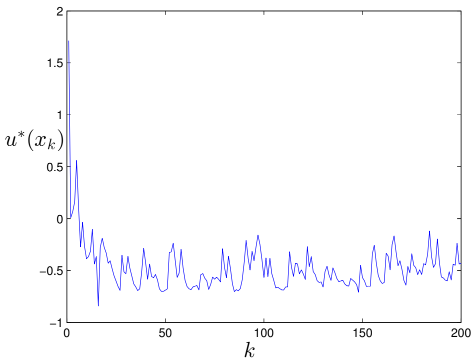


Figure 5.11: Optimal control law solves the linear programming problem (5.8) with random behavior of disturbances.

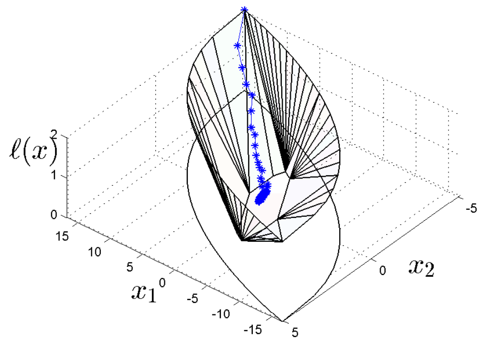


Figure 5.12: Strict decrease of $\ell(x)$ over $\mathcal{X}\backslash\Omega$ along the state.

5.3 Cascaded convex liftings based method

This section presents another method to design robust control in the presence of bounded additive disturbances and polytopic model uncertainties. Unlike the convex liftings based method presented in Section 5.2 which can only guarantee closed loop stability over a $\lambda-$contractive region for $0 \leq \lambda < 1$, this method can guarantee robust stability over the maximal controllable set as defined in Definition 2.4.2, known to be larger than the maximal $\lambda-$contractive set. The methodology still relies on a suitable control Lyapunov function. It is denoted as the *cascaded convex liftings based method* in this manuscript.

Again, consider the linear system (5.4) affected by bounded additive disturbances and polytopic uncertainties (5.5). Also, this system is assumed to be subject to constraints (5.6). Similar to Section 5.2, let Ω denote a full-dimensional robust positively invariant set associated with an unconstrained control law $u = Kx \in \mathbb{U}$ and $\mathcal{X}$ denote a domain of attraction. This domain of attraction is chosen as the $N-$steps controllable set i.e. $\mathcal{X} = \mathcal{K}_N(\Omega)$. As mentioned before, a construction of convex liftings as in Algorithm 5.1 over the domain of attraction $\mathcal{K}_N(\Omega)$ leads to difficulties ensuring strict decrease of such a convex lifting. Therefore, another construction for a Lyapunov candidate is necessary.

It is worth emphasizing that one can build an appropriate optimization problem by imposing:

$$x_k \in \mathcal{K}_i(\Omega),\ A(k)x_k + B(k)u_k + w_k \in \mathcal{K}_{i-1}(\Omega),\ \forall w_k \in \mathbb{W},\ [A(k)\ B(k)] \in \Psi.$$

Clearly, this construction can ensure that the state will be driven to Ω after N steps. Also, such a construction can guarantee the recursive feasibility due to the construction of $\mathcal{K}_N(\Omega)$. However, this construction has never been shown to be associated with a Lyapunov function. This limitation will be solved in the proposed procedure. Furthermore, such a control Lyapunov function may be non-convex which is a new feature to our best knowledge.

5.3.1 Cascaded convex liftings construction

This subsection aims to construct a cascade of convex liftings which serves as a control Lyapunov function. This construction is summarized in Algorithm 5.3.

Algorithm 5.3 Construction of a cascade of convex liftings

Input: A given robust positively invariant set $\Omega = \mathcal{K}_0(\Omega) \subset \mathbb{R}^{d_x}$, the domain of attraction $\mathcal{X} = \mathcal{K}_N(\Omega) \subset \mathbb{R}^{d_x}$ with a given $N \in \mathbb{N}_{>0}$ and a scalar $c > 0$.
Output: A cascade of convex liftings $\{\ell_i(x)\}_{i\in\mathcal{I}_N}$.

1: **For** $i = 1 : N$
2: $V_1 = \mathcal{V}(\mathcal{K}_{i-1}(\Omega)) \subset \mathbb{R}^{d_x}$, $\widehat{V}_1 = \left\{ \begin{bmatrix} v \\ (i-1)c \end{bmatrix} \mid v \in V_1 \right\} \subset \mathbb{R}^{d_x+1}$.
3: $V_2 = \mathcal{V}(\mathcal{K}_{i}(\Omega)) \subset \mathbb{R}^{d_x}$, $\widehat{V}_2 = \left\{ \begin{bmatrix} v \\ ic \end{bmatrix} \mid v \in V_2 \right\} \subset \mathbb{R}^{d_x+1}$.
4: $\Pi_i = \operatorname{conv}\left(\widehat{V}_1 \cup \widehat{V}_2\right)$.
5: Solve the parametric linear programming problem:

$$z^*(x) = \min_z z \quad \text{s.t.} \quad \begin{bmatrix} x^T & z \end{bmatrix}^T \in \Pi_i. \tag{5.24}$$

6: $\ell_i(x) = z^*(x)$.
7: **End**

Note that, unlike Algorithm 5.1, Algorithm 5.3 constructs a cascade of convex liftings. Each convex lifting is defined for two successive controllable sets $(\mathcal{K}_{i-1}(\Omega),\, \mathcal{K}_i(\Omega))$. This operation is carried out via steps 2–3, where the vertices of $\mathcal{K}_{i-1}(\Omega)$ are lifted to the same height e.g. $(i-1)c$, whereas the vertices of $\mathcal{K}_i(\Omega)$ are lifted to a higher level e.g. ic. Subsequently, a convex lifting for this pair of regions is computed via parametric linear programming at step 5. Such a convex lifting is denoted by $\ell_i(x)$. This procedure is repeated till the last pair of successive controllable regions $(\mathcal{K}_{N-1}(\Omega),\, \mathcal{K}_N(\Omega))$. Note also that through this construction, the domain of a convex lifting $\ell_i(x)$ is restricted in $\mathcal{K}_i(\Omega)$.

An important property of $\ell_i(x)$ is presented below.

Lemma 5.3.1 *The convex liftings generated from Algorithm 5.3 i.e.* $\ell_i(x)$ *are convex and continuous. Furthermore, for each* $i \in \mathcal{I}_N$, *one has* $ic \geq \ell_i(x) > (i-1)c$ *for all* $x \in \mathcal{K}_i(\Omega)\backslash\mathcal{K}_{i-1}(\Omega)$ *and* $\ell_i(x) = (i-1)c$ *for all* $x \in \mathcal{K}_{i-1}(\Omega)$.

Proof: The proof for the first claim is directly deduced from Lemma 5.2.1.

For the second claim, consider an $\ell_i(x)$, for $i \in \mathcal{I}_N$. Also, let $\left\{\mathcal{X}_j^{(i)}\right\}_{j\in\mathcal{I}_{N(i)}}$ denote the polyhedral partition associated with $\ell_i(x)$. In an explicit form, $\ell_i(x)$ is defined as follows:

$$\ell_i(x) = (a_j^{(i)})^T x + b_j^{(i)} \ \text{ for } \ x \in \mathcal{X}_j^{(i)}.$$

Clearly, there exists a $j \in \mathcal{I}_{N(i)}$ such that $\mathcal{X}_j^{(i)} = \mathcal{K}_{i-1}(\Omega)$, since $\mathcal{K}_{i-1}(\Omega)$ corresponds to a facet of Π_i. It is well known that any $x \in \mathcal{K}_{i-1}(\Omega)$ can be written as a convex combination of the vertices of $\mathcal{K}_{i-1}(\Omega)$, i.e.

$$x = \sum_{v\in\mathcal{V}(\mathcal{K}_{i-1}(\Omega))} \alpha(v)v, \ \ \alpha(v) \geq 0, \ \ \sum_{v\in\mathcal{V}(\mathcal{K}_{i-1}(\Omega))} \alpha(v) = 1.$$

This leads to the following for $x \in \mathcal{K}_{i-1}(\Omega)$:

$$\begin{aligned}
\ell_i(x) = (a_j^{(i)})^T x + b_j^{(i)} &= (a_j^{(i)})^T \sum_{v\in\mathcal{V}(\mathcal{K}_{i-1}(\Omega))} \alpha(v)v + b_j^{(i)} \\
&= \sum_{v\in\mathcal{V}(\mathcal{K}_{i-1}(\Omega))} \alpha(v)\left((a_j^{(i)})^T v + b_j^{(i)}\right) \\
&= \sum_{v\in\mathcal{V}(\mathcal{K}_{i-1}(\Omega))} \alpha(v)(i-1)c = (i-1)c.
\end{aligned}$$

Also, due to the convexity of $\ell_i(x)$, for any $x \in \mathcal{K}_i(\Omega)$ it can be observed that:

$$\begin{aligned}
\ell_i(x) &= \ell_i\left(\sum_{v\in\mathcal{V}(\mathcal{K}_i(\Omega))} \alpha(v)v\right) \\
&\leq \sum_{v\in\mathcal{V}(\mathcal{K}_i(\Omega))} \alpha(v)\ell_i(v) = \sum_{v\in\mathcal{V}(\mathcal{K}_i(\Omega))} \alpha(v)ic = ic.
\end{aligned}$$

To complete the proof, it is necessary to prove that $\ell_i(x) > (i-1)c$ for all $x \in \mathcal{K}_i(\Omega)\backslash\mathcal{K}_{i-1}(\Omega)$. In fact, consider any $s \in \mathcal{I}_{N(i)}$, $s \neq j$, due to the properties of a convex lifting defined in Definition 4.2.3:

$$\begin{aligned}
(a_s^{(i)})^T x + b_s^{(i)} &> (a_j^{(i)})^T x + b_j^{(i)} \ \text{ for } \ x \in \mathcal{X}_s^{(i)}\backslash\mathcal{K}_{i-1}(\Omega) \\
(a_s^{(i)})^T x + b_s^{(i)} &= (a_j^{(i)})^T x + b_j^{(i)} \ \text{ for } \ x \in \mathcal{X}_s^{(i)}\cap\mathcal{K}_{i-1}(\Omega).
\end{aligned} \tag{5.25}$$

Note that $\ell_i(x) = (i-1)c$ for all $x \in \mathcal{K}_{i-1}(\Omega)$ and $\mathcal{K}_{i-1}(\Omega)$ is full-dimensional, lead to the fact that:

$$a_j^{(i)} = 0, \quad b_j^{(i)} = (i-1)c. \tag{5.26}$$

From (5.25) and (5.26), it follows that

$$\ell_i(x) = (a_s^{(i)})^T x + b_s^{(i)} > (i-1)c \quad \text{for all} \quad x \in \mathcal{X}_s^{(i)} \backslash \mathcal{K}_{i-1}(\Omega). \tag{5.27}$$

It is worth emphasizing that (5.27) holds true for all $s \in \mathcal{I}_{N^{(i)}}$, $s \neq j$, meaning $\ell_i(x) > (i-1)c, \quad$ for all $\; x \in \mathcal{K}_i(\Omega) \backslash \mathcal{K}_{i-1}(\Omega)$. The proof is complete. □

The following property is a direct consequence of Lemma 5.3.1

Lemma 5.3.2 *For each $i \in \mathcal{I}_N$ and $0 \leq \beta \leq 1$, the following holds true:*

$$\ell_i(\beta x) \leq \beta \ell_i(x) + (1-\beta)(i-1)c, \quad \textit{for any} \quad x \in \mathcal{K}_i(\Omega). \tag{5.28}$$

Proof: Indeed, due to the convexity of $\ell_i(x)$ over $\mathcal{K}_i(\Omega)$, it follows that:

$$\ell_i(\beta x) = \ell_i(\beta x + (1-\beta)0) \leq \beta \ell_i(x) + (1-\beta)\ell_i(0). \tag{5.29}$$

Note that $0 \in \text{int}(\Omega)$, then $0 \in \text{int}(\mathcal{K}_i(\Omega))$ for all $i \in \mathcal{K}_i(\Omega)$. It can be deduced that $\ell_i(0) = (i-1)c$. Therefore, (5.29) leads to the conclusion. □

5.3.2 Robust control design procedure

Based on the above construction, this subsection aims to present a so-called *cascaded convex liftings based method* for robust control design for the linear system (5.4). For ease of presentation, define the following augmented system:

$$\begin{aligned} x_{k+1} &= A(k)x_k + B(k)u_k + w_k \\ z_{k+1} &= \alpha_k z_k \end{aligned} \tag{5.30}$$

where $z_k \in \mathbb{R}$ denotes an auxiliary state variable, $\alpha_k \in \mathbb{R}_+$ denotes an auxiliary control variable. Still, u_k, x_k, w_k are subject to constraints (5.6) and $[A(k) \;\, B(k)]$ belongs to Ψ defined in (5.5).

Without loss of generality, the domain of attraction in this section is restricted to $\mathcal{K}_N(\Omega)$. Define the following optimization problem:

$$\begin{aligned} \left[\alpha_k^* \; (u_k^*)^T\right]^T &= \arg \min_{\left[\alpha_k \; u_k^T\right]^T} \alpha_k \\ \text{s.t.} \;\; \ell_i(A(k)x_k + B(k)u_k + w_k) &\leq \alpha_k z_k, \;\; \alpha_k \geq 0, \;\; u_k \in \mathbb{U} \\ \forall \, [A(k) \;\, B(k)] \in \mathcal{V}(\Psi), \;\; &\forall w_k \in \mathcal{V}(\mathbb{W}). \end{aligned} \tag{5.31}$$

First, an intermediate but important result is introduced below.

Theorem 5.3.3 *Consider the system (5.30) and the optimization problem (5.31). The followings hold true:*

1. *if* $i = 1$ *and* $x_k \in \mathcal{K}_1(\Omega)$ *then* (5.31) *is feasible for any finite* $z_k \in \mathbb{R}$ *and* $z_{k+1} = 0$,
2. *if* $z_k \leq 0$ *and* $x_k \in \mathcal{K}_i(\Omega)$ *then* (5.31) *is infeasible for all* $i \in \mathcal{I}_N \backslash \{1\}$,
3. *if* $0 < z_k \leq (i-1)c$, $i \geq 2$ *and* $x_k \in \mathcal{K}_i(\Omega)$ *then* $\alpha_k^* \geq 1$,
4. *if* $z_k > (i-1)c$ *and* $x_k \in \mathcal{K}_i(\Omega)$ *then* $z_{k+1} = (i-1)c$ *and* $\alpha_k^* < 1$.

Proof: 1. If $i = 1$, then for any $x_k \in \mathcal{K}_1(\Omega)$ there exists $u(x_k) \in \mathbb{U}$ such that

$$A(k)x_k + B(k)u(x_k) + w_k \in \Omega, \ \forall [A(k) \ B(k)] \in \Psi, \ \forall w_k \in \mathbb{W}.$$

Note also that

$$0 = \ell_1(A(k)x_k + B(k)u(x_k) + w_k) \geq \ell_1(A(k)x_k + B(k)u_k^* + w_k) \geq 0.$$

Therefore, the optimization problem (5.31) is feasible only when $\alpha_k^* = 0$ or $z_k = 0$ leading to $z_{k+1} = \alpha_k^* z_k = 0$. Claim 1 is proved.

2. From the proof of Lemma 5.3.1, $\ell_i(x) \geq (i-1)c > 0$ over $\mathcal{K}_i(\Omega)$ and for all $i \in \mathcal{I}_N \backslash \{1\}$. Accordingly, if $z_k \leq 0$, constraint

$$0 < \ell_i(A(k)x_k + B(k)u_k + w_k) \leq \alpha_k z_k \leq 0$$

is infeasible for all $\alpha_k \geq 0$, leading to the infeasibility of (5.31). Therefore, claim 2 is proved.

4. Again, from Lemma 5.3.1, $\ell_i(x_k) \geq (i-1)c$ for all $x_k \in \mathcal{K}_i(\Omega)$. As defined before, for any $x_k \in \mathcal{K}_i(\Omega)$, there exists $u(x_k) \in \mathbb{U}$ such that

$$A(k)x_k + B(k)u(x_k) + w_k \in \mathcal{K}_{i-1}(\Omega), \ \forall [A(k) \ B(k)] \in \Psi, \ \forall w_k \in \mathbb{W}.$$

Also, it can be observed that

$$(i-1)c = \ell_i(A(k)x_k + B(k)u(x_k) + w_k) \geq \ell(A(k)x_k + B(k)u_k^* + w_k) \geq (i-1)c.$$

Therefore, $\ell(A(k)x_k + B(k)u_k^* + w_k) = (i-1)c$ for all $[A(k) \ B(k)] \in \Psi$ and for all $w_k \in \mathbb{W}$. Accordingly, since $(i-1)c = \alpha_k^* z_k = z_{k+1}$, then $z_k > (i-1)c$ leads to $\alpha_k^* < 1$. Claim 4 is completely proved.

3. Since $(i-1)c = \alpha_k^* z_k = z_{k+1}$, if $0 < z_k \leq (i-1)c$, then clearly $\alpha_k^* \geq 1$. The proof for claim 3 is complete. □

It is worth emphasizing that the choice of a suitable initial condition for z_k is also of importance for the feasibility of the optimization problem (5.31). This formulation will be the basis in the cascaded convex liftings based method. Apart from the aim to design robust control for the linear system (5.4), it is also important to show that z_k will serve as a Lyapunov candidate. Accordingly, to prove

---

Algorithm 5.4 Cascaded convex liftings based design procedure

---

Input: Convex liftings $\{\ell_i(x)\}_{i\in\mathcal{I}_N}$ and an unconstrained robust control: $u_k = Kx_k$ and $\mathcal{K}_N(\Omega)$.

Output: A robust control $u^*(x_k)$ and α_k^*.

1: **If** $k = 0$ **then** Initialize $z_0 = Nc$ and any $x_0 \in \mathcal{K}_N(\Omega)$.

2: **If** $k \leq N-1$ **then** Solve the following optimization problem:

$$\begin{aligned} \left[\alpha_k^* \; (u_k^*)^T\right]^T &= \arg \min_{\left[\alpha_k \; u_k^T\right]^T} \alpha_k \\ \text{s.t. } &\ell_{N-k}(A(k)x_k + B(k)u_k + w_k) \leq \alpha_k z_k, \;\; \alpha_k \geq 0, \;\; u_k \in \mathbb{U}, \\ &\forall\, [A(k)\;\; B(k)] \in \mathcal{V}(\Psi), \;\; \forall w_k \in \mathcal{V}(\mathbb{W}). \end{aligned} \tag{5.32}$$

3: $\quad u^*(x_k) = u_k^*$.

4: **Else** $u^*(x_k) = Kx_k$.

5: **End**

6: $k \leftarrow k+1$, Return to step 2.

---

strict decrease of z_k is a critical step. From four different initial conditions for z_k in Theorem 5.3.3, the last inclusion turns out to fulfill this requirement. This aspect will be clarified in the sequel.

Based on the construction of cascaded convex liftings in Algorithm 5.3 and the above results, the design procedure is summarized in Algorithm 5.4.

The following theorem clarifies that the control law designed from Algorithm 5.4 guarantees closed-loop stability.

Theorem 5.3.4 *Consider the linear system* (5.4) *subject to constraint* (5.6) *and model uncertainties* (5.5)*. The control law computed from Algorithm 5.4 ensures robust stability in the sense of Lyapunov. Moreover, z_k serves as a Lyapunov function.*

Proof: It is important to prove that $z_{k+1} < z_k$ for $0 \leq k \leq N-1$. Indeed, for any $x_0 \in \mathcal{K}_N(\Omega)$, according to claim 4 in Theorem 5.3.3:

$$\begin{aligned} z_0 = Nc > (N-1)c = \ell_N(A(0)x_0 + B(0)u^*(x_0) + w_0) = \alpha_0^* z_0 = z_1, \\ \forall\, [A(0)\;\; B(0)] \in \Psi, \;\; \forall w_0 \in \mathbb{W}. \end{aligned} \tag{5.33}$$

Similar to this inclusion, it can also be proved that

$$z_k = (N-k)c > (N-k-1)c = z_{k+1} \;\; \text{for all} \;\; 0 \leq k \leq N-1.$$

Otherwise, for $k \geq N$ $z_k = 0$ due to the fact that $x_k \in \Omega$. Therefore, z_k serves as a Lyapunov candidate, meaning closed-loop stability in the sense of Lyapunov. □

Remark 5.3.5 Note that the initial condition for z_0 in Algorithm 5.4 aims to fulfill the condition of claim 4 for all $i \in \mathcal{I}_N$. Clearly, any $z_0 \geq Nc$ is also possible.

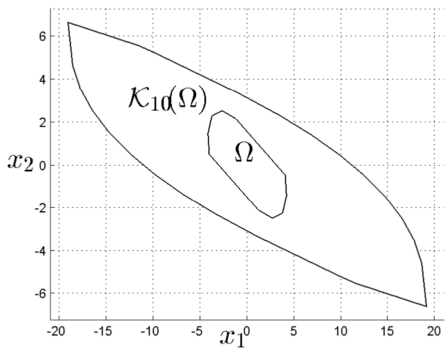


Figure 5.13: The maximal robust positively invariant set Ω and $\mathcal{K}_{10}(\Omega)$.

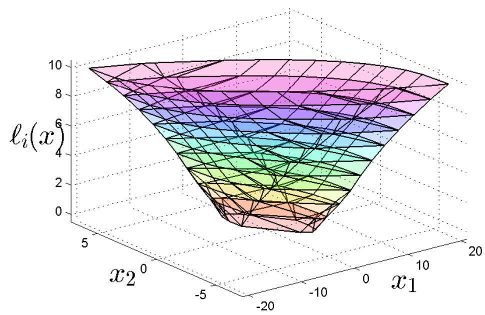


Figure 5.14: Cascaded convex liftings.

5.3.3 Illustrative example

To illustrate this cascaded convex liftings based method, consider again the double integrator system (5.22) subject to constraint (5.23). Unconstrained control law is chosen as: $u_k = [-0.6514 \ \ -1.3142]\, x_k$. Associated with this unconstrained control, the maximal robust positively invariant set Ω computed by the algorithm proposed in Gilbert and Tan [1991] and $\mathcal{K}_{10}(\Omega)$ are presented in Figure 5.13. Also, the cascade of convex liftings computed by Algorithm 5.4 with $c = 1$ is shown in Figure 5.14. Finally, Figure 5.15 illustrates the dynamics of z_k along the state depicted by the blue curve, proving that z_k tends to 0 after 10 steps. In other words, optimal control laws designed by Algorithm 5.4 ensure robust stability in the sense of Lyapunov.

5.4 Inverse optimality to improve robustness in constrained control

This section presents an idea for use of inverse optimality for control design of constrained linear systems. Although, the control design does not provide a complete methodology for tuning, it points to new open problems and promising advantages in control design.

Again the linear system (5.4) is considered. It is assumed to be affected by model uncertainties Ψ as in (5.5) and subject to constraints (5.6).

The idea for design procedure is summarized as follows:

1. Compute a robust unconstrained control law $u_k = Kx_k$ which can cope with polytopic uncertainties (5.5) and the given bounded additive disturbances.
2. Choose a nominal system $[A\ B] \in \Psi$.
3. Use this nominal system and robust unconstrained control $u_k = Kx_k$ to compute weighting matrices Q, R for a nominal MPC.
4. Use $[A\ B]$ and these matrices Q, R to solve a nominal MPC problem and obtain explicit solution.
5. Analyze the robustness of the resulted control law.

The first step can be computed as in Subsection 2.4.1. For a given nominal system

$$x_{k+1} = Ax_k + Bu_k,$$

where $(A,\ B) \in \Psi$ is controllable and a given stabilizing control law $u_k = -Kx_k$, an inverse optimality problem aims to find a positive semidefinite $Q \geq 0$ and a

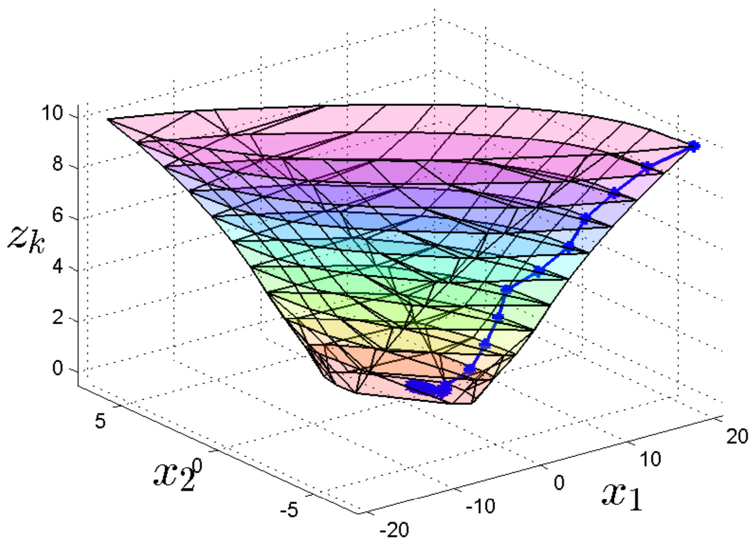


Figure 5.15: Dynamic of z_k along the state.

positive definite $R > 0$ such that $u_k = -Kx_k$ minimizes the following quadratic infinite criterion:

$$J_{Q,R} = \sum_{k=0}^{\infty} x_k^T Q x_k + u_k^T R u_k. \tag{5.34}$$

To solve this problem, Larin [2003] proposed a formulation by using linear matrix inequalities, this formulation is recalled as follows:

$$\min_{\lambda,S,Y,R,P} \lambda$$

s.t:

$$\begin{aligned} P &\geq 0 \\ A^T PA - P - K^T RK - K^T B^T PBK &\leq 0 \\ \begin{bmatrix} Y & S \\ S^T & I \end{bmatrix} &\geq 0 \\ Y &\leq \lambda I \\ S = RK + B^T PBK - B^T PA. \end{aligned} \tag{5.35}$$

The matrices Q, R, P stem from the Riccati equation:

$$A^T PA - P - K^T RK - K^T B^T PBK + Q = 0.$$

In fact, the resolution of (5.35) amounts to an approximation of the solution instead of the exact one. The Riccati equation is not directly solved but the error between the left and the right hand side is minimized. The second LMI in (5.35) relies on the assumption that $Q \geq 0$. From the solution of the Riccati equation:

$$K = (B^T PB + R)^{-1} B^T PA,$$

it follows that S has to be zero, so this LMI problem aims at minimizing λ in order to ensure that S is as close as possible to zero by the relationship: $\lambda I \geq Y \geq SS^T \geq 0$.

Based on the resolution of inverse optimality, the fourth step is to use these inclusions to design a PWA control law which can guarantee model uncertainties and bounded additive disturbances in a subset of the state space. The usual design procedure is recalled here with a prediction horizon $N \in \mathbb{N}_{>0}$ with respect to the nominal system chosen at step 2:

$$\min_{\mathbf{u}} J(x_k, \mathbf{u}) = \min_{\mathbf{u}} \sum_{i=0}^{N-1} (x_{k+i|k}^T Q x_{k+i|k} + u_{k+i|k}^T R u_{k+i|k}) + x_{k+N|k}^T P x_{k+N|k},$$

subject to

$$\begin{aligned}
&x_{k+i|k} \in \mathbb{X},\ \ u_{k+i|k} \in \mathbb{U},\ \ \forall\ 0 \leq i \leq N-1,\\
&x_{k+N|k} \in \mathbb{X}_f,
\end{aligned}$$

where Q, R are the solution from the LMI problem (5.35), $\mathbb{X}_f$ denotes the terminal constraints set to ensure stability for the nominal system and the optimization argument is defined as follows:

$$\mathbf{u} = \left[u_{k|k}^T \ldots u_{k+N-1|k}^T\right]^T.$$

Accordingly, the first input $u_k^* = \mathbf{u}^*(1 : d_u, \cdot)$ is applied to the system. Let $u_{pwa}(x)$ denote this explicit optimal controller defined over a feasible region $\mathcal{X}$. It can be observed that the control law associated with the central region i.e. the region containing the origin, is in the form $u_k = -Kx_k$ where K is the unconstrained robust control law gain computed at step 1. Therefore, there exists a subset of this central region over which this unconstrained control law is able to cope with the model uncertainties Ψ and the given bounded additive disturbances. The maximal robust positively invariant set associated with this control law is a candidate for this subset. A computation for this invariant set has been presented in Algorithm 2.1. Further, the control law $u_{pwa}(x)$ computed from the inverse optimality procedure, possesses a *robustness margin* which can be readily computed as in Chapter 3. These observations are stated in the following theorem.

Theorem 5.4.1 *Given system* (5.4), $u_{pwa}(x)$ *is the piecewise affine regulator, obtained as the solution of a nominal MPC with weighting parameters obtained via inverse optimality, then there exists* at least one *realization* $\left[\bar{A}\ \ \bar{B}\right] \in \Psi$ *such that the feasible set* $\mathcal{X}$ *is positively invariant and it contains a local region* $\mathcal{X}_{rob}$ *which is positively invariant for all* $[A(k)\ \ B(k)] \in \Psi$ *and all* $w_k \in \mathbb{W}$.

Proof: $\left[\bar{A}\ \bar{B}\right]$ denotes the nominal system by which $u_{pwa}(x)$ is synthesized. Accordingly, $u_{pwa}(x)$ can stabilize this nominal system in the feasible region $\mathcal{X}$. The set of these realizations for which $u_{pwa}(x)$ can stabilize is defined as *robustness margin* and is computed in Section 3.3.

For the second claim, it is known that there always exists a robust positively invariant set associated with a stabilizing control law. Particularly, the unconstrained control law is designed at step 1 to cope with the given polytopic uncertainties and bounded additive disturbances. □

It is worth recalling that this design can only improve the robustness of a PWA controller designed from a nominal MPC problem in a subset of central region. This has potentials to be completed with a re-tuning of the designed control law by adjustment of the nominal system $[A\ B]$, used in this design procedure. The main open problem is to find criteria for the characterization of the nominal system at step 2 such that the robustness margin of the resulted explicit controller is maximal. Also, as applied to many different systems, inverse optimality should be exploited to design robust control for linear system affected by polytopic uncertainties and bounded additive disturbances.

Chapter 6

Conclusions

This thesis has discussed certain topics related to implementation and design of PWA control laws.

6.1 Contributions

Implementation of PWA control laws:

- The first chapter of this thesis has discussed characterization and computations of the robustness and fragility margins for a given continuous PWA control law. Two different approaches have been presented for these computations. These margins were obtained as polyhedra. With respect to PWA controllers' implementation, these margins provide valuable tools for the evaluation of PWA controllers under finite precision arithmetic while closed loop stability is still guaranteed.
- The second chapter aims at finding solution to inverse parametric linear/quadratic programming problem. This solution was presented relying on convex liftings concept. Some advantages of this solution can be outlined as follows:
  - Implementing online the recovered optimization problem was shown to be useful to reduce online optimization time in comparison with the original optimization problem. This implementation can avoid substantial memory to store state space partition and control law gains.
  - The construction of a convex lifting is also of help to solve the point-location problem without storing the state space partitions. Accordingly, one can implement this convex lifting and the given PWA controller. The point-location problem reduces to a simple evaluation i.e. determine the maximal value among the affine functions composing this convex lifting at the current state. The index of this affine function is later used to find corresponding affine controller to be evaluated.

This allows the PWA controllers to be implemented into low-cost platforms.

Linear model predictive control

— Based on the result of inverse parametric linear/quadratic programming problem via convex liftings, a theoretical result in the case of linear model predictive control has been proved. Accordingly, it has been shown that any continuous PWA controller can be equivalently obtained via a linear MPC problem with the control horizon at most equal to 2 prediction steps.

Design of robust controllers

Two methods to design robust control laws have been put forward for linear system affected by bounded additive disturbances and polytopic model uncertainties.

— Robust control design based on convex liftings has been shown to be simple and able to design both implicit and explicit controllers. Such a convex lifting has been shown to be a control Lyapunov function defined over the maximal $\lambda-$contractive set for a $0 \leq \lambda < 1$. Also, this methods has been shown to ensure the recursive feasibility and robust stability in the sense of Lyapunov.

— Robust control design based on a cascade of convex liftings has been characterized. This method extends the feasible region to the $N-$steps controllable set known not to be contractive. Accordingly, an auxiliary variable has been presented to emulate a Lyapunov function. Namely, this auxiliary is non-negative, strictly decreasing for N first sampling instants and stays at 0 afterwards.

6.2 Future works

For future work, some directions can be outlined as follows:

Robust control based on convex liftings

Convex liftings have been shown to be control Lyapunov function defined over a contractive set. Also, a cascade of convex liftings, but not convex, has been of use to design robust control over the $N-$steps controllable set. Some open ideas can similarly be exploited:

— Construct a convex lifting defined over the $N-$steps controllable set as a control Lyapunov function. Many studies have focused on this problem by separating the $N-$steps controllable set into a convexly liftable partition and searching for a stabilizing PWA controller defined over this partition with respect to a chosen control Lyapunov function. However, these methods cannot guarantee the feasibility since they did not exploit the property of the $N-$steps controllable set. Similar to a convex lifting defined over a contractive set, this Lyapunov candidate should ensure the feasibility of the method. Later, such a convex lifting should be of use to design both implicit and explicit robust controllers.

— Based on the same methodology, such a control Lyapunov function should be constructed over an ellipsoidal contractive region and of use to design robust control.

— Output feedback control can also be studied based on these convex liftings based methods.

Piecewise affine systems

Applying the above convex lifting based methods for generic piecewise affine systems can also be exploited.

Model predictive control

Nonlinear model predictive control has reached a certain maturity in numerical solvers. However, its geometric properties have not been studied yet at the same level of understanding as linear MPC. In case of linear model predictive control, it is shown that geometrical properties of linear constrains can be exploited to find so-called explicit controllers. Another particular class of constraints should also be exploited i.e. mixed linear and quadratic constraints. Their geometrical properties should be applied to MPC.

Chapter 7

Appendices

Contents

This chapter aims at presenting some extensions of the results presented in Chapter 4. These are mostly extracted from Nguyen et al. [2015b,d].

7.1 Applications to parametric linear/quadratic programming problems

The aim of this subsection is to present the main results related to parametric linear/quadratic programming problems which can be subsequently related to control design problems. The definition of a parametric linear/quadratic programming problem was introduced in Section 4.3.1.

Recall that a parametric linear programming problem can be written in the following form:

$$u^*(x) = \arg\min_u C^T u \ \text{ s.t. } Gu \leq W + Ex, \tag{7.1}$$

where $u \in \mathbb{R}^{d_u}, x \in \mathbb{R}^{d_x}$ denote the decision variable and the parameter, respectively. It is shown in Gal [1995] that the optimal cost function $C^T u^*(x)$ is a continuous, convex PWA function. We recall this result below.

Theorem 7.1.1 *The optimal cost function of* (7.1) *i.e.* $C^T u^*(x)$, *is a continuous, convex, PWA function.*

Proof: The proof is based on Theorems IV-3 and IV-4 in Gal [1995]. □

Suppose optimal solution to (7.1) is denoted as follows

$$u^*(x) = H_i x + G_i \quad \text{for} \quad x \in \mathcal{X}_i.$$

It is worth emphasizing that $C^T u^*(x)$ is not a convex lifting defined over the associated partition, in many cases. Accordingly, one cannot directly use this optimal cost function for efficient implementation of PWA controller defined over this partition to avoid the storage of the state space partition, as advocated in Baotic et al. [2008]. More precisely, the key point is to write a continuous, convex, real-valued PWA function in an equivalent form as follows:

$$\begin{aligned} C^T u^*(x) &= C^T(H_i x + G_i) \text{ if } x \in \mathcal{X}_i, \\ &= \max_{j \in \mathcal{I}_N} C^T(H_j x + G_j). \end{aligned} \tag{7.2}$$

In case $C^T u^*(x)$ is a convex lifting for the cell complex $\{\mathcal{X}_i\}_{i \in \mathcal{I}_N}$, efficient implementation of PWA controller associated with this cell complex is summarized in Algorithm 7.1.

Otherwise, if there exist two regions $(\mathcal{X}_i, \mathcal{X}_j)$, $i \neq j$, $(i, j) \in \mathcal{I}_N^2$ such that

$$\begin{aligned} (C^T H_i, C^T G_i) &= (C^T H_j, C^T G_j), \\ (H_i, G_i) &\neq (H_j, G_j), \end{aligned}$$

Algorithm 7.1 Efficient implementation of PWA controllers

1: Store (H_i, G_i) and $(C^T H_i, C^T G_i)$
2: At each sampling time, measure the state x
3: Find the index $i \in \mathcal{I}_N$ such that:

$$C^T H_i x + C^T G_i = \max_{j \in \mathcal{I}_N} C^T(H_j x + G_j).$$

4: Inject controller $u = H_i x + G_i$.
5: Return to step 2.

Algorithm 7.1 is not applicable. To illustrate the singularity of this case, consider the following parametric linear programming problem:

$$\min_z z$$
$$\begin{bmatrix} 0.3310 & -0.7486 \\ 0.1286 & 0.6497 \\ -0.0738 & -0.5372 \\ -0.4472 & 0 \\ -0.1881 & 0 \\ -0.2703 & 0.2899 \\ -0.2872 & 0.2482 \\ -0.1881 & 0 \\ -0.0823 & -0.5171 \\ -0.2448 & -0.1599 \end{bmatrix} \begin{bmatrix} z \\ u \end{bmatrix} \leq \begin{bmatrix} -0.0152 \\ 0.0132 \\ -0.7892 \\ 0 \\ 0.9407 \\ 0.8250 \\ 0.7993 \\ -0.9407 \\ -0.7973 \\ -0.7950 \end{bmatrix} x + \begin{bmatrix} 0.5743 \\ 0.7491 \\ -0.2884 \\ -0.8944 \\ -0.2822 \\ -0.4029 \\ -0.4659 \\ -0.2822 \\ -0.3002 \\ -0.5315 \end{bmatrix}. \tag{7.3}$$

Optimal solution to the parametric linear programming problem (7.3) is presented below:

$$\begin{bmatrix} z^*(x) \\ u^*(x) \end{bmatrix} = \begin{cases} \begin{bmatrix} -5 \\ -2.1908 \end{bmatrix} x + \begin{bmatrix} 1.5 \\ -0.1039 \end{bmatrix} & \text{for } \mathcal{X}_1 = \{x \in \mathbb{R} \mid -0.3 \leq x \leq -0.1\} \\ \begin{bmatrix} 0 \\ 1.4692 \end{bmatrix} x + \begin{bmatrix} 2 \\ 0.2621 \end{bmatrix} & \text{for } \mathcal{X}_2 = \{x \in \mathbb{R} \mid -0.1 \leq x \leq 0\} \\ \begin{bmatrix} 0 \\ 4.9723 \end{bmatrix} x + \begin{bmatrix} 2 \\ 0.2621 \end{bmatrix} & \text{for } \mathcal{X}_3 = \{x \in \mathbb{R} \mid 0 \leq x \leq 0.1\} \\ \begin{bmatrix} 5 \\ -2.6842 \end{bmatrix} x + \begin{bmatrix} 1.5 \\ 1.0278 \end{bmatrix} & \text{for } \mathcal{X}_4 = \{x \in \mathbb{R} \mid 0.1 \leq x \leq 0.2\} \\ \begin{bmatrix} 5 \\ 0.7461 \end{bmatrix} x + \begin{bmatrix} 1.5 \\ 0.3417 \end{bmatrix} & \text{for } \mathcal{X}_5 = \{x \in \mathbb{R} \mid 0.2 \leq x \leq 0.3\}. \end{cases}$$

The optimal cost function is shown in Figure 7.1, whereas optimal solution to (7.3) is visualized in Figure 7.2. Notice that this optimal cost function is not a convex lifting for the associated cell complex since the same affine function $z^*(x) = 2$, is defined over two neighboring regions $[-0.1\ 0]$ and $[0\ 0.1]$. Note also that if one implements Algorithm 7.1, Step 3 results in either $i = 2$ or $i = 3$, while controllers associated with $\mathcal{X}_2, \mathcal{X}_3$ are different. In this case, clearly this algorithm can return different values of controllers at the current state. Accordingly, the given PWA controller is not properly implemented. It is worth emphasizing that optimal solution of (7.3) is not unique.

We will prove next that if the uniqueness of the optimal solution to a parametric linear programming problem is fulfilled, then the optimal cost function represents a convex lifting for the associated cell complex.

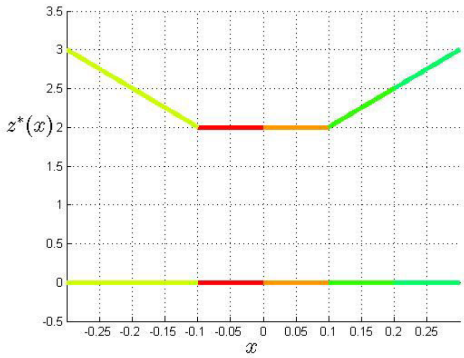


Figure 7.1: The optimal cost function of the parametric linear programming problem (7.3).

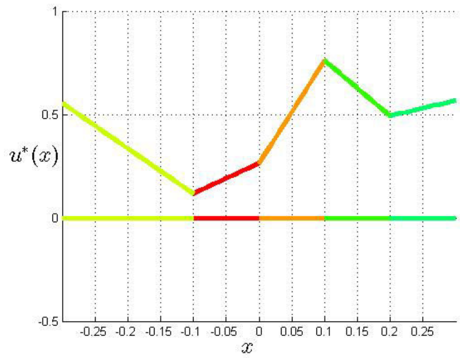


Figure 7.2: Optimal solution to the parametric linear programming problem (7.3).

Theorem 7.1.2 *If the optimal solution to a parametric linear programming problem is unique, then the associated parameter space partition admits affinely equivalent polyhedra.*

Proof: As per Theorem 7.1.1, the optimal cost function of (7.1) is a continuous, convex, PWA function defined over the associated partition. Thus, it suffices to prove that such a case like (7.3) cannot happen. More precisely, we will prove that if the optimal solution to (7.1) is unique, then the existence of two regions $\mathcal{X}_i, \mathcal{X}_j$, $i \neq j$, $(i,j) \in \mathcal{I}_N^2$ such that

$$(H_i, G_i) \neq (H_j, G_j),\ (C^T H_i, C^T G_i) = (C^T H_j, C^T G_j),$$

is impossible.

In fact, suppose the converse situation takes place. Consider $x_1 \in \operatorname{int}(\mathcal{X}_i)$, $x_2 \in \mathcal{X}_j$ and a scalar $\alpha \in [0,1]$. Due to the convexity of $C^T u^*(x)$, advocated in Theorem 7.1.1, we can see that

$$C^T u^*(\alpha x_1 + (1-\alpha)x_2) \leq \alpha C^T(H_i x_1 + G_i) + (1-\alpha)C^T(H_j x_2 + G_j). \quad (7.4)$$

If we choose α close to 1 such that $\alpha x_1 + (1-\alpha)x_2 \in \mathcal{X}_i$, then

$$C^T u^*(\alpha x_1 + (1-\alpha)x_2) = C^T(H_i(\alpha x_1 + (1-\alpha)x_2) + G_i). \quad (7.5)$$

Note also that according to the assumption $(C^T H_i, C^T G_i) = (C^T H_j, C^T G_j)$, it follows that:

$$\alpha C^T(H_i x_1 + G_i) + (1-\alpha)C^T(H_j x_2 + G_j) = C^T H_i(\alpha x_1 + (1-\alpha)x_2) + C^T G_i. \quad (7.6)$$

Also, since $H_ix_1 + G_i, H_jx_2 + G_j$ satisfy the constraint set in (7.1), so does $\alpha(H_ix_1 + G_i) + (1-\alpha)(H_jx_2 + G_j)$. According to (7.4), (7.5), (7.6), $\alpha(H_ix_1 + G_i) + (1-\alpha)(H_jx_2 + G_j)$ is also an optimal solution to (7.1). Due to the uniqueness of the optimal solution to (7.1), we obtain the following:

$$H_i(\alpha x_1 + (1-\alpha)x_2) + G_i = \alpha(H_ix_1 + G_i) + (1-\alpha)(H_jx_2 + G_j),$$

leading to:

$$H_ix_2 + G_i = H_jx_2 + G_j. \tag{7.7}$$

It is worth emphasizing that (7.7) holds true for all $x_2 \in \mathcal{X}_j$. Since $(H_i, G_i) \neq (H_j, G_j)$, the set of $x \in \mathbb{R}^{d_x}$ satisfying $H_ix + G_i = H_jx + G_j$ represents a polyhedron of dimension lower than d_x, while $\mathcal{X}_j$ is a full dimensional polyhedron in $\mathbb{R}^{d_x}$. This is clearly contradictory. Therefore, the initial hypothesis is not true. In other words, in case the optimal solution to (7.1) is unique, the optimal cost function $C^T u^*(x)$ describes a convex lifting for the associated cell complex, thus leading to the existence of an affinely equivalent polyhedron. □

As mentioned previously, the parameter space partition, associated with the optimal solution to a *parametric quadratic programming problem*, is not convexly liftable in many cases. However, based on the lifting properties, a result related to parametric linear/quadratic programming problem can be stated as follows:

Corollary 7.1.3 *Any parameter space polyhedral partition associated with optimal solution of a parametric linear/quadratic programming problem can be subdivided such that the internal boundaries of the initial partition are preserved and the convex liftability of the new cell complex is guaranteed.*

Proof: The proof is straightforward from Theorem 4.4.10. □

7.2 Inverse parametric linear programming for continuous set-valued maps

This section presents some additional results of inverse parametric linear/quadratic programming problem for the class of continuous piecewise affine set-valued maps. Accordingly, some auxiliary notations about set-valued analysis need to be recalled.

Let X, Y be two metric spaces, a *set-valued map* is denoted by $F : X \rightsquigarrow Y$ i.e. mapping each point of X to a subset of Y. Also, the definition of the graph of F denoted by $\mathrm{Graph}(F)$, is defined as follows:

$$\mathrm{Graph}(F) := \{(x, y) \in X \times Y \mid y \in F(x)\}.$$

A set-valued map $F : X \rightsquigarrow Y$ is called *upper semicontinuous* at $x \in X$ if for any open neighborhood $\mathcal{U}$ of $F(x)$ i.e. $F(x) \subset \mathcal{U}$, there exists an open neighborhood $\mathcal{M}$ of x such that $F(x') \subset \mathcal{U}$ for all $x' \in \mathcal{M}$. Moreover, F is called *lower semicontinuous* if for any open set $\mathcal{N} \subset Y$ and $\mathcal{N} \cap F(x) \neq \emptyset$, there exists an open neighborhood $\mathcal{M}$ of x such that $F(x') \cap \mathcal{N} \neq \emptyset$ for all $x' \in \mathcal{M}$. Finally, F is called *continuous* if it is both upper semicontinuous and lower semicontinuous.

Based on these notions, a piecewise affine set-valued map is defined as follows. Given a polyhedral partition $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ of a polyhedron $\mathcal{X} \subseteq \mathbb{R}^{d_x}$, a set-valued map $F : \mathcal{X} \rightsquigarrow \mathcal{Y} \subseteq \mathbb{R}^{d_u}$ is called *piecewise affine* if over each region $\mathcal{X}_i$ in the given polyhedral partition, $\mathrm{Graph}(F|_{\mathcal{X}_i})$ is a polyhedron, where $F|_{\mathcal{X}_i}$ represents set-valued map F restricted in $\mathcal{X}_i$. Mathematically, if the graph of F over $\mathcal{X}_i$ is written in the following form:

$$\mathrm{Graph}(F|_{\mathcal{X}_i}) := \{(x, y) \in \mathcal{X}_i \times \mathcal{Y} \mid G_i y \leq W_i + E_i x\},$$

then $F|_{\mathcal{X}_i}$ can be defined as follows:

$$F|_{\mathcal{X}_i}(x) := \{y \in Y \mid G_i y \leq W_i + E_i x\}, \quad \text{for } \; x \in \mathcal{X}_i.$$

If F is a continuous PWA set valued map, then for any pair of neighboring regions $(\mathcal{X}_i, \mathcal{X}_j)$:

$$F|_{\mathcal{X}_i}(x) = F|_{\mathcal{X}_j}(x) \;\; \text{for } \; x \in \mathcal{X}_i \cap \mathcal{X}_j.$$

Recall that we aim to find an optimization problem such that the given PWA set-valued map is a subset of the set of optimal solutions to this recovered optimization problem. In case of continuous PWA set-valued maps, it is possible to prove that the set of optimal solution to the recovered optimization problem coincides with the given continuous PWA set-valued map. Similar to the recovery of a PWA function, it is assumed that the given partition $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ is convexly liftable. For ease of presentation, let $\ell(x) : \mathcal{X} \rightarrow \mathbb{R}$ denote a convex lifting for the given partition i.e.

$$\ell(x) = a_i^T x + b_i \;\; \text{for } \; x \in \mathcal{X}_i. \tag{7.8}$$

Define the following set:

$$\begin{aligned}
&V = \bigcup_{i\in\mathcal{I}_N} \mathcal{V}(\mathrm{Graph}(F|_{\mathcal{X}_i})), \; R = \bigcup_{i\in\mathcal{I}_N} \mathcal{R}(\mathrm{Graph}(F|_{\mathcal{X}_i})),\\
&V_{[x^T\, u^T\, z]^T} = \left\{ \begin{bmatrix} v \\ z(v) \end{bmatrix} \middle| \begin{aligned} & v \in V, \\ & z(v) = \begin{bmatrix} a_i^T & 0_{d_u}^T \end{bmatrix} v + b_i \text{ if } v \in \mathcal{V}(\mathrm{Graph}(F|_{\mathcal{X}_i})) \end{aligned} \right\}\\
&R_{[x^T\, u^T\, z]^T} = \left\{ \begin{bmatrix} r \\ \widehat{\ell}(r) \end{bmatrix} \middle| \begin{aligned} & r \in R, \\ & \widehat{\ell}(r) = \begin{bmatrix} a_i^T & 0_{d_u}^T \end{bmatrix} r \text{ if } r \in \mathcal{R}(\mathrm{Graph}(F|_{\mathcal{X}_i})) \end{aligned} \right\}\\
&\Pi = \mathrm{conv}(V_{[x^T\, u^T\, z]^T}) \oplus \mathrm{cone}(R_{[x^T\, u^T\, z]^T}).
\end{aligned} \tag{7.9}$$

Then, a solution to inverse optimality problem for the class of continuous PWA set-valued maps can be stated in the following theorem.

Theorem 7.2.1 *Given a convexly liftable partition $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ of a polyhedron $\mathcal{X} \subseteq \mathbb{R}^{d_x}$ over which a continuous PWA set-valued map $F_{pwa} : \mathcal{X} \to \mathcal{Y} \subseteq \mathbb{R}^{d_u}$ is defined, F_{pwa} is the image via the orthogonal projection onto $\mathbb{R}^{d_u}$ of a subset of the optimal solutions to the following optimization problem:*

$$\min_{[u^T\ z]^T} z \ \ \textit{such that} \ \ \begin{bmatrix} x^T & u^T & z \end{bmatrix}^T \in \Pi, \tag{7.10}$$

where Π is computed from (7.9).

Proof: The proof is similar to the one of Theorem 4.5.17. □

Note that it is also possible to prove that the given continuous PWA set-valued map coincides with the image via the orthogonal projection onto $\mathbb{R}^{d_u}$ of the set of optimal solutions to the above recovered optimization problem. This proof needs further developments and does not enter in the mainline of this thesis. For simplicity, we only state a weaker version of this result.

7.3 Recognition of AW Voronoi diagrams and AW Delaunay decompositions

As presented in Subsection 4.4.1, a polyhedral partition admits a convex lifting, should be an additively weighted Voronoi diagram or an additively weighted Delaunay decomposition. Motivated from these necessary and sufficient conditions, it is important to recognize them. More specially, identifying suitable sites and associated weights will also be of interest in specific applications.

7.3.1 Additively weighted Voronoi diagrams

The present subsection aims to recall the definition of an additively weighted (AW) Voronoi diagram. Let $\rho(s, x)$ denote the Euclidean distance between two points $x, s \in \mathbb{R}^d$, then the additively weighted distance between x and s with respect to a weight $w \in \mathbb{R}$ of the point s can be described by: $\rho^2(s, x) - w$. Moreover, given a set of N points denoted by $S = \{s_1, \ldots, s_N\} \subset \mathbb{R}^d$, a point x belongs to the *Voronoi domain* of the site $s_i \in S$ with the weight w_i if and only if:

$$\rho^2(s_i, x) - w_i \leq \rho^2(s_j, x) - w_j, \quad \forall j \in \mathcal{I}_N.$$

An example is illustrated in Figure 7.3 wherein the sites have been coordinated respectively at $C_1 = [-1; 2]$, $C_2 = [2; 1]$, $C_3 = [-2; 1]$, $C_4 = [-2; -1]$, $C_5 =$

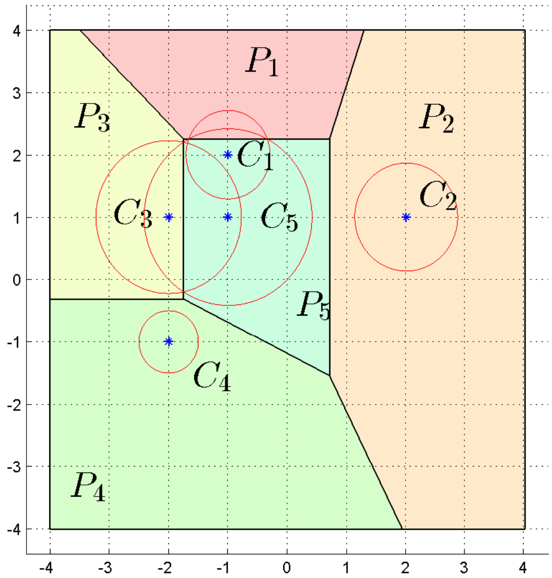


Figure 7.3: An additively weighted Voronoi diagram.

$[-1; 1]$ and their weights are $w_1 = 0.5, w_2 = 0.75, w_3 = 1.5, w_4 = 0.25, w_5 = 2$, respectively. Note also that the hyperplane separating two neighboring regions is orthogonal to the line connecting their associated sites as seen in this example. We use (C, r) to denote a circle centered at C with the radius r. Consider a point $x_0 \in P_4$, then the distance from x_0 to the intersection of $(C_4, \sqrt{w_4})$ and a line through x_0 tangent to $(C_4, \sqrt{w_4})$ is smaller than the one from x_0 to the intersection of $(C_3, \sqrt{w_3})$ and a line through x_0 tangent to $(C_3, \sqrt{w_3})$.

As an inverse problem, given a polyhedral partition $\{\mathcal{X}_i\}_{i \in \mathcal{I}_N}$ of a polyhedron $\mathcal{X} \subset \mathbb{R}^d$, the question is whether or not this partition is an additively weighted Voronoi diagram. Suppose it is, then each region $\mathcal{X}_i$ in this partition has to be a *Voronoi domain* of a site denoted by s_i with respect to an additive weight w_i. This constraint can be mathematically described as follows:

$$\rho^2(s_i, x) - w_i \leq \rho^2(s_j, x) - w_j, \ \forall x \in \mathcal{X}_i \text{ and } \forall j \in \mathcal{I}_N. \tag{7.11}$$

Further, there exist N sites s_i associated with N weights w_i such that constraint (7.11) holds true $\forall i \in \mathcal{I}_N$. Therefore, the goal is to find such s_i and w_i for every $i \in \mathcal{I}_N$. The weights w_i are not necessarily positive, they can admit negative values. If these weights are identical, then this partition is called Voronoi diagram. Moreover, these sites s_i do not necessarily lie in the interior of their corresponding Voronoi domains. Still, the notion of additively weighted Voronoi diagram is a

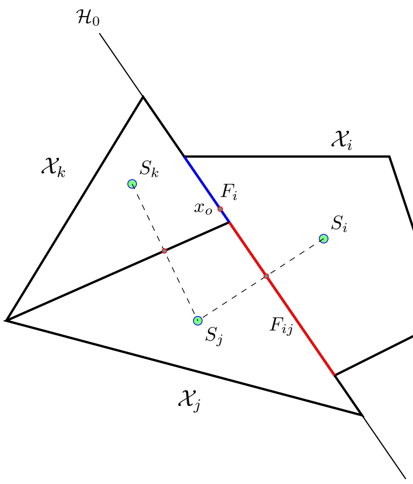


Figure 7.4: An illustration of a polyhedral partition not satisfying the facet-to-facet property.

simple generalization of *power diagram* studied in Aurenhammer [1987c]. In fact, power diagram requires that the weights w_i have to be strictly positive, it can trivially be seen that if a strictly negative, small enough constant, denoted by c, is added to two sides of (7.11), such that $c - w_i$ and $c - w_j$ are strictly negative, then the condition for a partition to be a power diagram is also fulfilled (see details in Aurenhammer [1987c, 1991]).

7.3.2 Recognition of AW Voronoi diagrams

Based on the properties of an additively weighted (AW) Voronoi diagram, this subsection aims to present an algorithm to verify whether the given partition of a polyhedron is an additively weighted Voronoi diagram. It can be observed that a polyhedral partition of a polyhedron, not satisfying the properties of a cell complex, is not an additively weighted Voronoi diagram. This observation is stated via the following proposition.

Proposition 7.3.1 *If a polyhedral partition of a polyhedron does not satisfy the properties of a cell complex, then it is not an additively weighted Voronoi diagram.*

Proof: Suppose a given polyhedral partition $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ of a polyhedron $\mathcal{X}$, does not satisfy the facet-to-facet property, but is an additively weighted Voronoi diagram. Then, there exists at least a pair of neighboring regions denoted by $\mathcal{X}_i, \mathcal{X}_j$ such that their intersection is a strict subset of one of their facets (an illustration can be found in Figure 7.4). Without loss of generality, we suppose that

$F_{ij} = \mathcal{X}_i \cap \mathcal{X}_j \subset F_i \in \mathcal{F}(\mathcal{X}_i)$. $\mathcal{X}_i$ and $\mathcal{X}_j$ are two Voronoi domains with two sites s_i, s_j and weights w_i, w_j, then inclusion (7.11) holds. Note also that the hyperplane denoted by $\mathcal{H}_0$ containing F_{ij} can be described by:

$$2(s_i^T - s_j^T)x = s_i^T s_i - s_j^T s_j + w_j - w_i. \tag{7.12}$$

Similarly, there exists another Voronoi domain neighbor to $\mathcal{X}_i$ denoted by $\mathcal{X}_k$ with a site s_k and a weight w_k such that: $F_{ik} = \mathcal{X}_i \cap \mathcal{X}_k \subset F_i$, then the hyperplane separating these two Voronoi domains can be described by:

$$2(s_i^T - s_k^T)x = s_i^T s_i - s_k^T s_k + w_k - w_i. \tag{7.13}$$

Clearly, the hyperplanes described by (7.12) and (7.13) are identical due to the fact that both of them contain F_i. It can be seen that every $x \in \mathcal{H}_0$ satisfies:

$$\begin{aligned} \rho^2(x, s_j) - w_j = \rho^2(x, s_i) - w_i, \\ \rho^2(x, s_i) - w_i = \rho^2(x, s_k) - w_k. \end{aligned} \tag{7.14}$$

Now, consider a point $x_o \in F_{ik} \subset \mathcal{H}_0$ but $x_o \notin \mathcal{X}_j$, then from (7.14) one can obtain:

$$\rho^2(x_o, s_j) - w_j = \rho^2(x_o, s_k) - w_k. \tag{7.15}$$

Note however that $x_o \in \mathcal{X}_k$ but $x_o \notin \mathcal{X}_j$, therefore

$$\rho^2(x_o, s_k) - w_k < \rho^2(x_o, s_j) - w_j. \tag{7.16}$$

(7.15) and (7.16) are evidently contradictory. $\square$

To recognize an additively weighted Voronoi diagram, an algorithm is required. This algorithm is based on the properties of an additively weighted Voronoi diagram shown in the preceding subsection and is presented in the sequel. It is shown in Aurenhammer [1987c] that given a set of finitely discrete points with their associated weights, there exists a unique decomposition which satisfies the properties of an additively weighted Voronoi diagram. Conversely, given an additively weighted Voronoi diagram, there may exist different sets of discrete sites with different sets of additive weights satisfying property (7.11). An algorithm for this recognition is presented in Algorithm 7.2.

As discussed before, an additively weighted Voronoi diagram does not need positive weights. However, to avoid the complication in implementing the cost function (7.19), the constraints $w_i \geq 0, \ \forall i \in \mathcal{I}_N$ can be added, then the cost function $f = \sum_{i \in \mathcal{I}_N} w_i$ replaces (7.19). Note that Algorithm 7.2 restricts to polytopic partitions. For polyhedral partitions, it is shown in Subsection 4.4.2.2 how to find a polytopic partition such that their convex liftability is equivalent.

Algorithm 7.2 Recognition of weighted Voronoi diagram

Input: A polytopic partition $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ of a polytope $\mathcal{X} \subset \mathbb{R}^d$.
Output: Is $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ an additively weighted Voronoi diagram? If yes, what are appropriate sites s_i and weights w_i?

1: Register all pairs of neighboring regions of $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$.
2: For each pair of neighboring regions $(\mathcal{X}_i, \mathcal{X}_j)$ with $(i,j) \in \mathcal{I}_N^2$:
— Add for every $v \in \mathcal{V}(\mathcal{X}_i \cap \mathcal{X}_j)$:

$$2(s_j^T - s_i^T)v = s_j^T s_j - s_i^T s_i + w_i - w_j. \tag{7.17}$$

— Add for every $u \in \mathcal{V}(\mathcal{X}_i)$ and $u \notin \mathcal{V}(\mathcal{X}_i \cap \mathcal{X}_j)$:

$$2(s_j^T - s_i^T)u < s_j^T s_j - s_i^T s_i + w_i - w_j. \tag{7.18}$$

3: Choose a cost function:

$$f = \sum_{i\in\mathcal{I}_N} |w_i|. \tag{7.19}$$

4: Solve the minimization problem of cost function (7.19) subject to constraints (7.17) and (7.18).
5: If this problem is feasible, then $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ is an additively weighted Voronoi diagram. Otherwise, it is not.

Practically, the strict inequality (7.18) can be transformed into inequality constraint to adapt to an optimization problem by adding a softening constant $a > 0$ on the left-hand side. Then, (7.18) can be relaxed as follows:

$$2(s_j^T - s_i^T)u + a \leq s_j^T s_j - s_i^T s_i + w_i - w_j. \tag{7.20}$$

This constant needs to be small enough to ensure that the constraints are not stringent. A value close to the numerical tolerant error may be appropriate. Step 5 of this algorithm can be explained via the following theorem.

Theorem 7.3.2 *A given cell complex $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ of a polytope $\mathcal{X} \subset \mathbb{R}^d$ is an additively weighted Voronoi diagram if and only if the optimization problem minimizing the cost function* (7.19) *subject to constraints* (7.17), (7.18) *is feasible.*

Proof: $\Longrightarrow$ If $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ is an additively weighted Voronoi diagram, then constraints (7.17) and (7.18) are clearly fulfilled. Therefore, the minimization problem with the cost function (7.19) subject to (7.17), (7.18) is feasible.
$\Longleftarrow$ Conversely, if this minimization problem is feasible, we need to prove that $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ is an additively weighted Voronoi diagram. In fact, consider a region $\mathcal{X}_i$

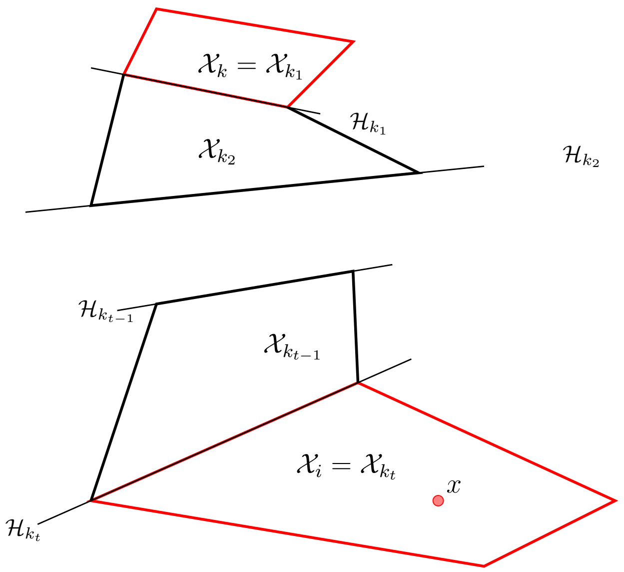


Figure 7.5: An illustration for the proof of Theorem 7.3.2.

and its neighboring regions $\mathcal{X}_{i_1}, \ldots, \mathcal{X}_{i_{n_i}}$ where $i_j \in \mathcal{I}_N$ and $j \in \mathcal{I}_{n_i}$, it is clear that for any pair $(\mathcal{X}_i, \mathcal{X}_{i_j})$:

$$\rho^2(x, s_i) - w_i \leq \rho^2(x, s_{i_j}) - w_{i_j}, \ \forall x \in \mathcal{X}_i, \ \forall j \in \mathcal{I}_{n_i}, \tag{7.21}$$

and the hyperplane which separates the halfspace containing $\mathcal{X}_i$ and the one containing $\mathcal{X}_{i_j}$, has the description:

$$\mathcal{H}_{i_j} := \left\{ x \mid 2(s_i^T - s_{i_j}^T)x = s_i^T s_i - s_{i_j}^T s_{i_j} + w_{i_j} - w_i \right\}.$$

We show now that $\rho^2(x, s_i) - w_i \leq \rho^2(x, s_k) - w_k, \ \forall k \in \mathcal{I}_N, \ \forall x \in \mathcal{X}_i$. Consider a region $\mathcal{X}_k$ with the site s_k and a weight w_k, for any point $x \in \mathcal{X}_i$ there exists a number of regions $\mathcal{X}_{k_1}, \ldots, \mathcal{X}_{k_t}$ such that $\mathcal{X}_{k_1} = \mathcal{X}_k, \mathcal{X}_{k_t} = \mathcal{X}_i$ and $\mathcal{X}_{k_j}, \mathcal{X}_{k_{j+1}}, \forall j = 1...t-1$ are neighbors such that the hyperplane separating them separates the whole space into two halfspaces: the first halfspace containing $\mathcal{X}_{k_j}$, the other containing $\mathcal{X}_{k_{j+1}}$ and x (an illustration is presented in Figure 7.5). Then it is clear that:

$$\rho^2(x, s_{k_j}) - w_{k_j} \geq \rho^2(x, s_{k_{j+1}}) - w_{k_{j+1}}, \tag{7.22}$$

and this inclusion holds true for all $j = 1...t-1$. Therefore, $\rho^2(x, s_{k_1}) - w_{k_1} \geq \rho^2(x, s_{k_t}) - w_{k_t}$, in other words, $\rho^2(x, s_k) - w_k \geq \rho^2(x, s_i) - w_i$. This completes the proof. $\square$

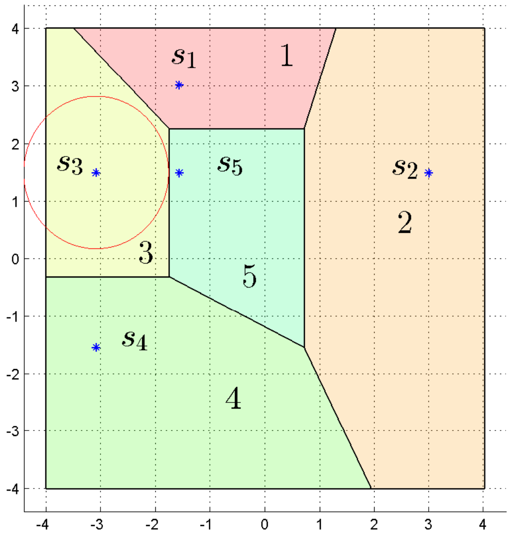


Figure 7.6: Result of Algorithm 7.2 for the partition in Figure 7.3.

To illustrate Algorithm 7.2, consider again the example in Figure 7.3. Algorithm 7.2 returns the following results:

$$\begin{aligned} s &= \begin{bmatrix} -1.5667 & 2.9833 & -3.0833 & -3.0833 & -1.5667 \\ 3.0083 & 1.4917 & -1.5417 & 1.4917 & 1.4917 \end{bmatrix}, \\ w &= \begin{bmatrix} 0 & 0 & 1.7442 & 0 & 0 \end{bmatrix}. \end{aligned}$$

This result is also presented in Figure 7.6.

7.3.3 Additively weighted Delaunay decompositions

Additively weighted (AW) Delaunay decomposition, considered as a generalization of Delaunay triangulation, is in fact the dual partition of an additively weighted Voronoi diagram (see Rybnikov [1999]). A vertex of an additively weighted Delaunay decomposition is a weighted point with an additive weight. Let $S \subset \mathbb{R}^d$ be a set of discrete points and $w(\cdot) : S \to \mathbb{R}$ be a real-valued function associated with the points in S, then a polytope P is called a Delaunay cell if:

— $\mathcal{V}(P) \subset S$,

— there exists a point denoted by $c(P) \in \mathbb{R}^d$ and $r(P) \in \mathbb{R}$ such that

$$\rho^2(v, c(P)) - w(v) = r(P), \ \forall v \in \mathcal{V}(P)$$

— $\forall s \in S, \ s \notin \mathcal{V}(P)$, $\rho^2(s, c(P)) - w(s) > r(P)$.

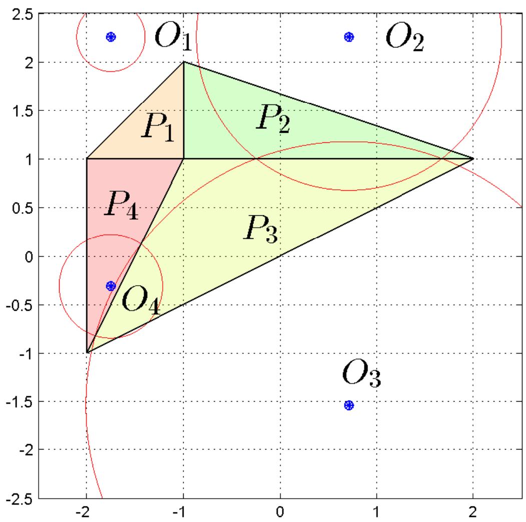


Figure 7.7: An additively weighted Delaunay decomposition.

An illustrative example can be found in Figure 7.7 where the given points and their weights in Figure 7.3 are reconsidered. The sites corresponding to the Delaunay cells P_1, P_2, P_3, P_4 have the coordinates $O_1 = [-1.75, 2.25]$, $O_2 = [0.7083, 2.25]$, $O_3 = [0.7083, -1.5417]$, $O_4 = [-1.75, -0.3125]$ and their associated radii $R_1 = 0.125, R_2 = 2.4809, R_3 = 7.3785, R_4 = 0.2852$. Note also that the circles in this figure are centered at O_1, O_2, O_3, O_4 with the radii equal to the root of R_1, R_2, R_3, R_4. It can be observed that the distance between point $[-2,\ 1]$ and a point, at which circle $(O_1, R_1^{0.5})$ and a tangent line through $[-2,\ 1]$ to this circle intersect, is the root of the weight of point $[-2,\ 1]$, according to the Pythagorean theorem. The same argument holds true for points $[-1,\ 1], [-1,\ 2]$.

Conversely, given a polytopic partition $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ of a polyhedron $\mathcal{X} \subset \mathbb{R}^d$, a question needs to be answered whether or not this partition is an additively weighted Delaunay decomposition. If yes, there exist a set of N discrete sites denoted as $C \subset \mathbb{R}^d$, a set of corresponding radii $R \subset \mathbb{R}$, and a real-valued function $w : \bigcup_{i\in\mathcal{I}_N} \mathcal{V}(\mathcal{X}_i) \to \mathbb{R}$ such that for each region $\mathcal{X}_i$ with its corresponding site $c_i \in C$ and a radius $r_i \in R$:

— $\rho^2(v, c_i) - w(v) = r_i,\ \ \forall v \in \mathcal{V}(\mathcal{X}_i)$,
— $\rho^2(u, c_i) - w(u) > r_i,\ \ \forall u \in \bigcup_{i\in\mathcal{I}_N} \mathcal{V}(\mathcal{X}_i),\ \ u \notin \mathcal{V}(\mathcal{X}_i)$.

Therefore, this question is equivalent to whether there exist such sites c_i with radii r_i and a function $w(\cdot)$ satisfying the above conditions. This question will be clarified in the next subsection.

7.3.4 Recognition of AW Delaunay decompositions

This subsection's goal is to present an algorithm to answer the certification problem of whether or not a given partition represents an additively weighted (AW) Delaunay decomposition.

We will answer this certification problem via the feasibility of an optimization problem. This optimization problem will be explicitly constructed and solved via bi-linear programming. Notice that if a given polytopic partition of a polytope fulfills the properties of an additively weighted Delaunay decomposition, then it has to be a cell complex. This comment is formally stated and proved via the following proposition.

Proposition 7.3.3 *Any polytopic partition of a polytope not satisfying the properties of a cell complex is not an additively weighted Delaunay decomposition.*

Proof: Given a polypotic partition $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ in $\mathbb{R}^d$, suppose this partition is not a cell complex but is an additively weighted Delaunay decomposition. From this assumption, there exists at least a pair of neighboring regions denoted $\mathcal{X}_i, \mathcal{X}_j$ whose facet-to-facet property does not hold. Denote also $F_{ij} = \mathcal{X}_i \cap \mathcal{X}_j$, $F_i \in \mathcal{F}(\mathcal{X}_i)$, $F_j \in \mathcal{F}(\mathcal{X}_j)$ such that $F_{ij} \subset F_i$ or $F_{ij} \subset F_j$, the sites of these two regions are denoted by c_i, c_j, with the corresponding radii r_i, r_j, respectively. Due to the properties of an additively weighted Delaunay decomposition, one has:

$$\begin{aligned} &\rho^2(v, c_i) - w(v) = r_i, \ \forall v \in \mathcal{V}(\mathcal{X}_i) \\ &\rho^2(v, c_j) - w(v) > r_j, \ \forall v \in \bigcup_{t\in\mathcal{I}_N} \mathcal{V}(\mathcal{X}_t), \ v \notin \mathcal{V}(\mathcal{X}_j). \end{aligned} \tag{7.23}$$

Similarly,

$$\begin{aligned} &\rho^2(v, c_j) - w(v) = r_j, \ \forall v \in \mathcal{V}(\mathcal{X}_j), \\ &\rho^2(v, c_i) - w(v) > r_i, \ \forall v \in \bigcup_{t\in\mathcal{I}_N} \mathcal{V}(\mathcal{X}_t), \ v \notin \mathcal{V}(\mathcal{X}_i). \end{aligned} \tag{7.24}$$

From (7.23) and (7.24), if $\mathcal{V}(\mathcal{X}_i) \cap \mathcal{V}(\mathcal{X}_j) = \emptyset$ then the hyperplane:

$$\mathcal{H}_0 = \left\{ x \mid 2(c_j^T - c_i^T)x = c_j^T c_j - c_i^T c_i + r_i - r_j \right\} \tag{7.25}$$

strictly separates $\mathcal{X}_i$ and $\mathcal{X}_j$, it is contradictory to the fact that $\mathcal{X}_i$ and $\mathcal{X}_j$ are neighbors. Therefore, there exists at least a common vertex $v \in \mathcal{V}(\mathcal{X}_i) \cap \mathcal{V}(\mathcal{X}_j)$. It can be observed that $F_i, F_j, F_{ij} \subset \mathcal{H}_0$.

Now, consider a vertex $u \in \mathcal{V}(F_{ij})$, such that either $u \in \mathcal{V}(\mathcal{X}_i)$, $u \notin \mathcal{V}(\mathcal{X}_j)$, or $u \in \mathcal{V}(\mathcal{X}_j)$, $u \notin \mathcal{V}(\mathcal{X}_i)$. Without loss of generality, suppose the former case happens, then the following holds true:

$$\rho^2(u, c_i) - r_i = w(u) < \rho^2(u, c_j) - r_j. \tag{7.26}$$

Algorithm 7.3 Recognition of weighted Delaunay diagram

Input: A polytopic partition $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ of a polytope in $\mathbb{R}^d$.
Output: Is $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ an additively weighted Delaunay decomposition? If yes, what are appropriate sites c_i, radii r_i and weights $w(\cdot)$?

1: For each region $\mathcal{X}_i$, add equality constraints:

$$\rho^2(v, c_i) - w(v) = r_i,\ \forall v \in \mathcal{V}(\mathcal{X}_i). \tag{7.28}$$

2: Register all pairs of neighboring regions of $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$.
3: For each pair of neighboring regions $(\mathcal{X}_i, \mathcal{X}_j)$ with $(i, j) \in \mathcal{I}_N^2$, add:

$$\rho^2(v, c_i) - w(v) > r_i,\ \forall v \in \mathcal{V}(\mathcal{X}_j), v \notin \mathcal{V}(\mathcal{X}_i) \tag{7.29}$$

4: Choose a cost function e.g.

$$f = \sum_{v\in\bigcup_{i\in\mathcal{I}_N}\mathcal{V}(\mathcal{X}_i)} |w(v)| \tag{7.30}$$

5: Solve the problem minimizing the cost function (7.30) subject to constraints (7.28) and (7.29).
6: If this problem is feasible, then $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$ is an additively weighted Delaunay decomposition. Otherwise, it is not.

Nevertheless, $F_{ij} \subset \mathcal{H}_0$ implies $u \in \mathcal{H}_0$, it leads to:

$$\rho^2(u, c_i) - r_i = \rho^2(u, c_j) - r_j. \tag{7.27}$$

It is clear that (7.26) and (7.27) are contradictory. This completes the proof. □

To recognize an additively weighted Delaunay decomposition, an algorithm needs to be presented for practical purposes. Notice that if the given cell complex is an additively weighted Delaunay decomposition, then there may exist different real-valued functions $w(\cdot)$ associated with the vertices of this cell complex, sites c_i and different radii r_i, $i \in \mathcal{I}_N$ corresponding to each Delaunay cell such that the properties of an additively weighted Delaunay decomposition are fulfilled. However, showing the existence of a function $w(\cdot)$, a set of sites c_i and a set of radii r_i, $i \in \mathcal{I}_N$ is enough for a conclusion. Moreover, function $w(\cdot)$ and radii r_i are not necessarily positive, they can get values in the real field. Algorithm 7.3 answers the certification question by providing, if there exists, such a feasible function $w(\cdot)$, such a set of points c_i and such a set of radii r_i.

It is remarked that $w(\cdot)$ does not necessarily take non-negative values over the set of vertices of $\{\mathcal{X}_i\}_{i\in\mathcal{I}_N}$, however to simplify the cost function, one can add the constraints $w(x) \geq 0, \forall x \in \bigcup_{i\in\mathcal{I}_N}\mathcal{V}(\mathcal{X}_i)$. Thereby, the cost function

$f = \sum_{x \in \bigcup_{i \in \mathcal{I}_N} \mathcal{V}(\mathcal{X}_i)} w(x)$, is equivalent to (7.30). Moreover, constraints (7.29) represent strict inequalities which can cause numerical errors for optimization solvers, therefore, a strictly positive constant denoted by a needs to be added on the right-hand side of (7.29) in order to adapt to an optimization problem subject to constraints, then constraints (7.29) are equivalent to $\rho^2(x, c_i) - w(x) \geq r_i + a$. Note however that this constant a needs to be chosen small enough such that the set of constraints is not over-stringent. A tolerant error may be preferable in this case.

The following theorem can serve as a clarification of Algorithm 7.3.

Theorem 7.3.4 *A given polytopic partition $\{\mathcal{X}_i\}_{i \in \mathcal{I}_N}$ of a polytope in $\mathbb{R}^d$ is an additively weighted Delaunay decomposition if and only if the optimization problem minimizing the cost function* (7.30) *subject to constraints* (7.28) *and* (7.29) *is feasible.*

Proof: $\Longrightarrow$ If the given polytopic partition is an additively weighted Delaunay decomposition, then it is clear that there exist a set of sites $c_i \in \mathbb{R}^d$, a set of radii $r_i \in \mathbb{R}$, $i \in \mathcal{I}_N$ and a real-valued function $w(\cdot) : \bigcup_{i \in \mathcal{I}_N} \mathcal{V}(\mathcal{X}_i) \to \mathbb{R}$ satisfying constraints (7.28) and (7.29). Therefore, the problem minimizing (7.30) subject to constraints (7.28) and (7.29) is feasible.
$\Longleftarrow$ Conversely, if the problem minimizing (7.30) subject to constraints (7.28) and (7.29) is feasible. Thus, constraints (7.28) and (7.29) hold true. We need to prove that any $v \in \bigcup_{t \in \mathcal{I}_N} \mathcal{V}(\mathcal{X}_t)$, $v \notin \mathcal{V}(\mathcal{X}_i)$, fulfills: $\rho^2(v, c_i) - w(v) > r_i$. Suppose $v \in \mathcal{V}(\mathcal{X}_k), i \neq k \in \mathcal{I}_N$.

Similar to the proof of Theorem 7.3.2, from (7.28), (7.29), there exists a number of regions $\mathcal{X}_{i_1}, \ldots, \mathcal{X}_{i_{n_i}}$ of $\mathcal{X}$ such that:

— $\mathcal{X}_{i_1} = \mathcal{X}_i$, $\mathcal{X}_{i_{n_i}} = \mathcal{X}_k$,
— $(\mathcal{X}_{i_j}, \mathcal{X}_{i_{j+1}}), j \in \mathcal{I}_{n_i - 1}$ are neighbors,
— the separating hyperplane $\mathcal{H}_j$ of $(\mathcal{X}_{i_j}, \mathcal{X}_{i_{j+1}})$, splits the whole space into two halfspaces: the first one contains $\mathcal{X}_{i_j}$, the second one contains $v, \mathcal{X}_{i_{j+1}}$.
— $v \notin \mathcal{H}_j$ for at least a $j \in \mathcal{I}_{n_i}$.

Note that $\mathcal{H}_j$ can be described as follows:

$$\mathcal{H}_j = \{x \mid 2(c_{i_{j+1}}^T - c_{i_j}^T)x = c_{i_{j+1}}^T c_{i_{j+1}} - c_{i_j}^T c_{i_j} + r_{i_j} - r_{i_{j+1}}\}.$$

As a consequence, the halfspace containing v can be described by:

$$\mathcal{H}_j^+ = \{x \mid 2(c_{i_{j+1}}^T - c_{i_j}^T)x \geq c_{i_{j+1}}^T c_{i_{j+1}} - c_{i_j}^T c_{i_j} + r_{i_j} - r_{i_{j+1}}\}.$$

$v \in \mathcal{H}_j^+$, satisfies:

$$2(c_{i_{j+1}}^T - c_{i_j}^T)v \geq c_{i_{j+1}}^T c_{i_{j+1}} - c_{i_j}^T c_{i_j} + r_{i_j} - r_{i_{j+1}}.$$

It leads to: $\rho^2(v, c_{i_j}) - r_{i_j} \geq \rho^2(v, c_{i_{j+1}}) - r_{i_{j+1}}$.

Due to $v \notin \mathcal{H}_j$ for at least a $j \in \mathcal{I}_{n_i}$, and the above inclusion holding for every $j \in \mathcal{I}_{n_i}$, we can conclude that $\rho^2(v, c_i) - r_i > \rho^2(v, c_k) - r_k = w(v)$. □

To illustrate Algorithm 7.3, consider again the partition in Figure 7.7. Algorithm 7.3 returns the set of vertices and their associated weights as follows:

$$V = \begin{bmatrix} -2 & -2 & -1 & -1 & 2 \\ 1 & -1 & 2 & 1 & 1 \end{bmatrix}, \; w = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

Also, the sites c_i and their corresponding radii are presented below:

$$c = \begin{bmatrix} -1.5 & 0.5 & 0.5 & -1.5 \\ 1.5 & 1.5 & -1 & 0 \end{bmatrix}, \; r = \begin{bmatrix} 0.5 & 2.5 & 6.25 & 1.25 \end{bmatrix}.$$

These results are also presented in Figure 7.8. The red circles are $(c_1, \sqrt{r_1})$, $(c_2, \sqrt{r_2})$, $(c_3, \sqrt{r_3})$, $(c_4, \sqrt{r_4})$.

Note that the numerical examples in this section are carried out in the environment of YALMIP and MPT 3.0 Herceg et al. [2013], Lofberg [2004]. Also, FMINCON is of use for the numerical examples.

Note also that the optimization problems in Algorithms 7.2, 7.3 are subject to bi-linear constraints. These constraints may also be non-convex. Therefore, they are quite computationally demanding. If the goal is only to determine whether or not a given partition is convexly liftable, one should use Algorithm 4.1.

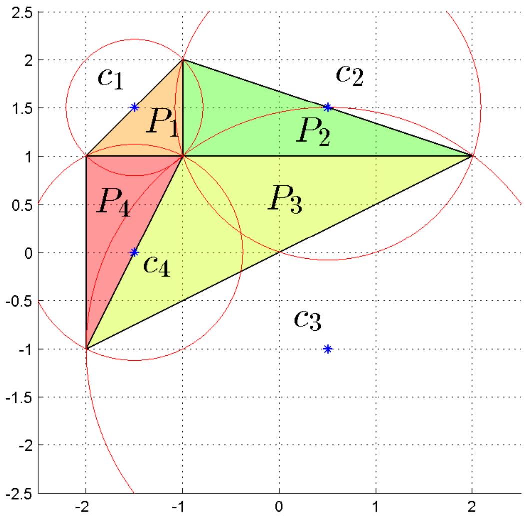


Figure 7.8: Result of Algorithm 7.3 for the partition in Figure 7.7.

Bibliography

O. Aichholzer, F. Aurenhammer, H. Krasser, and P. Brass. Pseudotriangulations from surfaces and a novel type of edge flip. *SIAM Journal on Computing*, 32 (6):1621–1653, 2003.

G. Z. Angelis. System analysis, modelling and control with polytopic linear models. 2001.

N. Athanasopoulos, G. Bitsoris, and M. Lazar. Construction of invariant polytopic sets with specified complexity. *International Journal of Control*, 87(8):1681–1693, 2014.

F. Aurenhammer. Criterion for the affine equivalence of cell complexes in r^d and convex polyhedra in r^{d+1}. *Discrete and Computational Geometry*, 2:49–64, 1987a.

F. Aurenhammer. Recognising polytopical cell complexes and constructing projection polyhedra. *Journal of Symbolic Computation*, 3:249–255, 1987b.

F. Aurenhammer. Power diagrams: properties, algorithms and applications. *SIAM Journal on Computing*, 16(1):78–96, 1987c.

F. Aurenhammer. Voronoi diagrams: A survey of a fundamental data structure. *ACM Comput. Surveys*, 23:345–405, 1991.

D. Avis and K. Fukuda. A pivoting algorithm for convex hulls and vertex enumeration of arrangements and polyhedra. *Discrete & Computational Geometry*, 8 (1):295–313, 1992.

M. Baes, M. Diehl, and I. Necoara. Every continuous nonlinear control system can be obtained by parametric convex programming. *IEEE tran on Automatic Control*, 53(8):1963–1967, Sept 2008.

M. Baotic, F. Borrelli, A. Bemporad, and M. Morari. Efficient on-line computation of constrained optimal control. *SIAM Journal on Control and Optimization*, 47(5):2470–2489, 2008.

B. Barmish and G. Leitmann. On ultimate boundedness control of uncertain systems in the absence of matching assumptions. *Automatic Control, IEEE Transactions on*, 27(1):153–158, Feb 1982.

A. Bemporad, M. Morari, V. Dua, and E. N. Pistikopoulos. The explicit linear quadratic regulator for constrained systems. *Automatica*, 38(1):3–20, 2002.

A. Bemporad, F. Borrelli, and M. Morari. Min-max control of constrained uncertain discrete-time linear systems. *Automatic Control, IEEE Transactions on*, 48 (9):1600–1606, 2003.

H. Benlaoukli and S. Olaru. Model predictive control–numerical methods for the invariant sets approximation. In *Numerical Analysis and Its Applications*, pages 192–199. Springer, 2009.

H. Benlaoukli, M. Hovd, S. Olaru, and P. Boucher. On the construction of invariant sets for piecewise affine systems using the transition graph. In *ICCA, International Conference on Control and Automation*, pages 122–127. IEEE, 2009.

G. Bitsoris. On the positive invariance of polyhedral sets for discrete-time systems. *Systems & control letters*, 11(3):243–248, 1988a.

G. Bitsoris. Positively invariant polyhedral sets of discrete-time linear systems. *International Journal of Control*, 47(6):1713–1726, 1988b.

G. Bitsoris and M. Vassilaki. Constrained regulation of linear systems. *Automatica*, 31(2):223–227, 1995.

F. Blanchini. Ultimate boundedness control for uncertain discrete-time systems via set-induced lyapunov functions. *Automatic Control, IEEE Transactions on*, 39(2):428–433, Feb 1994.

F. Blanchini. Set invariance in control. *Automatica*, 35(11):1747–1767, 1999.

F. Blanchini and S. Miani. *Set-theoretic methods in control*. Springer, 2008.

S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

S. P. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear matrix inequalities in system and control theory*, volume 15. SIAM, 1994.

T. F. Chan. Rank revealing qr factorizations. *Linear Algebra and Its Applications*, 88:67–82, 1987.

D. Chmielewski and V. Manousiouthakis. On constrained infinite-time linear quadratic optimal control. In *Decision and Control, 1996., Proceedings of the 35th IEEE Conference on*, volume 2, pages 1319–1324. IEEE, 1996.

H. Crapo and W. Whiteley. Plane self stresses and projected polyhedra 1: the basic pattern. *Structural Topology*, 19:55–73, 1993.

H. Crapo and W. Whiteley. Spaces of stresses, projections and parallel drawings for spherical polyhedra. *Contributions to Algebra and Geometry*, 35(So. 2): 259–281, 1994.

J. Daafouz and J. Bernussou. Parameter dependent lyapunov functions for discrete time systems with time varying parametric uncertainties. *Systems & control letters*, 43(5):355–359, 2001.

J. R. De Loera, S. Francisco, and A. Jesús. Triangulations. structures for algorithms and applications. *Algorithms and Computation in Mathematics*, 25, 2010.

P. Dorato. Non-fragile controller design: an overview. In *Proceedings of the American Control Conference*, volume 5, pages 2829–2831. IEEE, 1998.

H. Edelsbrunner and R. Seidel. Voronoi diagrams and arrangements. *Discrete Comput. Geom.*, 1986.

C. Feller, T. A. Johansen, and S. Olaru. An improved algorithm for combinatorial multi-parametric quadratic programming. *Automatica*, 49(5):1370–1376, 2013.

R. A. Freeman and P. V. Kokotovic. Inverse optimality in robust stabilization. *SIAM Journal on Control and Optimization*, 34(4):1365–1391, 1996.

K. Fukuda. cdd/cdd+ reference manual. *Institute for Operations Research, ETH-Zentrum*, 1997.

T. Gal. *Postoptimal analyses, parametric programming and related topics*. Walter de Gruyter, 1995.

B. D. George and B. C. Eaves. Fourier-motzkin elimination and its dual. *Journal of Combinatorial Theory, Series A*, 14(3):288 – 297, 1973. ISSN 0097-3165.

E. G. Gilbert and K. T. 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.

G. H. Golub and C. F. Van Loan. *Matrix computations*, volume 3. JHU Press, 2012.

A. Grancharova and T. A. Johansen. *Explicit Nonlinear Model Predictive Control*. Springer, 2012.

B. Grünbaum. *Convex polytopes*. Wiley Interscience, 1967.

M. Gulan, N. A. Nguyen, S. Olaru, P. Rodriguez-Ayerbe, and B. Rohal'-Ilkiv. Implications of inverse parametric optimization in model predictive control. In S. Olaru, A. Grancharova, and F. L. Pereira, editors, *Developments in Model-Based Optimization and Control*. Springer, 2015.

P.-O. Gutman and M. Cwikel. Admissible sets and feedback control for discrete-time linear dynamical systems with bounded controls and states. *Automatic Control, IEEE Transactions on*, 31(4):373–376, 1986.

P.-O. Gutman and M. Cwikel. An algorithm to find maximal state constraint sets for discrete-time linear dynamical systems with bounded controls and states. *Automatic Control, IEEE Transactions on*, 32(3):251–254, 1987.

D. Hartvigsen. Recognizing voronoi diagrams with linear programming. *ORSA J. Comput.*, 1992.

W. P. M. H. Heemels, B. De Schutter, and A. Bemporad. Equivalence of hybrid dynamical models. *Automatica*, 37(7):1085–1091, 2001.

A. Hempel, P. Goulart, and J. Lygeros. On inverse parametric optimization with an application to hybrid system control. *Automatic Control, IEEE Transactions on*, 60(4):1064–1069, April 2015.

J. C. Hennet. Discrete time constrained linear systems. *Control and Dynamic Systems*, 71:157–213, 1995.

M. Herceg, M. Kvasnica, C. Jones, and M. Morari. Multi-Parametric Toolbox 3.0. In *Proc. of the European Control Conference*, pages 502–510, Zürich, Switzerland, July 17–19 2013. http://control.ee.ethz.ch/~mpt.

M. Hovd and S. Olaru. Relaxing pwq lyapunov stability criteria for pwa systems. *Automatica*, 2012.

M. Hovd, S. Olaru, et al. Piecewise quadratic lyapunov functions for stability verification of approximate explicit mpc. *Modeling, Identification and Control*, 31(2):45–53, 2010.

T. A. Johansen. On multi-parametric nonlinear programming and explicit nonlinear model predictive control. In *Decision and Control, Proceedings of the 41st IEEE Conference on*, volume 3, pages 2768–2773, Dec 2002.

T. A. Johansen. Approximate explicit receding horizon control of constrained nonlinear systems. *Automatica*, 40(2):293 – 300, 2004.

R. E. Kalman. When is a linear control system optimal? *Journal of Fluids Engineering*, 86(1):51–60, 1964.

L. H. Keel and S. P. Bhattacharyya. Robust, fragile, or optimal? *Transactions on Automatic Control*, pages 1098–1105, 1997.

E. C. Kerrigan. *Robust constraint satisfaction: Invariant sets and predictive control*. PhD thesis, University of Cambridge, 2001.

H. K. Khalil. Nonlinear systems. *Prentice Hall, Inc*, page 323, 2002.

E. Kofman, H. Haimovich, and M. M. Seron. A systematic method to obtain ultimate bounds for perturbed systems. *International Journal of Control*, 80 (2):167–178, 2007.

I. Kolmanovsky and E. G. Gilbert. Theory and computation of disturbance invariant sets for discrete-time linear systems. *Mathematical Problems in Engineering*, 4(4):317–367, 1998.

H. Kong, G. Goodwin, and M. M. Seron. A revisit to inverse optimality of linear systems. *International Journal of Control*, 85(10):1506–1514, 2012.

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

M. Krstic and Z. H. Li. Inverse optimal design of input-to-state stabilizing nonlinear controllers. *Automatic Control, IEEE Transactions on*, 43(3):336–350, 1998.

M. Krstic and P. Tsiotras. Inverse optimal stabilization of a rigid spacecraft. *Automatic Control, IEEE Transactions on*, 44(5):1042–1049, 1999.

M. Kvasnica. *Real-time model predictive control via multi-parametric programming: theory and tools*. VDM-Verlag, 2009.

M. Kvasnica and M. Fikar. Clipping-based complexity reduction in explicit mpc. *Automatic Control, IEEE Transactions on*, 57(7):1878–1883, 2012.

M. Kvasnica, J. Hledík, I. Rauová, and M. Fikar. Complexity reduction of explicit model predictive control via separation. *Automatica*, 49(6):1776–1781, 2013.

V. Larin. About the inverse problem of optimal control. 2003.

C. W. Lee. P.l.-spheres, convex polytopes, and stress. *Discrete Comput Geom*, 1996.

J. Lofberg. Yalmip : a toolbox for modeling and optimization in matlab. In *Computer Aided Control Systems Design, 2004 IEEE International Symposium on*, pages 284–289, Sept 2004.

C. Løvaas, M. M. Seron, and G. C. Goodwin. Inverse minimax optimality of model predictive control policies. *Systems & Control Letters*, 58(1):31–38, 2009.

J. C. Maxwell. On reciprocal diagrams and diagrams of forces. *Philosophical Magazine*, ser. 4. 27:250–261, 1864.

H. N. Nguyen. *Constrained Control of Uncertain, Time-Varying, Discrete-Time Systems: An Interpolation-Based Approach*. Lecture Notes in Control and Information Sciences, 2014.

H. N. Nguyen, P. O. Gutman, S. Olaru, and M. Hovd. Explicit constraint control based on interpolation techniques for time-varying and uncertain linear discrete-time systems. In *World Congress*, volume 18, pages 5741–5746, 2011.

N. A. Nguyen, S. Olaru, P. Rodriguez-Ayerbe, M. Hovd, and I. Necoara. Constructive solution of inverse parametric linear/quadratic programming problems. a. URL `https://hal-supelec.archives-ouvertes.fr/hal-01207234/document`.

N. A. Nguyen, S. Olaru, P. Rodriguez-Ayerbe, and M. Kvasnica. Convex liftings-based robust control design. *Submitted*, b.

N. A. Nguyen, S. Olaru, G. Bitsoris, P. Rodriguez-Ayerbe, and M. Hovd. Explicit fragility margins for pwa control laws of discrete-time linear systems. In *13th European Control Conference, Strasbourg, France*, 2014a.

N. A. Nguyen, S. Olaru, P. Rodriguez-Ayerbe, and M. Hovd. An inverse optimality argument to improve robustness in constrained control. In *IFAC World Congress*, volume 19, pages 1631–1636, 2014b.

N. A. Nguyen, S. Olaru, P. Rodriguez-Ayerbe, M. Hovd, and I. Necoara. On the lifting problems and their connections with piecewise affine control law design. In *European Control Conference, Strasbourg, France*, 2014c.

N. A. Nguyen, S. Olaru, P. Rodriguez-Ayerbe, M. Hovd, and I. Necoara. Inverse parametric convex programming problems via convex liftings. In *19th IFAC World Congress, Cape Town, South Africa*, 2014d.

N. A. Nguyen, S. Olaru, G. Rodriguez-Ayerbe, Bitsoris, P., and M. Hovd. Explicit robustness and fragility margins for linear discrete systems with piecewise affine control law. *Automatica*, 2015a.

N. A. Nguyen, S. Olaru, and P. Rodriguez-Ayerbe. Any discontinuous pwa function is optimal solution to a parametric linear programming problem. In *54th IEEE Conference on Decision and Control, Osaka, Japan*, 2015b.

N. A. Nguyen, S. Olaru, and P. Rodriguez-Ayerbe. Inverse parametric linear/quadratic programming problem for continuous pwa functions defined on polyhedral partitions of polyhedra. In *54th IEEE Conference on Decision and Control, Osaka, Japan*, 2015c.

N. A. Nguyen, S. Olaru, and P. Rodriguez-Ayerbe. Recognition of additively weighted voronoi diagrams and weighted delaunay decompositions. In *European Control Conference, Linz, Austria*, 2015d.

N. A. Nguyen, S. Olaru, and P. Rodriguez-Ayerbe. On the complexity of the convex liftings-based solution to inverse parametric convex programming problems. In *European Control Conference, Linz, Austria*, 2015e.

N. A. Nguyen, S. Olaru, and P. Rodriguez-Ayerbe. Robust control design based on convex liftings. In *the 8th IFAC Symposium on Robust Control Design*, 2015f.

N. A. Nguyen, S. Olaru, P. Rodriguez-Ayerbe, M. Hovd, and I. Necoara. Fully inverse parametric linear/quadratic programming problems via convex liftings. In S. Olaru, A. Grancharova, and F. L. Pereira, editors, *Developments in Model-Based Optimization and Control*. Springer, 2015g.

S. Olaru and D. Dumur. A parameterized polyhedra approach for explicit constrained predictive control. In *Decision and Control, 43rd IEEE Conference on*, volume 2, pages 1580–1585, 2004.

S. Olaru and D. Dumur. Avoiding constraints redundancy in predictive control optimization routines. *Automatic Control, IEEE Transactions on*, 50(9):1459–1465, 2005.

S. Olaru and D. Dumur. On the continuity and complexity of control laws based on multiparametric linear programs. In *Decision and Control, 45th Conference on*, pages 5465–5470. IEEE, 2006.

S. Olaru, J. A. De Doná, M. M. Seron, and F. Stoican. Positive invariant sets for fault tolerant multisensor control schemes. *International Journal of Control*, 83 (12):2622–2640, 2010.

S. Olaru, N. A. Nguyen, G. Bitsoris, P. Rodriguez-Ayerbe, and M. Hovd. Explicit robustness margins for discrete-time linear systems with pwa control. In *17th International Conference on System Theory, Control and Computing, Sinaia, Romania*, 2013.

E. Ostertag. *Mono-and Multivariable Control and Estimation: Linear, Quadratic and LMI Methods*, volume 2. Springer Science & Business Media, 2011.

E. N. Pistikopoulos, M. C. Georgiadis, and V. Dua. *Multi-parametric programming*. Wiley-vch, 2007.

S. V. Raković and M. Lazar. The minkowski–lyapunov equation for linear dynamics: Theoretical foundations. *Automatica*, 50(8):2015–2024, 2014.

S. V. Rakovic, P. Grieder, M. Kvasnica, D. Q. Mayne, and M. Morari. Computation of invariant sets for piecewise affine discrete time systems subject to bounded disturbances. In *Decision and Control, 2004. CDC. 43rd IEEE Conference on*, volume 2, pages 1418–1423. IEEE, 2004.

S. V. Rakovic, E. C. Kerrigan, K. I. Kouramas, and D. Q. Mayne. Invariant approximations of the minimal robust positively invariant set. *Automatic Control, IEEE Transactions on*, 50(3):406–410, 2005.

S. V. Rakovic, E. C. Kerrigan, D. Q. Mayne, and J. Lygeros. Reachability analysis of discrete-time systems with disturbances. *Automatic Control, IEEE Transactions on*, 51(4):546–561, 2006.

J. Richalet, A. Rault, J. L. Testud, and J. Papon. Model predictive heuristic control: Applications to industrial processes. *Automatica*, 14(5):413–428, 1978.

R. T. Rockafellar. *Convex analysis*. Number 28. Princeton university press, 1970.

C. Rowe and J. Maciejowski. Tuning mpc using h_∞ loop shaping. In *American Control Conference, 2000. Proceedings of the 2000*, volume 2, pages 1332–1336. IEEE, 2000.

W. J. Rugh and J. S. Shamma. Research on gain scheduling. *Automatica*, 36(10): 1401–1425, 2000.

K. Rybnikov. *Polyhedral Partitions and Stresses*. PhD thesis, Queen University, Kingston, Ontario, Canada, 1999.

R. Schneider. *Convex bodies: the Brunn–Minkowski theory*, volume 151. Cambridge University Press, 2013.

A. Schrijver. *Theory of linear and integer programming*. John Wiley & Sons, 1998.

A. Schulz. *Lifting Planar Graphs to Realize Integral 3-Polytopes and Topics in Pseudo-Triangulations*. PhD thesis, Fachbereich Mathematik und Informatik der Freien Universit at Berlin, 2008.

F. Scibilia, S. Olaru, and M. Hovd. Approximate explicit linear mpc via delaunay tessellation. In *Control Conference (ECC), 2009 European*, pages 2833–2838. IEEE, 2009.

M. M. Seron, G. C. Goodwin, and J. A. Doná. Characterisation of receding horizon control for constrained linear systems. *Asian Journal of Control*, 5(2): 271–286, 2003.

J. Spjøtvold, E. C. Kerrigan, C. N. Jones, P. Tøndel, and T. A. Johansen. On the facet-to-facet property of solutions to convex parametric quadratic programs. *Automatica*, 42(12):2209–2214, 2006.

F. Stoican and S. Olaru. *Set-theoretic Fault-tolerant Control in Multisensor Systems*. John Wiley & Sons, 2013.

A. Suardi, S. Longo, E. C. Kerrigan, and G. A. Constantinides. Robust explicit mpc design under finite precision arithmetic. In *Proceedings of the 19th World Congress The International Federation of Automatic Control, Cape Town, South Africa*, 2014.

M. Sznaier and M. J. Damborg. Suboptimal control of linear systems with state and control inequality constraints. 1987.

F. Tahir and I. M. Jaimoukha. Robust positively invariant sets for linear systems subject to model-uncertainty and disturbances. In *Nonlinear Model Predictive Control*, volume 4, pages 213–217, 2012.

P. Tøndel, T. A. Johansen, and A. Bemporad. An algorithm for multi-parametric quadratic programming and explicit mpc solutions. *Automatica*, 39(3):489–497, 2003.

M. Vassilaki and G. Bitsoris. Constrained regulation of linear continuous-time dynamical systems. *Systems & Control Letters*, 13(3):247–252, 1989.

M. Vassilaki, J. C. Hennet, and G. Bitsoris. Feedback control of linear discrete-time systems under state and control constraints. *International Journal of control*, 47(6):1727–1735, 1988.

Y. Wang and S. Boyd. Fast model predictive control using online optimization. *Control Systems Technology, IEEE Transactions on*, 18(2):267–278, 2010.

K. Zhou, J. C. Doyle, K. Glover, et al. *Robust and optimal control*, volume 40. Prentice Hall New Jersey, 1996.

G. M. Ziegler. *Lectures on polytopes*, volume 152. Springer Science & Business Media, 1995.

Title : Explicit robust constrained control for linear systems : analysis, implementation and design based on optimization

Keywords : inverse optimality, piecewise affine functions, robustness/fragility margins

Abstract : Piecewise affine (PWA) feedback control laws have received significant attention due to their relevance for the control of constrained systems, hybrid systems; equally for the approximation of nonlinear control. However, they are associated with serious implementation issues. Motivated from the interest in this class of particular controllers, this thesis is mostly related to their analysis and design.

The first part of this thesis aims to compute the robustness and fragility margins for a given PWA control law and a linear discrete-time system. More precisely, the robustness margin is defined as the set of linear time-varying systems such that the given PWA control law keeps the trajectories inside a given feasible set. On a different perspective, the fragility margin contains all the admissible variations of the control law coefficients such that the positive invariance of the given feasible set is still guaranteed. It will be shown that if the given feasible set is a polytope, then so are these robustness/fragility margins.

The second part of this thesis focuses on inverse optimality problem for the class of PWA controllers. Namely, the goal is to construct an optimization problem whose optimal solution is equivalent to the given PWA function. The methodology is based on *convex lifting*: an auxiliary $1-$dimensional variable which enhances the convexity characterization into recovered optimization problem. Accordingly, if the given PWA function is continuous, the optimal solution to this reconstructed optimization problem will be shown to be unique. Otherwise, if the continuity of this given PWA function is not fulfilled, this function will be shown to be one optimal solution to the recovered problem.

In view of applications in linear model predictive control (MPC), it will be shown that *any continuous PWA control law can be obtained by a linear MPC problem with the prediction horizon at most equal to 2 prediction steps*. Aside from the theoretical meaning, this result can also be of help to facilitate implementation of PWA control laws by avoiding storing state space partition.

Another utility of *convex liftings* will be shown in the last part of this thesis to be a control Lyapunov function. Accordingly, this convex lifting will be deployed in the so-called *robust control design based on convex liftings* for linear system affected by bounded additive disturbances and polytopic uncertainties. Both implicit and explicit controllers can be obtained. This method can also guarantee the recursive feasibility and robust stability. However, this control Lyapunov function is only defined over the maximal $\lambda-$contractive set for a given $0 \leq \lambda < 1$ which is known to be smaller than the maximal controllable set. Therefore, an extension of the above method to the $N-$steps controllable set will be presented. This method is based on a *cascade of convex liftings* where an auxiliary variable will be used to emulate a Lyapunov function. Namely, this variable will be shown to be non-negative, to strictly decrease for N first steps and to stay at 0 afterwards. Accordingly, robust stability is sought.

Université Paris-Saclay
Espace Technologique / Immeuble Discovery
Route de l'Orme aux Merisiers RD 128 / 91190 Saint-Aubin, France

ÉCOLE DOCTORALE
Sciences et technologies de l'information et de la communication (STIC)

Titre : Commande robuste, explicite pour des systèmes linéaires: analyse, implémentation et synthèse fondée sur l'optimalité.

Mots clés : optimalité inverse, fonctions affines par morceaux, marges de robustesse/ fragilité

Résumé : Les lois de commande affines par morceaux ont attiré une grande attention de la communauté d'automatique de contrôle grâce à leur pertinence pour des systèmes contraints, systèmes hybrides également pour l'approximation de commandes non-linéaires. Pourtant, leur mise en œuvre est soumise à quelques difficultés. Motivé par l'intérêt à cette classe de commandes, cette thèse porte sur leur analyse, mise en œuvre et synthèse.

La première partie de cette thèse a pour but le calcul de la marge de robustesse et de la marge de fragilité pour une loi de commande affine par morceaux donnée et un système linéaire discret. Plus précisément, la marge de robustesse est définie comme l'ensemble des systèmes linéaires à paramètres variants que la loi de commande donnée garde les trajectoires dans de la région faisable. D'ailleurs, la marge de fragilité comprend toutes les variations des coefficients de la commande donnée telle que l'invariance de la région faisable soit encore guarantie. Il est montré que si la région faisable donnée est un polytope, ces marges sont aussi des polytopes.

La deuxième partie de ce manuscript est consacrée au problème de l'optimalité inverse pour la classe des fonctions affines par morceaux. C'est-à-dire, l'objective est de définir un problème d'optimisation pour lequel la solution optimale est équivalente à la fonction affine par morceaux donnée. La méthodologie est fondée sur le *convex lifting*, i.e., un variable auxiliaire, scalaire, qui permet de définir un ensemble convex à partir de la partition d'état de la fonction affine par morceaux donnée. Il est montré que si la fonction affine par morceaux donnée est continue, la solution optimale de ce problème redéfini sera unique. Par contre, si la continuité n'est pas satisfaite, cette fonction affine par morceaux sera une solution optimale parmi les autres du problème redéfini. En ce qui concerne l'applications dans la commande prédictive, il sera montré que *n'importe quelle loi de commande affine par morceaux continue peut être obtenue par un autre problème de commande prédictive avec l'horizon de prédiction au plus égal à* 2. A côté de cet aspect théorique, ce résultat sera utile pour faciliter la mise en oeuvre des lois de commandes affines par morceaux en évitant l'enregistrement de la partition de l'espace d'état.

Dans la dernière partie de ce rapport, une famille de *convex liftings* servira comme des sfonctions de Lyapunov. En conséquence, ce "*convex lifting*" sera déployé pour synthétiser des lois de commande robustes pour des systèmes linéaires incertains, également en présence de perturbations additives bornées. Des lois implicites et explicites seront obtenues en même temps. Cette méthode permet de guarantir la faisabilité récursive et la stabilité robuste. Cependant, cette fonction de Lyapunov est limitée à l'ensemble $\lambda -$contractive maximal avec une constante scalaire $0 \leq \ \lambda \ < \ 1$, qui est plus petit que l'ensemble controllable maximal. Pour cette raison, une extension de cette méthode pour l'ensemble controllable de $N -$ pas, sera présentée. Cette méthode est fondée sur des convex liftings en cascade où une variable auxiliaire sera utilisée pour servir comme une fonction de Lyapunov. Plus précisément, cette variable est non-négative, strictement décroissante pour les N premiers pas et égale toujours à $0 -$ après. Par conséquent, la stabilité robuste est guarantie.

Université Paris-Saclay
Espace Technologique / Immeuble Discovery
Route de l'Orme aux Merisiers RD 128 / 91190 Saint-Aubin, France