

Robustesse de la commande pr´edictive explicite

Rajesh Koduri

▶ **To cite this version:**

Rajesh Koduri. Robustesse de la commande pr´edictive explicite. Autre. Université Paris Saclay (COmUE), 2017. Français. NNT : 2017SACLC052 . tel-01668603

HAL Id: tel-01668603
https://theses.hal.science/tel-01668603

Submitted on 20 Dec 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

NNT : 2017SACLC052

THÈSE DE DOCTORAT
DE
L'UNIVERSITÉ PARIS-SACLAY
PRÉPARÉE À
CENTRALESUPÉLEC

ECOLE DOCTORALE N 580
Sciences et Technologies de l'Information et de la Communication (STIC)
Spécialité de doctorat: AUTOMATIQUE

Par

Rajesh Koduri

Robustness of Explicit Model Predictive Controllers

Thèse présentée et soutenue à Gif-sur-Yvette, le 29 Septembre 2017:

Composition du Jury:

Mme. Alexandra Grancharova	UCTM Sofia	Rapporteur
M. George Bitsoris	University of Patras Athens	Rapporteur
M. Shyam Kamal	IIT Varanasi	Examinateur
M. Carlos Eduardo Trabuco Dórea	UFRN Brazil	Examinateur
M. Hugues Mounier	CentraleSupélec/L2S	Président du jury
M. Sorin Olaru	CentraleSupélec/L2S	Co-encadrant de thèse
M. Pedro Rodriguez-Ayerbe	CentraleSupélec/L2S	Directeur de thèse

To my parents, my brother, sister-in-law and Esha.

Abstract

The control design techniques for linear or hybrid systems under constraints lead often to off-line state-space partitions with non-overlapping convex polyhedral regions. This corresponds to a piecewise affine (PWA) state feedback control laws associated to polyhedral partition of the state-space. Such control laws can be effectively implemented on hardwares for real-time control applications. However, the robustness of the explicit solutions depends on the accuracy of the mathematical model of the dynamical systems. The uncertainties in the system model pose serious challenges concerning the stability and implementation of the piecewise affine control laws. Motivated by the challenges facing the explicit solutions for the uncertainties in the dynamical systems, this thesis is mostly related to their analysis and re-design.

The first part of this thesis aims to compute robustness margins for a given nominal PWA control law obtained for a linear discrete-time system. Classical robustness margins i.e., gain margin and phase margin, consider the gain variation and phase variation of the model for which the stability of the closed loop is preserved. In this thesis work, an attempt to find the same kind of margin for a piecewise affine (PWA) controller is made. Starting from the invariance property of the closed loop obtained involving a discrete dynamic model and PWA controller in a convex region of the state space, we calculate the two robustness margins preserving this invariance property. The first one will be denoted as gain margin corresponding to the variation of the gain of the model guaranteeing the invariance. The second one, denoted the robustness margin against first order neglected dynamics will correspond to the slowest first order neglected dynamics allowed in the system preserving the invariance property. Next, we compute three robustness margins for a PWA contractive control law and linear discrete-time system. The first robustness margin is characterized for the polytopic uncertainty affecting the dynamical system model parameters. The other two robustness margins are concerned with the gain and phase variations for the contractive PWA control law for which the contractive state trajectories are achieved.

The second part of the thesis aims to consider perturbation in the representation of the vertices of the polyhedral regions. The idea behind this is to perform a quantization operation on the representation of the state-space regions and the associated PWA control laws in order to reduce the hardware requirements in terms of processor speed and memory unit. The quantized state-space partitions lose some of the important properties of the explicit controllers: ”non-overlapping”, ”convexity” and ”invariant characterization”. How the perturbation affects the polyhedral regions and invoke overlapping to the modified polyhedral regions is first shown. The

major contribution of this work is to analyze to what extent the non-overlapping and the invariance characteristics of the PWA controller can be preserved when perturbation takes place on the vertex representation. We determine two different sets called vertex-sensitivity region and sensitivity margin region to characterize admissible perturbation preserving the non-overlapping and the invariance property of the controller respectively. Finally, we show how to perturb multiple vertex sequentially and reconfigure the polyhedral regions to the perturbed vertices.

The third part aims to analyze the complexity of the explicit solutions in terms of computational time and memory storage requirements. In order to understand the computational complexity of the evaluation of the PWA functions, we propose two objectives, the first objective being the comparison made between the sequential and parallel evaluations of the PWA functions for the Alternating Direction Method of Multiplier (ADMM) algorithm. The second objective is to compare the computational complexity of the parallel evaluations of the PWA functions for the Progressive Hedging Algorithm (PHA) on the Central Processing Unit (CPU) and Graphical Processing Unit (GPU).

Résumé

Les techniques de conception de commandes pour les systèmes linéaires ou hybrides soumis à des contraintes linéaires conduisent souvent à des lois de commande linéaires par morceaux. Ces lois linéaires sont calculées hors ligne et sont définies dans des régions polyédrales convexes de l'espace d'état. Ceci correspond à des lois de commande par retour d'état affines par morceaux (PWA) associées à une partition polyédral de l'espace d'état. De telles lois de commande peuvent être implémentées efficacement sur des matériels spécifiques pour des applications de commande en temps réel. Cependant, la robustesse des solutions explicites dépend de la précision du modèle mathématique des systèmes dynamiques. Les incertitudes du modèle posent de sérieux problèmes concernant la stabilité et la mise en œuvre des lois de commande affines par morceaux. Motivés par les défis auxquels font face les solutions explicites face à des incertitudes des systèmes dynamiques, cette thèse est principalement liée à leur analyse et à leur configuration.

La première partie de cette thèse vise à calculer les marges de robustesse pour une loi de commande PWA nominale donnée pour un système linéaire à temps discret. Les marges de robustesse classiques, c'est-à-dire la marge de gain et la marge de phase, considèrent la variation de gain et la variation de phase du modèle pour lesquelles la stabilité de la boucle fermée est préservée. Dans ce travail de thèse, une tentative de trouver le même type de marge pour un contrôleur affine par morceaux (PWA) est faite. En partant de la propriété d'invariance de la boucle fermée obtenue avec un modèle dynamique discret et un contrôleur PWA dans une région convexe de l'espace d'état, deux marges de robustesse préservant cette propriété d'invariance sont calculées. La première sera noté marge de gain et correspond à la variation du gain du modèle garantissant l'invariance. La seconde, noté marge de robustesse face à une dynamique négligée du premier ordre, correspond à la plus lente dynamique négligée du premier ordre préservant la propriété d'invariance. Ensuite, trois marges de robustesse sont calculées pour une loi de PWA et un système à temps discret linéaire, avec la caractéristique d'avoir des trajectoires contractives dans l'espace d'état. La première marge de robustesse est caractérisée pour l'incertitude polytopique affectant les paramètres du modèle dynamique qui préservent la vitesse convergence des trajectoires. Les deux autres marges de robustesse concernent les variations de gain et de dynamique négligée de premier ordre pour lesquelles les trajectoires d'état contractives sont préserves.

La deuxième partie de la thèse vise à considérer les perturbations dans la représentati-

on des sommets des régions polyédrales de la loi de commande. L'idée derrière ceci est d"effectuer une opération de quantification sur la représentation des régions de l'espace d'état et des lois de commande PWA associées afin de réduire les exigences matérielles en termes de vitesse du processeur et de la capacité de mémoire. Les partitions de l'espace d'état quantifiées perdent certaines des propriétés importantes des contrôleurs explicites, à savoir le non-recouvrement des régions, la convexité des partitions et l'invariance. Premièrement, comment la quantification des sommets affecte les régions polyédrales et comment la propriété de non-recouvrement peut être perdue est montré. Dans une deuxième étape, on discute dans quelle mesure les caractéristiques de non-recouvrement et d'invariance du contrôleur PWA peuvent être préservées quand la perturbation a lieu sur la représentation du sommet est analysée. Deux ensembles différents appelés région de sensibilité au sommet et région de marge de sensibilité pour caractériser la perturbation admissible en préservant respectivement la propriété de non-recouvrement et d'invariance du contrôleur sont calculées. Finalement, il est montré comment perturber séquentielle ment plusieurs sommets et reconfigurer les régions polyédrales obtenues.

La troisième partie vise à analyser la complexité des solutions explicites en termes de temps de calcul et de stockage mémoire. Les solutions explicites deviennent très complexes, dès que l'ordre du modèle ou l'horizon de prédiction sont grandes. Dans cette partie, le problème d'optimisation amenant à la loi de commande PWA est partitionné en utilisant l'algorithme ADMM (Alternating Direction Method of Multiplier). Afin de comprendre la complexité de calcul de l"évaluation des fonctions PWA, deux objectifs sont proposés. Le premier est la comparaison entre les évaluations séquentielles et parallèles des fonctions PWA pour l'algorithme ADMM. Le deuxième objectif est de comparer la complexité computationnelle des évaluations parallèles des fonctions PWA pour l'algorithme de couverture progressive (PHA) sur l'unité centrale de traitement (CPU) et l'unité graphique de traitement (GPU).

Motivation de la thèse

Dans les travaux récents de [ONB+13, NOBRA14], les auteurs ont proposé une approche géométrique générique pour dériver la marge de robustesse pour la loi de commande PWA par rapport aux propriétés d'invariance positive obtenue avec le modèle nominal. Cette approche est adaptée à la particularité des caractéristiques affines par morceaux de la dynamique en boucle fermée et diffère d'une analyse de robustesse classique. L'objectif de cette approche géométrique est la description d'un ensemble polyédral dans l'espace des paramètres décrivant la variation paramétrique admissible sans perte de la propriété d'invariance. Il convient de mentionner que cette procédure d'analyse proposée dans [ONB+13, NOBRA14] gère la robustesse / fragilité de l'invariance positive mais ne prolonge pas l'analyse à la convergence (propriétés de stabilité asymptotique). En outre, les travaux mentionnés n'abordent pas les problèmes concernant la modification du gain ou des dynamiques négligées. L'analyse de la robustesse du contrôleur pour le système affecté par la dynamique négligée est très importante dans le contexte de l'inadéquation du modèle ou des

paramètres du modèle inexacts. Par ailleurs, l'analyse des caractéristiques du correcteur PWA en ce qui concerne les perturbations sur les sommets des partitions de l'espace d'état du correcteur est considérée comme vitale à de nombreuses fins. L'objectif est d'analyser l'effet d'une opération de "quantification" sur la partition de l'espace d'état donnée par la représentation de sommets des polyèdres. Cela permet de réduire l'espace mémoire nécessaire pour stocker le correcteur. En effet, la complexité des solutions explicites augmente avec l'inclusion des contraintes imposées aux systèmes dynamiques. Dans un tel cas, effectuer une opération de quantification sur les sommets de la partition polyédral aidera efficacement les exigences matérielles.

C'est un phénomène largement accepté que les solutions explicites souffrent encore de la complexité computationnelle pour des systèmes dynamiques à moyenne échelle (un système avec 5 états sera considéré comme entrant dans cette catégorie) et pour un long horizon de prédiction, (par exemple 10).

De plus, la commande prédictive explicite a gagné en réputation pour sa simple évaluation des lois de commande. Actuellement les, recherches ont été plus axées sur la réduction de la complexité de la solution explicite hors ligne du problème mp-QP. Il est avéré que la réduction de la complexité du problème mp-QP entraîne moins d'exigences matérielles pour la mise en œuvre mais entraine une perte de performance. Par contre, Il n'y a pas de recherche notable à mentionner en ce qui concerne l'évaluation en ligne des fonctions PWA pour les solutions complexes. Ces faits ont déclenché le présent travail de recherche qui vise à analyser les méthodes itératives de résolution de problèmes mp-QP au moyen des solutions explicites.

Objectifs

Dans cette section sont énumérés les objectifs abordés dans cette thèse et la façon dont sont présentés dans le manuscrit.

Chapitre 3: Marges de robustesse pour une loi de commande explicite par morceaux

Dans ce chapitre, les propriétés de robustesse inhérentes d'une commande prédictive explicite, décrite comme une loi de commande affine par morceaux (PWA) pour une classe de systèmes linéaires à temps discret, est considérée. Premièrement une méthode numérique pour calculer une marge de gain définie pour un système à temps discret stabilisé par une commande continue affine par morceaux est présentée. La marge de gain est un polytope qui caractérise les variations des gains du système préservant les caractéristiques d'invariance de la boucle fermée. Deuxièmement, l'influence d'une dynamique négligée du premier ordre est analysée. La marge de robustesse du contrôleur par rapport à la dynamique négligée du premier ordre

correspond à un ensemble caractérisant les paramètres de la dynamique négligée préservant la propriété d'invariance de la boucle fermée.

Marge de gain:

Cette section, présente l'influence 'effet de la modification du gain par rapport aux caractéristiques en boucle fermée obtenues avec la loi de commande explicite affine par morceaux. L'énoncé du problème de la marge de gain en boucle fermée est donné dans les équations ci-dessous et est également illustré dans la Figure 1.

$$\begin{aligned} x(k+1) &= Ax(k) + B(I_m + diag(\delta_K))u_{pwa}(x(k)) \\ u_{pwa}(x(k)) &= F_i x(k) + g_i \end{aligned}$$

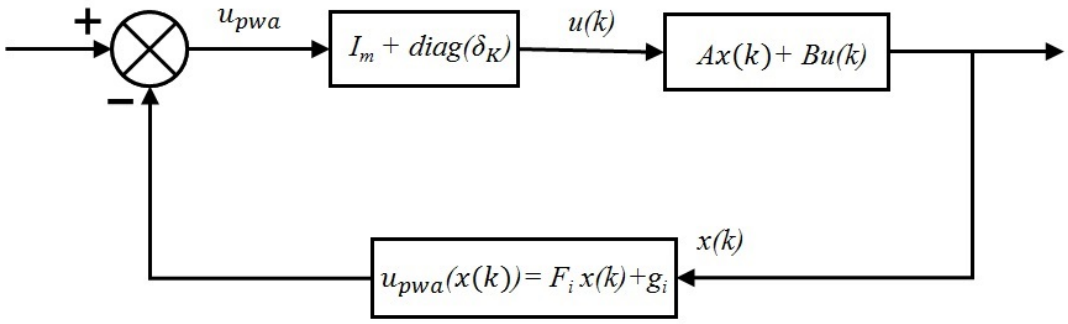


Figure 1: Modification du gain du système.

La construction de l'ensemble des marges de gain sera basée sur la représentation des sommets des régions formant la partition du correcteur PWA, nommé $\mathcal{R}$. $\mathcal{R}$ est une polytope dans l'espace d'état où la loi de commande est définie. Par ailleurs, $\mathcal{R}$ est invariant, c'est-à-dire, que les trajectoires commençant en $\mathcal{R}$ restent en $\mathcal{R}$.

Définition 1 *Considérons un système linéaire à temps discret avec une loi de commande continue PWA telle que l'ensemble $\mathcal{R}$ dans l'espace d'état soit positivement invariant. La marge de gain est représentée par l'ensemble $\mathcal{K}$, pour lequel*

$$x(k+1) = Ax(k) + B(I_m + diag(\delta_K))u_{pwa}(x(k)) \in \mathcal{R},\ \forall x(k) \in \mathcal{R} \text{ et } \delta_K \in \mathcal{K} \subset \mathbb{R}^m.$$

L'ensemble $\mathcal{K}$ est un ensemble qui contient les des variations de gain telles que pour tout point à l'intérieur de l'ensemble $\mathcal{K}$, les caractéristiques d'invariance de l'ensemble $\mathcal{R}$ sont préservées.

Dynamique négligée du premier ordre:

Dans cette partie la robustesse du contrôleur PWA nominal, défini sur l'ensemble polyédral R, face à des dynamiques négligées du premier ordre est analysée. L'approche est de présenter un ensemble admissible pour les variations des paramètres de

la dynamique négligé pour lesquelles l'invariance de l'ensemble R est préservée. Le problème est formulé pour la dynamique du premier ordre qui perturbe le système en boucle fermée, comme le montre la Figure 2. Le système linéaire nominal en

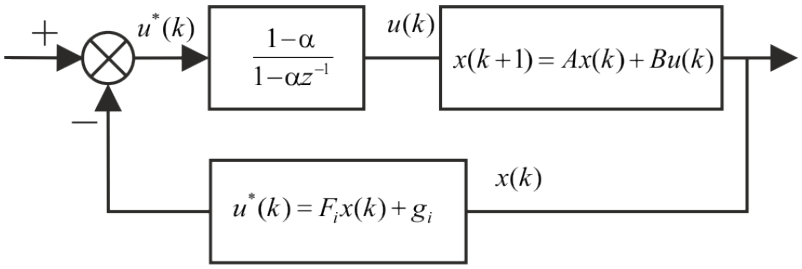


Figure 2: Dynamique négligée dans la boucle fermée.

temps discret et la loi de commande PWA à retour d'état peuvent être écrits sous la forme:

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) \\ u_{pwa}(x(k)) &= u^*(k) = F_i x(k) + g_i \end{aligned}$$

La dynamique négligée peut être représentée par une équation du premier ordre et elle peut s'écrire sous la forme:

$$\begin{aligned} x_I(k+1) &= \alpha x_I(k) + (1-\alpha)u^*(k) \\ u(k) &= \alpha x_I(k) + (1-\alpha)u^*(k) \end{aligned}$$

La dynamique négligée dans le modèle en boucle fermée peut être décrite par:

$$\begin{aligned} x(k+1) &= Ax(k) + Bx_I(k+1) \\ x(k+1) &= Ax(k) + \alpha B x_I(k) + B(1-\alpha)u^*(k) \end{aligned}$$

Définition 2 *Considérons un système discret linéaire affecté par la dynamique négligée du premier ordre et stabilisé via une loi de commande par retour d'état PWA. La marge de robustesse face pour à la dynamique négligée du premier ordre est caractérisée par un ensemble $\Omega^\alpha \in \mathbb{R}^m$ qui contient les valeurs du paramètre α(pour chaque entrée) de sorte que la propriété d'invariance de l'ensemble $\mathcal{R}$ soit assurée.*

Marges de robustesse pour les lois de commande explicites affines par morceaux

Dans ce chapitre, un problème de programmation quadratique multiparamétrique est formulé afin de calculer une loi de commande PWA qui impose la convergence des trajectoires d'état en boucle fermé. A partir de la loi de commande λ-contractive, nous présentons trois marges de robustesse pour le système stabilisé par cette loi de commande affine par morceaux. Tout d'abord, la marge de robustesse pour un système à temps discret affecté par des incertitudes paramétriques est présentée. La marge de robustesse notée Ω_{rob} est définie comme un sous-ensemble paramétrique et se présente sous la forme d'un ensemble polyédral. Pour tous les modèles appartenant à l'ensemble polyédral Ω_{rob}, la convergence des trajectoires d'état est garantie. Deuxièmement, une méthode numérique pour calculer une marge de gain

définie pour un système à temps discret stabilisé par une commande PWA contractive est présentée. La marge de gain désirée prend la forme d'un ensemble qui caractérise les variations admissibles de gain du système préservant les caractéristiques de convergence de la boucle fermée. Troisièmement, un ensemble admissible qui caractérise les variations de paramètres pour une dynamique négligée du premier ordre garantissant les caractéristiques de convergence de la boucle fermée est présenté.

Marge de robustesse paramétrique pour les ensembles λ contractifs:

Dans cette section, nous considérons l"incertitude paramétrique sur les matrices A et B du modèle du système linéaire discret. Un ensemble est introduit dans l'espace des paramètres,

$$\Omega = Conv\{[A_1\ B_1] \cdots [A_L\ B_L]\}.$$

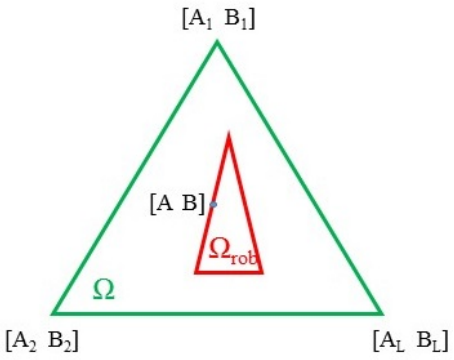


Figure 3: Description polytopique du modèle.

Définition 3 *Considérons un ensemble polytopique dans l'espace des paramètres du modèle. Le problème de la marge de robustesse est de calculer le plus grand sous-ensemble Ω_{rob} pour lequel les trajectoires obtenues avec la loi de commande PWA donnée par $u_{pwa}(x(k))$ et définie sur l'ensemble polyédrique $\mathcal{R}$ soient λ contractives.*

Marge de gain pour les ensembles λ contractifs:

Définition de la marge de gain définie pour les systèmes linéaires à temps discret stabilisés à l'aide d'une loi de commande PWA λ contractive. La construction de la marge de gain est similaire à celle présentée dans la section précédente.

Définition 4 *Considérons un système linéaire à temps discret avec une loi de commande PWA contractive continue, telle que l'ensemble d'états $\mathcal{R}$ soit λ-contractif. La marge de gain est représentée par l'ensemble $\mathcal{K}$, tel que:*

$$\begin{aligned} x(k+1) \ &= Ax(k) + B(I_m + diag(\delta_K))u_{pwa}(x(k)) \in \lambda\beta\mathcal{R}, \forall x(k) \in \mathcal{R}, \\ &\qquad \delta_K \in \mathcal{K} \subset \mathbb{R}^m \ et\ \lambda, \beta \in [0,1]. \end{aligned}$$

Marge de robustesse face à des dynamiques négligée pour les ensembles λ contractifs:

Cette section présente une analyse de la robustesse du contrôleur PWA face à des dynamiques négligées du premier ordre.

Définition 5 *Considérons un système linéaire à temps discret affecté par la dynamique négligée du premier ordre commande par un contrôleur à retour d'état PWA contractif. La marge de robustesse pour la dynamique négligée du premier ordre est caractérisée par un ensemble $\Omega^{\alpha} \in \mathbb{R}$ qui contient les valeurs du paramètre α pour lesquels la propriété de λ-contractif est préservée.*

Chapitre 5: Précision dans la représentation des partitions polyédrales et la fragilité de la commande PWA

Le cadre du présent chapitre est celui d'un système linéaire à temps discret commandé par une loi de commande explicite par morceaux. L'objectif est d'analyser la modification des partitions polyédrales en cas de perturbation sur la représentation des sommets des partitions. La contribution principale de ce travail est d'analyser dans quelle mesure les caractéristiques de non-recouvrement et d'invariance du contrôleur PWA peuvent être préservées lorsqu'une perturbation sur la représentation des sommets des partitions a lieu. Le chapitre est divisé en cinq sections, qui correspondent avec la procédure proposée pour la prise en compte des quantifications dans la représentation des sommets.

1. Caractériser les perturbations admissibles sur chaque sommet pris indépendamment dans la représentation des sommets. La perturbation est considérée comme admissible si elle préserve la propriété de non-recouvrement dans l'espace d'état. La perturbation admissible est caractérisée par un ensemble appelé régions de sensibilité aux sommets.

2. Fournir une méthode de mise à jour de la frontière de l'ensemble faisable pour la perturbation admissible en mettant à jour les sommets considérés comme sensibles. La mise à jour préserve la propriété de non-recouvrement et d'invariance.

3. Troisièmement, à partir du domaine réalisable reconfiguré, analyse des partitions polyédriques en considérant chacun des sommets de la partition qui ne sont pas placés aux frontières du domaine réalisable. Un ensemble appelé marge de sensibilité par rapport à l"invariance en boucle fermée est déterminé. Cet ensemble caractérise les perturbations admissibles sur la représentation des sommets préservant les caractéristiques de non- recouvrement et d'invariance du contrôleur PWA.

4. Quatrièmement, calcule des partitions polyèdres perturbées pour tous les sommets internes de l'ensemble faisable terminant séquentiellement la transformation des régions polyédriques d'origine en une nouvelle région polyédrique pour tous les sommets de l'ensemble faisable.

5. Cinquièmement, à partir d'un système linéaire stabilisé à l'aide d'une loi de commande affine par morceaux (PWA), tous les sommets de l'ensemble faisable sont misent à jours. Les perturbations des sommets sont réalisées en assurant les propriétés de non-recouvrement et de λ-contraction du contrôleur PWA.

Chapitre 6: Evaluation de la fonction PWA sur CPU et GPU pour ADMM et PHA

Ce chapitre cherche des solutions pour la mise en œuvre des solutions explicites pour des dynamiques linéaires discrètes avec des horizons de prédiction plus longs. Dans une deuxième partie des modèles dynamiques linéaires avec des incertitudes probabilistes sont considérés. Une solution basée sur des solutions explicites est présentée. Les solutions explicites sont une fonction algébrique simple, c'est-à-dire, qui fonctionne dans un cadre de multiplication matrice-matrice ou matrice-vecteur. De telles expressions sont parallélisables par l'unité centrale de traitement (CPU) et il est également possible d'activer l'unité de traitement graphique (GPU) pour décomposer les grandes matrices en sous-processeurs disponibles, afin d'obtenir des opérations plus rapides. Afin de permettre la mise en ouvre MPC explicite pour un horizon de prédiction plus long, la technique sur laquelle nous nous appuyons s'appelle le fractionnement d'opérateur (operator splitting). La division est effectuée sur l"horizon de prédiction et nous avons un nombre de sous-problèmes égal à la longueur de l'horizon de prédiction. En utilisant les solutions explicites, N+1 sous-problèmes (longueur de l'horizon de prédiction) sont réduits à trois sous-problèmes. L'approche ADMM (Alternating Direction Method of Multipliers) est utilisée pour satisfaire les contraintes de consensus survenues lors du découplage et pour la convergence. Cette approche incorporée à la formulation EMPC est résolue avec CPU. Enfin, nous comparons la complexité computationnelle de la formulation ci-dessus pour CPU en plus de la formulation MPC explicite générale (sans fractionnement temporel).

La deuxième partie du chapitre étudie l'implantation de l'algorithme PHA (Progressive Hedging Algorithm) avec des solutions explicites. Ce type d'algorithme est utilisé pour calculer la commande prédictive pour des systèmes affectés par des perturbations probabilistes. En résume, l'idée principale de ce chapitre est d'analyser la complexité des solutions explicites en termes de complexité de calcul et de stockage mémoire. L'objectif de ce chapitre est répertorié ci-dessous.

1. La comparaison des évaluations séquentielles et parallèles des fonctions PWA pour l'algorithme ADMM.
2. La comparaison des évaluations parallèles des fonctions PWA pour le PHA sur CPU et GPU.

Acknowledgements

The completion of this thesis has been achieved with the help of several people to whom I would like to express my gratitude and my warmest thanks.

Foremost, I would like to express my sincere gratitude to my thesis supervisors Prof. Pedro RODRIGUEZ-AYERBE and Prof. Sorin OLARU for their constant support, ideas and precious discussions. Throughout this thesis, their ideas and commitment have navigated me in right direction and supported my first steps in the research world. I am indebted to them for all these supports. Only some words are few to say all these.

My thanks are due to Assoc Prof. Alexandra Grancharova and Prof. George Bitsoris for their time spent on the evaluation of this thesis. Also, my thanks are given to Assoc Prof. Carlos Eduardo Trabuco Dórea, Dr. Shyam Kamal and Prof. Hugues Mounier for their acceptance to be examiners of my thesis.

I would like to thank Dr. Michal KVASNICA for supervising my first secondment at STUBA Bratislava where I learned many things from the works of him and his students. Also, I would like to express my warmest gratitude to Dr. Colin Jones for supervising my second secondment at EPFL Switzerland and helped setting the objective of the final chapter of thesis. My thanks go to Harsh Shukla, for his patience and dedication, with whom I collaborated for the final chapter of this thesis work. I am thankful to Prof. Morten Hovd for sharing ideas and also reviewing some of our works.

Also, I truly appreciate and highly regard the professors and research students from the TEMPO (Training on Embedded Predictive Control and Optimization) -A Marie Curie Initial Training Network project for all the meetings, presentations, professional development trainings, research sharing and project review. Our special thanks go to TEMPO Project coordinator Prof. Tor Arne Johansen for providing us this wonderful opportunity of being a part of such a diverse and intellectual group. Our association with the TEMPO group enabled us to work and collaborate with professors and students from top universities in the European Union.

My heartiest thanks to my former and current doctoral colleagues at CentraleSupelec Mohammad Laraba, Iris Ballesteros, Abid Rahman, Ngoc Anh Nguyen, Djawad Hamache, Mircea Dumitru, Minh Tri Nguyen, Nicolo Gionfra, Sophie Frasnedo, Guillaume Avrin, Miassa Taleb, Seif Eddine, Guillaume Jeanne, Andrea Beciu, Fethi

Benyoubi, Abdelkrim Bahloul, Binh Quang, Nolwenn Briquet, Nathan Michel, Mert Safa, Gauthier Rousseau, Ion Ghita, Fetra Rasoanarivo and Dory Merhy for their tremendous support, sharing experiences and knowledges in works and life.

Last but not the least, I thank my parents and my brother for their support, care and encouraging words that certainly acted as a paddle and propelled me to have a smooth sail in my academics.

I acknowledge the several friends and well wishers, whom I have not mentioned above and whose best wishes have always encouraged me.

Contents

Abstract **v**

Résumé **vii**

Acknowledgements **xv**

List of Figures **xxiv**

List of Tables **xxv**

1 Introduction **1**

1.1 Motivation of the Thesis 4

1.2 Objectives 5

1.3 Publications 7

2 Background Theory **9**

2.1 Convex Sets and Convex functions 10

2.2 Set theoretic definitions 11

2.3 Polyhedra and Polytope 11

2.4 Linear Systems 13

2.5 Set Invariance 14

2.6 Contractive Sets 15

2.7 Model Predictive Control 15

2.7.1 Mathematical Formulation of MPC 16

2.8 Explicit Model Predictive Control 18

2.8.1 Real-time implementation of EMPC solutions: 21
2.9 Tools for multi-parametric programming 22
2.10 Conclusion 22

3 Robustness margins for piecewise affine explicit control law 23
3.1 Introduction 23
3.2 Preliminaries 24
3.3 Gain Margin 26
3.3.1 Problem Formulation 26
3.3.2 Construction/Constructive results 28
3.3.3 Computation results 29
3.3.4 Example 30
3.4 First order neglected dynamics 39
3.4.1 Problem Formulation 40
3.4.2 Construction of vertex representation for the augmented state 41
3.4.3 Controller for extended state system 41
3.4.4 Admissible set for the first order neglected dynamics: 42
3.4.5 Example 44
3.5 Conclusion 44

4 Robustness margins for contractive piecewise affine explicit control laws 47
4.1 Introduction 47
4.2 Control design for controlled λ-contractive set 48
4.3 Main Results 52
4.3.1 Parametric margin for λ-controlled contractive sets 52
4.3.2 Gain margin for λ-controlled contractive sets 55
4.3.3 Neglected dynamics margin for λ-controlled contractive sets . 56
4.4 Examples 57
4.4.1 Robustness parametric margin 57

4.4.2 Gain Margin . . . 61
4.4.3 Robustness Margin for First order neglected dynamics . . . 62
4.5 Conclusions . . . 63

5 Precision in polyhedral partition representation and the fragility of PWA control 65

5.1 Introduction . . . 65
5.2 Motivation and Problem Formulation . . . 68
5.3 Treatment of a vertex considered independently - Polyhedral overlapping . . . 73
5.3.1 Characterization of the vertex sensitivity . . . 74
5.3.2 Admissibility restriction with respect to the input constraints 78
5.4 Impact of Vertex Perturbation on the invariance characterization . . . 79
5.4.1 Perturbations of vertices on the frontier of the feasible domain $\mathcal{R}$ . . . 79
5.4.2 Margin of Convexity . . . 82
5.4.3 Treatment of one inner vertex for non-overlapping and invariance 87
5.5 Treatment of multiple vertex perturbation . . . 89
5.6 Example . . . 90
5.7 Contractivity . . . 95
5.8 Conclusion . . . 99

6 PWA function evaluations on CPU and GPU for ADMM and PHA 101

6.1 Introduction . . . 101
6.2 Time-splitting approach for Explicit MPC . . . 104
6.2.1 Off-line Computation . . . 106
6.2.2 On-line computation and evaluation . . . 108
6.2.3 Evaluation of PWA functions for Matrix Input . . . 109
6.2.4 Examples . . . 110
6.3 Multi-Stage Stochastic Programming . . . 114
6.3.1 Decomposition of Scenario tree . . . 115

6.3.2 Progressive Hedging Algorithm . . . 115
6.3.3 Example . . . 118
6.4 Conclusions . . . 122

7 Conclusion **123**
7.1 Summary of Thesis Achievements . . . 123
7.2 Future Works . . . 124

Bibliography **127**

A Appendices **135**
A.1 Explicit MPC . . . 135
A.1.1 Explicit solutions with Critical regions . . . 135
A.1.2 Explicit solutions without Critical regions . . . 137

List of Figures

1 Modification du gain du système. x

2 Dynamique négligée dans la boucle fermée. xi

3 Description polytopique du modèle. xii

2.1 An illustrative example for explicit model predictive control. 21

3.1 Illustration of the set $\mathcal{R}$ with its vertices. 25

3.2 Illustration of the set $\mathcal{R}_i$ with its vertices 25

3.3 Representation of Gain Margin in closed-loop 26

3.4 Illustration of the Optimal feedback law. 31

3.5 Representation of Polyhedral partition. 31

3.6 state trajectories for $\delta_K = 0$. 32

3.7 state trajectories for $\delta_K = 0.3$. 32

3.8 state trajectories for $\delta_K = -0.25$ 33

3.9 States and control input simulations for different gains. 34

3.10 Polyhedral partitions for different prediction horizons. 35

3.11 Illustration of set $\mathcal{R}$ for different prediction horizons 35

3.12 Illustration of sets $\mathcal{R}_i$ and $\mathcal{R}$ 36

3.13 $\mathcal{K}$ set for $u_{pwa}(x(k))$ 37

3.14 $\mathcal{K}$ set for $u^{\mu}_{pwa}(x(k))$ 37

3.15 Polyhedral partition and the gain margin set $\mathcal{K}$ for prediction horizon $N_p = 5$. 38

3.16 Polyhedral partition and the gain margin set $\mathcal{K}$ for prediction horizon $N_p = 10$. 38

3.17 State trajectories and control input simulation for $N_p = 5$ for different gain margins 39

3.18 Closed loop system with first order neglected dynamics 40

3.19 Representation of polyhedral partition with 13 regions. 44

3.20 State trajectories for $\alpha = 0.325$. 45

3.21 Simulations of state trajectories and control input for different α values. 45

4.1 Illustrative representation of the sets Ω and Ω_{rob}. 52

4.2 State space partition with β as parameter 57

4.3 Projected state space partition x_1, x_2. 58

4.4 Robustness margin for contractive and invariant set in the plane of ζ_1, ζ_2. 58

4.5 Trajectories for different nominal systems for the same initial state. . 59

4.6 Simulation for the state trajectories and control input for an initial state $x_0 = [60 \ -80]^T$. 59

4.7 State space partition for $N_p = 10$ 60

4.8 Robustness margin for contractive and invariant set for $N_p = 10$. . . . 60

4.9 Projected state space partition with state trajectories for different δ_K values. 61

4.10 Simulation for state trajectories, control input and β for different δ_K values. 62

4.11 Representation of the set $\mathcal{R}$ with the simulation of state trajectories for all the frontier vertices as initial states for different gain margins δ_K. 63

5.1 2-D polyhedral representation before and after perturbation on the half-space representation. 69

5.2 2-D polyhedral representation before and after perturbation of the vertex representation. 71

5.3 2-D polyhedral representation before and after perturbation of the vertex representation 73

5.4 Polyhedral partition with four regions, the vertex of interest v and the vertex sensitivity region Ψ^v are shown. 76

5.5 The vertex v denoted by a black dot in Fig. 3.4 (a) is perturbed to $\hat{v}$ changing the regions $\mathcal{R}_1, \mathcal{R}_2, \mathcal{R}_3, \mathcal{R}_4$ to $\hat{\mathcal{R}}_1, \hat{\mathcal{R}}_2, \hat{\mathcal{R}}_3, \hat{\mathcal{R}}_4$. 76

5.6 Classification of the vertices of $\mathcal{P}_N(\mathcal{R})$. 79

5.7 Illustration for perturbation of the non-extreme vertex. 83

5.8 Representation of the set $\mathcal{R}$ and its image $\mathcal{F}_{pwa}$ are represented by contour in dashed lines and the contour in full lines respectively. The colored polytopes apart from white and gray ones represent the vertex sensitivity for the vertices depicted in blue dots 84

5.9 Representation of the set $\mathcal{R}$ given by contour in dashed red line. The solid polytopes inside the set $\mathcal{R}$ are the images $\mathcal{F}$ of the set $\mathcal{R}$ for each iteration in Algorithm 5.4.1. 86

5.10 Representations of the set $\mathcal{R}$ and the output of the Algorithm 5.4.1 $\mathcal{R}^\alpha$ are depicted in blue and red polytope respectively. 86

5.11 Polyhedral partition with 13 regions, before and after perturbation of all the frontier vertices. 87

5.12 In the subplots, the polyhedral regions $\hat{\mathcal{R}}_i$ are presented with the vertex sensitivity and invariant-vertex sensitivity sets depicted in red and green color respectively. The dot and the $\times$ in the subplots are the vertex candidate and their new positions 92

5.13 The states trajectories for the polyhedral partition, outcome of the Algorithm 5.5.1 as show in Fig 5.12i, for the vertices that lie on the boundary of the polytope. 94

5.14 The perturbed polyhedral regions $\mathcal{P}_N(\hat{\mathcal{R}})$ for the quantization function with a random variable $\|\Delta\bar{v}_j\|_\infty \leq 0.2$ are outlined with blue lines. The polyhedral regions $\mathcal{P}_N(\mathcal{R})$ are given in red lines. 94

5.15 The states trajectories for the polyhedral partition, outcome of the Algorithm 5.5.1 as show in Figure 5.14, for the vertices that lie on the boundary of the polytope. 95

5.16 The contractive set $\beta\mathcal{R}$ is given by a polytope outlined in blue color and the vertex of interest is denoted by a black dot. The polytope outlined in red color is representing the set $\beta\lambda\mathcal{R}$. The green polytope is the vertex sensitivity region Ψ^v for the vertex given in black dot. . 96

5.17 Representation of polyhedral partitions with 11 regions in the $[x,\ \beta]$-space. 97

5.18 The projection of Figure 5.17 onto x-space, now the newly formed polyhedral partitions has 24 regions. 97

5.19 The approximated polyhedral partition $\mathcal{P}_N(\hat{\mathcal{R}})$ after displacing all the vertices in the set $\mathcal{V}(\mathcal{R})$. 98

5.20 The state trajectories for the vertices in the approximated polyhedral partition showing all the trajectories are λ-contractive. 99

5.21 The time-simulation presenting the evolution of the β parameters for all the state trajectories depicted in Figure 5.20. 99

6.1 Representation of ADMM algorithm 107

6.2 Scenario tree decomposition for robust horizon $N_r = 3$. 115

6.3 Computational time comparison between CPU and GPU 120

6.4 Simulation results of the states, input and random disturbance for the EMPC problem with PHA and the standard EMPC. The initial condition $x_0 = [40,\ -20]^T$ is chosen for the simulations. 121

List of Tables

3.1 Gain Margin for different prediction horizons 33

3.2 Information of the gain margin computation for different prediction horizons. 38

5.1 This table represents the vertex candidates for each iteration and their new position along with their Chebychev radius. The last column shows the indexes of the subset of regions that are impacted by the perturbation of the vertex 93

5.2 This table represents the vertex co-ordinates for each iteration and their new position. The second column shows the values of the β parameter associated to the vertex co-ordinates. 98

6.1 EMPC without time splitting 111

6.2 EMPC with time splitting 111

6.3 Computational time and the ADMM final iteration counter number for MTS, MTP and C-S for different prediction horizon 111

6.4 EMPC without time splitting 112

6.5 EMPC with time splitting for different ρ constants 113

6.6 Computational time and the ADMM final iteration counter number for MTS and C-S for different prediction horizon and for different ρ constants 113

6.7 EMPC with PHA 119

6.8 EMPC with PHA 120

Notation

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

A vector is noted $x \in \mathbb{R}^n, x = [x_1, \cdots, x_n]^T$.
For a given matrix $A \in \mathbb{R}^{n \times m}$, then $vec(A)$ represents the vector composed of the columns of the matrix A,

$$vec(A) = [A^T(\cdot, 1), \cdots, A^T(\cdot, m)]^T \in \mathbb{R}^m,$$

here $A(i, \cdot)$ denote the i^{th} row of matrix A and $A(\cdot, j)$ denotes the j^{th} column of matrix A. Also, if $n = m$ the matrix A is called square matrix. An identity matrix is represented by I_n where the subscript n denotes the dimension of that matrix.
$\mathbf{1}$ is a vector with all its components equal to 1.
I denotes an identity matrix and I_n means that $I \in \mathbb{R}^{n \times n}$.
For a given $N \in \mathbb{N}_{>0}$, $\mathcal{I}_N$ denotes the set of integers of indexes,

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

A *mapping function f:* $\mathcal{R}^n \to \mathbb{R}^m$ is said to be positively homogeneous of the first degree, if $f(ax) = af(x),\ \forall a \in \mathbb{R}_+$ and $\forall x \in \mathbb{R}^n$.

The unit simplex in $\mathbb{R}^L$ is defined as,

$$\mathcal{S}_L = \{x \in \mathbb{R}_+^L | \mathbf{1}^T x = 1\}.$$

For a given vector $x \in \mathbb{R}^n$, we denote the p-norm of the vector x by $||x||_p$. In the following we define the norms that are used in this thesis.
-$p = 1,\ ||x||_1 = \sum_{i=1}^n |x_i|$,
-$p = 2,\ ||x||_2 = \sqrt{x^T x}$,
-$p = \infty,\ ||x||_\infty = \max_{i \in \mathcal{I}_n} |x_i|$.

Chapter 1

Introduction

Contents

1.1 Motivation of the Thesis 4

1.2 Objectives 5

1.3 Publications 7

The optimal control problems (OCP) [Pon87, Bel13, Ber95], represent an important part of modern control theory, and have gained prominence in the vast field of control system research by means of several techniques and methodologies that emerged from the original problem formulations (finite-time optimal control, receding horizon optimal control, model predictive control) or the plethora of dynamical models to be considered in the optimal control design (nonlinear, hybrid, impulsive, constrained, etc). The common objective of the OCP's is to determine a sequence of optimal control inputs for a dynamical systems (linear or non-linear) while satisfying a set of physical constraints.

Model Predictive Control (MPC) [MRRS00, RM12], recognized as one of the most studied control law algorithm besides the conventional controllers, can be classified as an advanced optimal control algorithm which has the ability to handle constraints on inputs, states and outputs. Standard MPC or optimization-based MPC has been widely applied to chemical and petro-chemical industries due to their handling of multi-variable nature [CB97]. The MPC algorithm is based on receding horizon technique consisting in the resolution of a finite-time open-loop optimal control problem (often stated in Quadratic Programming terms for linear prediction models). The growing maturity of the stability, recursive feasibility and optimality of the MPC algorithms has been implemented across the spectrum of control applications ranging from aeronautical, electrical, automotive, etc [QB03]. Recently the MPC algorithm has also been adopted in the areas of quantitative finance and financial engineering for investment and stock portfolio risk reduction [HKD$^+$06, DO15]. Moreover, such applications are the evidence of the growing clout and prominence of the MPC in the modern control theory and control-related applications. In reference with the stability of the MPC problems, a vast amount of research has been conducted for ensuring the stability of the closed-loop dynamical systems. One of the notable

approaches in accordance with the stability of the MPC problems includes the terminal set and terminal cost function to the optimization problem. The inclusion of the terminal set ensures that the evolution of the state trajectories of the dynamical systems converge to the same set which is a subset of the state constraint set fulfilling the physical (hard) constraints placed on the manipulated variables (e.g., actuators, valves, breaking system, etc.,) [MRRS00]. The terminal set can be usually represented by convex regions around the equilibrium point and takes the form of polyhedron or ellipsoid. In general the terminal sets characterized by the polyhedral are the most commonly used as they lead to linear constraints in the related optimization problems. In particular, the constraints represented by polyhedral sets within the optimization problem with a quadratic cost function leads to a Quadratic Programming (QP) problem while the ellipsoidal sets and quadratic cost function lead to a Quadratic constrained Quadratic Programming (QCQP) problem [RM12]. Both classes fall within the convex optimization framework and have appealing structural properties (uniqueness, optimality conditions, etc).

Although there are many advantages to the standard MPC as comparison to the conventional controllers in terms of optimality and multi-variability, implementing an MPC algorithm to obtain the optimal control input on-line by solving for instance Quadratic Programming (QP) problems is time consuming. Particularly, when it comes to system with fast dynamics the real-time update of the measurements and the resolution of optimization problems need to provide solutions at very high sampling rates. This on-line computational complexity can be overcome by transforming the QP problem to a multi-parametric Quadratic Programming (mp-QP) problem [PGD07] and solving it off-line at the control design stage. This approach of solving the mp-QP problem off-line is also called Explicit MPC (EMPC), where the computation of the optimal control input is reduced to a simple evaluation of algebraic functions stored in a look-up table [AB09, BMDP02, SDDG00, OD04]. The resulting solution of such problems is usually a set of linear piecewise affine (PWA) functions defined over the system state-space partitions. EMPC control laws can be easily evaluated and implemented on-line for systems with extremely fast dynamics as long as the state-space models are of small dimensions. Recently, explicit solutions (mp-QP) have been widely recognized for their ability of separation of the computational complexity between off-line and on-line which is not the case with standard MPC. The references of these works [BMDP02, AB09] show how the mp-QP problems formulated for constrained linear discrete-time time-invariant systems for finite or infinite horizon optimal control problem are solved off-line. In particular, the polyhedral partition is conducted based on the critical region search satisfying the KKT (Karush-Kuhn-Tucker) optimality conditions [BMDP02], the critical regions are represented by the linear inequality constraints which can be termed as half-space representation or simply H-representation. From a different perspective, in the work [MJ12] it was shown how the mp-QP problem can be solved for the piecewise affine function defined over the state-space partition enumerated using a vertex representation. Sub-optimality of the mp-QP solution is presented in [BF02] by relaxing the KKT optimality conditions or by approximating the polyhedral partition in [SOH09]. Nevertheless, the solution set of these methods comprise state space partition where the sub polyhedral sets are characterized by half-space repre-

sentation.

From a purely on-line evaluation (of the optimal control law) point of view, the polyhedral partition represented by the half-space and vertex characterization can be compared in terms of on-line computational complexity and off-line data storage. Speaking about the on-line evaluation of the PWA control laws, an algorithm is employed to conduct a sequential search through the state-space regions for an initial state or measured state (in case state estimation is considered) [BMDP02]. Usually, this way of finding the regions that contain the given point sequentially is time consuming. The evaluation of PWA control procedure expressed in the paper [TJB03] indicates the construction of a binary search tree and it is shown that the binary search tree method takes less time and requires smaller storage capacity than that of the sequential search methods. Some noticeable advantages of the binary search tree method are (i) the computational time is logarithmic in the number of regions associated to its PWA functions, (ii) the search is also efficient in the event of overlapping regions and holes in the polyhedral partitions. Though explicit control law has favorable advantages over the standard MPC, it also comes up with high computational cost for higher order systems and/or for large prediction horizon.

Explicit control laws can be effectively implemented on micro-controller circuits or on Field-Programmable Gate Array (FPGA) for a wide-range of control applications [JJST07, IK15]. The cost of the micro-controller circuits or FPGA depend on the size of the memory unit or Arithmetic and Logic Unit (ALU) associated with it. For storing a relatively small state-space partitions and the PWA control laws associated with each partition, the cost of the micro-controller required might be reasonable. But for storing considerably large state-space partitions and PWA functions, a truncation operation (interpreted also as a quantization process on the control law parameters) should be performed on the representation of the state-space regions and the associated PWA control laws in order to reduce the hardware costs [KZC15]. The post-quantization state-space regions and post-quantization affine control laws associated with the regions adversely impact the accuracy of the control laws. Moreover, the modified regions might lose the non-overlapping property of the state-space partition. Such a phenomenon will lead to non-uniqueness of the feedback control, discontinuities and the associated loss in stability, performance and real-time certifications.

The uncertainties in model parameters and the influence of disturbances make the plant to differ from the mathematical model. Since the mathematical model serves the purpose of prediction, the prediction will be less accurate as compared to the real plant. Since this millennium, much research has been devoted to tackle uncertainties in dynamic system for standard MPC [BM99, GKMS14, MRFA06]. The Robust MPC problem formulation for the linear discrete-time time-invariant systems affected by structured feedback uncertainty is presented in [KBM96]. The well known research works for the Robust MPC (RMPC) problems for linear dynamical systems with polytopic uncertainty / additive disturbance noise / bounded disturbances / stochastic disturbances are provided in the works of Robust Min-Max ap-

proach [SM98, KM03], robust tube based MPC [LCRM04, MKF11], Scenario-based or Multi-stage MPC [BB09, Luc14]. Although these Robust MPC can tackle the uncertainty to the accepted levels in terms of stability and optimality, particularly the later two RMPC's problem demand great computational efforts for real-time implementation. With regards to the impact of uncertainty in the explicit MPC, very few research has been conducted.

From the analysis point of view, it is important to commensurate the capacity of the control law to cope with disturbances, neglected dynamics or uncertain parameters. These characteristics are denoted in control theory as robustness of the controller. In the context of robustness analysis of PWA controllers, very few contributions have been made. Some noticeable recent works include an analysis procedure proposed in [ONB+13] and [NOBRA14], which handles the robustness/fragility of the positive invariance for the dynamics affected by uncertain parameters. On the other hand, the robustness analysis can be connected to the works on the robustification of the explicit controllers. The reference [RAO13] for example shows how to improve the robustness of the controller by retuning. It is worth to be mentioned that the analysis of a nominal PWA control and its retuning is essentially different approach from a robust control design. It is known that a robust PWA control as for example Robust Explicit Model Predictive Control (REMPC) synthesis [PFKP09] can account for uncertainties based on dynamic programming approach but the associated computational complexity is exponential with respect to the nominal case. The same thing can be said about the robust explicit model predictive control with contractive set based on variable-structure control law for linear polytopic uncertain system as presented in [YBHJ03].

1.1 Motivation of the Thesis

In the recent work of [ONB+13, NOBRA14], the authors proposed a generic geometrical approach to derive the robustness margin for PWA control law with respect to the positive invariance properties of the nominal system. This approach is adapted to the particularity of the piecewise-affine characteristics of the closed loop dynamics and differs from a classical robustness analysis. The ultimate objective of this geometrical approach for computing the robustness margin for explicit MPC is the description of a polyhedral set in the parameter space. It is worth to be mentioned that this analysis procedure proposed in [ONB+13, NOBRA14] handles the robustness/fragility of the positive invariance but does not extend the analysis to the convergence (asymptotic stability properties). Also, the mentioned works do not tackle the problems concerning the input noise or neglected dynamics. The analysis of the robustness of the controller for the system affected by neglected dynamics are very important in the context of model mismatch or inaccurate system parameters.

Analyzing the PWA characteristics with respect to perturbations on vertices of the system state partitions is considered vital for many purposes. It could help the online PWA controller to tackle considerable perturbation on the system states. The most resourceful purpose is to perform a "quantization" operation on the state-space

partition given by vertex representation. The complexity of the explicit solutions increases with the inclusion of the constraints imposed on the dynamical systems. In such a case, performing a quantization operation on the vertices of the polyhedral partition will effectively help the micro-controller to bring down the memory concerning the hardware requirements.

It is a widely accepted phenomenon that explicit solutions still suffer from the computational complexity while handling a dynamical system with somewhat a low-medium scale system (a system with 5 states will be considered to fall within this category) and for an acceptable prediction horizon (for instance finite horizon of length 10).

Moreover, the explicit MPC has gained reputation for its simple evaluation of control laws, researchers/scholars have been more interested in bringing down the complexity of solving the mp-QP problem off-line. It is a fact that bringing down the complexity of the mp-QP problem will result in less requirement of the data structure of memory unit. There is no notable research to be mentioned regarding the on-line evaluation of the PWA functions for large data.

These facts triggered the present research work which aims to extend the constructive analysis procedure based on the prevalent problems that exist in the research domain of the explicit controller.

1.2 Objectives

In the current section, we would like to list the goals that we aim to address in this thesis work and the way they structure the document chapter-wise. In this thesis, we consider the inherent robustness properties of explicit predictive control described as piecewise affine (PWA) control laws for a class of linear discrete-time systems.

Chapter 3:

1. First, we aim to develop a numerical method to compute a **gain margin** for a discrete-time system stabilized by a continuous PWA affine dynamics with respect to the invariance property. The desired gain margin is a set which characterize a variations of system gains preserving the invariant characteristics of the controller.

2. Second, we aim to establish a similar type of analysis for the linear dynamic system affected by **first order neglected dynamics**. The robustness margin of the controller against the neglected dynamics is defined by a set which characterizes a range of parameters of a first order neglected dynamics in the model preserving the invariance of the controller.

Chapter 4:

1. A methodological objective in relationship with MPC design is to replace the terminal set and terminal cost function with the controlled λ-contractive sets for constructing explicit control laws for discrete time systems assuring a λ-contractivity of the trajectories. In this framework we aim to establish a parametrization expressing the dependence on the state vector and use this as an additional factor for constructing the PWA control law.

2. Secondly, the initial λ-contractive controller can be seen as a possible decision variable to extend the explicit robustness margin to contractive framework. In order to exploit this idea, the use of a polytopic description of the plant will allow to express the robustness margin as a subset of parameters for which the λ-contractivity is maintained.

3. Third goal is to compute a gain margin in this framework for a discrete-time system controlled by a contractive PWA control law. The desired gain margin for the contractive controller has to preserve the λ-contractive properties of the controller.

4. Fourth, we aim to extend the first order neglected dynamics analysis presented in Chapter 3 for the contractive controller.

Chapter 5:

In this chapter, we consider perturbation on the vertices of the polyhedral partitions obtained from solving a mp-QP problem for a finite time OCP considering a linear discrete-time time-invariant system. Starting from a problem formulation concerning the perturbation on the vertices of the polyhedral partition, we fix the following objectives:

1. First, we aim to derive a polyhedral set called vertex sensitivity region that characterizes all admissible perturbations on the associated vertex. The vertex sensitivity region has to ensure the **non-overlapping property of the PWA controller** in the event of a vertex perturbation.

2. Second, the invariant property of the controller when perturbing vertices that lie on the frontier of the feasible domain needs to be analyzed. The vertices on the feasible domain characterize the invariant set and perturbing those vertices have an impact concerning the invariance and the convex properties of the PWA controller.

3. Third, we fix as objective to construct the vertex-invariant region, subset of the vertex sensitivity region, that characterize all admissible perturbation on the interior vertices of the polyhedral partition guaranteeing the invariant characteristics of the controller. The validation of the control inputs constraints needs to be considered at this stage.

4. Fourth, we aim to analyze the perturbation on the vertices for the contractive PWA controller ensuring jointly the non-overlapping and contractive properties of the controller.

Chapter 6:

In this chapter, we enable explicit solutions for longer prediction horizons and we also treat linear dynamics with probabilistic uncertainties controlled using explicit control laws. The objectives of this chapter are:

1. To explore a time-splitting operation carried out through a finite-time prediction horizon. The mp-QP sub-problems resulting from the time-splitting operator are grouped into three sub-problems. The Alternative Direction Method of Multipliers (ADMM) algorithm will be investigated to obtain the consensus constraints occurred during the decoupling and for ensuring the convex convergence.

2. The on-line evaluation of the control law based on the ADMM algorithm needs to be considered. This submodule is aimed at comparing the computational results between the sequential and the parallel evaluations of the ADMM algorithm.

3. On a different perspective, we aim to treat the bounded probabilistic disturbance for the linear discrete-time systems. Here, a scenario tree problem for different disturbance realizations can be constructed. The scenario tree mp-QP problem is known to be hard to solve. In order to address the complexity arising from the scenario tree, we propose to decompose the scenario tree into sub-problems. Finally the Progressive Hedging Algorithm (PHA) will be considered for the satisfaction of the non-anticipativity constraints developed from the scenario decomposition, (the PHA algorithm assures convex convergence for the sub-problems).

4. In this framework, the ultimate goal is to compare the computational complexity of the parallel evaluation of the PHA algorithm for Central Processing Unit (CPU) and Graphical Processing Unit (GPU).

1.3 Publications

Accepted publications:

- R. Koduri, P. Rodriguez-Ayerbe and S. Olaru, *Robustness margin for piecewise affine explicit control law*, IEEE 55th Conference on Decision and Control (CDC), 2016. url: `http://ieeexplore.ieee.org/document/7798610/`

- R. Koduri, P. Rodriguez-Ayerbe, S. Olaru and M. Hovd, *Explicit robustness margin for contractive piecewise affine control laws*, 20th International Conference on System Theory, Control and Computing (ICSTCC), 2016. url: `http://ieeexplore.ieee.org/document/7790767/`

- R. Koduri, S. Olaru and P. Rodriguez-Ayerbe, *Sensitivity of Piecewise Control Laws with Respect to Perturbation of the State-Space Partition*, 21st International Conference on Control Systems and Computer Science (CSCS), 2017. url: `http://ieeexplore.ieee.org/document/7968537/`

- R. Koduri, S. Olaru and P. Rodriguez-Ayerbe, *On the Precision in Polyhedral Partition Representation and the Fragility of PWA Control*, 56th IEEE Conference on Decision and Control, 2017.

In preparation:

- R. Koduri, H. Shukla and C. Jones, *PWA function evaluations on CPU and GPU forADMM and PHA* (Conference).

- R. Koduri, P. Rodriguez-Ayerbe, S. Olaru and M. Hovd, *Robustness margins for piecewise affine and contractive piecewise affine explicit control laws* (Journal).

- R. Koduri, S. Olaru and P. Rodriguez-Ayerbe, *Precision in polyhedral partition representation and the fragility of PWA control* (Journal).

Presentation to symposium:

- R. Koduri, S. Olaru and P. Rodriguez-Ayerbe, *Explicit robustness margin for contractive piecewise affine control laws*, 4th European Conference on Computational Optimization, EUCCO 2016, Leuven.

Chapter 2

Background Theory

Contents

2.1 Convex Sets and Convex functions 10

2.2 Set theoretic definitions 11

2.3 Polyhedra and Polytope 11

2.4 Linear Systems 13

2.5 Set Invariance 14

2.6 Contractive Sets 15

2.7 Model Predictive Control 15

2.7.1 Mathematical Formulation of MPC 16

2.8 Explicit Model Predictive Control 18

2.8.1 Real-time implementation of EMPC solutions: 21

2.9 Tools for multi-parametric programming 22

2.10 Conclusion 22

The purpose of this chapter is to present some of the fundamental definitions, concepts and notations which will be used throughout the thesis. The convex sets, convex optimization and their applications in the constrained control theory are well explained and presented in [Lei81, BV04]. The commonly used set theoretic operators in this thesis are defined as well as their applications ranging from constructing the control law to analyzing the robustness of the control law. Next, the definitions of the positively invariant sets and contractive sets subject to constraints are presented. Linear Model Predictive Control (LMPC) and its advantages over the classical linear controllers is briefly discussed along with its structural properties. The construction of the MPC with the terminal set and terminal cost function is summarized in this chapter. Finally, the explicit MPC is introduced with an example and the implementation of explicit controllers on real-time application is also briefly explained.

2.1 Convex Sets and Convex functions

A set $\mathcal{P}$ is called a convex set if for any two points $x, y \in \mathcal{P}$, their convex combination is a point in $\mathcal{P}$. Formally, the line segment between these points can be expressed as a subset of $\mathcal{P}$:

$$\{\gamma x + (1-\gamma)y | x, y \in \mathcal{P} \text{ and } \forall \gamma \in [0,\ 1]\} \subset \mathcal{P}. \tag{2.1}$$

Given a finite set of points $x_1, \cdots, x_d$ in a convex set $\mathcal{P}$, then the set $\mathcal{P}$ must contain any convex combination, the convex combination of those points can be expressed as:

$$x = \sum_{i=1}^{d} \gamma_i x_i \text{ s.t } \forall \gamma_i \geq 0 \text{ and } \sum_{i=1}^{d} \gamma_i = 1. \tag{2.2}$$

The term $\sum_{i=1}^{d} \gamma_i x_i$ in (2.2) is called a *convex combination*.

Let $x_1, \cdots, x_d$ be an arbitrary set of points, then the *convex hull* of these finite points is the set obtained by taking all the possible convex combinations,

$$conv\{x_1, \cdots, x_d\} = \Big\{ \sum_{i=1}^{d} \gamma_i x_i \text{ s.t } \forall \gamma_i \geq 0 \text{ and } \sum_{i=1}^{d} \gamma_i = 1 \Big\}. \tag{2.3}$$

Here, *conv* denotes the convex hull.

Definition 2.1.1 *A function $f : \mathcal{P} \in \mathbb{R}^n \to \mathbb{R}$ defined over a convex set $\mathcal{P}$ is said to be a convex function if for any $x, y \in \mathcal{P}$ and $\gamma \in [0, 1]$, then*

$$f(\gamma x + (1-\gamma)y) \leq \gamma f(x) + (1-\gamma) f(y). \tag{2.4}$$

Definition 2.1.2 *A function $f : \mathcal{P} \in \mathbb{R}^n \to \mathbb{R}$ is said to be a strictly convex function if for any $x, y \in \mathcal{P}$ and $\gamma \in [0, 1]$, then*

$$f(\gamma x + (1-\gamma)y) < \gamma f(x) + (1-\gamma) f(y). \tag{2.5}$$

Definition 2.1.3 *Let $\mathcal{P}$ be a symmetric convex set in $\mathbb{R}^n$, then a function $\mathcal{M}_{\mathcal{P}}(x)$ is called a Minkowski functional of $\mathcal{P}$, if*

$$\mathcal{M}_{\mathcal{P}}(x) = inf\{\gamma \in \mathbb{R}_+ |\ x \in \gamma \mathcal{P}\}. \tag{2.6}$$

Lemma 2.1.1
Let $\mathcal{P}$ be a convex set containing 0 as an interior point. Then the Minkowski functional $\mathcal{M}_{\mathcal{P}}$ of $\mathcal{P}$ satisfies [Lue97]:
- $\mathcal{M}_{\mathcal{P}}$ is continuous,
- $\mathcal{M}_{\mathcal{P}}$ is piecewise linear [1]*,*
- $\infty > \mathcal{M}_{\mathcal{P}}(x) \geq 0, \forall x \in \mathcal{X}$,
- $\mathcal{M}_{\mathcal{P}}(\alpha x) = \alpha \mathcal{M}_{\mathcal{P}}(x), for\ \alpha > 0$,
- $\mathcal{M}_{\mathcal{P}}(x_1 + x_2) \leq \mathcal{M}_{\mathcal{P}}(x_1) + \mathcal{M}_{\mathcal{P}}(x_2)$.

Note:
i) Let $\mathcal{R}_i, i \in \mathcal{I}_d$ be a finite collection of convex sets, then their intersection $\mathcal{R} = \cap_{i \in \mathcal{I}_d} \mathcal{R}_i$ is also convex.
ii) Let $\mathcal{R}$ be a convex set, then $\gamma \mathcal{R}$ is also convex for all $\gamma \in \mathbb{R}$.

[1] A piecewise linear function is a real-valued function defined on the real numbers, whose graph is composed of straight line sections.

2.2 Set theoretic definitions

Definition 2.2.1 *Given two sets $\mathcal{P} \in \mathbb{R}^n$ and $\mathcal{Q} \in \mathbb{R}^n$,*
- the intersection, denoted by $\mathcal{P} \cap \mathcal{Q}$, of $\mathcal{P}$ and $\mathcal{Q}$ is defined as

$$\mathcal{P} \cap \mathcal{Q} = \{x \in \mathbb{R}^n | x \in \mathcal{P} \text{ and } x \in \mathcal{Q}\}.$$

– the union, denoted by $\mathcal{P} \cup \mathcal{Q}$, of $\mathcal{P}$ and $\mathcal{Q}$ is defined as

$$\mathcal{P} \cup \mathcal{Q} = \{x \in \mathbb{R}^n | x \in \mathcal{P} \text{ or } x \in \mathcal{Q}\}.$$

– the set difference, denoted by $\mathcal{P} \setminus \mathcal{Q}$, of $\mathcal{P}$ and $\mathcal{Q}$ is defined as

$$\mathcal{P} \setminus \mathcal{Q} = \{x \in \mathcal{P} | x \notin \mathcal{Q}\}.$$

Definition 2.2.2 *Given two sets $\mathcal{P} \in \mathbb{R}^n$ and $\mathcal{Q} \in \mathbb{R}^n$, the Minkowski sum, denoted by $\mathcal{P} \oplus \mathcal{Q}$, of $\mathcal{P}$ and $\mathcal{Q}$ is defined as the set*

$$\mathcal{P} \oplus \mathcal{Q} = \{p + q |\ p \in \mathcal{P},\ q \in \mathcal{Q}\}.$$

Definition 2.2.3 *Given two sets $\mathcal{P} \in \mathbb{R}^n$ and $\mathcal{Q} \in \mathbb{R}^n$, the Pontryagin Difference, denoted by $\mathcal{P} \ominus \mathcal{Q}$, of $\mathcal{P}$ and $\mathcal{Q}$ is defined as the set*

$$\mathcal{P} \ominus \mathcal{Q} = \{x \in \mathbb{R}^n |\ x + \psi \in \mathcal{P},\ \forall \psi \in \mathcal{Q}\}.$$

Definition 2.2.4 *Given a polyhedron $\mathcal{P}$ the orthogonal projection mapping of a set $\mathcal{P} \subset \mathbb{R}^{c1}$ onto a subspace $\mathbb{R}^{c1} \subseteq \mathbb{R}^{c2}$ for $c1 > c2$ is defined as*

$$Proj_{\mathbb{R}^{c2}}\ \mathcal{P} = \{x \in \mathbb{R}^{c2} | \exists y \in \mathbb{R}^{c1-c2},\ s.t.\ \begin{bmatrix} x \\ 0 \end{bmatrix} + \begin{bmatrix} 0_{c2} \\ y \end{bmatrix} \in \mathcal{P}\}. \tag{2.7}$$

here, the projection mapping will be considered to operate on the first $c2$ coordinates of $\mathbb{R}^{c1}$.

Note:
i) The intersection of two or more convex sets is also convex.
ii) The set difference between/among two or more convex sets is not convex.
iii) The Minkowski sum of two or more convex sets is also convex.
iv) The Pontryagin Difference between/among two or more convex sets is also convex.
iv) The orthogonal projection of a convex set onto a subspace is also convex.

2.3 Polyhedra and Polytope

Definition 2.3.1 *A hyperplane is defined as the set $\{x \in \mathbb{R}^n |\ c^T x = \gamma\ s.t\ \gamma \in \mathbb{R}, c \in \mathbb{R}^n\}$ and is the set of all points in $\mathbb{R}^n$ that divides $\mathbb{R}^n$ into two half-spaces.*

Definition 2.3.2 *A half-space is given by the set* $\{x \in \mathbb{R}^n |\ c^T x \leq \gamma\ s.t\ \gamma \in \mathbb{R}, c \in \mathbb{R}^n\}$ *which characterizes all the points lying on one side of the hyperplane* $\{x \in \mathbb{R}^n |\ c^T x = \gamma\}$.

Definition 2.3.3 *A convex polyhedron is the intersection of a finite number of open or closed half-spaces.*

Such a definition is called *half-space representation* or simply *H-representation* of the polyhedron.

Consider a polyhedron $\mathcal{P}$ whose closed half-spaces can be written as a system of linear inequalities,

$$\mathcal{P} = \{x \in \mathbb{R}^n | Ax \leq b\}. \tag{2.8}$$

A bounded polyhedron can also be defined as a convex hull of finite set of points $x_1, \cdots, x_d$

$$\mathcal{P} = conv\{x_1, x2, \cdots, x_d\} \tag{2.9}$$

and such a representation is called *vertex representation* or *V-representation.*

Definition 2.3.4 *A closed and bounded polyhedron is called a polytope.*

Definition 2.3.5 *A point* x *in a polyhedron* $\mathcal{P}$ *is called an extreme point of* $\mathcal{P}$ *if it can be represented as strict* $(0 < \gamma < 1)$ *convex combination of two distant points in* $\mathcal{P}$.

Definition 2.3.6 *A set* $\mathcal{P} \subset \mathbb{R}^n$ *is called a proper* $\mathcal{C}$*-set if it is convex, closed, compact and contains the origin in its interior.*

Definition 2.3.7 *A set* $\mathcal{P} \in \mathbb{R}^n$ *is called a compact set if and only if the set* $\mathcal{P}$ *is closed and bounded.*

Notations: For a given set $\mathcal{P}$,
$-int(\mathcal{P})$ denotes the interior of $\mathcal{P}$.
$-bd(\mathcal{P})$ denotes the set of points which lies on the boundary of the set $\mathcal{P}$.
$-ext(\mathcal{P})$ denotes the set of extreme points of the set $\mathcal{P}$.
$-Card(\mathcal{P})$ denotes the cardinal number of $\mathcal{P}$.
$-\mathcal{V}(\mathcal{P})$ denotes the set of its vertices.

Proposition 2.3.1
Let $\mathcal{P}$ *be a compact set, then the following statements hold:*
- A closed subset of a compact set is also a compact set.
- A finite union of compact sets is compact.
- A finite intersection of compact sets is also compact.

2.4 Linear Systems

In this section, we briefly describe the different classes of linear systems which are very commonly used for practical applications.
Consider a linear discrete-time time-invariant systems with no input. Such a dynamical system is commonly denoted as *autonomous system*,

$$x_{k+1} = Ax_k \tag{2.10a}$$
$$y_k = Cx_k \tag{2.10b}$$

Next, consider a linear discrete-time time-invariant system with input variables,

$$x_{k+1} = Ax_k + Bu_k \tag{2.11a}$$
$$y_k = Cx_k \tag{2.11b}$$

and the counterpart affected by *additive disturbance*,

$$x_{k+1} = Ax_k + Bu_k + w_k \tag{2.12a}$$
$$y_k = Cx_k \tag{2.12b}$$

Similarly, the state-space dynamical systems subject to *probabilistic disturbances* will be denoted as follows,

$$x_{k+1} = Ax_k + Bu_k + Ed_k \tag{2.13a}$$
$$y_k = Cx_k \tag{2.13b}$$

this case is mainly employed in Chapter 6.

In Eq (2.10)-(2.13), $x_k \in \mathbb{R}^n$ denotes the system state at time k, $u_k \in \mathbb{R}^m$ denotes the input variables, $y_k \in \mathbb{R}^{n_y}$ denotes the output variables, $w_k \in \mathbb{R}^{n_w}$ denotes the additive disturbances and $d_k \in \mathbb{R}^{n_d}$ denotes the probabilistic disturbance variables. The matrices $A \in \mathbb{R}^{n\times n}, B \in \mathbb{R}^{n\times m}, C \in \mathbb{R}^{n_y\times n}$ and $E \in \mathbb{R}^{n\times n_d}$ are given with appropriate dimensions.

The states, control variables, additive disturbances and probabilistic disturbances are subject to constraints :

$$x_k \in \mathcal{X} \subset \mathbb{R}^n, u_k \in \mathcal{U} \subset \mathbb{R}^m, w_k \in \mathcal{W} \subset \mathbb{R}^{n_w}, d_k \in \mathcal{D} \subset \mathbb{R}^{n_d} \tag{2.14}$$

The state constraint set $\mathcal{X}$, input constraint set $\mathcal{U}$, bounded additive disturbances set $\mathcal{W}$ and the probabilistic disturbances set $\mathcal{D}$ are proper $\mathcal{C}$-sets.

A special class of linear discrete-time system subject to parametric uncertainty, input constraints and state constraints will be of interest in Chapter 4. It is expressed as follows:

$$x_{k+1} = Ax_k + Bu_k, \tag{2.15}$$

where $[A\ B]$ belongs to a *polytopic uncertainty set* Ω

$$\Omega = conv\{[A_1\ B_1], \cdots, [A_L\ B_L]\}. \tag{2.16}$$

The polytopic nature of the uncertainty means that any $[A\ B] \in \Omega$ can be expressed as:

$$[A\ B] = \sum_{i=1}^{L} \gamma_i [A_i\ B_i], \tag{2.17}$$

for $\gamma_i \geq 0, \forall i \in \mathcal{I}_L$ and $\sum_{i=1}^{L} \gamma_i = 1$. The system states and control variables are subject to constraints:

$$x_k \in \mathcal{X} \subset \mathbb{R}^{n_x},\ u_k \in \mathcal{U} \subset \mathbb{R}^{n_m}.$$

The state and control constraints sets $\mathcal{X}$,$\mathcal{U}$ are proper $\mathcal{C}$-sets.

2.5 Set Invariance

Set invariance in constrained control problem has been extensively studied and developed in the past few decades [Bit88b, VB89, Bla99]. Set invariance in control theory is widely used for restricting the state dynamics in a subset (of a state constraint set) for any admissible[2] control input. Here, the admissible control input indicates the control law that satisfies the relevant control constraints while keeping the state trajectories in a precomputed subset. Remarkable results on admissible set and feedback control law for discrete-time linear system subject to bounded input and state constraints were presented in early 1980's [GC86, Bit88a, BV95].

The computation of the invariant set has been further extended to uncertain dynamical system (2.17) with constrained control input [GG85]. The author of this paper [San94] presented a positively invariant sets for discrete-time system with disturbance of the form (2.13). These days, the inclusion of the invariant set for a constrained control (generally based on LP or QP) for any classes of dynamical system (linear or non-linear) with or without disturbances is very common. The purpose of incorporating the invariant set in the control synthesis is to keep the future trajectories inside the subset if the initial state belongs to that set.

In the following, we define the invariant sets for different classes of linear dynamical systems:

Definition 2.5.1 *A set $\mathcal{P} \subset \mathcal{X}$ is called positively invariant with respect to the system (2.10) if for all $x_k \in \mathcal{P}$, it follows $Ax_k \in \mathcal{P}$, $\forall k \in \mathbb{N}_{\geq 0}$.*

Definition 2.5.2 *A set $\mathcal{P} \subset \mathcal{X}$ is called controlled positively invariant with respect to the system* (2.11) *in closed loop if there exists a feedback control law u_k^* such that for all $x_k \in \mathcal{P}$ then $Ax_k + Bu_k^* \in \mathcal{P}$, $\forall k \in \mathbb{N}_{\geq 0}$.*

Definition 2.5.3 *A set $\mathcal{P} \subset \mathcal{X}$ is called robustly controlled positively invariant with respect to the system (2.12) in closed loop if there exists a feedback control law u_k^* such that for all $x_k \in \mathcal{P}$ and $w_k \in \mathcal{W}$, the solution is such that $Ax_k + Bu_k^* + w_k \in \mathcal{P}$, $\forall k \in \mathbb{N}_{\geq 0}$.*

[2]admissible control input indicates the control law satisfies the relevant control constraints besides keeping the state trajectories in a precomputed subset.

2.6 Contractive Sets

The inclusion of the invariant set in control design does not necessarily ensure the convergence of the future trajectories to the origin. It should also be noted that there are systems, which tends to exhibit oscillating dynamics or limit cycles. For such systems, the addition of invariant set keeps the future trajectories inside the feasible set but the convergence of the state trajectories to the origin is not guaranteed by the simple existence of invariant set. These drawbacks of the invariant set can be overcome by considering a particular kind of invariant sets called *contractive sets*.

An interesting result on the construction of contractive sets for a discrete-time linear system was presented in [DH99]. The results are based on iterative procedure with respect to the bounded state and input constraint sets. This iterative procedure to compute the contractive set is further investigated with the results published in [HOB14]. In this work, the construction of the set is based on the eigenvectors of the system matrix A. Such contractive sets are used in this thesis to analyze the robustness margin for uncertain system which can be observed in Chapter 4.

In the following, we will extensively use the contractive set for different classes of linear system based on the following definitions.

Definition 2.6.1 *A $\mathcal{C}$-set $\mathcal{P} \subseteq \mathcal{X}$ is called λ-contractive for a discrete-time system of the form* (2.10) *with contraction factor $\lambda \in$ [0,1) if $\forall x_k \in \mathcal{P}$ then $Ax_k \in \lambda\mathcal{P}$, $\forall k \in \mathbb{N}_{\geq 0}$.*

Definition 2.6.2 *A $\mathcal{C}$-set $\mathcal{P} \subseteq \mathcal{X}$ is called controlled λ-contractive with respect to the system* (2.11) *and a contraction factor $\lambda \in$ [0,1) if and only if there exists an admissible control law u_k^*, such that $\forall x_k \in \alpha P$ then $Ax_k + Bu_k^* \in \lambda\alpha\mathcal{P}$, $\forall \alpha \in [0, 1]$, $\forall k \in \mathbb{N}_{\geq 0}$.*

2.7 Model Predictive Control

Model predictive control (MPC) also referred to as receding horizon control is the most popular control algorithm technique besides the conventional controllers such as proportional integral derivative controllers (PID controllers). But unlike the PID controllers which are often used to stabilize single-input single-output (SISO) system, MPC has the ability to deal with multi-input multi-output (MIMO) dynamical systems. The constraints on the system states, inputs and outputs are naturally considered at the design stage on the MPC compared to the PID controllers. These advantages of the MPC over PID controller makes it widely acceptable and it has been adopted in a wide range of fields including automation, electrical and chemical industries. The objective of the MPC is to compute an optimal control sequence over its horizon by solving an optimization problem for each time step. For linear prediction models, the optimization problem consists of a Linear Programming (LP) or Quadratic Programming (QP) problem incorporated with the future dynamics

of the system states, which are predicted with the help of a system model based on the current state measurements.

Functioning of Model Predictive Control:

In MPC, a popular term called *prediction horizon* is the horizon over which the future states of the system are predicted using a mathematical model. The working principle of the MPC is described below,

1. Starting from the current state measurement, the future dynamics of the system are predicted over the prediction horizon.
2. Define an optimization problem with the future system dynamics and taking the constraints on the states and inputs variables into consideration.
3. The optimization problem is solved for the optimal control sequence over the horizon.
4. Only the first element in the control sequence is applied to the real plant and the remaining elements in the control sequence can be discarded.
5. Recede the horizon (prediction window) to the next time step all by restarting the procedure from step 1.

2.7.1 Mathematical Formulation of MPC

Let us consider a linear discrete-time system given by,

$$x_{k+1} = Ax_k + Bu_k \tag{2.18a}$$
$$y_k = Cx_k \tag{2.18b}$$

The system state and input variables are subject to constraints,

$$\mathcal{X} = \{x : H_x x \leq h_x, H_x \in \mathbb{R}^{p_x \times n}, h_x \in \mathbb{R}^{p_x}\},\ and \tag{2.19a}$$
$$\mathcal{U} = \{u : H_u u \leq h_u, H_u \in \mathbb{R}^{p_u \times m}, h_u \in \mathbb{R}^{p_u}\}. \tag{2.19b}$$

where the matrices H_x, H_u and the vectors h_x, h_u are assumed to be constant, and $\mathcal{X} \subset \mathbb{R}^n$ and $\mathcal{U} \subset \mathbb{R}^m$. Based on these definitions it becomes clear that the state and input constraints sets $\mathcal{X}$ and $\mathcal{U}$ are polyhedral sets.
Consider now the quadratic cost function for the standard MPC,

$$J(x_{k|k}, U^*) = V(x_{k+N_p|k}) + \sum_{i=0}^{N_p-1} L(x_{k+i|k}, u_{k+i|k}) \tag{2.20a}$$
$$s.t. \quad x_{k+i+1|k} = Ax_{k+i|k} + Bu_{k+i|k},\ \ i = 0, \cdots, N_p - 1 \tag{2.20b}$$
$$x_{k|k} = x_k \tag{2.20c}$$
$$x_{k+N_p|k} \in \mathcal{X}_f \tag{2.20d}$$
$$u_{k+i|k} \in \mathcal{U},\ i = 0, \cdots, N_p - 1 \tag{2.20e}$$
$$x_{k+i|k} \in \mathcal{X},\ i = 1, \cdots, N_p \tag{2.20f}$$

Here N_p is the length of the prediction horizon. The variables x_{k+i}, u_{k+i} denote the predicted system state and the control input to be optimized respectively at time $k+i$. The state sequence $x_{k+1|k}, x_{k+2|k}, \cdots, x_{k+N_p|k}$ is predicted using the model given in (2.20b) based on the state measurements available at time k. Also, the optimal control sequence obtained from the open-loop optimal control problem (2.20a) can be written as:

$$U^* = [u^{*T}_{k|k}, u^{*T}_{k+1|k}, \cdots, u^{*T}_{k+N_p-1|k}]^T \in \mathbb{R}^m. \tag{2.21}$$

The stage cost function is denoted by $L(x_{k+i|k}, u_{k+i|k})$ and it can take for example a weighted 2-norm form (often associated with the energy of the respective signal),

$$L(x_{k+i|k}, u_{k+i|k}) = x^T_{k+i|k} Q x_{k+i|k} \ + \ u^T_{k+i|k} R u_{k+i|k}.$$

In addition, the terminal cost function $V(x_{k+N_p|k})$ taking a similar 2-norm form is expressed as:

$$V(x_{k+N_p|k}) = x^T_{k+N_p|k} P x_{k+N_p|k}.$$

Alternatively, the stage and terminal cost functions taking an $1/2/\infty$-norm form is usually expressed in the form:

$$L(x_{k+i|k}, u_{k+i|k}) = ||Q x_{k+i|k}||_p + ||R u_{k+i|k}||_p,$$
$$V(x_{k+N_p|k}) = ||P x_{k+N_p|k}||_p,$$

here p=$1/2/\infty$.

$J(x_k, u^*_k)$ is the resulting optimal cost function. The weight matrices $Q = Q^T \geq 0$ and $P = P^T \geq 0$ are positive semi-definite matrices, and $R = R^T > 0$ is a positive definite matrix which defines the performance index of the optimization problem. The stability of the nominal MPC problem is ensured by placing a terminal constraint set $\mathcal{X}_f$ on the final state x_{N_p} given in (2.20d) and by adding the terminal cost function $V(x_{k+N_p|k})$ to the optimization problem [MRRS00].

The terminal constraint set $\mathcal{X}_f$ and terminal cost function $V(x_{k+N_p|k})$ in the optimization formulation (2.20) are used to ensure the recursive feasibility and subsequently the stability of the nominal closed-loop system $x(k+1) = Ax(k) + Bu^*_k$ over the feasible domain of the optimization problem (2.20). One of the practical constructions of terminal constraint is a positively invariant set $\mathcal{X}_f$ and this is achieved by adding the constraint $x_{N_p} \in \mathcal{X}_f$ to the problem (2.20). The terminal weight P is chosen as the solution of the Algebraic Riccati equation given with the system matrix pair (A, B) and the weights (R, Q),

$$P = (A + BK_{LQR})^T P (A + BK_{LQR}) + K^T_{LQR} R K_{LQR} + Q, \tag{2.22}$$

here K_{LQR} is the Linear Quadratic Regulator (LQR) gain,

$$K_{LQR} = -(R + B^T P B)^{-}1 B^T P A. \tag{2.23}$$

In summary, starting from the measured state at time k, MPC solves the following optimization problem,

$$U^* = \underset{u}{argmin} J(x_{k|k}, U^*)$$

subject to (2.20b)-(2.20f). From the optimal control sequence (2.21), only the first control $u^*_{k|k}$ is applied to the dynamical system or real plant. The procedure is repeated with the next measured state.

2.8 Explicit Model Predictive Control

In recent years, Explicit MPC [AB09, BMDP02, TJB03, RAO13, GJ12, OD04, OD05, SGDD03] has gained attention for its on-line functioning as evaluation of simple algebraic expressions. The central idea of EMPC is to transform the standard MPC optimization (2.20) to a multi-parameter programming problem. Such MPC problem posses a piecewise affine structure. In order to define the EMPC problem as multi-parametric Quadratic programming (mp-QP) problem, we try to transform the standard optimization problem, for the nominal linear time-invariant system

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

subject to the state constraint set $\mathcal{X}$, control input constraint set $\mathcal{U}$ and the terminal set $\mathcal{X}_f$, (2.20) to a mp-QP problem taking the form,

$$J(x, U) = \min_{U} \frac{1}{2} U^T H U + x^T F U + x^T Y x, \tag{2.24}$$

here $x = x_k$ and $U = [u_k, u_{k+1}, \cdots, u_{k+N_p-1}]^T$.

With respect to the constraints,

$$GU \leq Ex + W \tag{2.25}$$

Starting from the cost function (2.20a) taking a 2-norm form,

$$J(x_k, U^*) = \; x^T_{k+N_p|k} P x_{k+N_p|k} + \sum_{i=0}^{N_p-1} [x^T_{k+i|k} Q x_{k+i|k} \; + \; u^T_{k+i|k} R u_{k+i|k}]. \tag{2.26}$$

The above cost function for the dynamical system (2.20b) can be expressed as:

$$J(x_k, U^*) = X_k^T \overline{Q} X_k + x_k^T Q x_k + U^T \overline{R} U \tag{2.27}$$

here

$$X_k = [x_{k+1|k}, \cdots, x_{k+N_p|k}]^T$$

And X_k can be written in the form,

$$X_k = \overline{A} x_k + \overline{B} U \tag{2.28}$$

here, $U = [u_{k|k}, \cdots, u_{k+N_p-1|k}]$,

$$\overline{A} = \begin{bmatrix} A \\ A^2 \\ \vdots \\ A^{N_p} \end{bmatrix} \in \mathbb{R}^{nN_p \times n}, \; \overline{B} = \begin{bmatrix} B & 0 & \cdots & 0 \\ AB & B & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ A^{N_p-1}B & \cdots & \cdots & B \end{bmatrix} \in \mathbb{R}^{nN_p \times mN_p}$$

$$\overline{Q} = \begin{bmatrix} Q & & & \\ & \ddots & & \\ & & Q & \\ & & & P \end{bmatrix} \in \mathbb{R}^{(nN_p)\times(nN_p)} \text{ and } \; \overline{R} = \begin{bmatrix} R & & \\ & R & \\ & & \ddots \\ & & & R \end{bmatrix} \in \mathbb{R}^{(mN_p)\times(mN_p)}.$$

Substituting (2.28) in (2.27), we obtain:

$$J(x_k, U) = (\overline{A}x_k + \overline{B}U)^T\overline{Q}(\overline{A}x_k + \overline{B}U) + x_k^T Q x_k + U^T\overline{R}U \tag{2.29}$$

Rewriting (2.29) to

$$J(x_k, U) = x_k^T(\overline{A}^T\overline{QA} + Q)x_k + 2x_k^T\overline{A}^T\overline{QB}U + U^T(\overline{R} + \overline{B}^T\overline{QB})U \tag{2.30}$$

Now equating (2.24) to (2.30), we have:

$$H = \overline{R} + \overline{B}^T\overline{QB} \in \mathbb{R}^{mN_p \times mN_p}$$

$$Y = \overline{A}^T\overline{QA} + Q \in \mathbb{R}^{n \times n}$$

$$F = 2\overline{A}^T\overline{QB} \in \mathbb{R}^{n \times mN_p}$$

Similarly the constraints (2.20d)-(2.20f) and the predicted output variable $y_{k+i|k}$ can be transformed to the form given in (2.25).

Solving the mp-QP problem (2.24) with the constraint set (2.25) yields a finite set of affine functions defined over the polyhedral partition of the parametric set $\mathcal{R}$, where $\mathcal{P}_N(\mathcal{R}) = [\mathcal{R}_1, \mathcal{R}_2 \cdots \mathcal{R}_N]$ is called the *polyhedral partition* of the set $\mathcal{R}$. The polyhedral sets $\mathcal{R}_i \subset \mathbb{R}^n$ are called *critical regions* or *components* of the partition and correspond to the activation of a particular subset of constraints in the optimization problem. The polyhedral regions $\mathcal{R}_i$ are non overlapping i.e., $int(\mathcal{R}_i) \cap int(\mathcal{R}_j) = \emptyset$, for $i \neq j$. The non-overlapping property ensures the unique solution for the state $x_k \in \mathcal{R}_i$, $\forall i \in \mathcal{I}_N$. Two neighboring regions $\mathcal{R}_i$ and $\mathcal{R}_j$ share some vertices or facets and thus the optimal solution over the set $\mathcal{R}$ enjoys continuity properties.

Definition 2.8.1 *A mapping function*

$$f_{pwa} : \mathcal{R}^n \to \mathbb{R}^m, \quad f_{pwa}(x) = \underline{A}_i x + \underline{b}_i, \forall x \in \mathcal{R}_i, \ i \in \mathcal{I}_N \tag{2.31}$$

defined over the polyhedral partition of the set $\mathcal{R}$ is called a piecewise affine function on the polyhedral partition.

Theorem 2.8.1 *[BMDP02] Consider the multi-parametric quadratic programming problem* (2.24)*, the constraint set* (2.25) *and let $H > 0$. Then the set of feasible parameters $\mathcal{R}$ is a closed and bounded convex set, the optimizer function $U^* : \mathcal{R} \to \mathbb{R}^m$ is continuous and piecewise affine.*

Theorem 2.8.2 *[BMDP02] Consider the piecewise affine function defined over the polyhedral partition of the set $\mathcal{R}$. Then the set $\mathcal{R}$ is closed and bounded, the polyhedral regions $\mathcal{R}_i, i \in \mathcal{I}_N$ are closed, bounded and non empty sets, the regions are non overlapping $int(\mathcal{R}_i) \cap int(\mathcal{R}_j) = \emptyset$, for $i \neq j, i, j \in \mathcal{I}_N$.*

The proof for the continuity of the piecewise affine controller and the convexity of the set $\mathcal{R}$ are shown in [BMDP02] and is not recalled here.

The explicit control law or the state feedback PWA control law, solution of (2.20), is synthesized in terms of the piecewise affine function defined over the polyhedral partition of the set $\mathcal{R}$ and it can be described by

$$u_{pwa}(x_k) = F_i x_k + g_i, \quad \forall x_k \in \mathcal{R}_i. \tag{2.32}$$

where $\mathcal{R}_i = \{x \mid H_i x \leq w_i\}$, $i \in \mathcal{I}_N$, $F_i \in \mathbb{R}^{m \times n}$ and $g_i \in \mathbb{R}^m$.

By substituting the piecewise affine state feedback control law (2.32) in the equation (2.18), we obtain

$$x_{k+1} = (A + BF_i)x_k + Bg_i, \ \forall x_k \in \mathcal{R}_i, \ \ i \in \mathbb{I}_N. \tag{2.33}$$

To illustrate the explicit MPC, consider this linear time-variant system:

$$\begin{aligned} x_{k+1} &= \begin{bmatrix} 1.2 & 0 \\ 0.5 & 1 \end{bmatrix} x_k + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u_k, \\ y_k &= [1 \ 0] x_k. \end{aligned}$$

The input and output variables are subject to constraints:

$$-5 \leq y_k \leq 5 \text{ and } \ -5 \leq u_k \leq 5$$

The input and state weighing matrices are given as follows:

$$Q = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \text{ and } R = 1.$$

The positively invariant set $\mathcal{X}_f$ ($\mathcal{X}_f$ is the polytopic approximation of the maximal admissible ellipsoidal invariant set obtained with infinite horizon) is chosen to be the terminal set for the optimization problem. The terminal matrix P computed from the Algebraic Riccati equation (2.22) is given below as:

$$P = \begin{bmatrix} 3.456 & 2.111 \\ 2.111 & 4.085 \end{bmatrix}$$

The length of the prediction horizon chosen is 5, and the constrained finite time optimal control problem for the discrete time system is rewritten as follows:

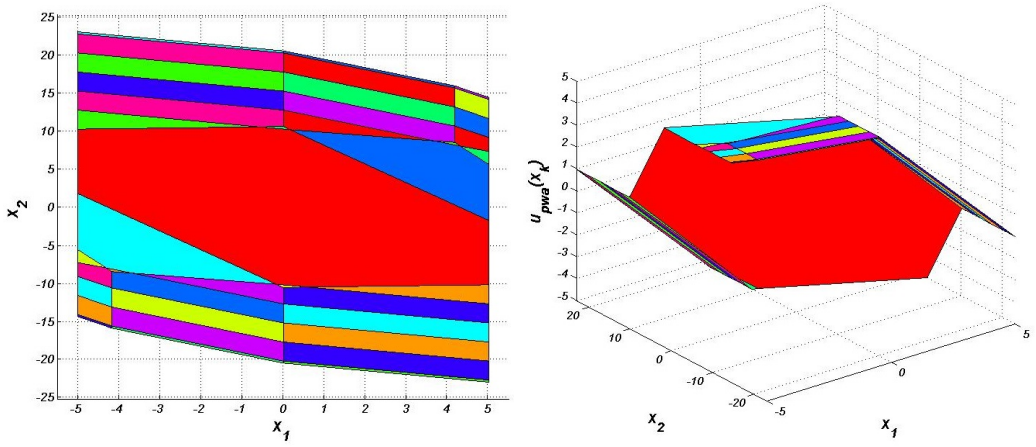
$$J(x_k, U^*) = \ x_{k+5|k}^T P x_{k+5|k} + \sum_{i=0}^{4} [x_{k+i|k}^T Q x_{k+i|k} \ + \ u_{k+i|k}^T R u_{k+i|k}].$$

The cost function matrices H, Y and F computed from (2.30) are as follows:

$$H = \begin{bmatrix} 162.126 & 116.136 & 79.562 & 49.599 & 25.667 \\ 116.136 & 88.211 & 59.606 & 37.519 & 19.635 \\ 79.562 & 59.606 & 44.977 & 27.453 & 14.599 \\ 49.599 & 37.519 & 27.453 & 21.064 & 10.407 \\ 25.666 & 19.635 & 14.599 & 10.400 & 8.913 \end{bmatrix}$$

$$Y = \begin{bmatrix} 140.827 & 26.557 \\ 26.557 & 9.085 \end{bmatrix} \text{ and } F = \begin{bmatrix} 210.914 & 152.047 & 103.508 & 64.095 & 32.912 \\ 37.524 & 25.366 & 16.067 & 9.151 & 4.222 \end{bmatrix}$$

Now solving the mp-QP problem with all these informations mentioned above and the constraint set (2.25), we obtain state feedback piecewise affine controller defined over the state-space partitions. The state partitions and the piecewise affine controller are illustrated in the Figure 2.1. The red polytope in the center of the set $\mathcal{R}$ represents the terminal set $\mathcal{X}_f$ and it is shown in Figure 2.1a. **Note:** Only the first optimal control input $u_{k|k}$ from the optimal control sequence U is defined over the state space partitions and it is illustrated in the Figure 2.1b.



H (a) Polyhedral partition for $N_p = 5$. (b) Piecewise affine controller.

Figure 2.1: An illustrative example for explicit model predictive control.

2.8.1 Real-time implementation of EMPC solutions:

The optimal solution of the explicit controllers, as mentioned before, is a simple algebraic expression. But, the computational complexity of the explicit controllers increases with the number of the controllers defined over the state space. For small number of controllers, finding the region that contains the given state x is computationally faster than solving a QP problem for the standard MPC for the optimal control input with the measured state. This advantage of the explicit controller has increased the attention in a wide range of fields.
Such controllers can be easily implemented on FPGA's or micro-controller chips. For example consider the dynamical example mentioned above. The solution set of the explicit controller consists of 39 continuous and piecewise affine controllers defined over 39 bounded polyhedral regions.
The optimal control input for the above dynamical system is given as follow:

$$u_{pwa}(x_k) = \left\{ \begin{array}{ll} F_1 x_k + g_1 & \text{if } x_k \in \mathcal{R}_1 \\ F_2 x_k + g_1 & \text{if } x_k \in \mathcal{R}_2 \\ \vdots & \\ F_{39} x_k + g_{39} & \text{if } x_k \in \mathcal{R}_{39} \end{array} \right\}$$

Definition 2.8.2 *Consider $x \in \mathcal{R}$ and the polyhedral partitions $\mathcal{R} = \cup_{i=1}^{N} \mathcal{R}_i$, then the point location function of the polyhedral partition of $\mathcal{R}$ is given by,*

$$i(x) = \min_i \; i \quad s.t \; x \in \mathcal{R}_i. \tag{2.34}$$

In order to implement the explicit solutions on FPGA's or micro-controller for any real-time applications, the following procedure has to be followed:

1. Off-line: Store all the polyhedral regions $\mathcal{R}_i$, the PWA control gains F_i and the constant components g_i in the memory unit of the micro-controllers.

2. On-line: The point location function is employed to find the polyhedral regions for the given parameter x. The position of the region that contains x is given by $i(x)$.

3. On-line: Evaluate the PWA control law $u_{pwa}(x_k) = F_{i(x_k)}x_k + g_{i(x_k)}$ based on the current state x_k.

The step 2 of this procedure is the one which deserves a careful implementation. Several methods have been proposed in the literature, from sequential search [TJB02], to binary search trees [TJB03], hash trees [BJJ12] or alternative solutions based on the resolution of a simple LP problem [NOBRA14].

2.9 Tools for multi-parametric programming

The multi-parametric programming emerged at the beginning of the years 2000 and has been accompanied by several numerical implementations available under different free or commercial packages. In the present thesis, the main package used in this respect is MPT Toolbox. MPT Toolbox was developed by a team of researchers leaded currently by M. Kvasnica and is publicly available [HKJM13]. The objective of this toolbox is to allow researchers/students to solve parametric programming problems in the Matlab environment. The toolbox covers a wide range of optimization solvers such as QP solvers, mp-QP solvers and allows the specification of dynamical models as either linear models, linear with parameter uncertainties, Hybrid system and allows different problem formulations as MPC, dynamic programming etc. The set theoretic operators such as set difference, intersection, Minkowski sum and etc. are also available in the toolbox. Throughout this thesis work, we use this toolbox extensively in order to solve the multi-parametric Quadratic Programming problem for explicit MPC but also as basic tool for polyhedral manipulation.

2.10 Conclusion

In this chapter, some of the fundamental definitions and notations of the convex sets, polyhedra and linear systems have been presented. A brief introduction on the set invariance theory and contractive sets has been presented with the definitions of the invariant set and contractive set for different classes of linear discrete-time systems. The objective, advantages, working principle and construction of the linear MPC have been briefly discussed. The transformation of the QP cost function of the MPC problem into mp-QP for the explicit MPC has been briefly explained. The toolbox that will be used extensively throughout this thesis has been also introduced.

Chapter 3

Robustness margins for piecewise affine explicit control law

Contents

3.1 Introduction 23
3.2 Preliminaries 24
3.3 Gain Margin 26
3.3.1 Problem Formulation 26
3.3.2 Construction/Constructive results 28
3.3.3 Computation results 29
3.3.4 Example 30
3.4 First order neglected dynamics 39
3.4.1 Problem Formulation 40
3.4.2 Construction of vertex representation for the augmented state 41
3.4.3 Controller for extended state system 41
3.4.4 Admissible set for the first order neglected dynamics: . . . 42
3.4.5 Example 44
3.5 Conclusion 44

3.1 Introduction

Classical robustness margins i.e., gain margin and phase margin, consider the gain variation and phase variation of the model for which the stability of the closed loop is preserved. The gain margin and phase margin have been in use for decades to understand the closed-loop behavior of any linear system given in the form of transfer functions [Oga01]. Such margins are also used to study the degree of robustness to process variations and model uncertainties [Oga01]. From the analysis point of

view, it is important to take into account the capacity of the control law to cope with disturbances, neglected dynamics or uncertain parameters. This characteristics is termed in control theory as robustness of the controller. In the context of robustness analysis of explicit PWA controller, very few contributions have been made. Few noticeable works include an analysis procedure proposed in [ONB+13, NOBRA14] handling the robustness/fragility of the positive invariance for the dynamics affected by uncertain parameters. On the other hand, there is a substantial work on the robustification of the explicit controllers. The reference [RAO13] shows how to improve the robustness of the controller taking disturbances into account in the design phase. In a different perspective [PFKP09] presented a robust explicit predictive control synthesis approach which accounts for uncertainties based on dynamic programming. The authors of [dlPBF04] show how the approximate multi-parametric programming problem can be implemented as robust MPC controller for linear systems with polytopic uncertainty. M.V Kothare et al [KBM96] incorporate the plant model/polytopic uncertainty in the description of robust MPC problem formulation involving LMI (Linear Matrix Inequalities) for robustly stabilizing the set of uncertain plants. However, these last developments can be considered mainly as robust design methodologies and not as robustness analysis tools "per se".

In this chapter, we consider the inherent robustness properties of an existing explicit predictive control described as piecewise affine (PWA) control law for a class of linear discrete-time systems. First, we present a numerical method to compute a gain margin set for a discrete-time system stabilized by a continuous PWA affine dynamics with respect to the invariance property. The desired gain margin set is a polytope which characterize the variations of system gains preserving the invariant characteristics of the closed-loop. Second, we analyze the dynamic system affected by first order neglected dynamics. The robustness margin of the controller against first order neglected dynamics correspond to a set characterizing the neglected dynamics parameters preserving the invariance property.

3.2 Preliminaries

Let us introduce some matrices which will be used to store the vertices (elements of the polyhedral regions) of the polyhedral sets $\mathcal{R}_i$ and their corresponding local control laws. These matrices are used throughout this chapter for the problem formulation.

Consider a polyhedral partition of the set $\mathcal{R} = \cup_{i=1}^{N} \mathcal{R}_i$. The vertex representation of the polyhedral sets $\mathcal{R}$ and $\mathcal{R}_i$ are given by,

$$\mathcal{R} = conv\{w_1, w_2 \cdots w_r\}, \tag{3.1a}$$

$$\mathcal{R}_i = conv\{w_{i1}, w_{i2} \cdots w_{ir_i}\}, i = 1 \cdots N. \tag{3.1b}$$

A matrix $W \in \mathbb{R}^{n \times r}$ will be used to store the vertices of the polyhedron $\mathcal{R} \subset \mathbb{R}^n$. An illustration of the polyhedron $\mathcal{R}$ is depicted in Figure 3.1.

$$W = [w_1, w_2, \cdots, w_r]. \tag{3.2}$$

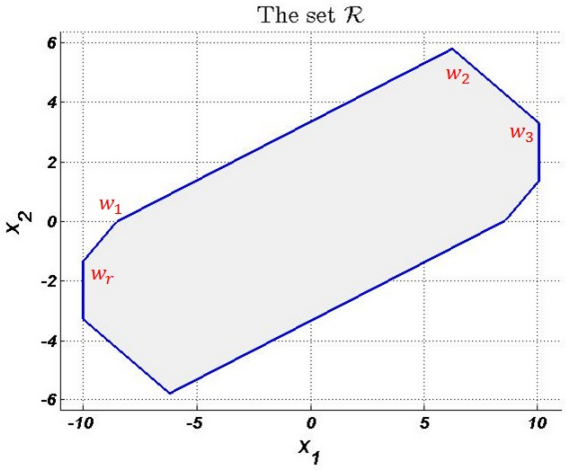


Figure 3.1: Illustration of the set $\mathcal{R}$ with its vertices.

Here, r represents the number of vertices in the set $\mathcal{R}$. We introduce a matrix V

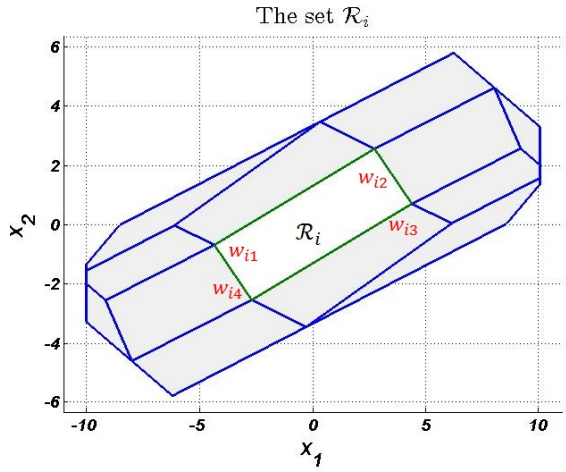


Figure 3.2: Illustration of the set $\mathcal{R}_i$ with its vertices

$\in \mathbb{R}^{n \times \sum_{i=1}^{N} r_i}$ to store the vertices of all the partition in the state space $\mathcal{R}_i \subset \mathcal{R}$, $\forall i \in \mathcal{I}_N$.

$$V = [w_{11}, w_{12}, \cdots, w_{ir_i}, \cdots, w_{Nr_N}]. \tag{3.3}$$

An illustration of the polytope $\mathcal{R}_i$ is shown in Figure 3.2 with its member vertices and for this particular example the matrix V has 2 rows and 52 columns. As stated before, neighboring partitions have vertices in common, and therefore, the repeated vertices are removed from the matrix and therefore V contains non-identical column vectors.

$$V = [w_1, w_2, \cdots, w_p] \in \mathbb{R}^{n \times p}, \tag{3.4}$$

where, p denotes the number of distinct vertices within the partition. With the help of affine mapping $f(x(k)) = u_{pwa}(x(k))$, a matrix $U \in \mathbb{R}^{m \times p}$ is used to store the control input for each column vector of the matrix V.

$$U = u_{pwa}(V). \tag{3.5}$$

3.3 Gain Margin

In this section, we examine the effect of the gain modification with respect to the closed-loop characteristics obtained with the piecewise affine explicit control law.

3.3.1 Problem Formulation

This subsection presents the formulation of a robustness measure in terms of a mp-QP problem. The criterion is the robustness margin notion which is understood as an allowable gain margin of a closed-loop PWA control law. The problem statement of the gain margin in closed-loop is given in the equations below and is also illustrated in Figure 3.3.

$$x(k+1) = Ax(k) + B(I_m + diag(\delta_K))u_{pwa}(x(k)) \tag{3.6a}$$
$$u_{pwa}(x(k)) = F_i x(k) + g_i \tag{3.6b}$$

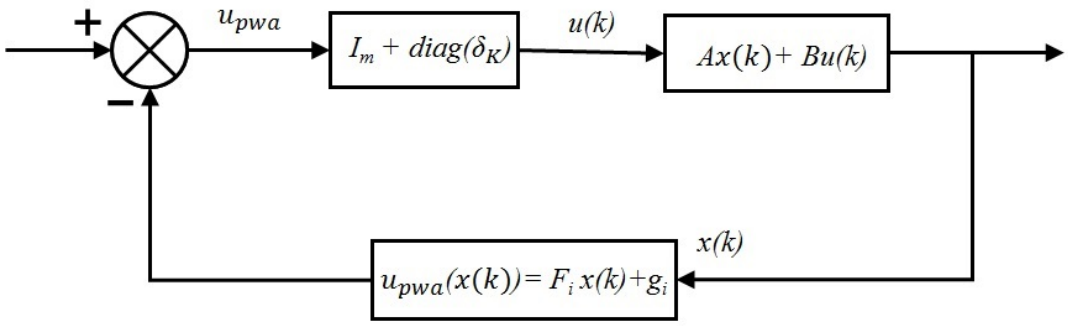


Figure 3.3: Representation of Gain Margin in closed-loop

In the following, we describe the gain margin set for the system (3.6a) stabilized with the help of a state feedback explicit control law (3.6b). The construction of the gain margin set will be based on vertex representation of the regions forming the partition $\mathcal{R}$.

Definition 3.3.1 *Consider a discrete time linear system* (3.6a) *with a continuous PWA control law* (3.6b)*, such that the set $\mathcal{R}$ in the state-space is positively invariant. The* Gain Margin *is represented by the set $\mathcal{K} \subset \mathbb{R}^m$, for which $x(k+1) = Ax(k) + B(I_m + diag(\delta_K))u_{pwa}(x(k)) \in \mathcal{R}$, $\forall x(k) \in \mathcal{R}$ and $\delta_K \in \mathcal{K} \subset \mathbb{R}^m$.*

The set $\mathcal{K} \subset \mathbb{R}^m$ is a set which contains the input channels gain variations δ_K such that for any point inside the set $\mathcal{K}$, the invariance characteristics of the set $\mathcal{R}$ is preserved.

Theorem 3.3.1 *Consider a discrete-time linear system* (3.6a) *with full column rank B and a piecewise affine state feedback control law defined in* (3.6b)*. The gain margin $\mathcal{K}$ of the controller is represented by the set,*

$$\mathcal{K} = \cap_{q=1}^{p} \mathcal{K}_q \tag{3.7}$$

Where $\mathcal{K}$ represents the gain margin set and $\mathcal{K}_q$ the local gain margin for the vertex w_q for some $q \in \mathcal{I}_p$.

Proof 3.3.1 *Starting from the PWA closed loop dynamics assuring the invariance of the set $\mathcal{R}$ given by,*

$$x(k+1) = Ax(k) + Bu_{pwa}(x(k)) \in \mathcal{R} \tag{3.8}$$

with the addition of δ_K in (3.8), $\forall \delta_K \in \mathcal{K}$ we obtain:

$$x(k+1) = Ax(k) + B(I_m + diag(\delta_K))u_{pwa}(x(k)) \in \mathcal{R}. \tag{3.9}$$

It is also possible to exploit the structure of the PWA closed-loop dynamics, by introducing the parametric variations on the control gains preserving the invariance and boundedness properties of the controller. Suppose that $\hat{u} = u_{pwa}(x(k)) + \delta u$, where the term $\delta_u \in \mathbb{R}^m$ describes admissible control input variations,

$$x(k+1) = Ax(k) + Bu_{pwa}(x(k)) + B\delta_u \in \mathcal{R}. \tag{3.10}$$

By relating (3.9) and (3.10) we obtain:

$$\begin{aligned} Ax(k) + Bu_{pwa}(x(k)) + B\delta_u = \\ Ax(k) + B(I_m + diag(\delta_K))u_{pwa}(x(k)). \end{aligned} \tag{3.11}$$

Actually,

$$B(u_{pwa}(x(k)) + \delta_u) = B(I_m + diag(\delta_K))u_{pwa}(x(k)). \tag{3.12}$$

And, considering B full column rank, we obtain:

$$u_{pwa}(x(k)) + \delta_u = (I_m + diag(\delta_K))u_{pwa}(x(k)). \tag{3.13}$$

By rewriting (3.13), by analogy to each column vector of the matrix V, $\forall q \in \mathcal{I}_p$(vertices in $\mathcal{R}_i$), we obtain:

$$\delta_{u_q} = diag(\delta_{K_q})u_{pwa}(w_q). \tag{3.14}$$

The admissible input δ_{u_q} belongs to a set $\Delta\mathcal{U}_q \subset \mathbb{R}^m$ and $\delta_{K_q} \in \mathcal{K}_q$. A matrix $M_q = diag(u_{pwa}(w_q)) \in \mathbb{R}^{m \times m}$ is uniquely defined based on the value of the control action $u_{pwa}(w_q)$ such that

$$\Delta\mathcal{U}_q \supseteq M_q\mathcal{K}_q. \tag{3.15}$$

Exploiting the polyhedral structure of the admissible input variations,

$$\Delta\mathcal{U}_q = \{u : |\hat{H}_u u \leq \hat{h}_u\}, \tag{3.16}$$

where $\hat{H} \in \mathbb{R}^{d_u \times m}$ and $\hat{h} \in \mathbb{R}^{d_u}$ one can obtain the local set of gain variation for the vertex w_q as:

$$\mathcal{K}_q = \{z \in \mathbb{R}^m | \exists u \in \Delta\mathcal{U}, M_q z = u\}. \tag{3.17}$$

Equation (3.17) *leads to:*

$$\mathcal{K}_q = \{z \in \mathbb{R}^m | \hat{H}_u M_q z \leq \hat{h}_u\}. \tag{3.18}$$

Finally,

$$\mathcal{K} = \cap_{q=1}^{p} \mathcal{K}_q. \tag{3.19}$$

The collection of sets $\mathcal{K}_q$ are independent of each vertex of the set in $\mathcal{R}_i$ and the intersection of these independent sets gives the global set $\mathcal{K}$. In order to compute $\mathcal{K}_q$, we first need to compute explicitly the sets $\Delta\mathcal{U}_q$.

Corollary 3.3.1 *The set $\mathcal{K}$ representing the gain margin is not empty.*

Proof 3.3.2 *Even if no control gain variation is admissible in (3.9), the null vector $\mathbf{0}_m \in \mathcal{K}$ is admissible as long as it corresponds to the set invariance of the original PWA control law.*

3.3.2 Construction/Constructive results

In this subsection, a description for the $\Delta\mathcal{U}_q$ set is constructed based on forward mapping of the vertex in the PWA partition.

Theorem 3.3.2 *Consider a linear discrete-time system* (3.6a) *stabilized by a piecewise affine control law* (3.6b). *The set $\Delta\mathcal{U}_q$ of admissible input variations at the vertex w_q is obtained by*

$$\Delta\mathcal{U}_q = Proj_U \mathcal{H}_q, \tag{3.20}$$

here, $U \in \mathbb{R}^m$ denotes the input space. The polyhedral set $\mathcal{H}_q$ is described by:

$$\mathcal{H}_q = \left\{ (\delta u, \gamma) \in \mathbb{R}^m \times \mathbb{R}^r, \text{ and } [A\ B] \begin{bmatrix} x(k) \\ u_{pwa}(x(k)) \end{bmatrix} + B\delta u = W\gamma | \mathbf{1}^T\gamma = 1 \right\}. \tag{3.21}$$

Proof 3.3.3 *Let us recall the equation (3.10) which resumes the positive invariance for the set $\mathcal{R}$,*

$$[A\ B] \begin{bmatrix} x(k) \\ u_{pwa}(x(k)) \end{bmatrix} + B\delta u \in \mathcal{R}. \tag{3.22}$$

From the definition of invariance, (3.22) can be expressed using the vertices of the set $\mathcal{R}$ as convex combination and corresponding variables $\gamma = [\gamma_1, \gamma_2, \cdots, \gamma_r]$ and equality constraints $\mathbf{1}^T\gamma = 1$ such that

$$[A\ B] \begin{bmatrix} x(k) \\ u_{pwa}(x(k)) \end{bmatrix} + B\delta u = W\gamma. \tag{3.23}$$

Remark 3.3.1 *The set in* (3.20) *characterizes input variations guaranteeing the positive invariance. The control action in itself is not modified but the variations of the system gain is related to the set defined in* (3.20). *This is the reason for not taking into account explicitly the input constraints in* (3.21).

3.3.3 Computation results

Based on the value of the control action $u_{pwa}(w_q)$, consider the objective of finding the $\Delta\mathcal{U}_q$ set for each column vector of the matrix V independently, $\forall q \in \mathcal{I}_p$.

$$Aw_q + Bu_{pwa}w_q + B\delta_u = W\gamma, \tag{3.24}$$

here $\gamma = [\gamma_1, \gamma_2, \cdots, \gamma_r]$ and $\mathbf{1}^T\gamma = 1$ with $\gamma_i \geq 0 \ \forall i \in \mathcal{I}_r$.
Clearly (3.24) can be represented by a system of linear equalities with the variable vector Γ, where the vector Γ contains the required variable δ_u along with the variables $[\gamma_1, \cdots, \gamma_{r-1}]$.

$$\Gamma = [\delta_u, \gamma_1, \gamma_2, \cdots, \gamma_{r-1}] \in \mathbb{R}^{m+r-1}, \tag{3.25a}$$

$$\gamma_r = 1 - \sum_{i=1}^{r-1}\gamma_i, \quad \gamma_i \geq 0. \tag{3.25b}$$

A linear programming problem with (3.24) and (3.25) as linear equality constraints can be setup,

$$\begin{bmatrix} \begin{bmatrix} B(1,\cdot) \\ \vdots \\ B(n,\cdot) \end{bmatrix} & -\hat{W} \end{bmatrix} \Gamma = w_r - (Aw_q + Bu_{pwa}(w_q)), \tag{3.26}$$

where $\hat{W}$ is defined by the notation $W = [w_1 \cdots w_r]$,

$$\hat{W} = [w_1 - w_r, \cdots, w_{r-1} - w_r] \in \mathbb{R}^{n\times r-1}.$$

The system of linear equalities (3.26) is defined in the form $H\Gamma = h$, where $H \in \mathbb{R}^{n\times(m+r-1)}$ and $h \in \mathbb{R}^{n\times 1}$, and has the solution

$$\Gamma = H_z t + h_z, \tag{3.27}$$

here $H_z \in \mathbb{R}^m$ is nothing but the orthonormal basis for the null space of matrix H and h_z is a particular solution for the linear programming problem presented in (3.26). Subsequently, we establish a polyhedral set from the matrices H_z, h_z with a variable t.

The set $\Delta\mathcal{U}_q$ can be determined by splitting (3.27) in two parts. The first part corresponds to $H_z^{(1)}t + h_z^{(1)} = \delta_u$. And recall the non-negativity constraints defined in (3.25b). A polyhedron is defined with the non-negativity property of the elements in the solution vector,

$$\Gamma(m+1 : m+r-1) = [\gamma_1, \cdots, \gamma_{r-1}].$$

The polyhedron is given in the form,

$$\mathcal{H}^{(2)} = \{t| \ -H_z^{(2)}t \leq h_z^{(2)}\}. \tag{3.28}$$

The matrices appearing in (3.27) are related to a decomposition:

$$\begin{aligned} H_z^{(1)} &= H_z(1:m,:), \\ h_z^{(1)} &= h_z(1:m), \\ H_z^{(2)} &= H_z(m+1:m+r-1,:) \ \textit{and}, \\ h_z^{(2)} &= h_z(m+1:m+r-1). \end{aligned}$$

And finally the set $\Delta\mathcal{U}_q$ is nothing but a linear transformation of the set $\mathcal{H}^{(2)}$,

$$\Delta\mathcal{U}_q = H_z^{(1)}\mathcal{H}^{(2)} \oplus h_z^{(1)}. \tag{3.29}$$

The vertices of $\Delta\mathcal{U}_q$ are computed by applying the transformation to all the vertices of the set $\mathcal{H}^{(2)}$.

Corollary 3.3.2 *The set $\mathcal{K}_q$ representing the gain matrix set is polyhedral $\forall q \in \mathcal{I}_p$.*

Proof 3.3.4 *The set $\mathcal{H}_q$ and $\Delta\mathcal{U}_q$ used in the description of $\mathcal{K}_q$ are polyhedral and, by this virtue, $\mathcal{K}_q$ inherits this property.*

Remark 3.3.2 *We note the analogy between the results reported in [NOBRA14] and (3.20). The work of [NOBRA14] focuses on the computation of fragility margin which is characterized for the parametric variations in the space of δ_{Fi} for each individual region. However, in the present framework we interpret those findings to compute the variations in the control input for each individual vertex in the set $\mathcal{R}$.*

Remark 3.3.3 *Regarding the computation of the set $\mathcal{K}$, one needs to solve p LP problems with a set of linear equations given in (3.26), where p is the number of non-identical vertices in the set $\mathcal{R}$.*

3.3.4 Example

Single Input system:

Consider a linear discrete-time system,

$$\begin{aligned} x(k+1) &= \begin{bmatrix} 1.2 & 0 \\ 0.8 & -1.1 \end{bmatrix} x(k) + \begin{bmatrix} 1.0 \\ 0.3 \end{bmatrix} u_k, \\ y_k &= [1\ 0]\ x(k). \end{aligned}$$

The weighing matrices of the states and input are chosen

$$Q = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \text{ and } R = 1$$

in the problem formation of the open-loop optimal control problem considered for Explicit MPC.
The prediction horizon chosen is 2 for simplicity of the partition and readability of the graphical illustrations. The input constraint $-5 \leq u_k \leq 5$ and the output constraint $-5 \leq y_k \leq 5$ are given as box constraints. Solving the mp-QP problem

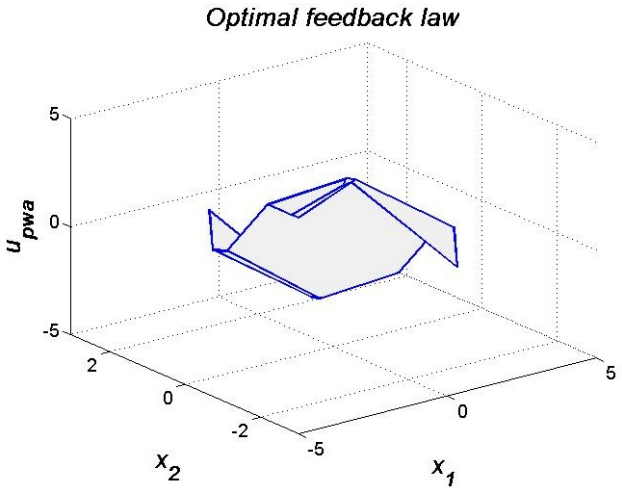


Figure 3.4: Illustration of the Optimal feedback law.

yields 11 controllers. Figure 3.4 shows the optimal feedback law u_{pwa} as a function of the states, the states partition is represented in x and y axis and the feedback control law imposed over the states is in z axis respectively.

The state partition of 11 resulting regions obtained as the solution of the mp-QP is presented in Figure 3.5. The state partition is invariant with respect to the obtained PWA control law and this is because the feasible set $\mathcal{R}$ is invariant to the PWA controller $u_{pwa}(x(k))$. The gain margin set $\mathcal{K}$ in (3.7) is computed and the

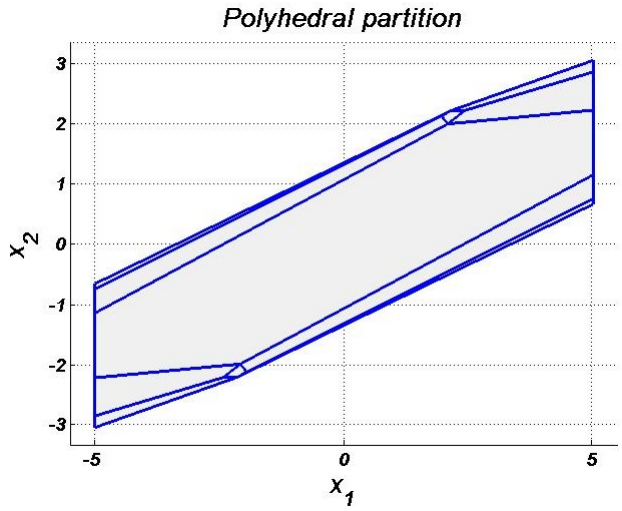


Figure 3.5: Representation of Polyhedral partition.

value of δ_K lies in the interval $[-0.2178,\ 0.3051]$. For any gain in the set $\mathcal{K}$ with respect to the PWA closed-loop formulation the invariance characteristics of the controller is preserved. In the Figure 3.6 the state trajectories simulated in closed loop with $\delta_K = 0$ for all the vertex entries of the matrix V as initial states are illustrated. State trajectories are simulated in closed loop with δ_K value such that $x(k+1) = Ax(k) + B(1+\delta_K)u_{pwa}(x(k))$. Figure 3.7 shows the state trajectories for $\delta_K = 0.3$ which is a value inside the computed gain margin and it is observed that the trajectories are inside the invariant set thus confirming the theoretical

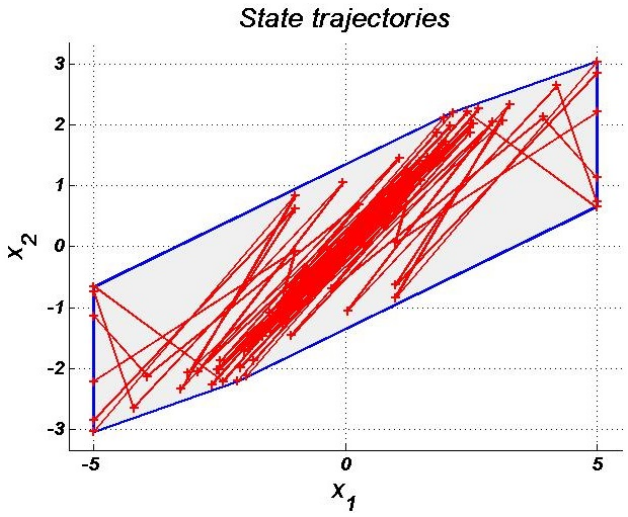


Figure 3.6: state trajectories for $\delta_K = 0$.

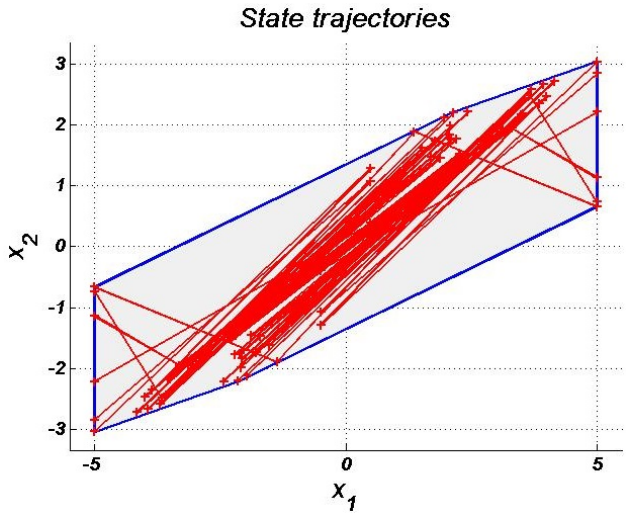
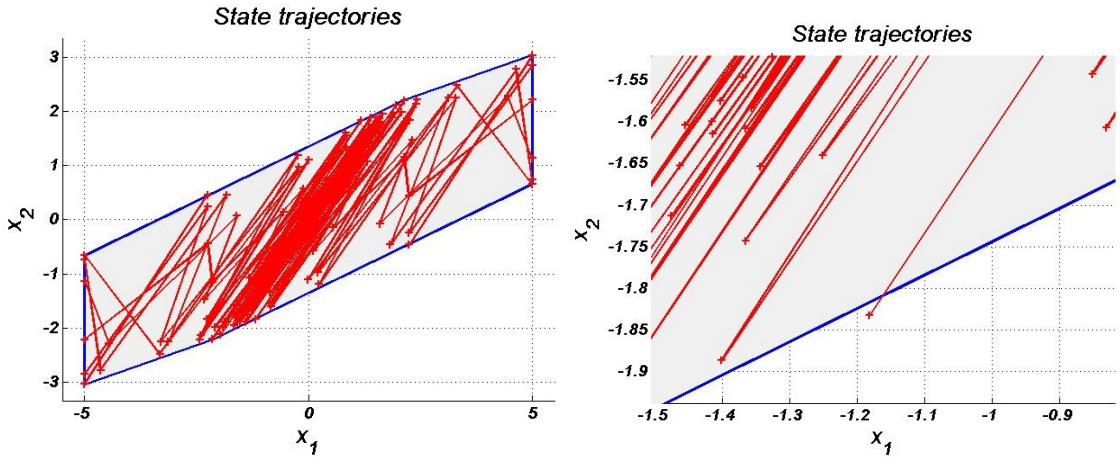


Figure 3.7: state trajectories for $\delta_K = 0.3$.

result. From Figure 3.8, it is noticeable that the controller is no more invariant for $\delta_K = -0.25$. Next, we simulate the state trajectories and control input for different gains $\delta_K \in \mathcal{K}$ in closed-loop. The gains considered for simulations are $\delta_K = 0.3, \delta_K = 0$ and $\delta_K = -0.2$. The state and control simulations for these gains in closed-loop are illustrated in Figure 3.9. The performance changes with gain because the response time will significantly vary for different gains. The range of gain variation that the system can tolerate is also observed from the Figure 3.9. For the gain $\delta_K = 0.3$, the state trajectories are converging relatively fast to the origin whereas the state trajectories for gain $\delta_K = -0.2$ in closed-loop are slowly approaching to the origin. The number of time samples taken for the system with $\delta_K = 0.3$ in closed-loop to settle to the origin is approximately 22. This performance can be observed from the subplots in the Figure 3.9. Finally, we compute the gain margin for the same system dynamics for different prediction horizons and with the same weight matrices Q and R. The different prediction horizons N_p considered for gain margin analysis are provided in the Table 3.1. The number of regions, the number of unique vertices in the set $\mathcal{R}_i$, computational time for obtaining the gain



(a) Full Image of the state trajectories. (b) Zoomed Image of 3.8a.

Figure 3.8: state trajectories for $\delta_K = -0.25$

Table 3.1: Gain Margin for different prediction horizons

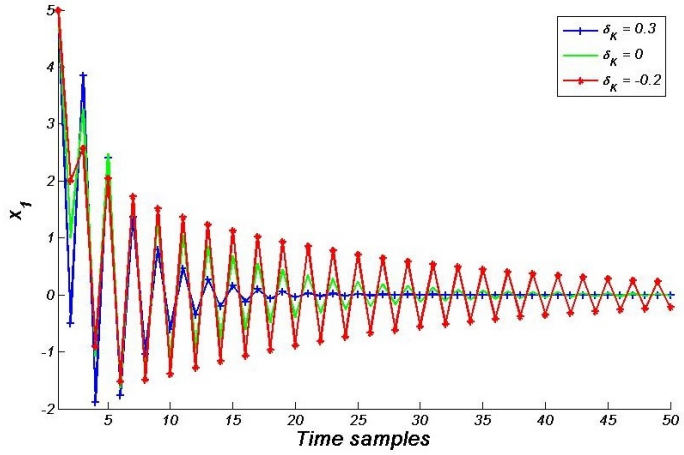
N_p	*Number of regions*(N)	*Number of vertices*(p)	*Computational Time in seconds*	*Gain Margin* $\mathcal{K}$
2	11	20	0.3	[-0.2178, 0.3051]
5	43	58	0.97	[-0.1231, 0.1611]
10	137	162	5.4	[-0.0945, 0.1280]

margin set $\mathcal{K}$ and the gain margin range for different prediction horizons are also mentioned in the Table 3.1.

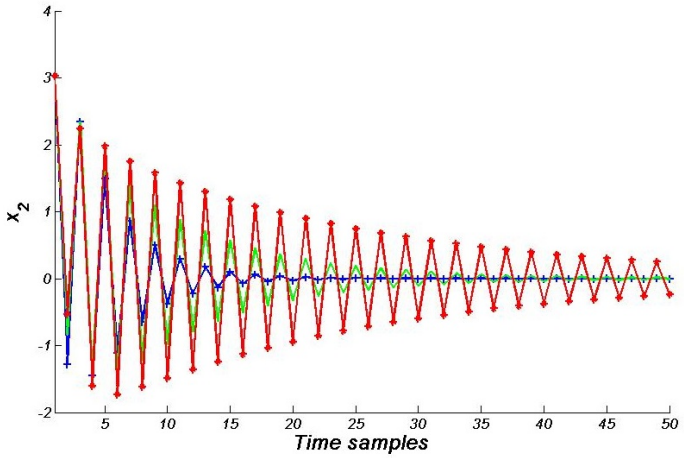
It can be observed that the gain margin set $\mathcal{K}$ shrinks with the increase in prediction horizon. It is a fact that the number of polyhedral regions increases with higher prediction horizon. The polyhedral partitions for prediction horizon 5 and 10 are illustrated in the Figure 3.10.

Note: For large prediction horizon the system in closed loop is usually slower and more robust than those of small ones. Similarly the feasible region is large for large prediction horizon and the feasible set $\mathcal{R}$ for different prediction horizons is illustrated in Figure 3.11.

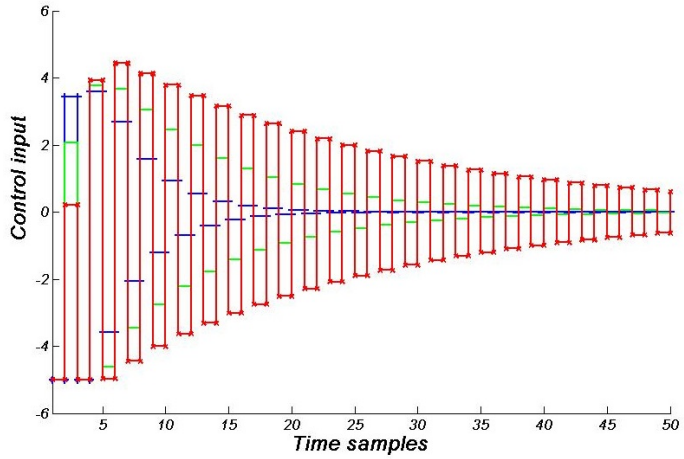
From the gain margin analysis, it should be stressed out the fallacious conclusion that if prediction horizon is large the gain margin will be small as it can be observed in the previous Table. It is important to recall that the margins are related to different PWA controllers and as a result their domain of validity is different rendering the comparison of gain margins for different horizon highly dependent on the topology of the partition. In order to illustrate that for large prediction horizon the gain margin is in fact large, we conduct an analysis based on different controller for different state partitions. Let us denote $\mathcal{R}^2$ and $\mathcal{R}^{10}$ as the feasible sets for prediction horizons $N_p = 2$ and $N_p = 10$ respectively. Similarly we denote u^2_{pwa} and u^{10}_{pwa} as



(a) State trajectories for the state x_1 for different gains.



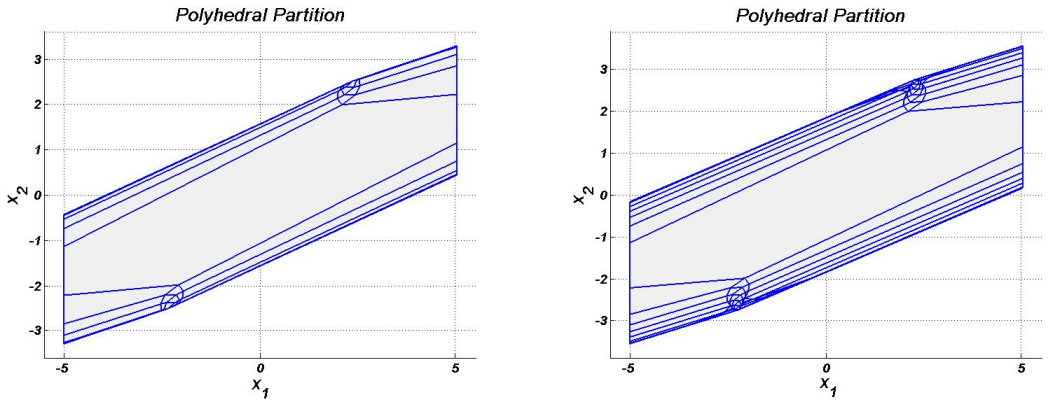
(b) State trajectories for the state x_2 for different gains.



(c) Control simulation for different gains.

Figure 3.9: States and control input simulations for different gains.

PWA controllers for $N_p = 2$ and $N_p = 10$ respectively. In the following, we establish two analysis:



(a) Polyhedral partition for $N_p = 5$. (b) Polyhedral partition for $N_p = 10$.

Figure 3.10: Polyhedral partitions for different prediction horizons.

Analysis 1 : Compute the Gain Margin for the following problem:

$$x(k+1) = Ax(k) + B(I_m + diag(\delta_K))u^{10}_{pwa} \in \mathcal{R}^{10}, \forall x(k) \in \mathcal{R}^2, \delta_K \in \mathcal{K}.$$

The gain margin $\mathcal{K}$ is computed and the value of δ_K lies between $[-0.4436,\ 0.4697]$.

Analysis 2 : Compute the Gain Margin for the following problem:

$$x(k+1) = Ax(k) + B(I_m + diag(\delta_K))u^{10}_{pwa} \in \mathcal{R}^2, \forall x(k) \in \mathcal{R}^2, \delta_K \in \mathcal{K}.$$

The gain margin $\mathcal{K}$ is computed and the value of δ_K lies between $[-0.2178,\ 0.4697]$.

Thus it can be observed that the gain margin obtained, for the above two analysis (Analysis 1 and Analysis 2), is larger as the prediction horizon increases than the gain margin obtained for $N_p = 2$ as shown in the Table 3.1.

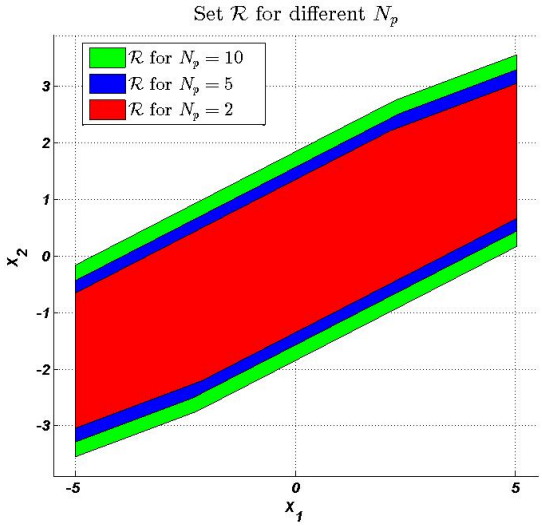


Figure 3.11: Illustration of set $\mathcal{R}$ for different prediction horizons

Multi Input system

Consider a linear discrete system with two inputs and two outputs,

$$x(k+1) = \begin{bmatrix} 1.2 & -1. & 0 \\ 0 & -1.2 & 0.5 \\ 0.2 & 0.4 & 0 \end{bmatrix} x(k) + \begin{bmatrix} 1.0 & 0.2 \\ 0.5 & 0 \\ 0 & 0.7 \end{bmatrix} u_k,$$

$$y_k = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} x(k).$$

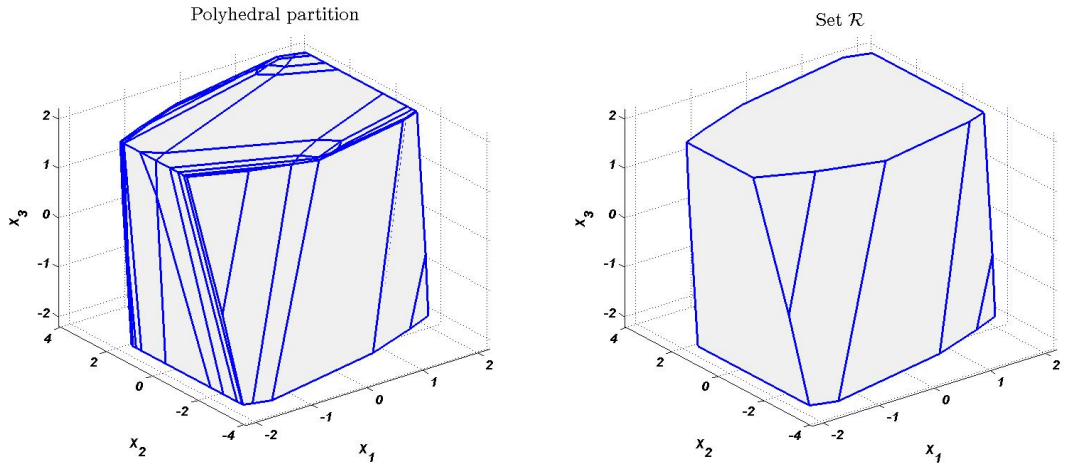
The weight applied on the control inputs and state vectors are

$$Q = \begin{bmatrix} 5 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \text{ and } R = \begin{bmatrix} 0.5 & 0 \\ 0 & 1 \end{bmatrix}.$$

The input constraints are $\begin{bmatrix} -2 \\ -2 \end{bmatrix} \leq u_k \leq \begin{bmatrix} 2 \\ 2 \end{bmatrix}$ and,

the output constraints $\begin{bmatrix} -2 \\ -2 \end{bmatrix} \leq y_k \leq \begin{bmatrix} 2 \\ 2 \end{bmatrix}$.

The prediction horizon chosen is 2. The resulting controller has 37 regions and it is illustrated in Figure 3.12a. The set $\mathcal{R}$ is illustrated in the Figure 3.12b.



(a) Representation of the state partition. (b) Representation of set $\mathcal{R}$.

Figure 3.12: Illustration of sets $\mathcal{R}_i$ and $\mathcal{R}$

Figure 3.13 shows the Gain margin set $\mathcal{K}$ and for all the points in this set the invariance property of the set $\mathcal{R}$ is guaranteed. However, the invariance property of the controller is fragile to unmeasured input perturbation. For instance, choosing $\delta_K = diag(-0.01, -0.01)$ results in the controller loosing the invariance.

To prove the positive side of gain margin analysis, a new explicit controller is chosen

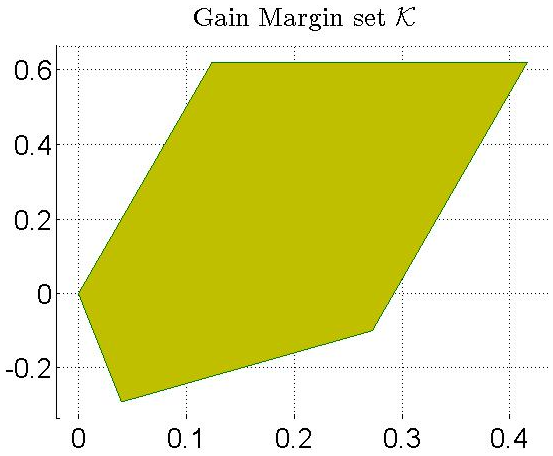


Figure 3.13: $\mathcal{K}$ set for $u_{pwa}(x(k))$

by adapting the gain with an admissible variation from the set $\mathcal{K}$, shown in Figure 3.13, and which preserves its definition domain over the set $\mathcal{R}$.

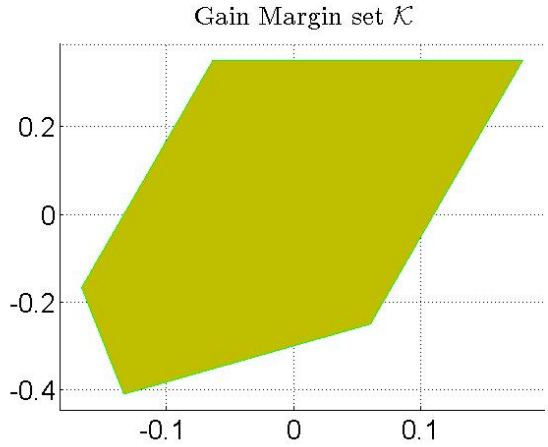


Figure 3.14: $\mathcal{K}$ set for $u^{\mu}_{pwa}(x(k))$

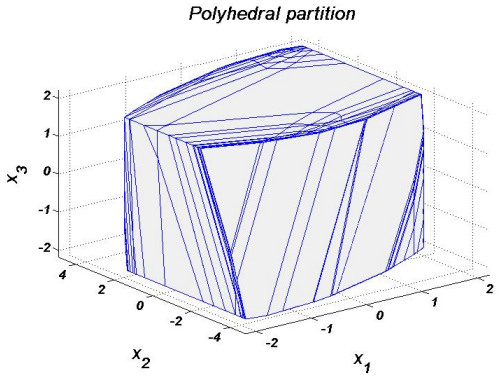
The new controller is adapted in a straightforward manner to:

$$u^{\mu}_{pwa}(x(k)) = (I_m + diag(\delta_K)) * (F_i x(k) + g_i),\ \forall i \in \mathcal{I}_N \tag{3.30}$$

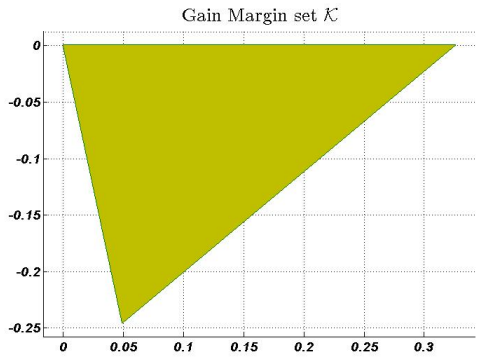
and $\delta_K = diag(0.2, 0.2)$. Subsequently, we computed the gain margin set for this controller and present an illustration of its corresponding admissible variation.
It is noticeable from the Figure 3.14 that the new controller is more robust to input disturbances and practically we obtained a translation of the point corresponding to the nominal controller without altering the shape of the set $\mathcal{R}$. The invariance property of the controller $u^{\mu}_{pwa}(x(k))$ is preserved. The new controller may violate the input constraints but the violation can be handled by saturating the control inputs.

N_p	*No of Regions*(N)	*No of Vertices*(p)	*Computational Time in seconds*	*Gain Margin* $\mathcal{K}$
2	37	110	5.1	Ref Figure 3.13
5	137	376	60.5	Ref Figure 3.15b
10	374	943	1400	Ref Figure 3.16b

Table 3.2: Information of the gain margin computation for different prediction horizons.

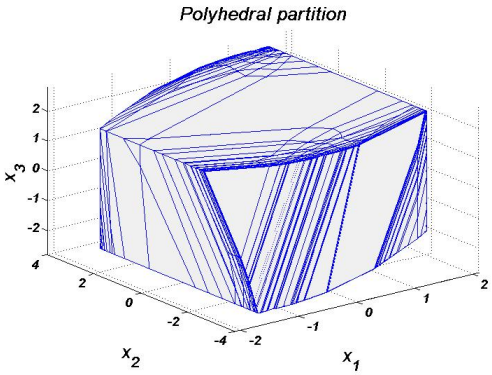


(a) Polyhedral partition for $N_p = 5$.

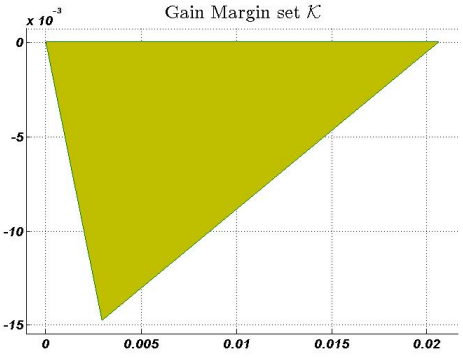


(b) $\mathcal{K}$ set for $N_p = 5$.

Figure 3.15: Polyhedral partition and the gain margin set $\mathcal{K}$ for prediction horizon $N_p = 5$.



(a) Polyhedral partition for $N_p = 10$.



(b) $\mathcal{K}$ set for $N_p = 10$.

Figure 3.16: Polyhedral partition and the gain margin set $\mathcal{K}$ for prediction horizon $N_p = 10$.

The mp-QP problem yields 137 controllers, for $N_p = 5$ and with same weight matrices Q and R, and its corresponding polyhedral partition is shown in Figure 3.15a. The gain margin set for this controller is computed and it is illustrated using the Figure 3.15b. Comparing the gain margin set from Figures 3.13 and 3.15a, it can be seen that the gain margin set for $N_p = 5$ is smaller than those obtained for $N_p = 2$. Similarly, prediction horizon length of 10 and the same weight matrices Q and R are considered and solving the mp-QP problem produces 374 controllers. The polyhedral partition and the gain margin set obtained for these configurations are illustrated in Figure 3.16. We use Table 3.2 to provide information about the number of vertices in the set $\mathcal{R}_i$ and the computation time taken to obtain the gain

margin sets for different prediction horizons. Finally, we simulate for the state trajectories and control inputs, for the controller obtained with $N_p = 5$, with different gain margins for the initial state $x(0) = [2\ 3\ 2]^T$. The state trajectories and the control input simulations for different gain margins are presented using the Figure 3.17.

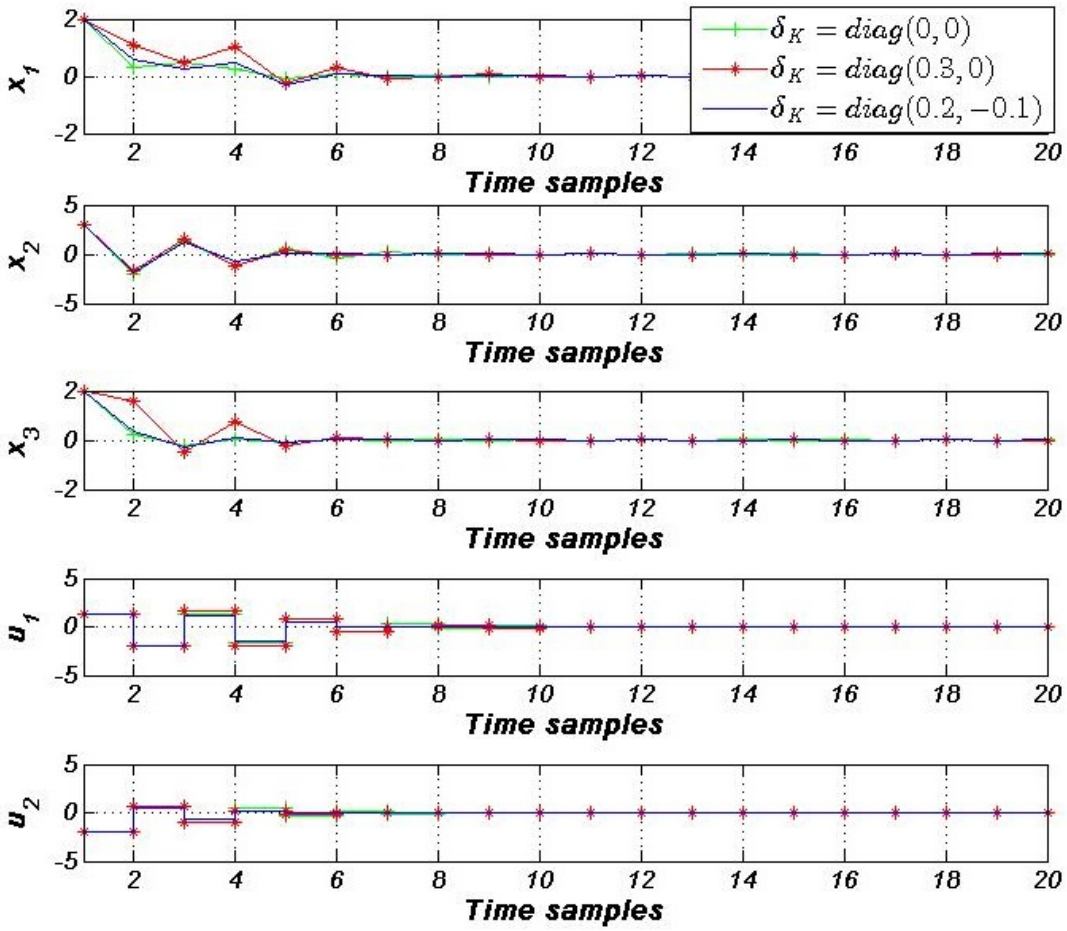


Figure 3.17: State trajectories and control input simulation for $N_p = 5$ for different gain margins

3.4 First order neglected dynamics

In the following, we analyze the robustness of the nominal PWA controller defined over the polyhedral set $\mathcal{R}$ affected by a first order neglected dynamics in closed loop. Our approach is to present an admissible set for the variations of the neglected parameters or variables in the dynamical model assuring the invariance of the set $\mathcal{R}$.

3.4.1 Problem Formulation

In this section, a problem is formulated for the first order dynamics that perturb the closed loop system as shown in Figure 3.18. The nominal linear discrete-time

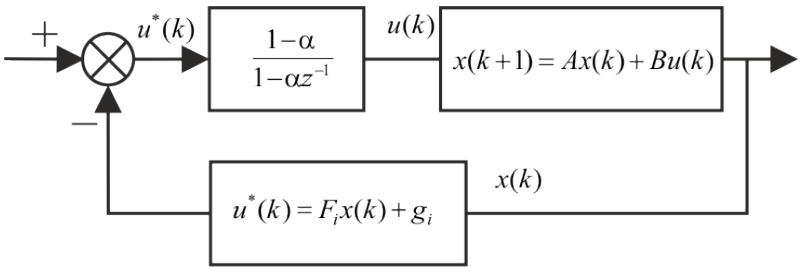


Figure 3.18: Closed loop system with first order neglected dynamics

system and the nominal feedback PWA control law can be written in the form,

$$x(k+1) = Ax(k) + Bu(k) \tag{3.31a}$$
$$u_{pwa}(x(k)) = u^*(k) = F_i x(k) + g_i \tag{3.31b}$$

The neglected dynamics can be represented by a first order equation and it can be written in the form,

$$x_I(k+1) = \alpha x_I(k) + (1-\alpha)u^*(k) \tag{3.32a}$$
$$u(k) = \alpha x_I(k) + (1-\alpha)u^*(k) \tag{3.32b}$$

The neglected dynamics in closed loop model can be described by,

$$x(k+1) = Ax(k) + Bx_I(k+1) \tag{3.33a}$$
$$x(k+1) = Ax(k) + \alpha B x_I(k) + B(1-\alpha)u^*(k) \tag{3.33b}$$

The augmented model is

$$x_e(k+1) = A_e x_e(k) + B_e u(k) \tag{3.34a}$$
$$y_e(k) = C_e x_e(k) \tag{3.34b}$$

and,

$$x_e = \begin{bmatrix} x \\ x_I \end{bmatrix}, A_e = \begin{bmatrix} A & \alpha B \\ 0_{1\times n} & \alpha \end{bmatrix}$$
$$B_e = \begin{bmatrix} B(1-\alpha) \\ 1-\alpha \end{bmatrix}, C_e = [C, 0] \tag{3.35}$$

where $x_e \in \mathbb{R}^{n_e}$, $A_e \in \mathbb{R}^{n_e \times n_e}$ and $B_e \in \mathbb{R}^{n_e \times m}$. In this study, we consider a single-input single-output system and therefore $n_e = n + 1$ and $m = 1$.

Starting from a piecewise affine control (3.31b) defined over the polyhedral partition of the set $\mathcal{R}$ with $\mathcal{R}_i \in \mathcal{R}\ \forall i \in \mathcal{I}_N$ for the linear discrete time system (3.31a)

the aim is to investigate the robustness of the control synthesis (3.31b) for the system (3.34) and (3.35) affected by the neglected dynamics.

The objective being to find the largest set of α assuring the invariance of the set $\mathcal{R}$, first we construct vertices for the augmented state with the help of the vertices in the set $\mathcal{R}_i$ $\forall i \in \mathcal{I}_N$. Second, we analyze the robustness of the control synthesis (3.31b) with respect to the invariance characteristics for the set $\mathcal{R}$.

3.4.2 Construction of vertex representation for the augmented state

Before entering into the technical details of the construction, we discuss about the extension of extreme point (vertices) for the augmented state (3.34).
Recall that in the previous subsection the vertices of the set $\mathcal{R}$ and $\mathcal{R}_i$ defined by the column vectors of matrices (3.2) and (3.4) were instrumental in the construction of the robustness analysis. The piecewise affine control law $u_{pwa}(x(k))$ for the column vectors of the matrix V is given by the matrix U (3.5) in the nominal case.

The construction of vertices for the augmented state x_I (which includes the neglected dynamics) can be done in two ways. Recall that the augmented state correspond to the past input $u(k-1)$ for $\alpha = 0$.

- Method 1: Initialize the values of the state $x_I(k)$ for each vertex corresponding to the column vectors of matrix V with the value given by the corresponding nominal piecewise affine control law. For instance, $w_{I_p} = u_{pwa}(w_p)$ $\forall p \in \mathcal{I}_p$.

- Method 2: Initialization of the state $x_I(k)$ for each vertex corresponding to the column vectors of matrix V can be carried out with the help of minimum and maximum values of the control input.

Method 2 produces twice the number of vertices as compared to method 1. For each column vector of matrix V, we obtain two different vertices i.e., the values of augmented state vector x_e for each non-identical vertices in $\mathcal{R}_i$ is $[w_p, u_{min}]$ and $[w_p, u_{max}]$ $\forall p \in \mathcal{I}_p$. In the following, we investigate the robustness problem for method 1.

3.4.3 Controller for extended state system

The controller for the extended model (3.34) can be represented by,

$$u^e_{pwa}(x_e(k)) = F^e_i x_e(k) + g^e_i \tag{3.36}$$

where $F^e_i = [F_i \; 0] \in \mathcal{R}^{m \times n_e}$ and $g^e_i = g_i \in \mathcal{R}^{m \times 1}$.

This is just an extension of (3.31b) to accommodate the extended state and therefore the characteristics of the original controller (3.31b) are preserved.

3.4.4 Admissible set for the first order neglected dynamics:

We introduce few matrices which will be used in the construction.
The extended model (3.34)-(3.35) can be written as a convex combination of the extreme realizations obtained with $\alpha = 0$ and $\alpha = 1$.

For $\alpha = 0$,

$$A_1^e = \begin{bmatrix} A & 0_{n\times m} \\ 0_{m\times n} & 0_{m\times m} \end{bmatrix}, \quad B_1^e = \begin{bmatrix} B \\ 1_{m\times m} \end{bmatrix}. \tag{3.37}$$

For $\alpha = 1$,

$$A_2^e = \begin{bmatrix} A & B \\ 0_{m\times n} & 1_{m\times m} \end{bmatrix}, \quad B_2^e = \begin{bmatrix} 0_{n\times m} \\ 0_{m\times m} \end{bmatrix}. \tag{3.38}$$

Therefore,

$$A_e = (1-\alpha)A_1^e + \alpha A_2^e,\ B_e = (1-\alpha)B_1^e + \alpha B_2^e. \tag{3.39}$$

Since our objective follows the computation of all the values of α assuring the invariance of the set $\mathcal{R}$, we need to analyze the state and input values in order to evaluate for the state vector $x(k+1)$ in (3.33).

$$A_1 = \begin{bmatrix} A & 0_{n\times m} \end{bmatrix} \in \mathbb{R}^{n\times n_e}, A_2 = \begin{bmatrix} A & B \end{bmatrix} \in \mathbb{R}^{n\times n_e}, B_1 = B, \text{ and } B_2 = [0_{n\times m}]. \tag{3.40}$$

Similar to (3.4), we introduce a matrix $V_e^{(1)} \in \mathbb{R}^{n_e\times p}$ which stores the vertices of the state vector $x(k)$ within the given partition and the PWA control input associated with it for all the non-identical vertices in $\mathcal{R}_i \in \mathbb{R}^n$, $i \in \mathcal{I}_N$ in order to create a matrix that contains the vertices for the extended state vector x_e.

$$V_e^{(1)} = [V, U] \tag{3.41}$$

In a second step, the values of the state vector $x_e(k+1)$ are found by exploiting (3.41) and (3.36) for column vectors of the matrix $V_e^{(1)}$ in a closed loop formulation. This leads to the matrix $V_e^{(2)} \in \mathbb{R}^{n_e\times p}$.

$$V_e^{(2)} = A_1^e V_e^{(1)} + B_1^e U \tag{3.42}$$

Similar to the notation in (3.5), a PWA image for the matrix $V_e^{(2)}$ can be found with the control law (3.36) and stored in a matrix $U_e^{(2)} \in \mathbb{R}^{m\times p}$.

$$U_e^{(2)} = u_{pwa}^e[V_e^{(2)}] \tag{3.43}$$

Definition 3.4.1 *Consider a linear discrete time system affected by the first order neglected dynamics (3.34)-(3.35) and stabilized via a PWA state feedback control law (3.36). A margin for first order neglected dynamics is characterized by a set* $\Omega^{\alpha} \in \mathbb{R}^{m}$ *which contains the values of parameter* α *(for each input channel) such that the invariance property of the set* $\mathcal{R}$ *is assured.*

Theorem 3.4.1 *Consider the extended system (3.34) subject to first order neglected dynamics stabilized by a piecewise affine control law (3.36). The admissible set of parameters for the neglected dynamics is given as the projection,*

$$\Omega^{\alpha} = Proj_{\alpha}\mathcal{T} \tag{3.44}$$

where $\mathcal{T}$ *denotes the polyhedral set:*

$$\mathcal{T} = \left\{ \begin{array}{l} (\alpha, \Gamma) \in \mathbb{R} \times \mathbb{R}^{r\times p} | \;\; (1-\alpha)(A_1 V_e^{(2)} + B_1 U_e^{(2)}) + \\ \alpha(A_2 V_e^{(2)} + B_2 U_e^{(2)}) = W\Gamma | \Gamma = [\gamma_1, \cdots, \gamma_p], \gamma_i \in \mathcal{S}_r, \forall i \in \mathcal{I}_p \end{array} \right\} \tag{3.45}$$

Proof 3.4.1 Ω^{α} *describes admissible set for neglected dynamics if* $\forall \alpha \in \Omega^{\alpha}$ *and selection of* $(x_e(0), u_e(0))$ *we have* $((1-\alpha)A_1 + \alpha A_2)x_e(k) + ((1-\alpha)B_1 + \alpha B_2)u^e_{pwa} \in \mathcal{R}$, $\forall k \in \mathbb{N}$.

We remark that the computation of the admissible set Ω^{α} *corresponds to the variations between the nominal model (3.31a) and the model affected by the neglected dynamics (3.34). In an equivalent form,*

$$\begin{array}{r} (1-\alpha)(A_1 x_e(k) + B_1(F_i^e x_e(k) + g_i^e)) + \\ \alpha(A_2 x_e(k) + B_2(F_i^e x_e(k) + g_i^e)) \in \mathcal{R} \end{array} \tag{3.46}$$

By replacing $x^e(k)$ *with the column elements of the matrix* $V_e^{(2)}$ *and similarly the PWA function* $u^e_{pwa} = F_i^e x^e(k) + g_i^e$ *with the column vectors of matrix* $U_e^{(2)}$*. Equation (3.46) is further modified into:*

$$\begin{array}{r} (1-\alpha)(A_1 V_e^{(2)}(:,q) + B_1 U_e^{(2)}(:,q)) + \\ \alpha(A_2 V_e^{(2)}(:,q) + B_2 U_e^{(2)}(:,q)) \in \mathcal{R} \end{array} \tag{3.47}$$

For $q \in \mathcal{I}_p$*, by representing the state vector* $x \in \mathcal{R}$ *as the convex combination of column vector of the matrix* W (3.2)*, one can write* $\sum_{l=1}^{r} \gamma_l W(:,l)$ *and* $\sum_{l=l}^{r} \gamma_l = 1$ *and introducing this term in (3.47) obtain*

$$\begin{array}{r} (1-\alpha)(A_1 V_e^{(2)}(:,q) + B_1 U_e^{(2)}(:,q)) + \\ \alpha(A_2 V_e^{(2)}(:,q) + B_2 U_e^{(2)}(:,q)) = W\gamma \end{array} \tag{3.48}$$

with a solution vector $\gamma = [\gamma_1, \cdots, \gamma_r] \in \mathbb{R}^r$.

If (3.48) holds for one column vector of matrix $V_e^{(2)}$*, then it holds for all the column vectors of the matrix* $V_e^{(2)}$ *and with the image of the piecewise affine control law defined by the matrix* $U_e^{(2)}$.

Equation (3.48) can be written as,

$$(1-\alpha)(A_1 V_e^{(2)} + B_1 U_e^{(2)}) + \alpha(A_2 V_e^{(2)} + B_2 U_e^{(2)}) = W\Gamma \tag{3.49}$$

with $\Gamma \in \mathbb{R}^{r\times p}$ *and the proof is completed.*

3.4.5 Example

Consider the discrete-time linear system given by,

$$x(k+1) = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} x(k) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(k),$$

$$y(k) = [1\ 0].$$

The output variable and control input are subject to the following polytopic constraints $-5 \leq y(k) \leq 5$ and $-1 \leq u(k) \leq 1$. The weight matrices on the states and control input considered in the MPC problem are given by $Q = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$ and $R = 0.1$ respectively. The prediction horizon chosen for this example is 2 and a positively invariant set as a terminal constraint set is added to the optimization formulation.

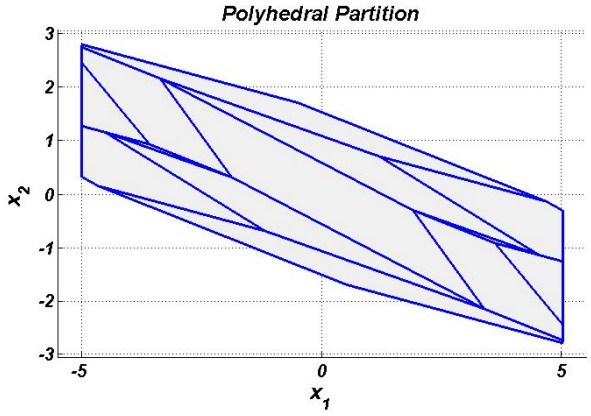


Figure 3.19: Representation of polyhedral partition with 13 regions.

The resulting mp-QP solution consists of 13 PWA control laws defined over the state-space partition. Figure 3.19 shows the state-space partition of 13 regions.

The robustness margin for the first order neglected dynamics is represented by the set $\Omega^{\alpha} = [0,\ 0.3255]$. The invariance property of the controller is preserved for any input filter with parameters in these values.

Figure 3.20 illustrates the state trajectories for the system (3.39) with $\alpha = 0.325$. The state trajectories for all the vertices are strictly inside the set $\mathcal{R}$ thus proving the Theorem 3.4.1. The states and control input simulations with $\alpha = 0$, $\alpha = 0.15$ and $\alpha = 0.3$ for the same initial condition $x_0 = [-5,\ 2.73]^T$ are presented in the Figure 3.21.

3.5 Conclusion

In this work the gain margin set assuring the invariance of the closed-loop for a linear discrete time system controlled by a piecewise affine control law has been computed.

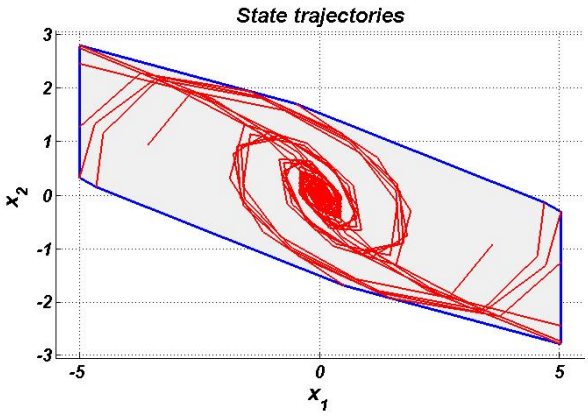


Figure 3.20: State trajectories for $\alpha = 0.325$.

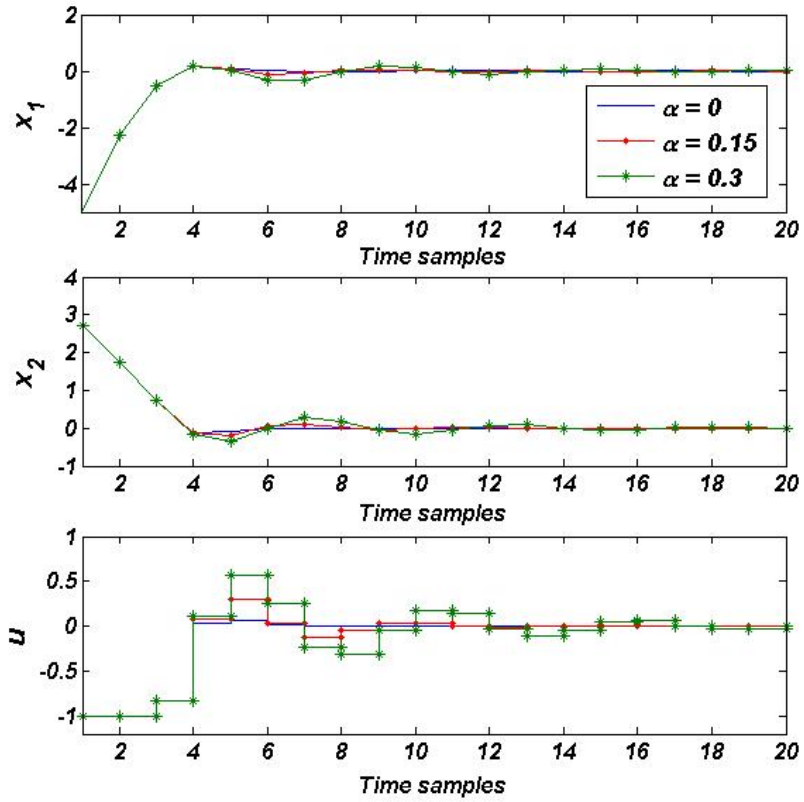


Figure 3.21: Simulations of state trajectories and control input for different α values.

For a discrete time system affected by first order neglected dynamics, a robustness margin has been also deduced assuring the invariance property. It is worth to be mentioned that the proposed analysis procedure does not extend to analysis of the convergence (asymptotic stability properties). For the systems affected by time-varying parametric uncertainty, it is important to analyze the convergence properties of the controller. This can be done by considering a contractive set as feasible domain. In the next chapter the robustness margin for contractive PWA controller for discrete-time systems influenced by parametric uncertainty is analyzed.

Chapter 4

Robustness margins for contractive piecewise affine explicit control laws

Contents

4.1 Introduction 47
4.2 Control design for controlled λ-contractive set 48
4.3 Main Results 52
4.3.1 Parametric margin for λ-controlled contractive sets 52
4.3.2 Gain margin for λ-controlled contractive sets 55
4.3.3 Neglected dynamics margin for λ-controlled contractive sets 56
4.4 Examples 57
4.4.1 Robustness parametric margin 57
4.4.2 Gain Margin 61
4.4.3 Robustness Margin for First order neglected dynamics . . 62
4.5 Conclusions 63

4.1 Introduction

In the context of stability of the model predictive control, the terminal cost function and the terminal set are widely adopted in the problem formulation of the MPC at the design stage to ensure the state trajectories converge to the origin. A remarkable study on the control invariant characterization of linear systems is proposed in [DH99]. In this reference, the construction of successively tighter outer approximations for controlled λ-contractive set is obtained via an iterative algorithm. This approach was further extended to construct a non-iterative controlled contractive set based on some conservative assumptions [HOB14]. In this chapter, we advocate

the use of a controlled λ-contractive set that guarantees contractivity for each time step of the closed loop.

Starting from these framework, from the analysis point of view, it is important to commensurate the capacity of the control law to cope with disturbances, neglected dynamics or uncertain parameters. This characteristic is denoted in control theory as robustness of the controller.

It is worth to be mentioned that the analysis of a nominal PWA control and its retuning is essentially different approach from a robust control design. It is known that a robust PWA control as for example robust explicit predictive control synthesis [PFKP09] can account for uncertainties based on dynamic programming approach but the associated computational complexity is exponential with respect to the nominal case. The same thing can be said about the robust explicit model predictive control with contractive set based on variable-structure control law for linear polytopic uncertain system as presented in [YBHJ03].

In this chapter, a multi-parametric quadratic programming problem is formulated in order to compute a PWA control law which enforces contractivity for the class of linear discrete-time systems. Starting from the λ-contractive control law, we present three robustness margins for the system stabilized by a contractive piecewise affine control law. First, the robustness margin for a discrete-time system affected by polytopic uncertainty is presented. The robustness margin denoted Ω_{rob} is defined as a subset of the parametric uncertainty set Ω and is shown to take the form of polyhedral set. For all the models belonging to the polyhedral set Ω_{rob}, the contractivity of the state trajectories is guaranteed in the presence of time-varying uncertainties. Second, we propose a numerical method to compute a gain margin set for a discrete-time system stabilized by a contractive PWA affine control. The desired gain margin takes the form of a set which characterizes admissible variations of system gains preserving the contractive characteristics of the controller. Third, we compute an admissible set which characterizes the parameter variations for the discrete-time system affected by first order neglected dynamics. For all the variations in this polyhedral set the contractive properties of the controller is preserved.

4.2 Control design for controlled λ-contractive set

Consider the discrete-time linear time-invariant system given by,

$$x(k+1) = Ax(k) + Bu(k). \tag{4.1}$$

Considering the class of PWA feedback laws in the large, we are interested in defining the notion of λ-contractiveness.

Definition 4.2.1 *A $\mathcal{C}$-set $\mathcal{P} \subseteq \mathcal{R}$ is called controlled λ-contractive with contraction factor $\lambda \in$ [0,1) if there exists an admissible control law $u_{pwa}(x(k))$ such that $\forall x(k) \in \beta\mathcal{P}$ then $Ax(k) + Bu_{pwa}(x(k)) \in \lambda\beta\mathcal{P}$, $\forall \beta \in [0,1]$ [HOB14].*

The controlled λ-contractive set represents an important notion that can be employed for stabilizing a constrained discrete-time linear systems. The properties of λ-contractive sets are enhanced versions of the positively invariant set. The polytopes generated from the contractive set are simpler than the reachable set and provide compact representations for the optimization based control design [HOB14].

In the context of constrained control, in which a contractive PWA control law stabilizes for the full range of state space, the sequence of control actions is given for each vertex, thus steering the extreme vertex of the contractive set polytopes to a strict subset of the contractive set. The contractive set $\mathcal{P} \subseteq \mathcal{R}$ recursively shrinks for each time step.

In this section, we discuss control design based on the explicit MPC law, but with an additional λ-contractive constraint imposed on the problem formulation. To facilitate an explicit control with controlled λ-contractive set, it is desirable to compute an initial contractive set that does not violate the original state constraints.

The controlled contractive sets of the shape specified in [HOB14] are considered. To calculate an initial set, let us consider the system (4.1) and the state constraints, with a matrix $A = V_A D_A V_A^{-1}$. Here V_A and D_A denote the matrices of eigenvectors and diagonal matrix with diagonal entries being the real eigenvalues of A corresponding to the Jordan decomposition of matrix A. The obtained initial set is symmetric and can be represented as,

$$\mathcal{P} = \left\{ x : \begin{bmatrix} V_A^{-1} \\ -V_A^{-1} \end{bmatrix} x \leq \begin{bmatrix} b_x \\ b_x \end{bmatrix}, V_A \in \mathbb{R}^{n \times n}, b_x \in \mathbb{R}^n \right\}. \tag{4.2}$$

Imposing contractiveness does not inherently require the set to be described in the form (4.2), this being a particular choice used in the present work to find a particularly simple controlled contractive set. The quadratic cost function for the controller stabilizing a linear discrete-time system given by (4.1) subject to constraints is formulated as,

$$J(x(k), U^*) = \min_{U^*} \sum_{i=1}^{N_p} ||Qx(k+i)||_2^2 + \sum_{i=0}^{N_p-1} ||Ru(k+i)||_2^2 \tag{4.3a}$$

$$s.t. \quad x(k+1) = Ax(k) + Bu(k), \;\; k = 0, \cdots, N_p - 1 \tag{4.3b}$$

$$x_0 = x(0) \tag{4.3c}$$

$$\begin{bmatrix} V_A^{-1} \\ \\ -V_A^{-1} \end{bmatrix} x(k+1) \leq \lambda\beta \begin{bmatrix} b_x \\ \\ b_x \end{bmatrix} \tag{4.3d}$$

$$u(k) \in \mathcal{U}, \; k = 0, \cdots, N_p - 1 \tag{4.3e}$$

$$x(k) \in \mathcal{X}, \; k = 1, \cdots, N_p \tag{4.3f}$$

Here λ is a pre-defined contractive factor, $\lambda \in [0, 1)$ and $\beta \in [0, 1]$ will be considered as a parameter. The weight matrices $Q = Q^T \geq 0$ and $R = R^T > 0$ are positive semi-definite and positive definite respectively which define the performance index of the optimization problem (4.3) following the classical predictive control design [MRRS00].

In (4.3a) we use an optimization criterion spanning over a multiple-time step horizon, while the contraction is only imposed for the first time step. The lack of supplementary state constraints is related to the fact that the contractive set constraint in (4.3d) should be designed to make (4.3f) redundant.

Now, one has to transform the problem (4.3) into a multi-parametric programming problem including the full vector of parameters. This complete mp-QP problem is formulated for the state vector x and β as an augmented parameter vector $\hat{x} = [x, \beta]^T$ leading to the compact cost formulation:

$$J(x(k), U^*) = \min_{U^*} \ \frac{1}{2} U^{*T} H U^* + \hat{x}^T E^T U^* + \frac{1}{2} \hat{x}^T Y \hat{x}. \tag{4.4}$$

The constraints for the state and input variables can be appended to the inequality constraint given below,

$$GU^* \leq D + S\hat{x}. \tag{4.5}$$

In (4.4) and (4.5), $\hat{x} = \hat{x}(k)$. Subsequently, the initial contractive set given in (4.2) is also extended to $\hat{x}$ space, by setting the bounds of β parameter between 0 and 1. This set is herewith denoted as $\hat{\mathcal{P}} \in \mathbb{R}^{n+1}$. It inherits the polyhedral structure and will be included in the problem formulation via a set of linear constraints with a linear dependence of the right hand side on the extended parameter vector $\hat{x}$.

The contractive set (4.3d) represented by the inequality constraints can be written within (4.5):

$$G = \begin{bmatrix} \begin{bmatrix} V_A^{-1} \\ \\ -V_A^{-1} \end{bmatrix} B, \begin{bmatrix} 0 \\ 0 \end{bmatrix}_{2n \times N_p - 1} \end{bmatrix},$$

$$S = \begin{bmatrix} \begin{bmatrix} -V_A^{-1} \\ \\ V_A^{-1} \end{bmatrix} A, \lambda \begin{bmatrix} b_x \\ \\ b_x \end{bmatrix} \end{bmatrix} \text{ and}$$

$$D = \begin{bmatrix} 0 \\ \\ 0 \end{bmatrix}_{2n \times 1}$$

From the problem formulation it can be noted that the state vector x and β are the new parameters of the mp-QP problem. The state space partition obtained from the problem (4.4) and (4.5) can be represented as a finite collection of regions in the extended $[x, \beta]$ space. This will be denoted next as

$$\hat{\mathcal{R}} = \cup_{i \in \mathcal{I}_{\hat{N}}} \hat{\mathcal{R}}_i \tag{4.6}$$

The control law obtained from the mp-QP formulation is given by,

$$\hat{u}_{pwa}(\hat{x}(k)) = \hat{F}_i \hat{x}(k) + \hat{g}_i, i \in \mathcal{I}_{\hat{N}} \tag{4.7}$$

In Eq (4.6)-(4.7), $\hat{N}$ represents the number of controllers.

In the following, we further explore the parametric dependence on β by exploiting the extended vector $[x, \beta]$-space and reduce it to the initial x-space by preserving the

piecewise affine formulation. This can be done with a particular choice of β which can be interpreted as an implicit function of x using in practice the Minkowski function $\mathcal{M}_p$ with respect to the initial contractive set $\mathcal{P}$. We recall here some of the basic properties.

Now, introduce a subset $\mathcal{H} \subset \mathbb{R}^{n+1}$ defined as the graph of the Minkowski function with respect to the set $\mathcal{P} \subset \mathbb{R}^n$.

$$\mathcal{H} = \left\{ \begin{bmatrix} x \\ \beta \end{bmatrix} : \mathcal{M}_{\mathcal{P}}(x) = \beta \right\} \tag{4.8}$$

The PWA function (4.7) can be restricted to the subset $\mathcal{H} \cap \hat{\mathcal{R}}$ and subsequently projected onto the original state space. This results in an explicit PWA function:

$$u_{pwa}(x(k)) = F_i x(k) + g_i, i \in \mathcal{I}_N, \text{for } x(k) \in \mathcal{R}_i \tag{4.9}$$

defined over a partition $\mathcal{R}$, $\mathcal{R} = \bigcup_{i \in \mathcal{I}_N} \mathcal{R}_i$. This design procedure is summarized in Algorithm 4.2.1.

Algorithm 4.2.1 Algorithm for the control law in x space

Input: $\hat{\mathcal{R}} \in \mathbb{R}^{n+1}$, $\mathcal{H} \in \mathbb{R}^{n+1}$, $\hat{F}_j \in \mathbb{R}^{m \times n+1}$, $\hat{g}_j \in \mathbb{R}^m$
Output: $\mathcal{R} \in \mathbb{R}^n$, $F_i \in \mathbb{R}^{m \times n}$, $g_i \in \mathbb{R}^m$
1: *Initialization* : $i = 0$
2: Obtain the polyhedral regions,
3: $\hat{\mathcal{R}} = \bigcup_{j \in \mathcal{I}_{\hat{N}}} \hat{\mathcal{R}}_j$, $\hat{\mathcal{R}}_j \in \mathbb{R}^{n+1}$
4: **for** $j = 1$ to $\hat{N}$ **do**
5: $P_{int} = \hat{\mathcal{R}}_j \bigcap \mathcal{H}$.
6: $P_{proj} = Proj(P_{int}, 1:n)$
7: **if** $(\dim(P_{proj} == n))$ **then**
8: $i = i + 1$
9: $\mathcal{R}_i = P_{proj}$
10: $F_i = \hat{F}_j(:, 1:n) + \mathcal{M}_{\mathcal{P}}(x) * \hat{F}_j(:, n+1)$
11: $g_i = \hat{g}_j$
12: **end if**
13: **end for**

The piecewise affine control law obtained for (4.4) and (4.5) $\hat{u}_{pwa}(\hat{x}(k)) = \hat{F}_j \hat{x}(k) + \hat{g}_j$, $j \in \mathcal{I}_{\hat{N}}$ is modified to $u_{pwa}(x(k)) = F_i x(k) + g_i$, $i \in \mathcal{I}_N$ as given in Algorithm 4.2.1. The projected polyhedron $\mathcal{R}$, $\mathcal{R} = \bigcup_{i \in \mathcal{I}_N} \mathcal{R}_i$ is obtained from the Algorithm.

Before entering into the robustness-related developments, let us explore that the projected polyhedron satisfies the following properties:
i) $\mathcal{R} = \cup_{i \in \mathcal{I}_N} \mathcal{R}_i$,
ii) $int(\mathcal{R}_i) \neq \emptyset, \forall i \in \mathcal{I}_N$,
iii) $int(\mathcal{R}_i) \cap int(\mathcal{R}_j) = \emptyset, \forall (i,j) \in \mathcal{I}_N^2$, and $i \neq j$.

4.3 Main Results

4.3.1 Parametric margin for λ-controlled contractive sets

In this section, we consider parametric uncertainty on the A and B matrices of the system model (4.1). A set Ω is introduced in the parameter space,

$$\Omega = Conv\{[A_1\ B_1] \cdots [A_L\ B_L]\}. \tag{4.10}$$

The nominal system is given by a convex combination,

$$[A\ B] = \sum_{i=1}^{L} \zeta_i [A_i\ B_i] \tag{4.11}$$

where $\zeta = [\zeta_1, \cdots, \zeta_L]^T \in \mathcal{S}_L$ and ζ_i is a non-negative scalar.

Definition 4.3.1 *Consider a polytopic set Ω in the model parameter space, the robustness margin problem is to compute the largest subset $\Omega_{rob} \subset \Omega$ for a given PWA control law $u_{pwa}(x(k))$ defined over the polyhedral set $\mathcal{R}$ such that this set is controlled λ-contractive with the system model (4.1).*

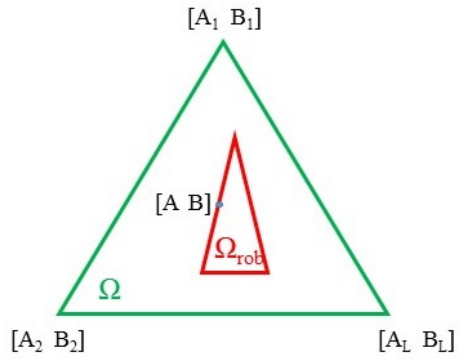


Figure 4.1: Illustrative representation of the sets Ω and Ω_{rob}.

The robustness margin for the projected polyhedral set $\mathcal{R}$ can be constructed by using the vertex or half-space representation. In this work we focus on the vertex representation and start by recalling the nominal system (4.1) subject to the constraint sets, $\mathcal{X}$ and $\mathcal{U}$, used for the design of a contractive piecewise affine control law.

The matrices $[A\ B]$ belong to the polytopic uncertain set Ω as defined in (4.10). Following the Definition 4.3.1, the robustness margin problem is to compute the subset $\Omega_{rob} \subset \Omega$ such that the closed loop dynamics obtained with the PWA control law defined over $\mathcal{R}$ is λ-contractive, that is:

$$x(k+1) = (A + BF_i)x(k) + Bg_i \in \lambda\beta\mathcal{R},\ \forall x(k) \in \beta\mathcal{R} \tag{4.12}$$

and $i \in \mathcal{I}_N$.
Let us define few important matrices which will be used in the following.

Recall the matrix V defined in (3.4) which stores all the non-identical vertices of the $\mathcal{R}_i$ with a cardinal number p. The value of parameter β is computed for each column vector in the matrix V,

$$\beta_i(w_i) = \mathcal{M}_{\mathcal{P}}(w_i), i \in \mathcal{I}_p. \tag{4.13}$$

It is possible to express (4.12) as a convex combination of the vertices of the polyhedral set $\mathcal{R}$,

$$\sum_{j=1}^{r} \gamma_j v_j, \gamma = [\gamma_1, \cdots, \gamma_r]^T \in \mathcal{S}_r, \mathbf{1}^T\gamma = 1.$$

Rewriting (4.12) we obtain:

$$(A + BF_i)x(k) + Bg_i = \lambda\beta(x(k))W\gamma. \tag{4.14}$$

Replacing (4.13) within (4.14) one can compute for each column vector in the matrix V with the corresponding vector γ_i. Storing the column vectors $\gamma_i, i = 1, \ldots, p$, a matrix $\Gamma \in \mathbb{R}^{r\times p}$ will be obtained

$$\Gamma = [\beta_1\gamma_1, \beta_2\gamma_2, \cdots, \beta_p\gamma_p]. \tag{4.15}$$

Finally, after defining a matrix $M \in \mathbb{R}^{n\times r}$ as a simple scaling of the vertices of the feasible set $M = \lambda W$ one can state the main result with respect to the robustness margin characterization.

Theorem 4.3.1 *Consider a discrete-time system (4.1) subject to a polytopic uncertainty and subject to the states and input constraints. The robustness margin for a given contractive piecewise affine control law is given by*

$$\Omega_{rob} = Proj_{\mathcal{S}_L}\mathcal{T} \tag{4.16}$$

where $\mathcal{T}$ represents the polyhedral set,

$$\mathcal{T} = \left\{ \begin{matrix} \{(\zeta, \Gamma) \in \mathcal{S}_L \times \mathbb{R}^{r\times p} | \mathbf{1}^T\Gamma = [\beta_1, \cdots, \beta_p], \\ \\ \sum_{j=1}^{L} \zeta_j(A_jV + B_jU) = M\Gamma\}. \end{matrix} \right\}. \tag{4.17}$$

Proof 4.3.1 *To prove the existence of robustness margin for the polyhedral set $\mathcal{R}$ whose control law is associated with controlled λ-contractive set, let us consider the closed loop formulation of the piecewise affine control law with the λ-contraction.*

$$(A + BF_i)x + Bg_i \in \lambda\beta\mathcal{R} \tag{4.18}$$

where λ denotes the contractive factor and the parameter $\beta(x) = \mathcal{M}_{\mathcal{P}}(x)$. We recall that $\forall (A, B) \in \Omega_{rob}, \forall x \in \mathcal{R}_i$, $\forall i \in \mathcal{I}_N$ and considering the polytopic uncertainty set Ω we can show that $\Omega_{rob} \subset \Omega$. And clearly, (4.18) can be written as

$$\sum_{j=1}^{L} \zeta_j (A_j + B_j F_i)x + \zeta_j B_j g_i \in \lambda\beta\mathcal{R}. \tag{4.19}$$

Now, simply the state vector, $x \in \mathcal{R}_i$, can be expressed as a convex combination of the vertices, $x = \sum_{l=1}^{r_i} \beta_l w_{il}$ for $\sum_{l=1}^{r_i} \beta_l = 1$, with ζ_j the elements of a vector $\zeta \in \mathcal{S}_L$, $\forall i \in \mathcal{I}_N$ and $\forall l \in \mathcal{I}_{r_i}$.

Subsequently replacing x with w_{il}, the β parameter for the vertex w_{il} is computed by $\beta_{il} = \mathcal{M}_{\mathcal{P}}(w_{il})$. Equivalently (4.19) is followed by,

$$\sum_{j=1}^{L} \zeta_j (A_j + B_j F_i)w_{il} + \zeta_j B_j g_i \in \lambda\beta_{il}\mathcal{R}. \tag{4.20}$$

Moreover, the inclusion can be explicitly described by the existence of $y_{il} \in \lambda\beta_{il}\mathcal{R}$ such that:

$$\sum_{j=1}^{L} \zeta_j (A_j + B_j F_i)w_{il} + \zeta_j B_j g_i = y_{il}. \tag{4.21}$$

The vector $y_{il} \in \mathbb{R}^n$ can be expressed as,

$$y_{il} = \lambda\beta_{il}[\mathcal{V}(\mathcal{R})]\gamma_{il} \quad for \;\; \gamma_{il} \in \mathcal{S}_r. \tag{4.22}$$

Substituting (4.22) in (4.21) and introducing a matrix $M \in \mathbb{R}^{n\times r}$, where $M = \lambda W$ we obtain,

$$\sum_{j=1}^{L} \zeta_j (A_j + B_j F_i)w_{il} + \zeta_j B_j g_i = M\beta_{il}\gamma_{il}. \tag{4.23}$$

From (4.23) it can be stated that if it holds $\forall l \in \mathcal{I}_{r_i}$, consequently it will hold for all the columns of the matrix V as given in (3.4). Exploiting the admissible input mapping of the columns of V as in U leads to the matrix formulation,

$$\sum_{j=1}^{L} \zeta_j A_j V + \zeta_j B_j U = M\Gamma. \tag{4.24}$$

It can be noticed that $\gamma_{il} \in \mathcal{S}_r$, that is, each column of matrix Γ is restricted to the simplex $\mathcal{S}_r$ multiplied by β_i, $\forall i \in \mathcal{I}_p$. The above derivations prove that the polyhedral set $\mathcal{T}$ in (4.17) represents a parametrized set of all the model uncertainties guaranteeing the controlled λ-contractivity of the closed loop for the dynamical system affected by uncertainties. Further the polyhedral set $\mathcal{T}$ is projected on the simplex function $\mathcal{S}_L$.

4.3.2 Gain margin for λ-controlled contractive sets

In the following, we describe an analysis of the gain margin set for the system (4.1) stabilized with the help of a state feedback contractive PWA control law. The construction of the gain margin is similar with the one presented in the previous section and the proofs will be omitted.

Definition 4.3.2 *Consider a discrete time linear system (4.1) with a continuous contractive PWA control law, such that the state space set $\mathcal{R}$ is controlled λ-contractive. The* Gain Margin *is represented by the set $\mathcal{K} \subset \mathbb{R}^m$, such that $x(k+1) = Ax(k) + B(I_m + diag(\delta_K))u_{pwa}(x(k)) \in \lambda\beta\mathcal{R}, \forall x(k) \in \mathcal{R}$, $\delta_K \in \mathcal{K} \subset \mathbb{R}^m$ and $\lambda, \beta \in [0,1]$.*

where λ denotes the contractive factor and the parameter $\beta(x) = \mathcal{M}_{\mathcal{P}}(x)$. The set $\mathcal{K} \subset \mathbb{R}^m$ is a set which contains the input channels gain variations δ_K such that for any point inside the set $\mathcal{K}$, the λ-contractive characteristics of the set $\mathcal{R}$ is preserved.

Theorem 4.3.2 *Consider a discrete-time linear system (4.1) with a contractive piecewise affine state feedback control. The gain margin $\mathcal{K}$ of the controller is defined by the set,*

$$\mathcal{K} = \cap_{q=1}^{p} \mathcal{K}_q, \tag{4.25}$$

where $\mathcal{K}$ represents the gain margin set and $\mathcal{K}_q$ the local gain margin for the vertex w_q for some $q \in \mathcal{I}_p$.

$$\mathcal{K}_q = \{z \in \mathbb{R}^m | \exists u \in \Delta\mathcal{U}_q,\ Mz = u\} \tag{4.26}$$

with

$$\Delta\mathcal{U}_q = \{u\ | \hat{H}_u u \leq \hat{h}_u\} \tag{4.27}$$

where $\hat{H}_u \in \mathbb{R}^{d_u \times m}$ and $\hat{h}_u \in \mathbb{R}^{d_u}$. $\Delta\mathcal{U}_q$ represents the set of admissible input variation for the vertex w_q preserving the contractivity.

The collection of sets $\mathcal{K}_q$ are independent for each vertex of the set in $\mathcal{R}_i$ and the intersection of these independent set gives the global set $\mathcal{K}$. In order to compute $\mathcal{K}_q$, we first need to compute explicitly the sets $\Delta\mathcal{U}_q$.

Theorem 4.3.3 *Consider a linear discrete-time system (4.1) stabilized by a contractive piecewise affine control law. The set $\Delta\mathcal{U}_q$ of admissible input variations at the vertex w_q is obtained by*

$$\Delta\mathcal{U}_q = Proj_{\mathcal{U}} \mathcal{H}_q \tag{4.28}$$

$\mathcal{U} \in \mathbb{R}^m$ denotes the input constraint set. The polyhedral set $\mathcal{H}_q$ is described by:

$$\mathcal{H}_q = \left\{ (\delta_u, \gamma) \in \mathbb{R}^m \times \mathbb{R}^r,\ and\ [A\ B] \begin{bmatrix} x(k) \\ u_{pwa}(x(k)) \end{bmatrix} + B\delta_u = \lambda\beta W\gamma, \mathbf{1}^T\gamma = 1 \right\} \tag{4.29}$$

Proof 4.3.2 *Let us recall the equation which resumes the contractive property for the set* $\mathcal{R}$*,*

$$[A\ B]\begin{bmatrix} x(k) \\ u_{pwa}(x(k)) \end{bmatrix} + B\delta_u \in \lambda\beta\mathcal{R} \tag{4.30}$$

Equation (4.30) can be expressed using the vertices of the set $\mathcal{R}$ *as convex combination and corresponding variables* $\gamma = [\gamma_1, \gamma_2, \cdots, \gamma_r]$ *and equality constraints* $\mathbf{1}^T\gamma = 1$ *such that*

$$[A\ B]\begin{bmatrix} x(k) \\ u_{pwa}(x(k)) \end{bmatrix} + B\delta_u = \lambda\beta W\gamma. \tag{4.31}$$

4.3.3 Neglected dynamics margin for λ-controlled contractive sets

In the section, we describe an analysis of the robustness of the nominal contractive PWA controller affected by a first order neglected dynamics in closed-loop formulation. Recall the augmented model given in (3.34) and (3.35).

$$x_e(k+1) = A_e x_e(k) + B_e u(k) \tag{4.32a}$$
$$y_e(k) = C_e x_e(k) \tag{4.32b}$$

Also, recall the extension of the PWA controller (for invariance) for the augmented model (4.32) given in (3.36). Similarly, we extend the contractive PWA control law given in (4.9) for the extended model,

$$u^e_{pwa}(x_e(k)) = F^e_i x_e(k) + g^e_i. \tag{4.33}$$

Similarly, recall all the supporting matrices $A_1, A_2, B_1, B_2, V^{(2)}_e, U^{(e)}_e$ mentioned in (3.40), (3.42) and (3.43) which will be used in the problem formulation.

Definition 4.3.3 *Consider a discrete-time linear system affected by the first order neglected dynamics* (4.32) *controlled via a contractive PWA state feedback controller* (4.33)*. A margin for first order neglected dynamics is characterized by a set* $\Omega^\alpha \in \mathbb{R}$ *which contains the values of parameter* α *such that* $((1-\alpha)A_1 + \alpha A_2)x_e(k) + ((1-\alpha)B_1 + \alpha B_2)u^e_{pwa}(x_e(k)) \in \lambda\beta\mathcal{R}, \forall\alpha \in \Omega^\alpha$*.*

Recall the definition of the matrix Γ given in (4.15) and let $M = \lambda W$. Now we present the theorem which characterize the variation of the parameter α for the dynamical system affected by first order neglected dynamics.

Theorem 4.3.4 *Consider the extended dynamical model* (4.32) *subject to the first order neglected dynamics controlled with the help of a contractive piecewise affine control law* (4.33)*. The admissible set of parameters for the neglected dynamics is given by the set,*

$$\Omega_\alpha = Proj_\alpha \mathcal{T}, \tag{4.34}$$

where $\mathcal{T}$ *denotes the polyhedral set:*

$$\mathcal{T} = \left\{ \begin{array}{c} \{(\alpha, \Gamma) \in \mathbb{R} \times \mathbb{R}^{r\times p} | \mathbf{1}^T\Gamma = [\beta_1, \cdots, \beta_p], (1-\alpha)(A_1V^{(2)}_e + B_1U^{(2)}_e) \\ + \alpha(A_2V^{(2)}_e + B_2U^{(2)}_e) = M\Gamma \end{array} \right\} \tag{4.35}$$

Proof 4.3.3 *The proof of this theorem is similar to the one presented in Proof 3.4.1.*

4.4 Examples

4.4.1 Robustness parametric margin

Consider a discrete-time linear system constructed from the uncertainty set described by:

$$\Omega = \text{conv}\left\{ \begin{array}{l} [A_1\ B_1] = \begin{bmatrix} 0.4546 & -0.0913 & 0.0849 \\ 0.1836 & 0.5389 & 0.0064 \end{bmatrix} \\ [A_2\ B_2] = \begin{bmatrix} 0.7326 & -0.0767 & 0.0609 \\ 0.1557 & 0.9909 & 0.0114 \end{bmatrix} \\ [A_3\ B_3] = \begin{bmatrix} 1.0866 & -0.0861 & 0.0823 \\ 0.1722 & 1.4323 & 0.0076 \end{bmatrix} \end{array} \right\}$$

with the nominal model chosen to be:

$$x_{k+1} = Ax(k) + Bu_k$$

with,

$$A = 0.3A_1 + 0.2A_2 + 0.5A_3 \text{ and}$$
$$B = 0.3B_1 + 0.2B_2 + 0.5B_3.$$

The input constraint is given by $-2 \leq u_k \leq 2$ and the state constraints are -100 $\leq$ [0 1]$x(k) \leq$ 100. The contractive factor λ chosen is 0.99. Unity weights are applied to the inputs and states penalties and the prediction horizon chosen is 2. Multi-Parametric Toolbox is used to obtain the state space partition. Figure 4.2 shows the state space partition of the above system and it has 11 regions.

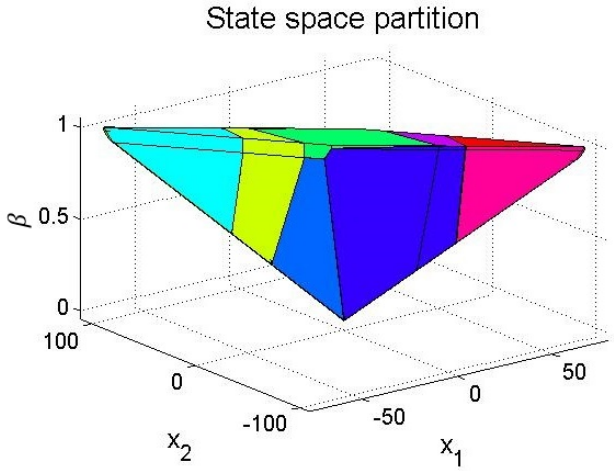


Figure 4.2: State space partition with β as parameter

Figure 4.3 shows the regions of the initial contractive set with the projected states [x_1, x_2]. Here the initial contractive set is divided into four regions and the hyperplane for the β parameter is calculated for each region. The projection is done using

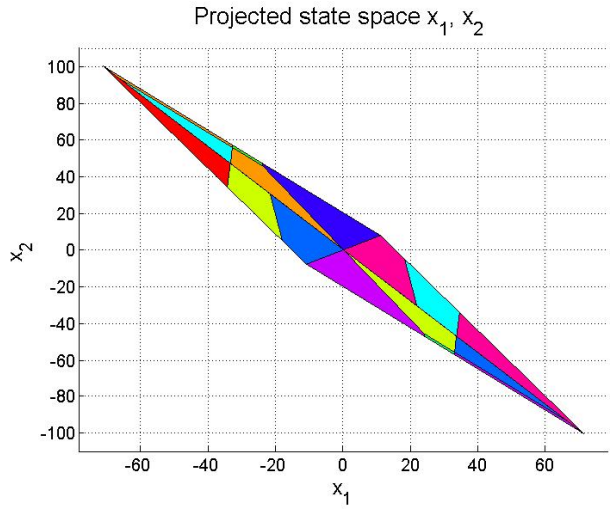


Figure 4.3: Projected state space partition x_1, x_2.

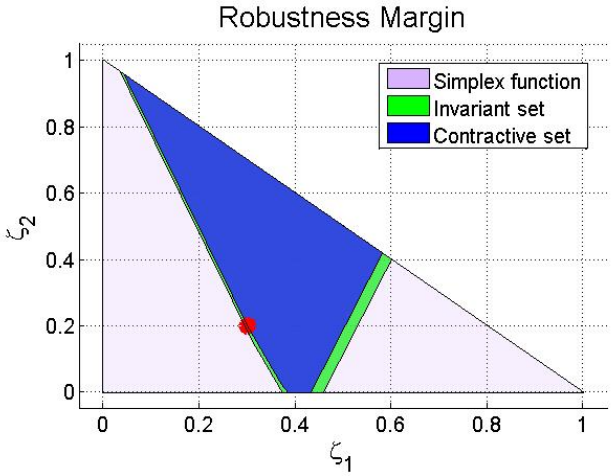


Figure 4.4: Robustness margin for contractive and invariant set in the plane of ζ_1, ζ_2.

the proposed algorithm in section IV. Figure 4.4 shows the robustness margin Ω_{rob} for the controlled λ-contractive set presented by the blue polytope and robustness margin for the controlled positively invariant set by the green polytope. The red dot denotes the considered nominal system where $\zeta_1 = 0.3$ and $\zeta_2 = 0.2$. For simplicity, the simplex function is presented only for ζ_1 and ζ_2 such that $\zeta_3 = 1$ - ζ_1 - ζ_2.

It is observed that the given control law guarantees the contractivity of the feasible region $\mathcal{R}$ only if the system is inside the blue polytope. Similarly, the control law cannot guarantee either the positive invariance or contractivity of $\mathcal{R}$ if the system parameters are away from the blue and green polytopes. For system inside the green polytope and outside the blue polytope the control law guarantees the invariance of the operating region $\mathcal{R}$.

Figure 4.5 shows the state trajectories for the same initial state $x_0 = [-70\ 98]^T$ for different systems. The contractive state trajectories for a system chosen from the robustness margin set with $\zeta_1 = 0.4, \zeta_2 = 0.4$ and $\zeta_3 = 0.2$ is illustrated in the

Figure 4.5. The state trajectories for the system with $\zeta_1 = 0.6, \zeta_2 = 0.4$ and $\zeta_3 = 0$, a system which belongs to the green polytope and does not exist in blue polytope, are invariant with respect to the invariant PWA controller. Similarly, the state trajectories for the system, which neither belongs to the blue and red polytope, are shown to be neither invariant nor contractive in the Figure 4.5. Figure 4.6, depicts

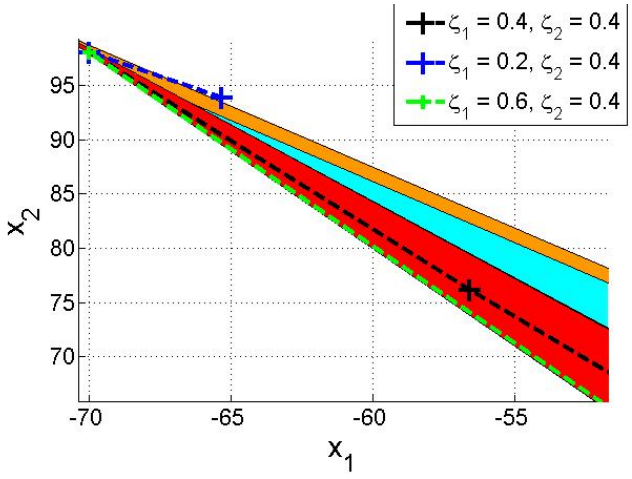


Figure 4.5: Trajectories for different nominal systems for the same initial state.

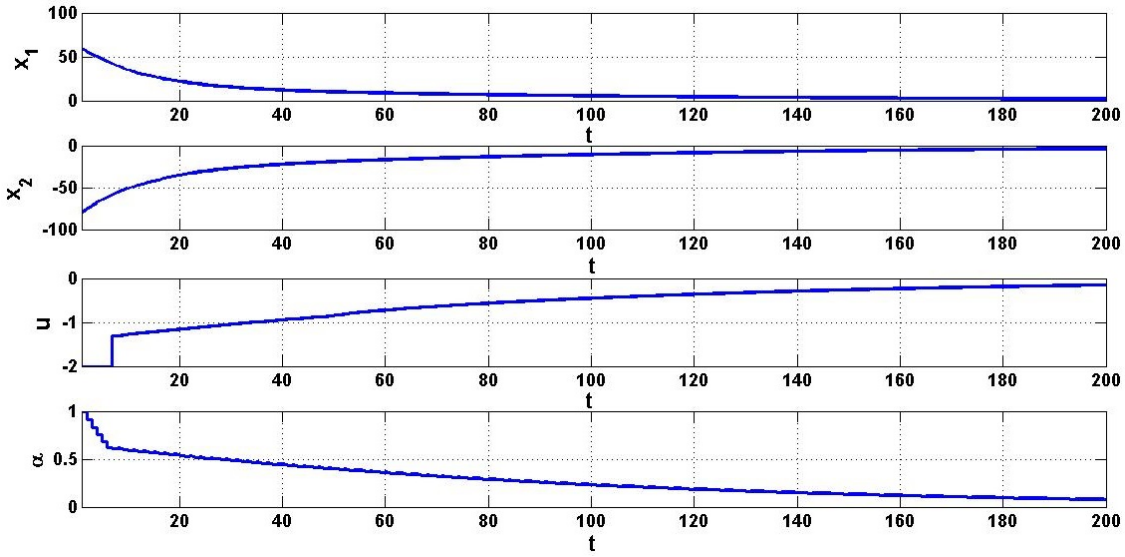
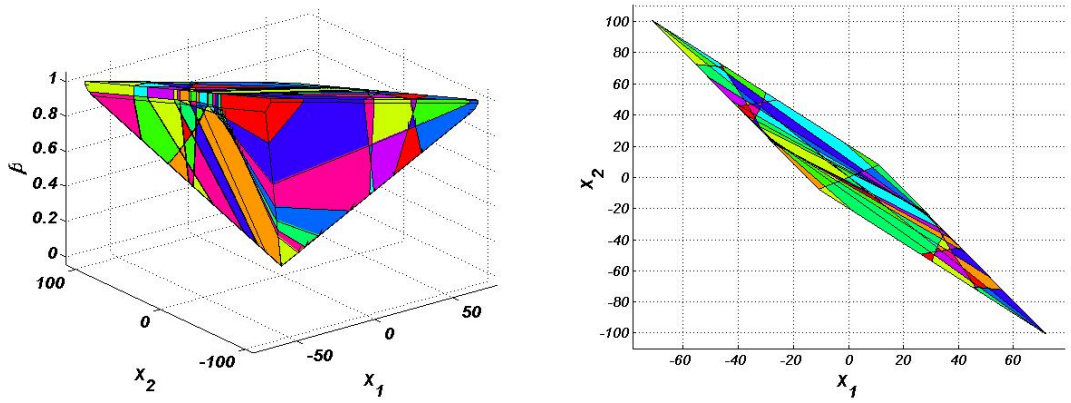


Figure 4.6: Simulation for the state trajectories and control input for an initial state $x_0 = [60 \ \ -80]^T$.

the simulation for the state trajectories and control input for a nominal system with $\zeta_1 = 0.3,\ \zeta_2 = 0.2$ and $\zeta_3 = 0.5$ and, for an initial state $x_0 = [60 \ \ -80]^T$. In the next, we obtain the polyhedral partitions on the $[x\ \beta]^T$ parameter space for the mp-QP problem for the prediction horizon $N_p = 10$ with the same system dynamics mentioned above. The contractive factor $\lambda = 0.99$ and the same weight matrices Q and R are considered for the mp-QP problem. The original polyhedral partition on the $[x\ \beta]^T$ space consists of 175 regions and it is illustrated in the Figure 4.7a. After projecting the $[x,\ \beta]^T$ state-space partition on the x-plane, we obtained 136 regions



(a) State space partition with β as parameter for $N_p = 10$.

(b) Projected state space partition x_1, x_2 for Figure 4.7a.

Figure 4.7: State space partition for $N_p = 10$

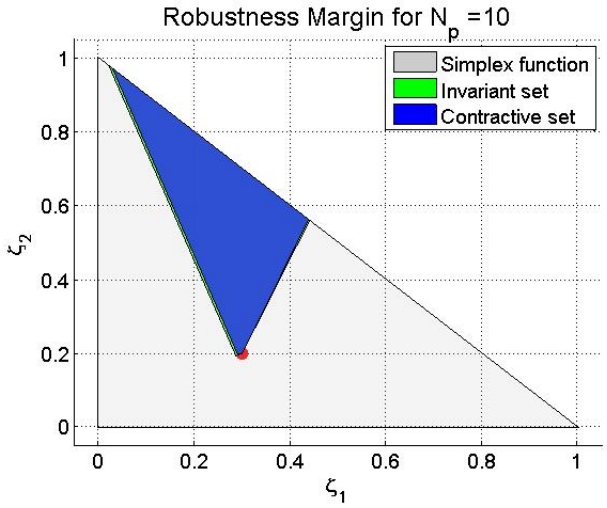


Figure 4.8: Robustness margin for contractive and invariant set for $N_p = 10$.

and the new controller for the projected polyhedral sets are also computed using the Algorithm 2.4.1. The projected state partition is depicted in the Figure 4.7b. Now, we compute the robustness margin for the projected state partition with the new controller with respect to both the invariance and the contractiveness as depicted in the Figure 4.8. The nominal system is marked with a red dot in the Figure 4.8. For all the systems in the contractive set denoted by the blue polytope the contraction of the state trajectories are guaranteed.

Note: For the MPC problem with the controlled-invariant set, it is a fact that the volume of the feasible region grows with the increase in the length of the prediction horizon. Unlike the invariant set, the objective of the contractive set that we consider here is to provide a one-step contractivity for the system dynamics. The contractive set is constructed based on the system state matrix A, and the length of the prediction horizon has no influence over the contractive feasible region. The results mentioned here for the different prediction horizon length ($N_p = 2$ and $N_p = 10$)

are intended to show the complexity of the PWA controller. It is shown from the above analysis that the robustness margin for the contractive set with $N_p = 10$ is smaller than that of the margin with $N_p = 2$.

4.4.2 Gain Margin

Consider a discrete-time linear system,

$$x(k+1) = \begin{bmatrix} 0.9 & 0.5 \\ 0.2 & 0.8 \end{bmatrix} x(k) + \begin{bmatrix} 1.0 \\ 0.2 \end{bmatrix} u(k)$$

The input constraint is given by $-2 \leq u(k) \leq 2$ and state constraint by $-5 \leq [0\ 1]x(k) \leq 5$. The prediction horizon chosen is 2 and unity weights are applied on the inputs and states penalties . The contraction factor $\lambda = 0.98$ is considered.

The gain margin set $\mathcal{K}$ in (4.25) is computed for the PWA control law assuring the contractivity characteristics of the controller. As a term of comparison, with respect to the invariance, the value of δ_K lies between $[-0.524,\ 1.554]$ while for the contractive controller, the value of δ_K lies between $[-0.467,\ 1.445]$.

Figure 4.9 shows the projected state partition on the $x-$plane with state simulations in closed loop for an initial state $x(0) = [0.355\ 5.0]^T$ for different δ_K values.

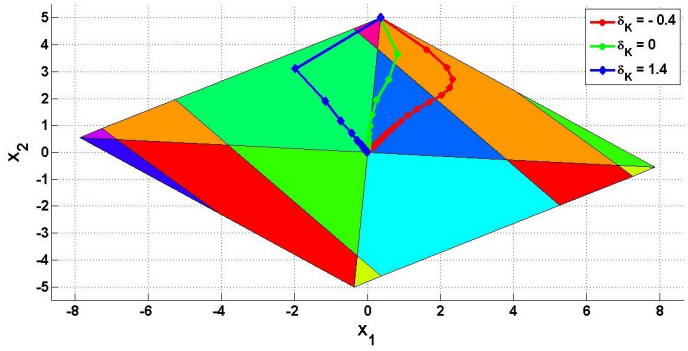


Figure 4.9: Projected state space partition with state trajectories for different δ_K values.

State trajectories, control inputs and β values are simulated in closed loop for an initial state $x_0 = [0.355\ 5.0]^T$ with different δ_K values, such that $x(k+1) = Ax(k) + B(1+\delta_K)u_{pwa}(x(k))$, for the contractive controller and it is shown in Figure 4.10. It is observed that the trajectories are λ-contractive thus confirming the theoretical result.

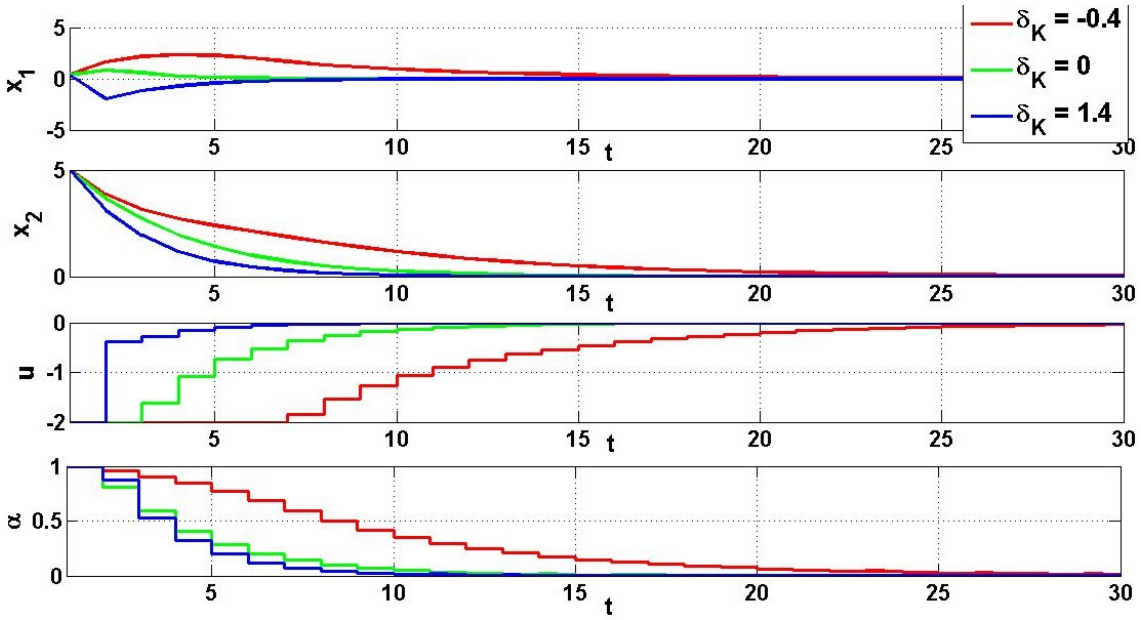
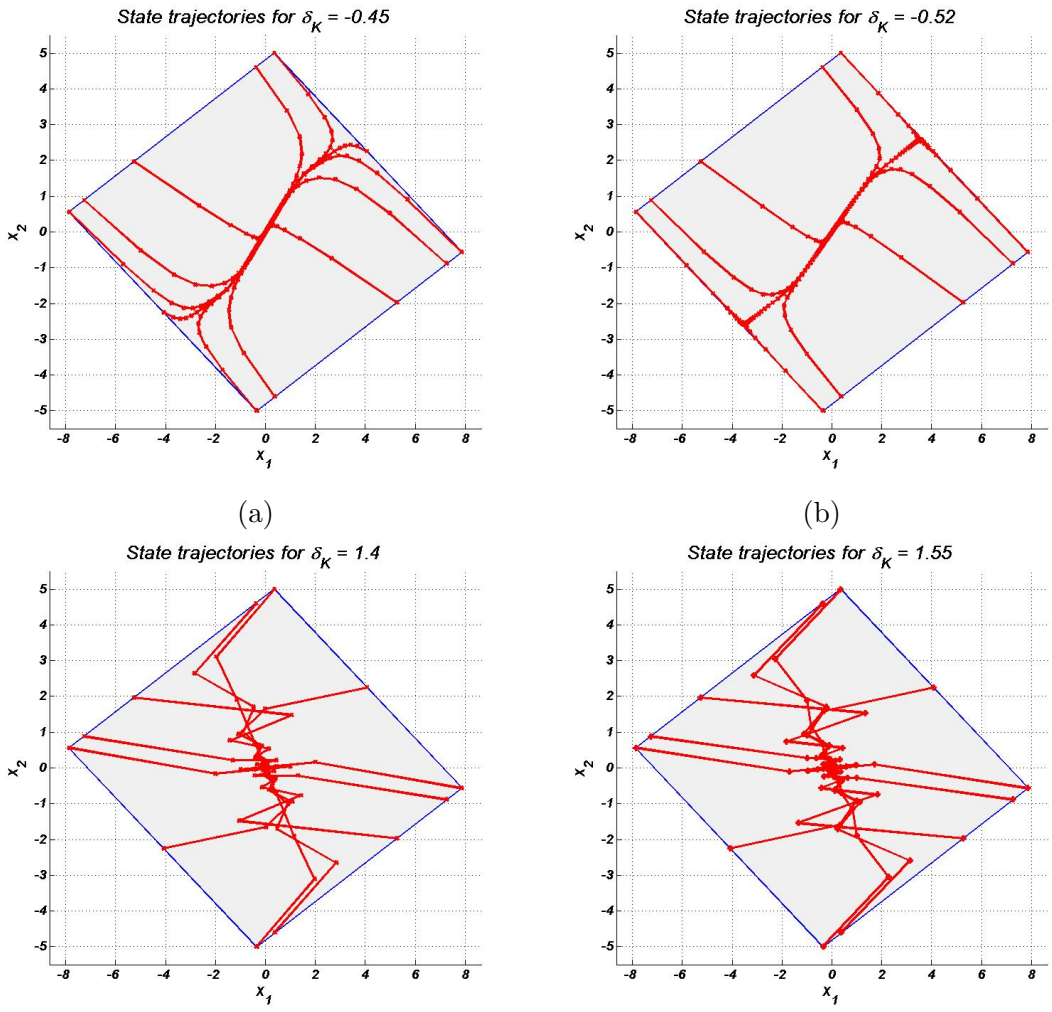


Figure 4.10: Simulation for state trajectories, control input and β for different δ_K values.

Finally, we simulate the state trajectories for the vertices on the frontier as initial states for different gain margins $\delta_K = -0.45$, $\delta_K = -0.52$, $\delta_K = 1.4$, and $\delta_K = 1.55$ and it is presented with the help of the Figure 4.11. For the state trajectories simulated with the gain margins $\delta_K = -0.45$ and $\delta_K = 1.4$ in closed-loop, the trajectories are λ-contractive. Similarly, for the trajectories with the gain margins $\delta_K = -0.52$ and $\delta_K = 1.55$, the invariance of the contractive properties of the PWA controller are guaranteed.

4.4.3 Robustness Margin for First order neglected dynamics

Consider the nominal system example given in the section 4.4.1 and we use the same weight matrices and same contractive factor λ. Then the margin for the first order neglected dynamics is represented by the set $\Omega^\alpha = [0,\ 0.4745]$. For all the α parameter values in the set Ω_α, the contractive property of the controller is preserved.



(a) (b) (c) (d)

Figure 4.11: Representation of the set $\mathcal{R}$ with the simulation of state trajectories for all the frontier vertices as initial states for different gain margins δ_K.

4.5 Conclusions

In this work the robustness margin assuring the contractivity of trajectories for a linear uncertain system controlled by a piecewise affine control law has been deduced. Starting with a nominal contractive PWA control law, this robustness margin consists in a subset of the uncertain polytopic description of the plant. For all the parameters within this set and the associated dynamical models, the trajectories are contractive. Moreover a gain robustness margin has also been computed. Finally, we also computed the margin for first order neglected dynamics for the contractive PWA controller.

Chapter 5

Precision in polyhedral partition representation and the fragility of PWA control

Contents

5.1 Introduction 65
5.2 Motivation and Problem Formulation 68
5.3 Treatment of a vertex considered independently - Polyhedral overlapping 73
5.3.1 Characterization of the vertex sensitivity 74
5.3.2 Admissibility restriction with respect to the input constraints 78
5.4 Impact of Vertex Perturbation on the invariance characterization 79
5.4.1 Perturbations of vertices on the frontier of the feasible domain $\mathcal{R}$ 79
5.4.2 Margin of Convexity 82
5.4.3 Treatment of one inner vertex for non-overlapping and invariance 87
5.5 Treatment of multiple vertex perturbation 89
5.6 Example 90
5.7 Contractivity 95
5.8 Conclusion 99

5.1 Introduction

Explicit PWA control laws can be efficiently implemented on hardware circuits for systems with fast dynamics and relatively small dimension of system states. Recently, such control laws have gained popularity for a wide range of real-time control

applications [JJST07, IK15, TBGRI16, LWM08, MDM09]. However, the adoption of such control laws pertains to the numbers of state space partitions and the piecewise affine control laws associated with those partitions. In order to exploit the computational advantages of the explicit controller, a "*truncation or quantization operation*" must be performed on the representation of the state space partitions and on their associated PWA controls. The implications of the quantized state partitions and the quantized PWA gains and offsets extend to affect control input accuracy, whose computations are based on point location functions, and the properties of the PWA controller. The quantized state partitions might also adversely affect the non-overlapping and non-emptiness characteristics of the PWA controller. In recent work [ONB+13, NORA+16], a geometrical approach to determine robustness/fragility margins with respect to the invariance characteristics of the PWA controller has been proposed. However their approach does not extend to the quantized state space partitions. In a recent study [KZC15], the accuracy of the explicit control input for the quantized regions and the quantized PWA control laws is analyzed in general to prove the scale of quantization required in order to obtain a certain degree of control accuracy. However, all these references build the control input analysis on the assumption that the modified state space regions are non-overlapping and thus they do not address one of the essential characteristics of the representation of the state partitions: the well-posedness and completeness of the polyhedral partition of the feasible domain.

The framework of the present chapter is the one of a linear discrete-time system controlled by piecewise affine explicit control law. It will be analyzed how the regions or polyhedral partitions change in the event perturbation on the vertex representation of the partitions occurs. The main contribution of this work is to analyze to what extend the non-overlapping and the invariance characteristics of the PWA controller can be preserved when a perturbation on the vertex representation of the partitions takes place. The following list is the proposed procedure or a list of parts of the chapter. The chapter is divided in five sections:

- Characterize admissible perturbations on each vertex taken independently in the vertex representation. The perturbation is considered admissible if it preserves the non-overlapping property of the state space. The admissible perturbation is characterized by a set called vertex-sensitivity regions.
- Provide a method for updating the frontier of the feasible set for admissible perturbation by updating the vertices which are considered to be sensitive. The updating preserves the non-overlapping and the invariance property of the controller.
- Third, starting from the updated feasible domain, we analyze the polyhedral partitions by considering each of the vertices of the partition which are not placed on the frontiers of the feasible domain. A set called sensitivity margin with respect to the closed-loop invariance is determined. This set characterizes admissible perturbations on the vertex representation preserving the non-overlapping and the invariance characteristics of the PWA controller.
- Fourth, we compute the perturbed polyhedral partitions for all the inner vertices of the feasible set sequentially completing the transformation of the orig-

inal polyhedral regions to a new polyhedral regions for all the vertices in the feasible set.

- Fifth, starting from a linear system stabilized with the help of a contractive piecewise affine (PWA) control law, all the vertices of the feasible set are perturbed. The perturbation of the vertices are conducted by assuring the non-overlapping and λ-contractive properties of the PWA controller.

The mathematical framework will be the same as in the previous chapter, we consider a linear discrete-time system given by,

$$x_{k+1} = Ax_k + Bu_k. \tag{5.1}$$

Definition 5.1.1 *Consider for a given $x \in \mathbb{R}^n$ and the polyhedral partitions $\mathcal{R} = \cup_{i=1}^{N} \mathcal{R}_i$, the point location function of the polyhedral partition of $\mathcal{R}$ is given by,*

$$x \to i(x) \text{ with } i : \mathcal{R} \to \mathbb{N}_{\leq N}. \tag{5.2}$$

Practically, $i(x)$ indicates (the unique) polyhedral region that contains x within the partition. Whenever x lies on the frontiers, there might be several polyhedral sets containing the point. In such cases, without loss of generality $i(x)$ is selected as the minimal index.

The feedback control law takes the form of a mapping u_{pwa}: $\mathcal{R} \to \mathbb{R}^m$

$$u_{pwa}(x_k) = F_{i(x)}x_k + g_{i(x)}, \quad x_k \in \mathcal{R}_{i(x)}. \tag{5.3}$$

defined over the polyhedral partition of the set $\mathcal{R} = \cup_{i \in \mathcal{I}_N} \mathcal{R}_i$.

Assumption 5.1.1

1. *The set $\mathcal{R}$ is a bounded polyhedron and $\mathcal{I}_N$ is finite.*
2. *$\mathcal{R}_i$ are polyhedral.*
3. *$int(\mathcal{R}_i) \cap int(\mathcal{R}_j) = \emptyset, \ \forall i \neq j, \ i, j \in \mathcal{I}_N^2$.*

With respect to the PWA function, the following assumptions hold

Assumption 5.1.2

1. *$\mathcal{R}$ is positively invariant with respect to $x_{k+1} = Ax_k + Bu_{pwa}(x_k)$.*
2. *The control law $u_{pwa}(x_k)$ is continuous.*

By construction of the standard MPC, the positive invariance is considered to be guaranteed and thus the assumptions above hold for the explicit MPC law (5.3).

5.2 Motivation and Problem Formulation

For real-time implementation of the PWA control law, three stages need to be considered:

(A) Off-line: Storage of the polyhedral regions $\mathcal{R}_i$, the PWA control gains F_i and affine components g_i.

(B) On-line: Use of a point location mechanism with respect to the parameter x and the polyhedral partitions $\mathcal{R} = \cup_{i=1}^{N} \mathcal{R}_i$, This can be assimilated to a function $x_k \to i(x_k)$.

(C) On-line: Evaluation of the PWA control law $u_{pwa}(x_k) = F_{i(x_k)}x + g_{i(x_k)}$ based on the current state x_k and the result of the previous stage of positioning.

In practice this evaluation procedure can fail due to several reasons.

(i) The precision of $\mathcal{R}_i$ representation.

(ii) Due to point location mismatch.

(iii) PWA control accuracy inflicted by the precision of representation of the control gain F_i and offset g_i.

The PWA control accuracy and the fragility issues of the gains and affine terms F_i and g_i have been extensively discussed in [ONB+13, NORA+16, KZC15]. The resulting solution obtained from the EMPC problem is a set of PWA functions defined over the polyhedral partition $\mathcal{P}_N(\mathcal{R})$ and their analysis in the point iii) above can be handled in the respective framework. However, the issues related with the representation and the closely related point location problems (items i) and ii) above) remain largely uncovered and will represent the main goal of the present chapter.

Before entering into the details of the main results, let us motivate the chosen approach by considering a polyhedral region $\mathcal{R}_i \subset \mathcal{R} \in \mathbb{R}^n$, $i \in \mathcal{I}_N$, and its half-space representation given by,

$$\mathcal{R}_i = \{x \mid h_{i,j}x \leq b_{i,j}, \forall i \in \mathcal{I}_N, \quad j = 1, \cdots, r_i\}. \tag{5.4}$$

Here, r_i denotes the number of closed half-spaces of the region $\mathcal{R}_i$. In order to analyze the sensitivity of the polyhedral partition representation and its implication on the PWA control, a perturbation in the representation of the half-space $\{h_{i,r_b} \leq b_{i,r_b}\}$, for one of the indices $r_b \in \mathcal{I}_{r_i}$ of the region $\mathcal{R}_i$ will be considered,

$$\hat{h}_{i,r_b} = h_{i,r_b} + \Delta h_{i,r_b} \text{ and } \hat{b}_{i,r_b} = b_{i,r_b} + \Delta b_{i,r_b} \tag{5.5}$$

which leads to a new polyhedral set:

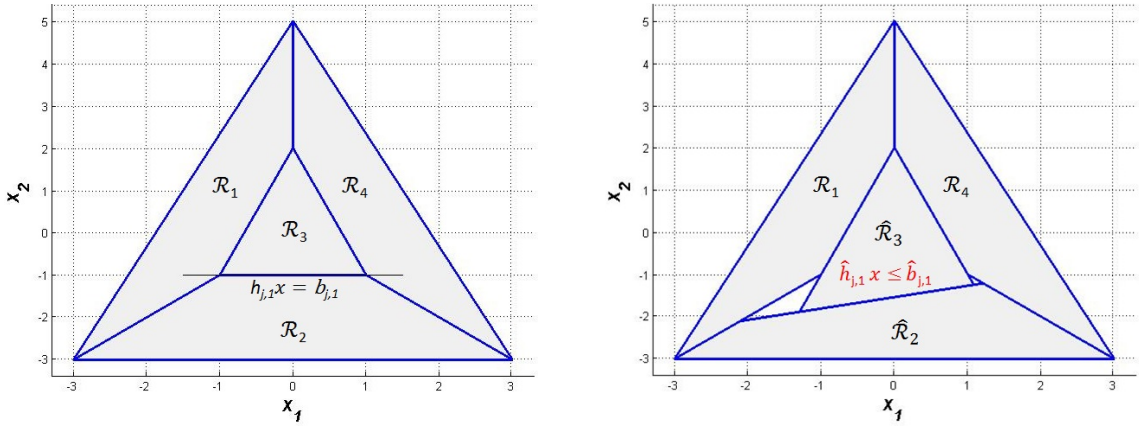
$$\hat{\mathcal{R}}_i = \{x \mid \hat{h}_{i,r_b}x \leq \hat{b}_{i,r_b}\}. \tag{5.6}$$

The perturbation of the half-space representation of the region $\mathcal{R}_i$ will concomitantly affect all the neighbor regions $\mathcal{R}_j$ sharing the respective frontier. As several neighboring regions are affected, the analysis of the effects on the partition will encounter structural problems:

1. Invalidation of the polyhedral partitions definition due to the violation of the property: $int(\hat{\mathcal{R}}_i) \cap int(\hat{\mathcal{R}}_j) = \emptyset, \ \forall i \neq j$.

2. $\mathcal{R} \backslash \left\{ \cup_{i=1}^{N} \hat{\mathcal{R}}_i \right\} \neq \emptyset$ even if $conv(\mathcal{R}) = conv(\cup_{i \in \mathcal{I}_N} \hat{\mathcal{R}}_i)$ posing an well-possessedness issue in the characterization of the polyhedral partition and subsequently in the PWA function evaluation (5.3).

The first type of problem arises from the asymmetric consideration of the perturbation in between neighboring regions while the second can take place even if the perturbation is treated similarly among the neighboring regions. Moreover, both phenomena lead to invalidation of the PWA control law defined over the partition $\hat{\mathcal{R}} = \cup_{i \in \mathcal{I}_N} \hat{\mathcal{R}}_i$. Particularly the second phenomenon leaves the point location function seemingly untraceable and this case is shown in Figure 5.1.



(a) 2-D Polyhedral with four regions $\mathcal{R} = \cup_{i=1}^{4} \mathcal{R}_i$ before perturbation of the half-space representation.

(b) Illustration of regions $\hat{\mathcal{R}}_1$, $\hat{\mathcal{R}}_2$, $\hat{\mathcal{R}}_3$ and $\hat{\mathcal{R}}_4$ after perturbation of $h_{3,1}x = b_{3,1}$. Such a reconfiguration produces holes in the feasible domain that will undermine the well-possessedness characteristics.

Figure 5.1: 2-D polyhedral representation before and after perturbation on the half-space representation.

The drawbacks demonstrated by the perturbation on the half-space representation are the consequence of the fact that the perturbations are not considered jointly for all half-spaces. This is due to the fact that the closed half-spaces of the regions $\mathcal{R}_i$ are treated independently at the level of each neighboring region and addressing perturbation on such representation is missing the interplay between regions in composing the polyhedral partition. These drawbacks lead us to the duality of the polyhedron representation where the problem can be reformulated.

Eq (5.4) can be given with equivalent vertex representation by virute of Motzkin duality [MRTT53]:

$$\mathcal{R}_i = Conv\{v_{i,1}, \cdots, v_{i,r_i}\},\ \forall i \in \mathcal{I}_N \tag{5.7}$$

here r_i is the number of vertices of $\mathcal{R}_i$. Now, consider a perturbation with respect to the vertex representation $v_{i,j},\ j \in \mathcal{I}_{r_i}$ of the region $\mathcal{R}_i$,

$$\hat{v}_{i,j} = v_{i,j} + \Delta v_{i,j},\ i \in \mathcal{I}_N,\ j \in \mathcal{I}_{r_i} \tag{5.8}$$

this will lead to a new polyhedral set:

$$\hat{\mathcal{R}}_i = Conv\{v_{i,1} + \Delta v_{i,1}, \cdots, v_{i,r_i} + \Delta v_{i,r_i}\}. \tag{5.9}$$

It becomes obvious that in this case $\mathcal{R} \setminus \left\{\cup_{i\in\mathcal{I}_N} \hat{\mathcal{R}}_i\right\} = \emptyset$ if the vertices on the frontier of $\mathcal{R}$ are not perturbed.

In order to provide a graphical interpretation of the problem under study let us consider the example of a simple continuous PWA function $f_{pwa}(x)$ defined over the interval $[-\mu,\ \mu] = [v_{2,1}, v_{3,2}] = \mathcal{R} \subset \mathbb{R}^n$.

$$f_{pwa}(x) = \begin{cases} \alpha x & \text{for} \quad x \in \mathcal{R}_1 = Conv\{v_{1,1}, v_{1,2}\}, \\ \bar{\beta} & \text{for} \quad x \in \mathcal{R}_2 = Conv\{v_{2,1}, v_{2,2}\}, \\ \bar{\gamma} & \text{for} \quad x \in \mathcal{R}_3 = Conv\{v_{3,1}, v_{3,2}\}. \end{cases} \tag{5.10}$$

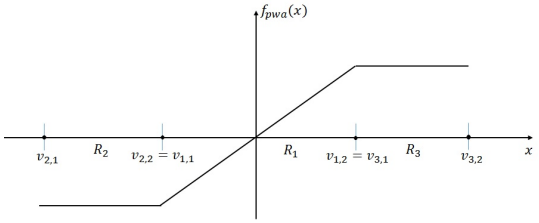
which can be simplified by the continuity between the neighboring regions: $v_{1,2} = v_{3,1}$ and $v_{2,2} = v_{1,1}$.

In Figure 5.2, the regions representation before and after the perturbation on the vertices is presented. In Figure 5.2a, the nominal three regions obtained from EMPC denoted by $\mathcal{R}_1, \mathcal{R}_2$ and $\mathcal{R}_3$, $\mathcal{R} = \mathcal{R}_1 \cup \mathcal{R}_2 \cup \mathcal{R}_3$ satisfy by definition the property $int(\mathcal{R}_i) \cap int(\mathcal{R}_j) = \emptyset,\ \forall i, j \in \mathcal{I}_3,\ i \neq j$.

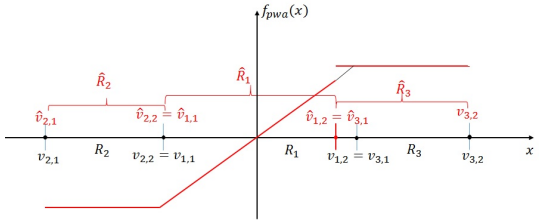
The vertices of the regions are denoted by v_{i,r_i}, here i and r_i denote the index of regions and the index of vertices for each region respectively. After introducing a perturbation Δv on the vertex $v_{1,2}$ and $v_{3,1}$ (recall that $v_{1,2} = v_{3,1}$) which falls in the regions $\mathcal{R}_1$ and $\mathcal{R}_3$, the perturbed vertex is denoted by $\hat{v}_{1,2}$ and $\hat{v}_{3,1}$ ($\hat{v}_{1,2} = v_{1,2} + \Delta v_{1,2},\ \hat{v}_{3,1} = v_{3,1} + \Delta v_{3,1},\ \ \hat{v}_{1,2} = \hat{v}_{3,1}$) as depicted in Figure 5.2b.

Consequently, one vertex displacement will influence the topology of the regions $\mathcal{R}_1$ and $\mathcal{R}_3$ leading to $\hat{\mathcal{R}}_1$ and $\hat{\mathcal{R}}_3$, while region $\mathcal{R}_2 = \hat{\mathcal{R}}_2$ remains unchanged. It can be noticed from Figure 5.2b that even after the perturbations of the regions, the polyhedral partition is preserved $\hat{\mathcal{R}} = \hat{\mathcal{R}}_1 \cup \hat{\mathcal{R}}_2 \cup \hat{\mathcal{R}}_3$ and in this particular example the non-overlapping property holds: $int(\hat{\mathcal{R}}_i) \cap int(\hat{\mathcal{R}}_j) = \emptyset, \forall i, j \in \mathcal{I}_3,\ i \neq j$.

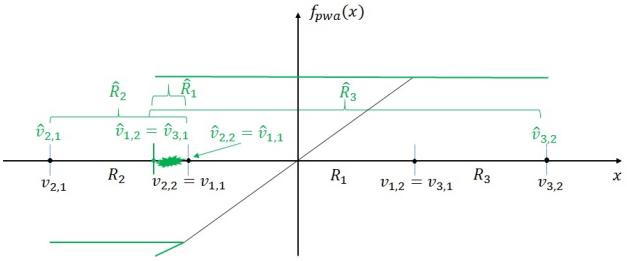
A consequence of the perturbation is the loss of continuity for the PWA function $f_{pwa}(x)$ defined over $\mathcal{R} = \hat{\mathcal{R}}_1 \cup \hat{\mathcal{R}}_2 \cup \hat{\mathcal{R}}_3$. Moreover, the control input depends on which region the state vector x falls into and in particular with the regions $\hat{R}_1$ and $\hat{R}_3$, the loss of continuity leads to a control action which is multivalued at the frontier of these regions. It is obvious that the modifications of the geometry of partition will lead to a loss of precision with respect to the nominal control input and consequently to a loss of performances. The loss of continuity is the price to be paid for the loss of precision in the partition representation and can be acceptable as



(a) 1-D Polyhedral with three regions $\mathcal{R}_1, \mathcal{R}_2$ and $\mathcal{R}_3$ and the corresponding PWA functions before perturbation.



(b) Three regions $\mathcal{R}_1, \mathcal{R}_2$ and $\mathcal{R}_3$ before perturbation of the vertex representation in black. Illustration of regions $\hat{\mathcal{R}}_1, \hat{\mathcal{R}}_2$ and $\hat{\mathcal{R}}_3$ after perturbation preserving the property, $int(\hat{\mathcal{R}}_i) \cap int(\hat{\mathcal{R}}_j) = \emptyset,\ \forall i, j \in \mathcal{I}_3, i \neq j$.



(c) Three regions $\mathcal{R}_1, \mathcal{R}_2$ and $\mathcal{R}_3$ before perturbation in black. Illustration of the regions $\hat{\mathcal{R}}_1, \hat{\mathcal{R}}_2$ and $\hat{\mathcal{R}}_3$ after perturbation with the particular configuration leading to: $int(\hat{\mathcal{R}}_i) \cap int(\hat{R}_j) \neq \emptyset,\ \forall i, j \in \mathcal{I}_3, i \neq j$.

Figure 5.2: 2-D polyhedral representation before and after perturbation of the vertex representation.

long as the control action is uniquely defined on the interior of the full-dimensional regions within the partition.

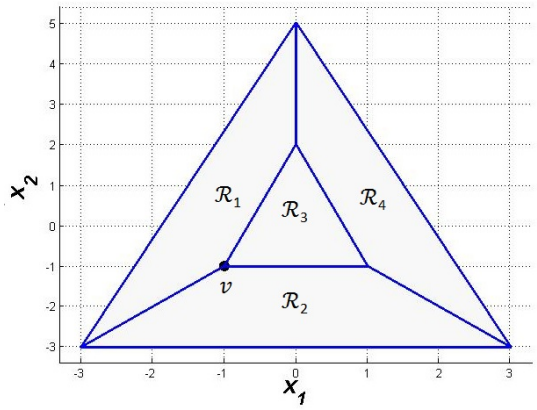
In the same framework, in Figure 5.2c, the vertex $v_{1,2}$ and $v_{3,1}$ are perturbed to $\hat{v}_{1,2}$ and $\hat{v}_{3,1}$ and moved onwards the region $\mathcal{R}_2$ adversely affecting all the three regions. The new regions denoted by $\hat{\mathcal{R}}_1, \hat{\mathcal{R}}_2$ and $\hat{\mathcal{R}}_3$ are now overlapping, i.e., $int(\hat{\mathcal{R}}_1) \cap int(\hat{\mathcal{R}}_2) \neq \emptyset$, $int(\hat{\mathcal{R}}_2) \cap int(\hat{\mathcal{R}}_3) \neq \emptyset$ and $int(\hat{\mathcal{R}}_1) \cap int(\hat{\mathcal{R}}_3) \neq \emptyset$.

This possible overlapping due to changes in the vertices of the polyhedral partition represents a critical structural change because the unicity of the control law is lost on a compact full-dimensional region of the state space. The non-uniqueness of the control action leads to behaviors which are difficult to characterize in terms of determinedness and loss of performance and thus should be avoided in the first place. This issue forms the basis for investigation in the present chapter and can be resumed by the need to characterize the limits of the perturbation which preserve the "non-overlapping" property of the polyhedral partition. In order to illustrate the partitions in this framework and present the obvious advantages of considering perturbation on the vertex representation, a similar partition to the one presented in Figure 5.1 is depicted in the Figure 5.3. This time it is obvious that using the dual representation of polyhedra and their perturbed version, the completeness of the partition is not lost. In general terms, the case $\mathcal{R} \setminus \left\{\cup_{i=1}^{N} \hat{\mathcal{R}}_i\right\} \neq \emptyset$ is avoided

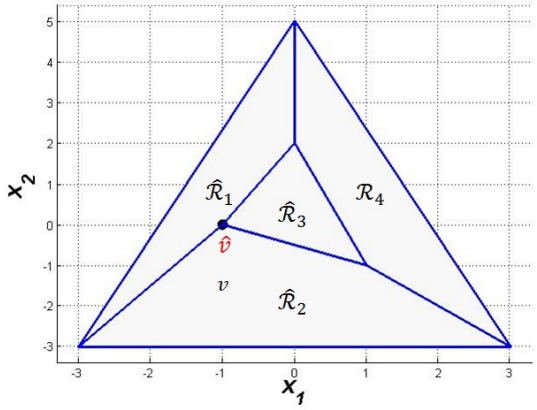
from the consequences of the perturbations in the polyhedral partition.

To resume, starting from the existence of the system in the form (5.1) stabilized by a PWA control law, the main objective is to discuss the impact of perturbations on the vertex representation of the polyhedral region by proposing:

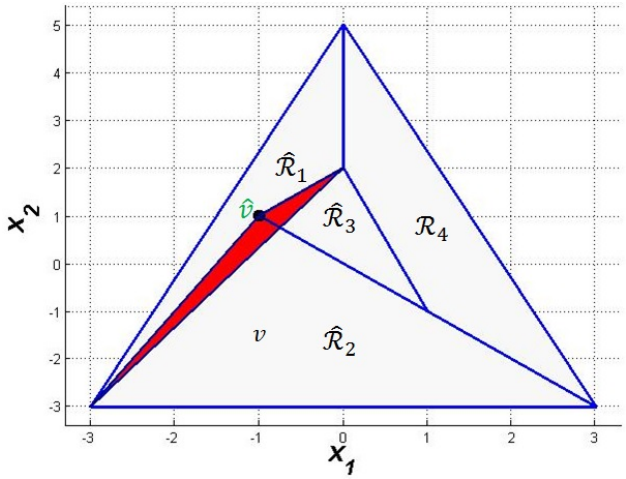
- An analysis of the admissible perturbations with respect to the overlapping characteristics of the PWA controller,
- An analysis of the admissible perturbations with respect to the invariance properties of the PWA controller.



(a) 2-D Polyhedral with four regions $\mathcal{R} = \cup_{i=1}^{4}\mathcal{R}_i$ before perturbation of the vertex representation.



(b) Illustration of regions $\hat{\mathcal{R}}_1$, $\hat{\mathcal{R}}_2$, $\hat{\mathcal{R}}_3$ and $\hat{\mathcal{R}}_4 = \mathcal{R}_4$ after perturbation preserving the property, $int(\hat{\mathcal{R}}_i) \cap int(\hat{\mathcal{R}}_j) = \emptyset,\ \forall i, j \in \mathcal{I}_4, i \neq j$. Such a reconfiguration of the partition is admissible from the point of view of point-location.



(c) Illustration of regions $\hat{\mathcal{R}}_1$, $\hat{\mathcal{R}}_2$, $\hat{\mathcal{R}}_3$ and $\hat{\mathcal{R}}_4 = R_4$ after perturbation with the particular configuration leading to: $int(\hat{\mathcal{R}}_i) \cap int(\hat{\mathcal{R}}_j) \neq \emptyset,\ \forall i, j \in \mathcal{I}_4, i \neq j$.

Figure 5.3: 2-D polyhedral representation before and after perturbation of the vertex representation

5.3 Treatment of a vertex considered independently - Polyhedral overlapping

In the following, a formal definition of the vertex sensitivity is provided focusing on the non-overlapping property of the polyhedral regions under the assumption that all the other vertices are fixed and only the vertex under study is subject to perturbations.

Definition 5.3.1 *Consider the partition $\mathcal{P}_N(\mathcal{R}) \in \mathbb{R}^n$ with each region given by its vertex representation $\mathcal{R}_i = Conv\{v_{i,1}, \cdots, v_{i,r_i}\},\ i \in \mathcal{I}_N$. Let $v \in \mathbb{R}^n$ be a vertex*

within $\mathcal{P}_N(\mathcal{R})$ and denote Θ^v as the set of indexes of polyhedral regions having v as a vertex:

$$\Theta^v = \{j \in \mathcal{I}_N |\ v \in \mathcal{V}(\mathcal{R}_j)\} \tag{5.11}$$

A compact set $\Psi \subset \mathcal{R} \subset \mathbb{R}^n$ is describing a vertex sensitivity for the vertex v if $v \in \Psi$ and for all $(v + \Delta v) \in \Psi$ the collection of sets

$$\begin{cases} \hat{\mathcal{R}}_j = Conv\{\mathcal{V}(\mathcal{R}_j) \setminus \{v\},\ v + \Delta v\},\ \forall j \in \Theta^v, \\ \hat{\mathcal{R}}_j = \mathcal{R}_j, \forall j \in \mathcal{I}_N \setminus \Theta^v \end{cases} \tag{5.12}$$

*represents a polyhedral partition: $\hat{\mathcal{P}}_N(\mathcal{R}) = \{\hat{\mathcal{R}}_1, \cdots, \hat{\mathcal{R}}_N\}$. The **sensitivity margin** for the vertex v is defined as the set Ψ^v containing any valid vertex sensitivity $\Psi \subset \Psi^v$.*

Given this formal definition, we concentrate next on the structural properties of this set and on its practical construction.

5.3.1 Characterization of the vertex sensitivity

In the next result, the structure of the sensitivity margin is stated while the proof will be constructed in such a way that the two scenarios of infeasible perturbations are enumerated and fully characterized. More than that, the set characterization will be constructive thus allowing the statement of a finite algorithmic procedure.

Theorem 5.3.1 *Consider the subset of regions $\mathcal{R}_j,\ j \in \Theta^v$ of $\mathcal{P}_N(\mathcal{R})$ such that $v \in \mathcal{V}(\mathcal{R}_j),\ \forall j \in \Theta^v$, then the vertex sensitivity margin Ψ^v is represented by a polyhedral set.*

Proof 5.3.1 *Let us describe the possible overlapping scenarios and collect the linear constraints imposed in order to avoid such configurations. We will concentrate on the set of indices within Θ^v identifying the polyhedral regions having v as a vertex. They will be used as long as the perturbation of the vertex v will directly affect these regions and consequently $\Psi^v \subseteq \bigcup_{j \in \Theta^v} \mathcal{R}_j$. For each $\mathcal{R}_j,\ j \in \Theta^v$, the half-space representation is given by:*

$$\mathcal{R}_j = \{x \mid H_j x \leq b_j\} \quad \textit{or implicitly,} \tag{5.13a}$$

$$\mathcal{R}_j = \{x \mid h_{j,r_b} x \leq b_{j,r_b},\ r_b = \{1, \cdots, r_j\}\}. \tag{5.13b}$$

Where r_j represents the number of half-space inequalities of $\mathcal{R}_j$.

Let us introduce a set Γ which collects the restrictions on the vertex perturbation. This set will be initialized with $\Gamma \leftarrow \mathcal{R}$.

First overlapping scenario: *Consider all the hyperplanes $h_{j,r_b} x = b_{j,r_b},\ j \in \Theta^v, r_b = 1, \cdots, r_j$ taken sequentially for each region $\mathcal{R}_j$ described by the half-space representation in* (5.13b) *and retain the inequalities $h_{j,r_b} x \leq b_{j,r_b}$ that are not saturated by the vertex v.*

$$\begin{cases} \forall j \in \Theta^v,\ r_b = 1, \cdots, r_j \text{ if } h_{j,r_b} v \neq b_{j,r_b} - \text{ then} \\ \qquad \Gamma \leftarrow \Gamma \cap \{h_{j,r_b} x \leq b_{j,r_b}\}. \end{cases} \tag{5.14}$$

Each of the linear inequalities in Γ *represents a constraint for the sensitivity set. Indeed, whenever the vertex* v *will be perturbed to a value* $\hat{v} \notin \Gamma$ *it will violate one of these half-spaces* $h_{j,r_b}\hat{v} \leq b_{j,r_b}$ *for some* $j \in \Theta^v$ *and* $r_b \in \{1, \ldots, r_j\}$*. Such a constraint violation will make one of the vertices of* $\mathcal{R}_j$ *redundant. Moreover, there exists at least one index* $j \in \Theta^v$ *such that the vertex representation is perturbed and* $\hat{v} \notin \bigcup_{j\in\Theta^v} \mathcal{R}_j$*. The consequence is that* $\hat{v} \in \bigcup_{j\in\mathcal{I}_N\setminus\Theta^v} \mathcal{R}_j$ *or* $\hat{v} \notin \mathcal{R}$*. The first case leads to overlapping and thus needs to be reinforced in the description of the admissible perturbation set* Ψ^v*. The second case invalidates the polyhedral partition of the original polytope* $\mathcal{P}_N(\mathcal{R})$*.*

Second overlapping scenario: *Consider a set constructed based on the region* $\mathcal{R}_j$ *and denote it as* $\overline{\mathcal{R}}_j,\ \forall\ j \in \Theta^v$ *using the following definition*

$$\overline{\mathcal{R}}_j = Conv\left\{\{v_{j,1}, \cdots, v_{j,r_{n_j}}\}\backslash v\right\} \tag{5.15}$$

here r_{n_j} *denotes the number of vertices of* R_j*. Based on this definition, the equivalent half-space representation of* $\overline{\mathcal{R}}_j$ *is*

$$\overline{\mathcal{R}}_j = \{x\ |\overline{h}_{j,b_i}x \leq \overline{b}_{j,b_i}, b_i = 1, \cdots, t_j\}. \tag{5.16}$$

Each of the closed half-space inequalities $\overline{h}_{j,b_i}x \leq \overline{b}_{j,b_i},\ j \in \Theta^v,\ b_i \in \{1, \cdots, t_j\}$ *from* (5.16) *will be analyzed with respect to* Γ*.*

If the intersection between the closed half-space given by the linear inequalities $\overline{h}_{j,b_i}x \leq \overline{b}_{j,b_i}$ *and the set* Γ *alters the shape of* Γ *then the elements of the opposite half-space i.e.,* $-\overline{h}_{j,b_i}x \leq -\overline{b}_{j,b_i}$ *will be stored as a restriction for the sensitivity set:*

$$\begin{cases} \forall j \in \Theta^v,\ b_i = 1, \cdots, t_j \ \text{ if } \Gamma \cap \{\overline{h}_{j,b_i}x \leq \overline{b}_{j,b_i}\} \neq \Gamma - \\ \qquad \text{then} \quad \Gamma \leftarrow \Gamma \cap \{-\overline{h}_{j,b_i}x \leq -\overline{b}_{j,b_i}\}. \end{cases} \tag{5.17}$$

Whenever a perturbation is chosen so that $\hat{v} \in int\{\overline{\mathcal{R}}_j\}$*, it invalidates (makes redundant) at least one vertex of* $\overline{\mathcal{R}}_j$*. This leads to an overlapping phenomenon and thus it is excluded by the mechanism described in* (5.17)*.*

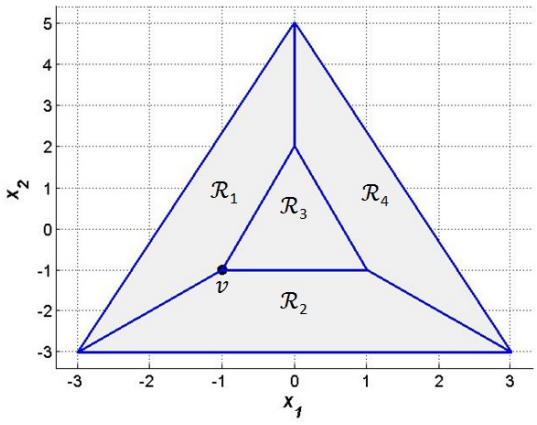
In order to complete the proof that Γ *represents the sensitivity set* Ψ^v*, it remains to show that there is no overlapping configuration excluded from the above scenarios. By the fact that the only perturbed vertex is* $\hat{v}$*, the overlapping can be evaluated exclusively by the positioning of* $\hat{v}$ *within* $\bigcup_{j\in\Theta^v} \mathcal{R}_j$*. The two scenarios are covering the case when the vertex itself is redundant or it renders redundant one of the existing vertices.*

Finally, the compact set that contains all possible variations for the given vertex v *preserving the overlapping property of the polyhedral partition with respect to the above scenarios can be resumed as:*

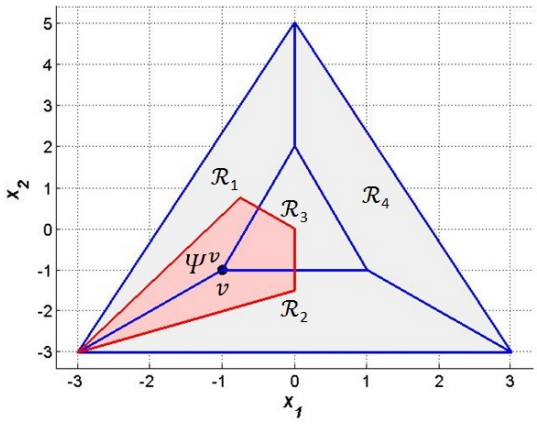
$$\Psi^v \leftarrow \Gamma. \tag{5.18}$$

In order to illustrate the result Figure 5.4a presents a polyhedral partition with four regions $\mathcal{R}_i, i = 1, \cdots, 4$ and the vertex of interest $v = [-1\ -1]^T$ is denoted by a black

dot. The vertex v belongs to three regions. In Figure 5.4b, the vertex sensitivity region Ψ^v is represented by a pink polytope and the vertex v can be settled to any of the points in the polytope Ψ^v in the event of reduced precision in the representation of the polytopic region.



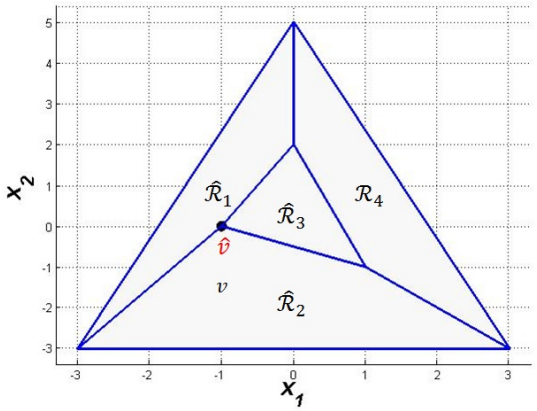
(a) Polyhedral with four regions $\mathcal{R}_1, \mathcal{R}_2, \mathcal{R}_3, \mathcal{R}_4$ and the black dot represents v.



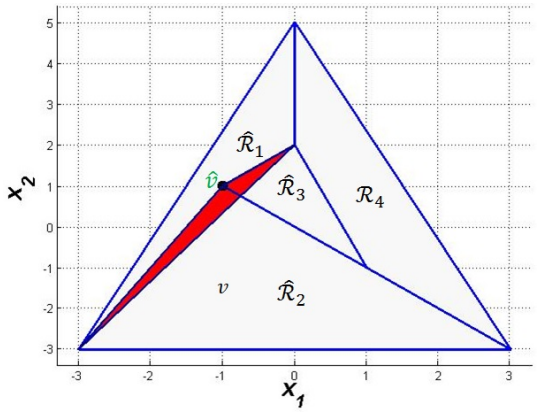
(b) The vertex of interest v and the vertex sensitivity region Ψ^v is shown.

Figure 5.4: Polyhedral partition with four regions, the vertex of interest v and the vertex sensitivity region Ψ^v are shown.

According to the structural result in Theorem 5.3.1, the vertex sensitivity region Ψ^v is a polytopic set. For any perturbed vertex $\hat{v}$ or point outside the red polytope in Figure 5.4, the non-overlapping property of the PWA control law is lost.



(a) After perturbation $\hat{v} \in \Psi^v$ the regions are changed and $int(\hat{\mathcal{R}}_i) \cap int(\hat{\mathcal{R}}_j) = \emptyset,\ \forall i, j \in J^v, i \neq j$.



(b) After perturbation $\hat{v} \notin \Psi^v$ $int(\hat{\mathcal{R}}_i) \cap int(\hat{\mathcal{R}}_j) \neq \emptyset,\ \forall i, j \in J^v, i \neq j$.

Figure 5.5: The vertex v denoted by a black dot in Fig. 3.4 (a) is perturbed to $\hat{v}$ changing the regions $\mathcal{R}_1, \mathcal{R}_2, \mathcal{R}_3, \mathcal{R}_4$ to $\hat{\mathcal{R}}_1, \hat{\mathcal{R}}_2, \hat{\mathcal{R}}_3, \hat{\mathcal{R}}_4$.

As expected, the new regions formed with the displaced vertex $\hat{v}$ guarantees the "non-overlapping" property of the polyhedral partition if $\hat{v} \in \Psi^v$. This observation is validated with the help of the Figure 5.5a and Figure 5.5b, where the polyhedral

regions are recreated by the displacement of vertex $\hat{v}$. In Figure 5.5a the vertex $v = [-1 \ \ -1]^T$ is displaced to vertex $\hat{v} = [-1 \ 0]^T \in \Psi^v$ which alters all the four regions with indices in Θ^v but still preserves the overlapping property i.e., $int(\hat{\mathcal{R}}_i) \cap int(\hat{\mathcal{R}}_j) = \emptyset, \ \forall i, j \in \mathcal{I}_4, i \neq j$. Conversely, in Figure 5.5b, it is clearly visible that the overlapping of the regions takes place since $\hat{v} = [-1 \ 1]^T \notin \Psi^v$.

The procedure for computing the vertex sensitivity region Ψ^v is resumed in Algorithm 5.3.1.

Algorithm 5.3.1 Algorithm for computing the vertex sensitivity set Ψ^v

Input: $\mathcal{R} = \cup_{i=1}^{N} \mathcal{R}_i, i \in \mathcal{I}_N$ and $v \in \mathcal{P}_N(\mathcal{R})$.
Output: Ψ^v

Initialization : $M = [\,], \ W = [\,]$. {% matrices storing the half-space description of Ψ^v}
Find the regions $\mathcal{R}_j$ that contain v, and store the indices j in Θ^v.
% Compute the half space representation of $\mathcal{R}_j$.
$\mathcal{R}_j = \{x \mid h_{j,r_i} x \leq k_{j,r_i}, r_i = 1, \cdots, r_j\}$
LOOP Process
for each $j \in \Theta^v$ **do**
  $V = [\,]$
  for $r_i = 1$ to r_j **do**
    if $h_{j,r_i} \times v \ < \ k_{j,r_i}$ **then**
      $M = [M; \ h_{j,r_i}]; \ W = [W; \ k_{j,r_i}]$
    end if
    if $v_{j,r_i} \neq v$ **then**
      $V = [V; v_{j,r_i}]$
    end if
  end for
  % Compute $\overline{\mathcal{R}}_j$ by vertex representation
  $\overline{\mathcal{R}}_j = Polyhedron(V)$
end for
% Compute $\tilde{\mathcal{R}}$ by half-space representation
$\tilde{\mathcal{R}} = Polyhedron(M, W)$
% t_j is the number of closed half-spaces of $\overline{\mathcal{R}}_j$
for $\forall j \in \Theta^v$ **do**
  for $b_i = 1$ to t_j **do**
    if $(\{\bar{h}_{j,b_i}, \bar{k}_{j,b_i}\} \cap \tilde{\mathcal{R}}) \notin \tilde{\mathcal{R}}$ **then**
      $M = [M; -\bar{h}_{j,b_i}]$
      $W = [W; -\bar{k}_{j,b_i}]$
    end if
  end for
end for
% Compute Ψ^v by half-space representation
$\Psi^v = Polyhedron(M, \ W)$

5.3.2 Admissibility restriction with respect to the input constraints

In this section, we discuss the violation of the control law occurred due to the perturbation of the vertices in the PWA control law definition. Let us consider the 1-D example from section 5.2. From Figure 5.2c, it can be inferred that when the vertex $v_{1,2}$ is perturbed to $\hat{v}_{1,2}$ the displaced vertex $\hat{v}_{1,2} \notin \mathcal{R}_1$ but $\hat{v}_{1,2} \in \hat{\mathcal{R}}_1$. The corresponding control law for the region $\mathcal{R}_1$ is given in the form of a gain αx as shown in (5.10). The feedback control law defined over the polyhedral partition of the set $\hat{\mathcal{R}}$ takes the form of a mapping $\hat{u}_{pwa} : \hat{\mathcal{R}} \to \mathbb{R}^m$,

$$\hat{u}_{pwa}(x_k) = F_i x_k + g_i, \ x_k \in \hat{\mathcal{R}}. \tag{5.19}$$

As a consequence, the displacement of the vertex $v_{1,1}$ outside the region $\mathcal{R}_1$ will cause the new control law $\hat{u}_{pwa}(x_k)$ to violate the input constraint set $\mathcal{U}$. In order to tackle such control violation issues, inflicted by the displacement of the vertex, we need to take into account the margin of control admissibility.

Definition 5.3.2 *The PWA function $u_{pwa} : \mathcal{R} \to \mathcal{U}$ and the counterpart $\hat{u}_{pwa} : \hat{\mathcal{R}} \to \mathbb{R}^m$ obtained by the perturbation of a single vertex $v \to \hat{v}$ is admissible if $\hat{u}_{pwa}(x) \in \mathcal{U}, \ \forall x \in \hat{\mathcal{R}}$.*

Theorem 5.3.2 *Let $u_{pwa}(x) : \mathcal{R} \to \mathcal{U}$ and v a vertex of $\mathcal{V}(\mathcal{P}_N(\mathcal{R}))$. The point $\hat{v}$ is admissible as a perturbation of v if it belongs to the set*

$$\Psi_u^v = \{\hat{v} \in \Psi^v | \ F_i\hat{v} + g_i \in \mathcal{U}, \forall i \in \Theta^v\}. \tag{5.20}$$

Proof 5.3.2 *For any given vertex in the set $\mathcal{V}(\mathcal{P}_N(\mathcal{R}))$ it follows that $v \in \mathcal{V}(\mathcal{R}_i)$ and there exists a PWA control law $u_{pwa}(v) = F_i v + g_i$, that satisfies the constraints $u_{pwa}(v) \in \mathcal{U}$ and moreover $Av + Bu_{pwa}(v) \in \mathcal{R}$, based on the controlled-invariance of the closed loop [Bla99].*

Now, let us introduce perturbation on any vertex v, such that for $v \to \hat{v}$, we have $\mathcal{P}_N(\mathcal{R}) \to \mathcal{P}_N(\hat{\mathcal{R}})$. With respect to the existing control law $F_i\hat{v} + g_i$ defined over the new polyhedral partitions $\mathcal{P}_N(\hat{\mathcal{R}})$, it cannot be assured that the control laws for the new set satisfy the control constraints at all times as long as all the PWA functions valid on the point $\hat{v}$ do not satisfy the input constraints. In other words, not all control laws for all the points in the vertex sensitivity set Ψ^v belong to the input constraint set $\mathcal{U}$. In order to guarantee the control law $u_{pwa}(\hat{v})$ is admissible at all times for any point in the set Ψ^v, we need to eliminate the points that violate the constraint set and remove them from the set Ψ^v which leads to the definition of the set Ψ_u^v in the statement of the theorem.

$$\Psi_u^v = \{\hat{v} \in \Psi^v | \ F_i\hat{v} + g_i \in \mathcal{U}, \forall i \in \Theta^v\}. \tag{5.21}$$

Corollary 5.3.1 *The set Ψ_u^v defining the perturbations of the vertex v which are admissible and non-overlapping from the point of view of partition $\mathcal{R}$ is a polyhedral set.*

Proof 5.3.3 *The set Ψ^v and $\mathcal{U}$ used in the description for Ψ_u^v are polyhedral and given the piecewise affine structure of the control mapping, it follows that Ψ_u^v inherits the convexity and the polyhedral structure.*

5.4 Impact of Vertex Perturbation on the invariance characterization

In this section, we bring into discussion the set invariance characterization in relationship with the PWA controller. The positive invariance of the closed-loop dynamics will be considered on top of the non-overlapping property of the PWA control function (Theorem 5.3.1) which retains a well-possessedness structural property. It is important to mention that we preserve the assumption that only one vertex is perturbed at the time, all the other vertices being maintained at the nominal values. This strong hypothesis will be relaxed later in section 5.5.

From *Theorem 5.3.1*, it is understood that the vertex sensitivity can be analyzed with respect to the admissible perturbation related to the non-overlapping characteristics for any single vertex of the polyhedral partition $\mathcal{P}_N(\mathcal{R})$. In order to incorporate the analysis of vertex sensitivity with respect to the invariance property of the PWA control law, we will have to make a difference among the vertices and the impact of their perturbation. The vertices that represent extreme points of the set $\mathcal{R}$ are particularly sensitive to perturbation taking into account that they characterize the controlled-invariant properties *per se*. Indeed, any perturbation to these vertices will change the topology of the boundary of the set $\mathcal{R}$ and potentially invalidate the positive invariance. The second class of vertices are those that are included in the strict interior of the set $\mathcal{R}$. In Figure 5.6, the classification of the vertices are illustrated.

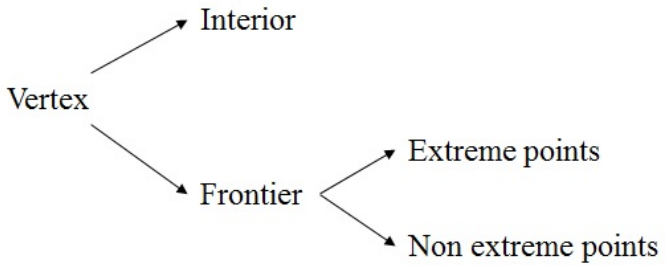


Figure 5.6: Classification of the vertices of $\mathcal{P}_N(\mathcal{R})$.

5.4.1 Perturbations of vertices on the frontier of the feasible domain $\mathcal{R}$

In the following we analyze the perturbation of vertices that represent extreme points of the set $\mathcal{R}$ (placed on the frontier of $\mathcal{R}$) and thus, by their repositioning lead to a reconstruction of the polyhedral partition $\mathcal{P}_N(\mathcal{R}) = \{\mathcal{R}_1, \cdots, \mathcal{R}_N\}$.

Consider the set $\mathcal{R} = \cup_{i=1}^{N} \mathcal{R}_i$, with $\mathcal{R}_i = Conv\{v_{i,1}, v_{i,2}, \cdots, v_{i,r_i}\}$. Let us define the set of vertices on the frontier of $\mathcal{R}$ as:

$$V = \{v \in \mathcal{R} : \exists i \text{ such that } v \in \mathcal{V}(\mathcal{R}_i) \text{ and } v \notin int(\mathcal{R})\}. \tag{5.22}$$

For the sake of notation, the set will be represented as, $V = \{v_1, v_2, \cdots, v_r\}$ with r the number of vertices, lying on the frontier of the set $\mathcal{R}$.

The analysis of perturbations in the representation of the set $\mathcal{R}$ all by assuring the non-overlapping and invariance characteristics of $\mathcal{P}_N(\mathcal{R})$ is directly related to the positioning of the frontier vertices and will be considered for each vertex in V taken independently. We start by recalling the closed-loop mapping for any point in the set $\mathcal{R}$ preserving the invariance characteristics of the PWA controller:

$$f_{pwa}(x) = Ax + Bu_{pwa}(x) \in \mathcal{R}. \tag{5.23}$$

Using (5.23), we can represent the image of the set $\mathcal{R}$ by,

$$\mathcal{F}_{pwa}(\mathcal{R}) = \{y \in \mathbb{R}^n | \exists x \in \mathcal{R} \text{ such that } y = f_{pwa}(x)\}. \tag{5.24}$$

In the work of Scibilia et al [SOH11], it has been shown that any approximation of $\mathcal{R}$ denoted by $\mathcal{R}^\alpha$ and which satisfies $\mathcal{R}^\alpha \subseteq \mathcal{R}$ and $\mathcal{R}^\alpha \supseteq \mathcal{F}_{pwa}$ preserves the invariance property of the closed loop. We aim to exploit the same principle in the framework of the vertex perturbations of the PWA control functions. We are interested in guaranteeing that the invariance holds with respect to a set $\mathcal{R}^\alpha$ defined in relationship with the existing PWA controller by perturbation of one of the frontier vertices $v \in V$ towards a point $\hat{v} \in \mathcal{R}$ thus leading to a novel (perturbed) set:

$$\begin{cases} \mathcal{R}_i^\alpha(v, \hat{v}) = conv\{\mathcal{V}(\mathcal{R}_i) \setminus v, \hat{v}\}, \forall i \in \mathcal{I}_N, \\ \mathcal{R}^\alpha(v, \hat{v}) = \cup_{i=1}^N \mathcal{R}_i^\alpha(v, \hat{v}), \\ \mathcal{P}_N(\mathcal{R}^\alpha(v, \hat{v})) = \{\mathcal{R}_1^\alpha(v, \hat{v}), \ldots, \mathcal{R}_N^\alpha(v, \hat{v})\}. \end{cases} \tag{5.25}$$

Theorem 5.4.1 *Let a dynamical system in the form* (5.1) *and the PWA control law* $u_{pwa}(x)$ (5.3) *defined over the set* $\mathcal{R}$ *and assuring its positive invariance in closed-loop. Given a set* $\mathcal{R}^\alpha \subset \mathcal{R}$*, the function* $\hat{u}_{pwa} : \mathcal{R}^\alpha \to \mathcal{U}$ *defined as* $\hat{u}_{pwa}(x) = u_{pwa}(x),\ \forall x \in \mathcal{R}^\alpha$ *ensures the positive invariance of* $\mathcal{R}^\alpha$ *with respect to* $x_{k+1} = Ax + B\hat{u}_{pwa}(x)$ *if* $\mathcal{R}^\alpha \supseteq \mathcal{F}_{pwa}$.

Proof 5.4.1 *See [SOH11] for the proof.*

Unfortunately, the theorem 5.4.1 is not offering the appropriate guarantees for the positive invariance of $\mathcal{R}^\alpha$ in closed loop with the perturbed PWA control law. The main reason is that after perturbation of a vertex of the set $\mathcal{R}^\alpha(v, \hat{v})$, the new PWA function is not guaranteed to preserve the relationship $u_{pwa}(x) = \hat{u}_{pwa}(x),\ \forall x \in \mathcal{R}^\alpha$ as stated in the Theorem above. The new partition $\mathcal{P}_N(\mathcal{R}^\alpha) \neq \mathcal{P}_N(\mathcal{R})$ and it differs in the regions affected by the perturbation of the vertex v as long as $\mathcal{R}_i^\alpha(v, \hat{v}) \neq \mathcal{R}_i, \forall i \in \Theta^v$. Explicitly, after perturbation, we have:

$$\hat{u}_{pwa}(x) = F_i x + g_i \text{ for } x \in \mathcal{R}_i^\alpha(v, \hat{v}), \tag{5.26}$$

and $\hat{u}_{pwa}(x) \neq u_{pwa}(x)$ when $x \in \mathcal{R}_i$ but $x \notin \mathcal{R}_i^\alpha$. This observation leads us to the statement of the main result where the following notation will be used:

$$\tilde{\mathcal{F}}(\mathcal{R}^\alpha) = \{y \in \mathcal{R} | \exists x \in \mathcal{R}^\alpha \text{ such that } y = Ax + B\hat{u}_{pwa}(x)\}. \tag{5.27}$$

Theorem 5.4.2 *Let $v \in V$ and its perturbation $\hat{v} = (v + \Delta v) \in \Psi^v$. The positive invariance properties of the set $\mathcal{R}^\alpha(v, \hat{v})$ with respect to $x_{k+1} = Ax + B\hat{u}_{pwa}(x_k)$ is guaranteed if $\mathcal{R}^\alpha(v, \hat{v}) \supseteq \mathcal{F}_{pwa}$ and $\tilde{\mathcal{F}}(\mathcal{R}^\alpha) \subset \mathcal{F}_{pwa}$.*

Proof 5.4.2 *From Theorem 5.3.1, it is known that the disturbances in Ψ^v guarantees the non-overlapping property of the partition for the vertex $v \in V$. This property ensures the well-possessedness of the PWA mapping $x_{k+1} = Ax + B\hat{u}_{pwa}(x_k)$. From hypotheses, the PWA control law u_{pwa} defined over the set $\mathcal{R}$ assures the positive invariance characteristics of the set $\mathcal{R}$ i.e.,*

$$x_{k+1} = Ax_k + Bu_{pwa}(x_k) \in \mathcal{R}. \tag{5.28}$$

Let us consider the perturbation on the vertex v, $v \to \hat{v}$ and the modified partition $\mathcal{P}_N(\mathcal{R}) \to \mathcal{P}_N(\mathcal{R}^\alpha)$. Even if the new PWA function $\hat{u}_{pwa}$ loses the continuity over $\mathcal{R}^\alpha$, it preserves the piecewise continuity. Thus the positive invariance can be analysed by considering the image of the vertex $\hat{v} \in \Psi^v$ and particularly by analyzing its image with respect to the set $\mathcal{F}_{pwa}$. In order to ensure the invariance characteristics of the approximated set $\mathcal{R}^\alpha$, we must ensure that the control law corresponding to the polyhedral regions of the set $\mathcal{R}_i^\alpha$ satisfy:

$$Ax + B\hat{u}_{pwa}(x) \in \mathcal{F}_{pwa}, \forall x \in \mathcal{R}^\alpha. \tag{5.29}$$

or equivalently if

$$\tilde{F}(\mathcal{R}^\alpha) \subset \mathcal{F}_{pwa}. \tag{5.30}$$

Taking into account that only one vertex v is affected by the perturbation, the variations that guarantee $\tilde{f}_{pwa}(\hat{v}) \in int(\mathcal{F}_{pwa})$ and $\hat{v} \in \Psi^v$ preserve the invariance. The collection of all the feasible vertex perturbation leads to the set:

$$\Pi^v = \{\hat{v} \in \{\Psi^v \setminus \mathcal{F}_{pwa}\} | \tilde{f}_{pwa}(\hat{v}) \in \mathcal{F}_{pwa}\}. \tag{5.31}$$

The compact set $\Pi^v \subset \Psi^v$ is describing the invariance-margin for a frontier vertex $v \in V$ and leads to the redefinition of the regions of the partition:

$$\begin{cases} \mathcal{R}_i^\alpha = conv\{\mathcal{V}(\mathcal{R}_i) \setminus v, \hat{v}\}, \ \forall i \in \Theta^v, \\ \mathcal{R}_i^\alpha = \mathcal{R}_i, \forall i \in \mathcal{I}_N \setminus \Theta^v \end{cases} \tag{5.32}$$

all by guaranteeing that $A\hat{v} + B(F_i\hat{v} + g_i) \in \mathcal{F}_{pwa}$.

Thus, we complete the proof that $\mathcal{R} \leftarrow \mathcal{R}^\alpha(v, \hat{v})$ represents the approximation of the set $\mathcal{R}$ whose vertices are perturbed and positioned in a way that the set $\mathcal{R}^\alpha$ assures the non-overlapping and the invariance property of the controller.

Remark 5.4.1 *The set Π^v in (5.31) provides a characterization of the disturbances for any vertex $v \in V$ but it should be noted that it is not a convex set. However, being constructed as the set difference of two polyhedral sets, it can be represented as a finite union of polyhedra.*

Remark 5.4.2 *The Theorem 5.4.2 provides a guarantee of positive invariance after the perturbation of a vertex $v \in V$ but does not guarantee the convexity of the set $\mathcal{R}^\alpha(v, \hat{v})$. This is particularly problematic as long as the polyhedral partition properties of the nominal PWA control might be lost. The next section provides a formal description of the conditions which enforce the convexity all by remaining tractable from the construction point of view.*

5.4.2 Margin of Convexity

Next, we define the set of vertices of the polyhedral set $\mathcal{R}$ as:

$$\breve{V} = \mathcal{V}(\mathcal{R}). \tag{5.33}$$

The set will be represented as $\breve{V} = \{v_1, v_2, \cdots, v_{\breve{r}}\}$ with $\breve{r}$ represents the number of vertices of $\mathcal{R}$. Note that the set V and $\breve{V}$ are not equivalent as long as there exist vertices of $\mathcal{R}_i$ which are located on the frontier of $\mathcal{R}$ but do not represent vertices of this later set. Now, we classify the vertices on the frontier of the set $\mathcal{R}$ into two sets. The first set is already defined in (5.33) while the second represents its complement with respect to V. This last set will contain the vertices which are not in the set $\breve{V}$ because they are not representing vertices of $\mathcal{R}$ even if they are positioned on the frontier of $\mathcal{R}$:

$$\tilde{V} = V \setminus \breve{V}. \tag{5.34}$$

Similarly, in order to facilitate the notation $\tilde{V} = \{v_1, v_2, \cdots, v_{\tilde{r}}\}$ with $\tilde{r}$ the number of vertices in this set.

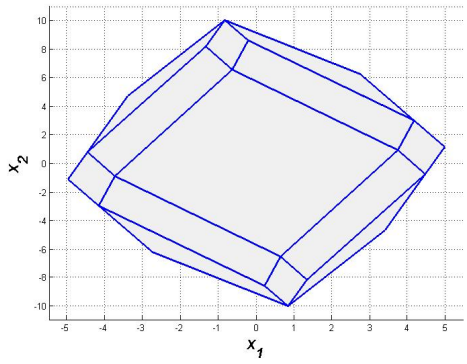
It can be observed from Theorem 5.4.2 that although the invariance is preserved after the perturbation, the polyhedral structure of the set $\mathcal{R}^\alpha$ is not necessarily preserved. This is due to the fact that perturbing any vertices from the frontier and particularly those for the set $\tilde{V}$ will result in lost of convexity when taking the poly-union of the polyhedral partition $\mathcal{R}^\alpha = \cup_{i=1}^{N} \mathcal{R}_i^\alpha$.

In order to exemplify this phenomenon, consider a polyhedral partition with 13 regions $\mathcal{R}_i$ as represented in Figure 5.7a. Next, we classify the frontier vertices into extreme and non-extreme vertices (contained in $\tilde{V}$) denoted with the help of red and blue dots respectively (those contained in $\breve{V}$). This classification can be observed from the Figure 5.7b along with the representation of the set $\mathcal{R}$. Now, we aim to analyze the perturbation of a non-extreme point $v \in \tilde{V}$ that satisfies $\hat{v} \in \Pi^v$ as given in (5.31). The non-extreme point v to be perturbed is denoted by a black dot and the perturbed vertex $\hat{v}$ is denoted by a red + symbol as illustrated in the Figure 5.7c. After the perturbation, the perturbed polyhedral regions $\mathcal{R}_i^\alpha(v, \hat{v})$ are computed and it is shown in the Figure 5.7d. It can be inferred from Figure 5.7d that the overall convexity of the poly-union of the regions $\mathcal{R}^\alpha$ is lost.

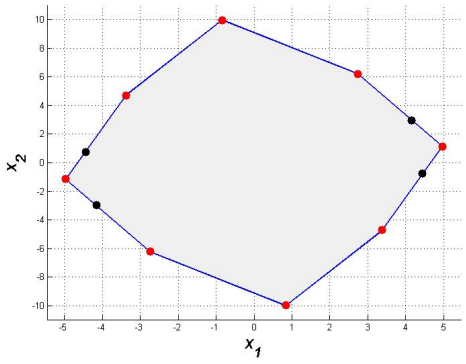
Theorem 5.4.3 *Let the system* (5.1) *and the PWA function* (5.3)*. The positive invariance with respect to the closed loop and the convexity of the feasible domain is guaranteed if a vertex $v \in V$ is perturbed towards $\hat{v} \in \Pi^v \setminus int(conv(V \setminus v))$.*

Proof 5.4.3 *The condition $\hat{v} \in \Pi^v$ was shown to guarantee the invariance. Secondly, $\hat{v} \notin int(conv(V \setminus v))$ implies that $\hat{v}$ is not redundant after perturbation. More than that, $\mathcal{R}^\alpha(v, \hat{v}) \subset \mathcal{R}$ and by virtue of the convexity of $\mathcal{R}$ it follows that $\mathcal{R} \supset conv\{\mathcal{R}^\alpha(v, \hat{v})\}$. Ultimately, $conv\{\mathcal{R}^\alpha(v, \hat{v})\} = \mathcal{R}^\alpha(v, \hat{v})$ as any point in the convex hull is part of $\mathcal{R}$ and it can be expressed as a convex combination involving the vertices of $(V \setminus v)$ and $\hat{v}$ all by preserving the non-overlapping property of the partition.*

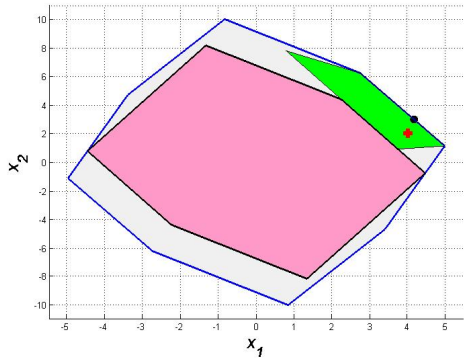
The previous Theorem offers a condition for preserving the convexity of the partition after perturbation. However, for the vertices in $\tilde{V}$ this reduces the set of admissible



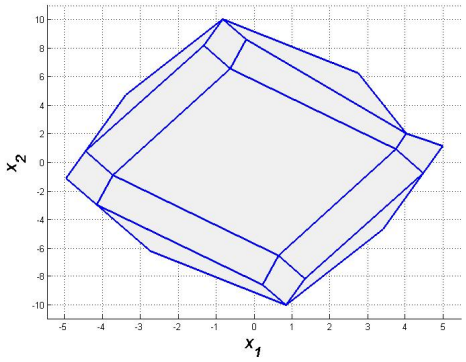
(a) Polyhedral partition with 13 regions $\mathcal{R}_i$, $\mathcal{R} = \cup_{i=1}^{N} \mathcal{R}_i$.



(b) Illustration of set $\mathcal{R}$ with extreme and non extreme points are denoted by red and black dots respectively.



(c) The pink polytope denotes the image set $\mathcal{F}_{pwa}$. The green polytope represents the sensitivity margin Ψ^v for the non extreme point v given in black dot. The red + symbol represents a point $\hat{v}$ chosen for exemplification $\hat{v}$.



(d) Illustration of $\mathcal{R}^\alpha = \cup_{i=1}^{N} \mathcal{R}_i^\alpha$ after perturbation.

Figure 5.7: Illustration for perturbation of the non-extreme vertex.

perturbations to a degenerate set. In order to offer a certain degree of freedom in the treatment of these vertices, the idea is to consider the perturbation of the vertices in the set $\breve{V}$ in a first stage and then to consider the perturbation of the vertices in the set $\tilde{V}$. Consider the convexity problem as stated for the example in Figure 5.7, we assume that all the vertices in the set $\breve{V}$ are perturbed and we have computed the final approximated set $\mathcal{R}^\alpha$ after perturbing the vertices sequentially. Starting from this approximated set given by the extreme points in vertex representation:

$$\mathcal{R}^\alpha = \{v_1^\alpha, \cdots, v_{r^\alpha}^\alpha\}. \tag{5.35}$$

In a second stage, the vertices in the set $\tilde{V}$ individually preserving the non-overlapping and the invariance property are constructed based on (5.31). Exploiting the geometrical structure of Π and the set $\mathcal{R}^\alpha$, we directly write down the constraints that satisfy the convexity property of the set $\mathcal{R}^\alpha$ and its polyhedral partition $\mathcal{P}_N(\mathcal{R}^\alpha)$ for the vertices $v \in \tilde{V}$.

$$\Pi = \{x \in \mathcal{R} | \hat{v} \in \Pi^v \setminus int(conv(V \setminus v))\}. \tag{5.36}$$

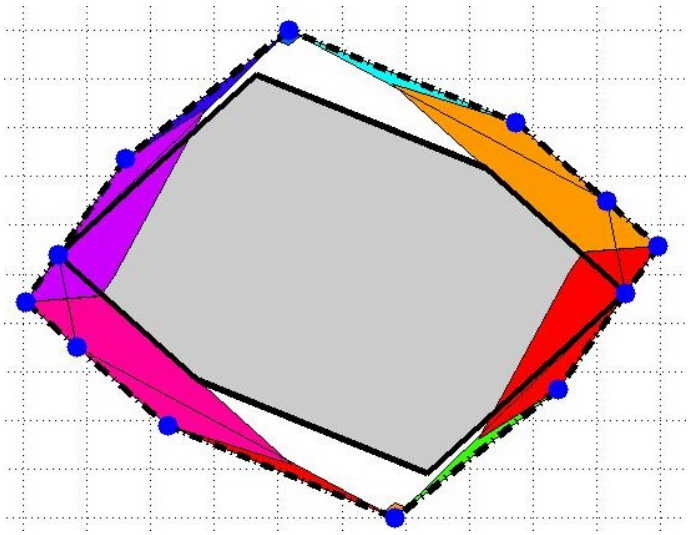


Figure 5.8: Representation of the set $\mathcal{R}$ and its image $\mathcal{F}_{pwa}$ are represented by contour in dashed lines and the contour in full lines respectively. The colored polytopes apart from white and gray ones represent the vertex sensitivity for the vertices depicted in blue dots

From Figure 5.8, it can be noted that several boundary vertices of the image set $\mathcal{F}_{pwa}$ lie on the frontier of the set $\mathcal{R}$ and perturbing those vertices will result in violating the relation $\mathcal{R}^{\alpha} \supseteq \mathcal{F}_{pwa}$, here $\mathcal{R}^{\alpha}$ denotes the approximated set for the perturbed vertex. These vertices have no full dimensional set of admissible perturbation in the first stage of the procedure if the respective vertices are treated in the first stage. It is worth to be mentioned that not all the points in V fall in this category. There after, we perturb any vertices $v \in V$ that satisfies the relation $\mathcal{R}^{\alpha}(v, \hat{v}) \supseteq \mathcal{F}_{pwa}$ and also satisfying the condition that the vertex to be strictly inside the vertex sensitivity region and preserve the convexity.

Remark 5.4.3 *Consider two vertices* $[v_1,\ v_2] \in V$*, the vertex sensitivity set* Ψ^{v_1} *for the vertex* v_1 *is computed assuming that the vertex* v_2 *is fixed and vice versa. Perturbing the vertex* v_1 *inside its vertex sensitivity set may invalidate the vertex sensitivity set computed for the vertex* v_2*. There after, we fix the position for the vertex after perturbation assuring the invariance and convexity, then recompute the set* $\mathcal{R}^{\alpha}$ *with its partition* $\mathcal{P}_N(\mathcal{R}^{\alpha})$*, here the perturbed set is assumed to be* $\mathcal{R}^{\alpha}$*. We proceed to deriving the image set for the new* $\mathcal{R}^{\alpha}$ *and recompute the vertex sensitivity sets for all the untreated vertices that lie on the boundary of the set* $\mathcal{R}^{\alpha}$ *and likewise perturb the vertices sequentially.*

Algorithm 5.4.1 Algorithm for computing the perturbed set $\mathcal{R}^{\alpha}$

Input: $\mathcal{R} = \cup_{i=1}^{N} \mathcal{R}_i, i \in \mathcal{I}_N$, $V \notin \ int(\mathcal{R})$
Output: $\mathcal{R}^{\alpha}$

Initialization : $\mathcal{R}^{\alpha} = \mathcal{R}$
LOOP Process
for $\gamma = 1$ to r **do**
  for $b_1 = 1$ to $r - \gamma - 1$ **do**
    Compute the vertex sensitivity set $\Psi^{v_{b_1}}$
  end for
  Compute the Image set $\mathcal{F}$
  for $b_2 = 1$ to $r - \gamma - 1$ **do**
    $v_{b_2}^{\alpha} = quantify(v_{b_2})$
    if $v_{b_2}^{\alpha} \in (\Psi^{v_{b_2}} \cap \mathcal{R})$ and $v_{b_2}^{\alpha} \notin int(\mathcal{F})$ **then**
      break
    end if
  end for
  for $t = 1$ to $length(\Theta^{v_{b_2}})$ **do**
    $\mathcal{R}_j^{\alpha} = Conv\{\mathcal{V}(\mathcal{R}_j) \setminus \{v_{b_2}\}, v_{b_2}^{\alpha}\}\ \forall j \in \Theta^{v_{b_2}}$
  end for
  $V(b_2, :) = [\ ]$
  $\mathcal{R} = \mathcal{R}^{\alpha}$
end for

The Algorithm 5.4.1 is responsible for computing the perturbed set $\mathcal{R}^{\alpha}$. In Algorithm 5.4.1, r denotes the number of vertices lying on the frontier of the set R. Figure 5.9 illustrates the set $\mathcal{R}$ in dashed red line along with the image $\mathcal{F}$ for each outer approximation vertex for each iteration in Algorithm 5.4.1. The image set $\mathcal{F}$ from the last iteration of the Algorithm is depicted by a green polytope in Figure 5.9. The final approximated set $\mathcal{R}^{\alpha}$ is given by a red polytope illustrated in Figure 5.10.

The polyhedral partition $\mathcal{P}_N(\mathcal{R})$ of the set $\mathcal{R}$ with 13 regions is shown in Figure 5.11a and the polyhedral partition of the set $\mathcal{R}^{\alpha}$ (outcome of the Algorithm 5.4.1) with all the displaced vertices is given in Figure 5.11b. The approximated set $\mathcal{R}^{\alpha}$ for the nominal PWA control law validates the invariance properties for all the displaced frontier vertices. Possible alternatives on the characterization of the PWA control law are enumerated in the following,

1. Starting from the PWA state-feedback dynamics that assures the controlled-invariance of the set $\mathcal{R}$ for the nominal system (1). Determine the maximum admissible set that allows perturbation on the state-space partition given by vertex representation, such that $A(v_i + \Delta v_i) + Bu_{pwa}(v_i + \Delta v_i) \in \mathcal{R}$, $\forall (v_i + \Delta v_i) \in \mathcal{R},\ i \in \mathcal{I}_r$. This particular problem leads to the computation of the robustness margin [ONB+13, NOBRA14, NORA+16]. However, this problem does not implicitly take into account the non-overlapping characterization of the PWA controller.

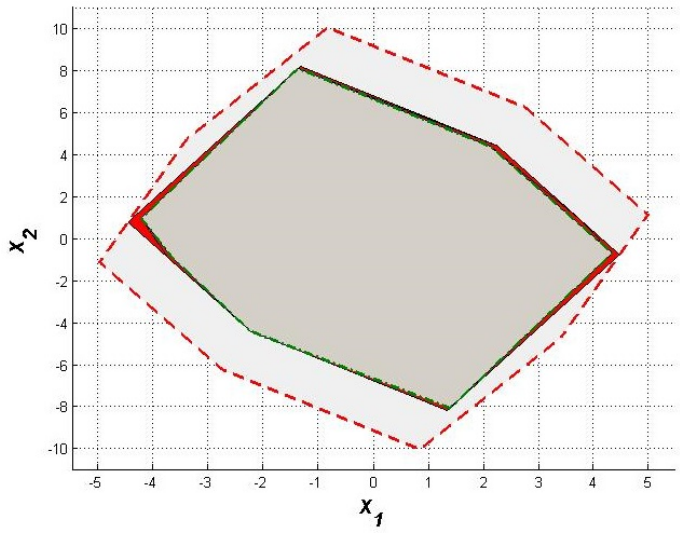


Figure 5.9: Representation of the set $\mathcal{R}$ given by contour in dashed red line. The solid polytopes inside the set $\mathcal{R}$ are the images $\mathcal{F}$ of the set $\mathcal{R}$ for each iteration in Algorithm 5.4.1.

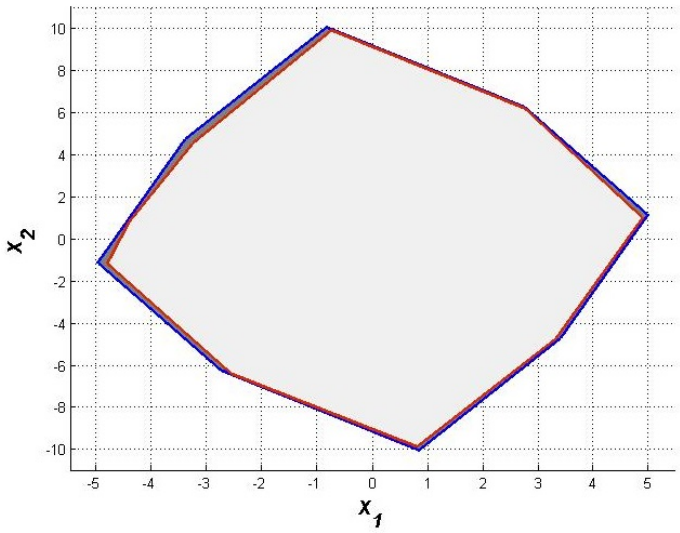
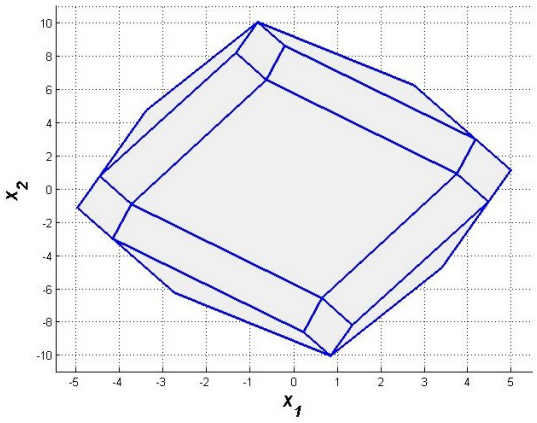


Figure 5.10: Representations of the set $\mathcal{R}$ and the output of the Algorithm 5.4.1 $\mathcal{R}^{\alpha}$ are depicted in blue and red polytope respectively.

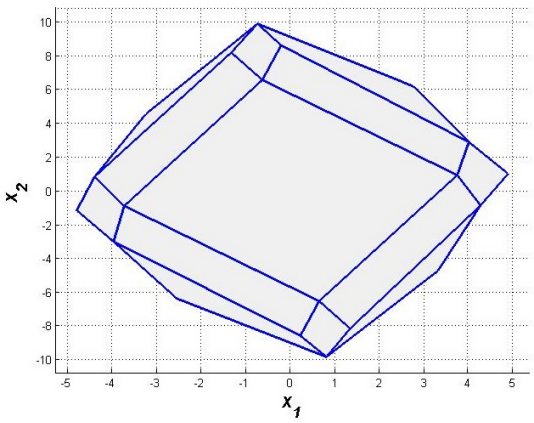
2. Extend the EMPC problem to controlled-λ contractivity with respect to the nominal system [HOB14]. The perturbation on the vertices should be chosen, such that $A(v_i + \Delta v_i) + Bu_{pwa}(v_i + \Delta v_i) \in \lambda\mathcal{R}$, $\hat{v}_i = v_i + \Delta v_i \in \mathcal{R}$, $\forall i \in \mathcal{I}_r$ and the magnitude of perturbed numerical value should be lesser than $1 - \lambda$. This methodology is discussed later in this chapter.

In the following we assume $\mathcal{R} \equiv \mathcal{R}^{\alpha}$ and we introduce a matrix $\bar{V} \in \mathbb{R}^{n \times p}$ to store all the vertices of all the polyhedral regions $\mathcal{R}_i, \forall i \in \mathcal{I}_N$ excluding the vertices on the boundary of the polyhedron $\mathcal{R}$,

$$\bar{V} = [\bar{v}_1, \bar{v}_2, \cdots, \bar{v}_p]. \tag{5.37}$$



(a) Polyhedral partitions with 13 regions (Input for the Algorithm 5.4.1).



(b) Polyhedral partition of the set $\mathcal{R}^{\alpha}$ after perturbing all the frontier vertices (Output from Algorithm 5.4.1).

Figure 5.11: Polyhedral partition with 13 regions, before and after perturbation of all the frontier vertices.

5.4.3 Treatment of one inner vertex for non-overlapping and invariance

In this subsection, we characterize the sensitivity margin with respect to the invariant property of the PWA controllers for the interior vertices of the polyhedral partition $\mathcal{P}_N(\mathcal{R}) \in \mathbb{R}^n$. Recall the vertex sensitivity margin Ψ^v from (5.18) which preserves the non-overlapping property of the polyhedral partition.

Definition 5.4.1 *Consider the polyhedral partition $\mathcal{P}_N(\mathcal{R}) \in \mathbb{R}^n$ with*

$$\mathcal{R}_i = Conv\{v_{i,1}, \cdots, v_{i,r_i}\},\ i \in \mathcal{I}_N, \tag{5.38}$$

$\bar{V} = [\bar{v}_1, \cdots, \bar{v}_p]$ and $\mathcal{R}$ is assumed to be controlled-invariant. For any $\bar{v} \in \bar{V}$ denote $\Lambda^{\bar{v}} \subset \mathcal{I}_N$ the subset of indexes of regions that satisfies $(\Psi^{\bar{v}} \cap \mathcal{R}_j) \neq \emptyset$:

$$\Lambda^{\bar{v}} = \{j \in \mathcal{I}_N | (\Psi^{\bar{v}} \cap \mathcal{R}_j) \neq \emptyset\}. \tag{5.39}$$

The set $\Upsilon^{\bar{v}} \subseteq \Psi^{\bar{v}} \subset \mathcal{R}$ is representing the invariance-vertex sensitivity for a given vertex $\bar{v}$ if $\forall(\bar{v} + \Delta\bar{v}) \in \Upsilon^{\bar{v}}$, the following properties hold for the newly constructed polyhedral partition $\mathcal{P}_N(\hat{\mathcal{R}}(\bar{v}, \bar{v} + \Delta\bar{v}))$:

1. *$\hat{\mathcal{R}}$ is controlled-invariant.*
2. *$\hat{\mathcal{R}} = \cup_{i=1}^{N} \hat{\mathcal{R}}_i$ is a polytope.*
3. *$int(\hat{\mathcal{R}}_i) \cap int(\hat{\mathcal{R}}_j) = \emptyset,\ \forall i, j \in \mathcal{I}_N, i \neq j.$*

Theorem 5.4.4 *Consider the subset of regions $\mathcal{R}_j$, $j \in \Lambda^{\bar{v}}$ that satisfies $(\Psi^{\bar{v}} \cap \mathcal{R}_j) \neq \emptyset$, of $\mathcal{P}_N(\mathcal{R})$ and $\mathcal{R}$ is assumed to be controlled-invariant then the invariance-vertex sensitivity for $\bar{v}$ is represented by a polyhedral set $\Upsilon^{\bar{v}}$.*

$$\Upsilon^{\bar{v}} = \Psi^{\bar{v}} \cap \{\bar{v} \mid A\bar{v} + B(F_j\bar{v} + g_j) \in \mathcal{R}, F_j\bar{v} + g_j \in \mathcal{U},\ \forall j \in \Lambda^{\bar{v}}\} \tag{5.40}$$

Proof 5.4.4 *Starting from the PWA state-feedback control assuring the invariance characteristics of the set $\mathcal{R}$,*

$$A\bar{v} + B(F_i\bar{v} + g_i) \in \mathcal{R}, \forall i \in \mathcal{I}_N, \tag{5.41}$$

and the vertex sensitivity described by the set $\Psi^{\bar{v}}$.
Now, locate the subset of regions of $\mathcal{P}_N(\mathcal{R})$ assuring the non-overlapping behavior for the vertex $\bar{v}$ and index it using a set $\Lambda^{\bar{v}}$.

Using (5.41) *and the description of $\Lambda^{\bar{v}}$, we are able to compute the sets denoted by $S_j^{\bar{v}}$ containing all the admissible points for the PWA controllers, given by the indexes in the set $\Lambda^{\bar{v}}$, such that the invariance property is preserved. By simply intersecting the set $S_j^{\bar{v}}$ and $\Psi^{\bar{v}}$ sequentially we obtain the invariance-vertex sensitivity set:*

$$\Upsilon^{\bar{v}} = \bigcap_{j \in \Lambda^{\bar{v}}} (S_j^{\bar{v}} \cap \Psi^{\bar{v}}) \tag{5.42}$$

It should be noted from the structural point of view that $\Upsilon^{\bar{v}}$ is a polyhedral set.

5.5 Treatment of multiple vertex perturbation

In the section, the objective is to relax in a sequential procedure the assumption of single vertex perturbation. Such an approach allows to perturb multiple vertices inside the polyhedral partition $\mathcal{P}_N(\mathcal{R})$. From the theoretical point of view we will relay on the proofs presented in the previous subsections for the perturbation of a single vertex assuring the non-overlapping characteristics of the set $\mathcal{R}$ and the invariance property of the PWA closed-loop dynamics hold, under the assumption that all the other vertices are fixed.

We propose an algorithm to consider perturbing the position of all the inner vertices sequentially as resumed in Algorithm 5.5.1. In the algorithm, p denotes the number of interior vertices of the set $\mathcal{R}$. The steps involved in transforming the set $\mathcal{R}$ to $\hat{\mathcal{R}}$ with respect to relocating the position of the vertices to the perturbed ones are described in the following.

- The first inner loop involving to compute the vertex sensitivity, sensitivity margin and Chebychev radius for the sensitivity margin set for all the inner vertices yet to be perturbed.
- Identify the vertex, also called as candidate, that has the least fragility by identifying the vertex that has the smallest Chebychev radius associated to its invariance-vertex sensitivity.
- The next step is to consider a quantization function as $f(\bar{v}) = \bar{v} + \Delta\bar{v}$, here $\Delta\bar{v}$ is a random vector satisfying $\|\Delta\bar{v}\|_\infty \leq 10^{-\epsilon}, \epsilon \in \mathbb{N}_+$.
- Update of the regions that satisfy the perturbed (quantized) vertex threshold.
- Reconstruct the set $\mathcal{R}$ and remove the candidate vertex from the matrix $\bar{V}$ that has been treated now and restart the same procedure for the remaining vertices.

Algorithm 5.5.1 Algorithm for computing the perturbed set $\hat{\mathcal{R}}$

Input: $\mathcal{R} = \cup_{i=1}^{N} \mathcal{R}_i, i \in \mathcal{I}_N$, $\bar{V} \in \ int(\mathcal{R})$.
Output: $\hat{\mathcal{R}}$
1: *Initialization* : $\hat{\mathcal{R}} = \mathcal{R}$
2: *LOOP Process*
3: **for** $\gamma = 1$ to p **do**
4: $Rad = [\ ]$
5: **for** $b_1 = 1$ to $p - \gamma - 1$ **do**
6: Compute the vertex sensitivity set $\Psi^{\bar{v}_{b_1}}$
7: Compute the sensitivity margin $\Upsilon^{\bar{v}_{b_1}}$
8: $Rad(b_1) = \Upsilon^{\bar{v}_{b_1}}$.chebychev.radius()
9: **end for**
10: $[,\ R] = sort(Rad)$
11: % Set the quantified value for $\bar{v}_{R(1)}$
12: $v_{qn} = quantify(\bar{v}_{R(1)})$
13: **if** $v_{qn} \notin \Upsilon^{\bar{v}_{R(1)}}$ **break end if**
14: **for** $t = 1$ to $length(\Theta^{\bar{v}_{R(1)}})$ **do**
15: $\hat{\mathcal{R}}_j = Conv\{\mathcal{V}(\mathcal{R}_j) \setminus \{\bar{v}_{R(1)}\}, v_{qn}\}\ \forall j \in \Theta^{\bar{v}_{R(1)}}$
16: **end for**
17: $\bar{V} = \bar{V} \setminus \bar{v}_{R(1)}$
18: $\mathcal{R} = \hat{\mathcal{R}}$
19: **end for**

5.6 Example

Consider the discrete-time linear system,

$$x_{k+1} = Ax_k + Bu_k \tag{5.43}$$

where,

$$A = \begin{bmatrix} 1.4 & 0 \\ 1.8 & -1.1 \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} 0.5 \\ 0.7 \end{bmatrix}$$

The constraints on the states are $-5 \leq [1\ 0\ ]x_k \leq 5$ and input constraint $-5 \leq u_k \leq 5$ will be considered in an MPC design. The weighing matrices $Q = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ and $R = 1$ are considered in the open-loop finite time optimization problem with prediction horizon $N_p = 2$.

Solving the EMPC problem using MPT 3.0 toolbox yields 13 affine controllers and its associated state space partitions. First, we approximated the set $\mathcal{R}$ by perturbing the vertices on the boundary of the set. A quantization function $f(v_j) = v_j + \Delta v_j,\ \forall j \in \mathcal{I}_r$ with a random variable $\|\Delta v_j\|_\infty \leq 10^{-2}$ is considered for perturbing the frontier vertices. A smaller quantizer function is chosen concerning the

volume of the operating domain. Choosing an aggressive quantizer may approximate the outer representation of the set $\mathcal{R}$ but the freedom of displacing the inner vertices closer to those of the outer ones should be taken into consideration from the performance point of view. However, the objective of the present numerical example is not to provide a minimal perturbation type of solution but to prove the validity of the proposed perturbation margins.

In the next step, we assume that the vertices on the boundary are fixed and we proceed to perturb the inner vertices in the set $\mathcal{R}$ sequentially as described in Algorithm 5.5.1. In the set $\mathcal{R}$ there are 8 inner vertices and we choose to manually displace them for this analysis and illustrative purpose (with quantizer we presented an analysis depicted in Figure 5.14). Figure 5.12 shows the functioning of the algorithm for each iteration. In the subplots from Figure 5.12, the polyhedral regions $\hat{\mathcal{R}}_i$ are presented with the vertex sensitivity and invariance-vertex sensitivity sets depicted in red and green color respectively, for the vertex that has the smallest Chebychev radius. The symbols dot and $\times$ in the subplots are the vertex candidate and the new position where the candidate will end up after perturbation. The positions of the vertex candidate for each iteration and their new position are presented in the TABLE 5.1. The regions that undergo transformation for each iteration are also given in the table along with the Chebychev radius for the candidate vertex. The numerical values of the vertices $\bar{v}$ are originally double precision representation but in the table we restricted the values to four decimal places due to space constraint.

Starting from Figure 5.12a, for the first vertex candidate, we perturb the vertex from $[-1.3314\ 8.1440]^T$ to the position $[-4.0, 1.6]^T$ there by affecting three regions with indexes 8, 12, 13. The next subplot shows the new polyhedral regions after perturbation. After the 8th iteration, the subplot 5.12i represents the final set $\hat{\mathcal{R}}$ that is the output of the algorithm 5.5.1.

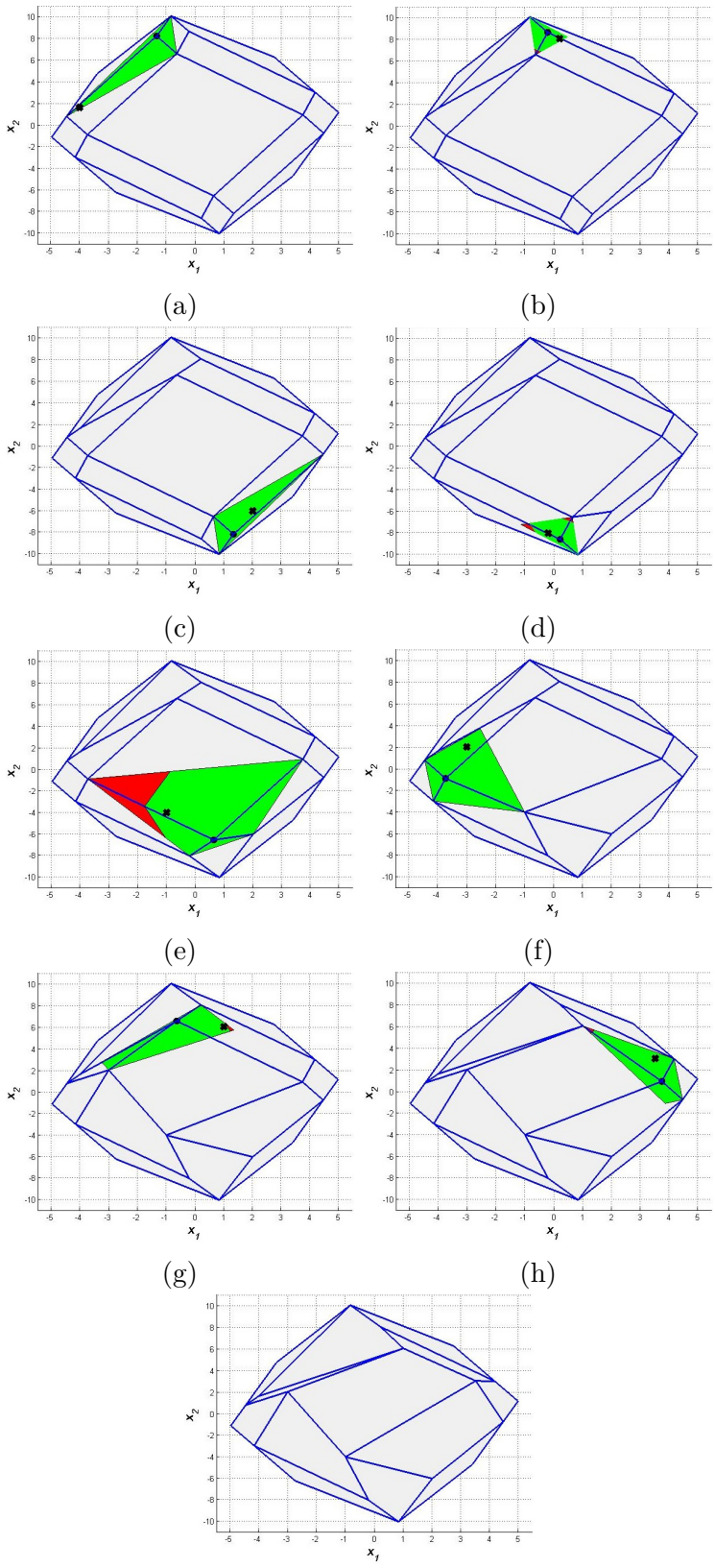


Figure 5.12: In the subplots, the polyhedral regions $\hat{\mathcal{R}}_i$ are presented with the vertex sensitivity and invariant-vertex sensitivity sets depicted in red and green color respectively. The dot and the $\times$ in the subplots are the vertex candidate and their new positions

γ	$\bar{v}$	*Chebychev radius*	$\bar{v} + \Delta\bar{v}$	$\Theta^{\bar{v}_{R(1)}}$
1	$[-1.3314, 8.1440]^T$	0.706	$[-4.0, 1.6]^T$	$\{8, 12, 13\}$
2	$[-0.2162, 8.5668]^T$	0.54	$[0.2, 8]^T$	$\{7, 11, 12\}$
3	$[1.3314, -8.1440]^T$	0.706	$[2, -6]^T$	$\{4, 6, 9\}$
4	$[0.2162, -8.5668]^T$	0.70	$[-0.2, -8]^T$	$\{1, 3, 4\}$
5	$[0.6361, -6.5235]^T$	2.128	$[-1, -4]^T$	$\{1, 2, 4, 6\}$
6	$[-3.7291, -0.9076]^T$	1.395	$[-3, 2]^T$	$\{1, 2, 5, 8\}$
7	$[-0.6361, 6.5235]^T$	1.02	$[1, 6]^T$	$\{2, 7, 8, 12\}$
8	$[3.7291, 0.9076]^T$	0.87	$[3.5, 3]^T$	$\{2, 6, 7, 10\}$

Table 5.1: This table represents the vertex candidates for each iteration and their new position along with their Chebychev radius. The last column shows the indexes of the subset of regions that are impacted by the perturbation of the vertex

From Figure 5.12, it is obvious from the subplots that no overlapping took place although a very aggressive perturbation has been tested for illustration. This validates one part of our work. In order to conclude on the closed loop behavior, we simulated for the state trajectories for the PWA controller for the outer vertices as initial states and this is presented in Figure 5.13. In the second analysis, we assume that the vertices on the frontier of the set $\mathcal{R}$ are fixed. A quantization function $f(\bar{v}_j) = \bar{v}_j + \Delta\bar{v}_j,\ \forall j \in \mathcal{I}_p$ with a random variable $\|\Delta\bar{v}_j\|_\infty \leq 0.2$ is considered for all the inner vertices in the set $\mathcal{R}$. In Figure 5.14, the polyhedral partition outlined with red lines are the ones from the approximated feasible set and the perturbed polyhedral regions $\mathcal{P}_N(\hat{\mathcal{R}})$ assuring the non-overlapping and the invariance properties for the inner vertices are illustrated in blue lines.

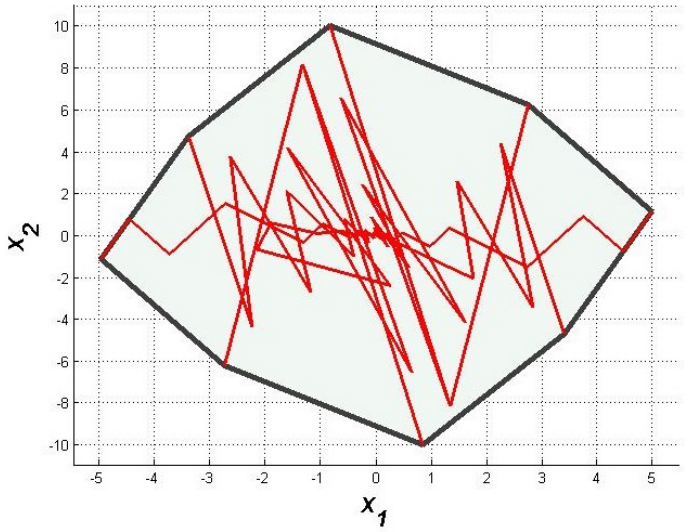


Figure 5.13: The states trajectories for the polyhedral partition, outcome of the Algorithm 5.5.1 as show in Fig 5.12i, for the vertices that lie on the boundary of the polytope.

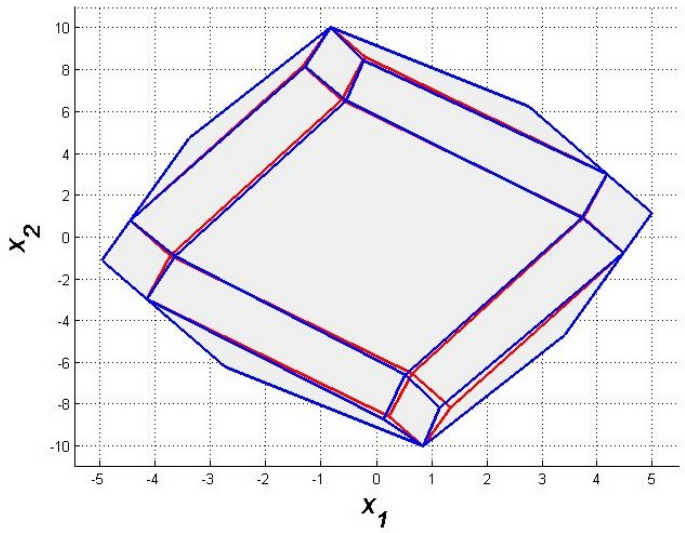


Figure 5.14: The perturbed polyhedral regions $\mathcal{P}_N(\hat{\mathcal{R}})$ for the quantization function with a random variable $\|\Delta\bar{v}_j\|_\infty \leq 0.2$ are outlined with blue lines. The polyhedral regions $\mathcal{P}_N(\mathcal{R})$ are given in red lines.

Figure 5.15 depicts the state trajectories for the perturbed polyhedral partition controlled by the nominal PWA control law for all the frontier vertices.

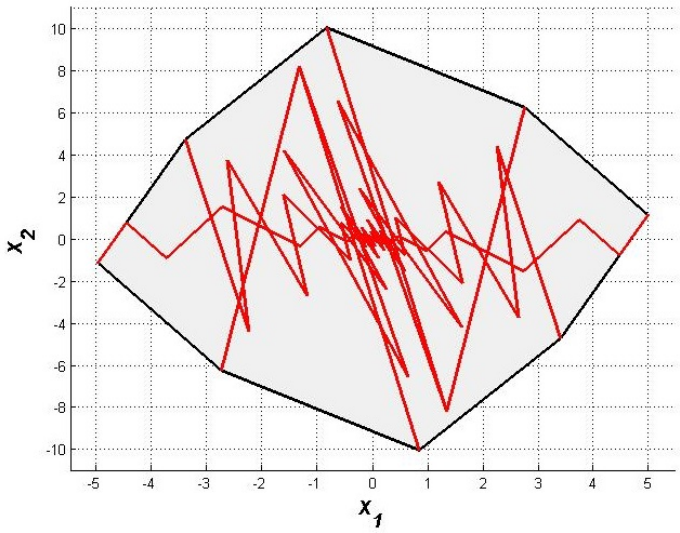


Figure 5.15: The states trajectories for the polyhedral partition, outcome of the Algorithm 5.5.1 as show in Figure 5.14, for the vertices that lie on the boundary of the polytope.

5.7 Contractivity

In this section, we aim to extend the sensitivity margin problems to the PWA control laws obtained based on controlled λ-contractivity sets. Starting from the contractive piecewise affine controller, with parameter β as a Minkowski functional of the contractive set $\mathcal{R}$, constructed upon the optimization-based formulation provided in Chapter 3 and characterized by the vertex margin set Ψ^v (5.18), we define the problem to be solved in the following.

Definition 5.7.1 *Consider a linear system and PWA control law defined over the polyhedral partition $\mathcal{P}_N(\mathcal{R}) \in \mathbb{R}^n$, where $\mathcal{R}$ is controlled λ-contractive set with respect to the closed-loop dynamics. Let $v \in \mathcal{V}(\mathcal{P}_N(\mathcal{R}))$ and denote J^v as the subset of indexes of regions having v as vertices:*

$$J^v = \{\theta \in \mathcal{I}_N | v \in \mathcal{V}(R_\theta)\}. \tag{5.44}$$

The set $\Upsilon_c^v \in \Psi^v$ is describing the sensitivity of the vertex $v \in \mathcal{V}(\mathcal{R}_i)$ with respect to the controlled λ-contractive set $\mathcal{R}$ if for any point $\hat{v} = v + \Delta v \in \Upsilon_c^v$, the collection of sets

$$\begin{aligned} &\hat{\mathcal{R}}_\theta = Conv\{\mathcal{V}(\mathcal{R}_\theta) \setminus \{v\},\ v + \Delta v\},\ \forall \theta \in J^v, \\ &\hat{\mathcal{R}}_\theta = \mathcal{R}_\theta, \forall \theta \in \mathcal{I}_N \setminus J^v \end{aligned} \tag{5.45}$$

represents a polyhedral partition: $\hat{\mathcal{P}}_N(\hat{\mathcal{R}}) = \{\hat{\mathcal{R}}_1, \cdots, \hat{\mathcal{R}}_N\}$, and the approximated set $\hat{\mathcal{R}} = \cup_{i=1}^N \hat{\mathcal{R}}$ is controlled λ-contractive and convex.

Theorem 5.7.1 *Consider a linear system and the PWA control law defined over $\mathcal{P}_N(\mathcal{R})$ where $\mathcal{R}$ is assumed to be controlled λ-contractive. The contractive-vertex for v is guaranteed within the set Υ_c^v,*

$$\Upsilon_c^v = \{\hat{v} \in \Psi^v |\ \hat{v} \notin int(\beta\lambda\mathcal{R}) \text{ with } \beta = \mathcal{M}_\mathcal{R}(\hat{v})\}. \tag{5.46}$$

Proof 5.7.1 *Starting from the contractive characterization of the PWA closed loop system, suppose that $v \in \beta\mathcal{R}$. Then it holds that*

$$Av + Bu_{pwa}(v) \in \beta\lambda\mathcal{R}. \tag{5.47}$$

From (5.47), it follows that for any vertex $v \in \beta\mathcal{R}$ the closed-loop successor state belongs to the set $\beta\lambda\mathcal{R}$. A simple way to enforce the λ-contractivity is to restrict the displacement of the vertex v such that

$$v + \Delta v \in \beta\mathcal{R} \text{ and } v + \Delta v \notin int(\beta\lambda\mathcal{R}). \tag{5.48}$$

Finally, taking the vertex sensitivity margin Ψ^v into consideration for avoiding the overlapping scenarios, the contractive margin set preserving the λ-contractivity properties can be given by:

$$\Upsilon_c^v = \{\hat{v} \in \Psi^v |\ \hat{v} \notin int(\beta\lambda\mathcal{R}) \text{ with } \beta = \mathcal{M}_{\mathcal{R}}(\hat{v})\}. \tag{5.49}$$

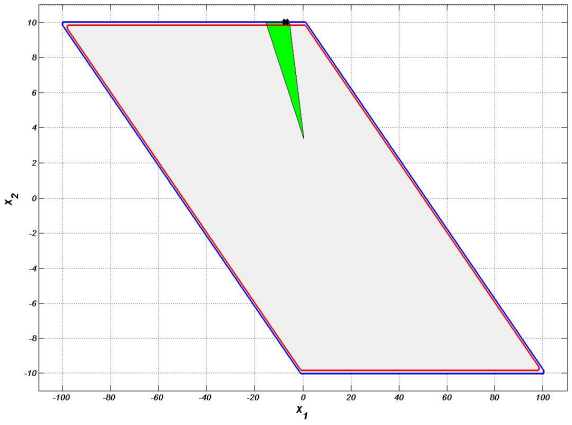


Figure 5.16: The contractive set $\beta\mathcal{R}$ is given by a polytope outlined in blue color and the vertex of interest is denoted by a black dot. The polytope outlined in red color is representing the set $\beta\lambda\mathcal{R}$. The green polytope is the vertex sensitivity region Ψ^v for the vertex given in black dot.

Figure 5.16 illustrates the computation of the contractive margin set for a frontier vertex of the set $\mathcal{R}$, where the contractive margin set Υ_c^v is given by the set intersection $\Psi^v \cap (\beta\mathcal{R} \setminus int(\beta\lambda\mathcal{R}))$.

Remark 5.7.1 *Perturbing the frontier vertices of the contractive set $\mathcal{R}$ may result in the loss of convexity while taking the poly-union of the polyhedral regions. The convexity of the poly-union $\hat{\mathcal{R}} = \cup_{i=1}^{N} \hat{\mathcal{R}}$ is preserved by following the procedure mentioned in the section 5.4.2 (Margin of Convexity).*

Example

Consider the discrete-time linear system,

$$x_{k+1} = Ax_k + Bu_k \tag{5.50}$$

where,

$$A = \begin{bmatrix} 0.9 & 0.5 \\ 0 & 0.8 \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} 0.4 \\ 0.2 \end{bmatrix}$$

and the constraints on the states are $-10 \leq [0\ 1\ ]x_k \leq 10$ and input constraint $-2 \leq u_k \leq 2$. The weighing matrices $Q = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ and $R = 1$ are considered in the finite time optimization problem. The contractive factor λ chosen is 0.99 for this example.

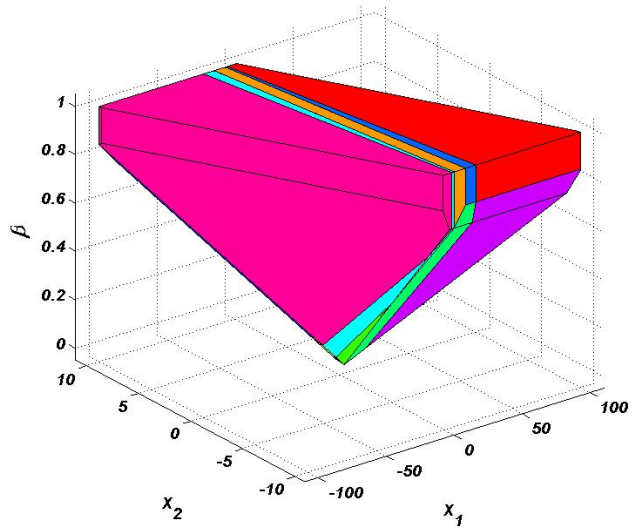


Figure 5.17: Representation of polyhedral partitions with 11 regions in the $[x,\ \beta]$-space.

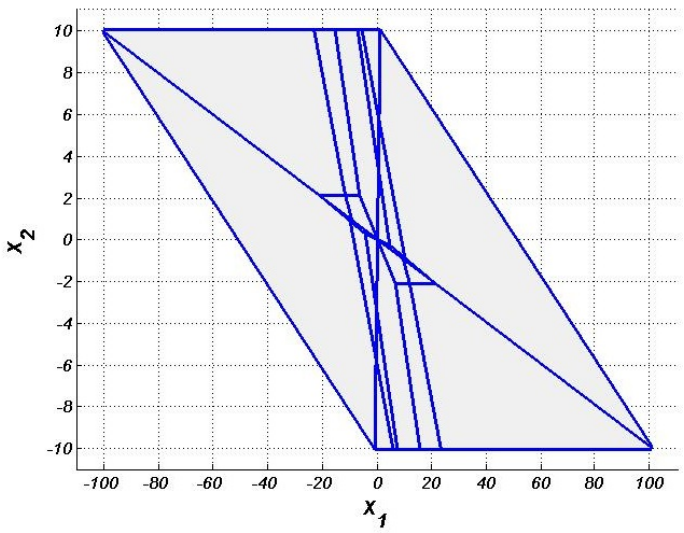


Figure 5.18: The projection of Figure 5.17 onto x-space, now the newly formed polyhedral partitions has 24 regions.

Figure 5.17 shows the polyhedral partition with 11 regions obtained from the mp-QP problem. This polyhedral partition is represented on an $[x,\ \beta]$-space. The projection of this polyhedral representation is carried out on x-plane with the help of the

algorithm mentioned in Chapter 3. The projected polyhedral partition consists of 24 regions and is shown in Figure 5.18. Starting from the polyhedral partition $\mathcal{P}_N(\mathcal{R})$

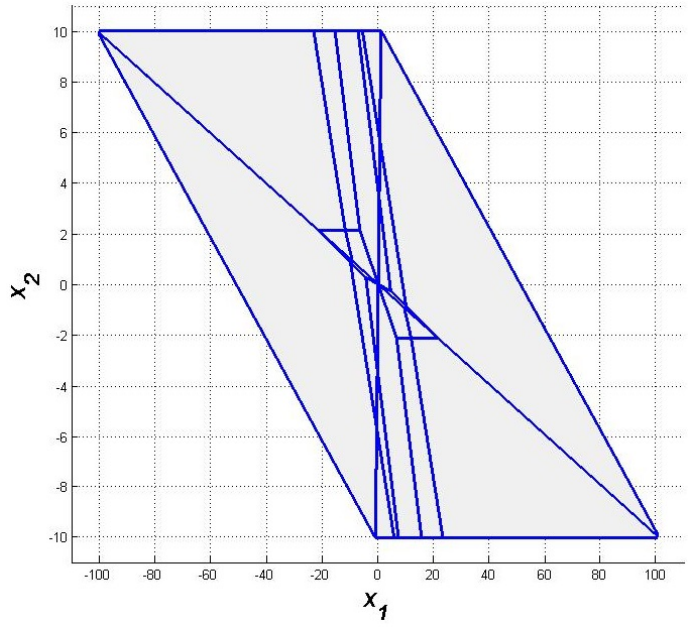


Figure 5.19: The approximated polyhedral partition $\mathcal{P}_N(\hat{\mathcal{R}})$ after displacing all the vertices in the set $\mathcal{V}(\mathcal{R})$.

illustrated in Figure 5.18, the displacement of the vertices are carried out such that the vertices $v_j, j \in \mathcal{I}_r$, where r is the total number of vertices in the set $\mathcal{V}(\mathcal{R}_i)$, belongs to the contractive-vertex sensitivity set $\Upsilon_c^{v_j}$. The newly constructed polyhedral partition after displacing all the vertices simultaneously is shown in Figure 5.19. The contraction factor being close to 1, the displacement of vertices is reduced. For exemplification we provide some vertices co-ordinates before the displacement, the factor β for these vertices and the vertices co-ordinates after displacement in Table 5.2

The quantized polyhedral partition $\mathcal{P}_N(\hat{\mathcal{R}})$ is controlled λ-contractive for the nominal PWA control law and this is proved by certificating the states trajectories emanating from all the vertices of the set $\mathcal{V}(\hat{\mathcal{R}})$. The state trajectories considering all the vertices in the set $\mathcal{V}(\hat{\mathcal{R}}_i)$ as initial states is illustrated in Figure 5.20. The contraction of the state trajectories given by the parameter β with respect to the time is show in Figure 5.21.

v	β	$v + \delta v$
$[-7.2595,\ 10.0000]^T$	1.0	$[-7.2611,\ 9.9911]^T$
$[-15.5547,\ 10.0000]^T$	1.0	$[-15.5510,\ 9.9953]^T$
$[100.0000,\ -9.9020]^T$	1.0	$[99.9962,\ -9.9004]^T$
$[-4.4460,\ 0.2616]^T$	0.0615	$[-4.4354,\ 0.2607]^T$

Table 5.2: This table represents the vertex co-ordinates for each iteration and their new position. The second column shows the values of the β parameter associated to the vertex co-ordinates.

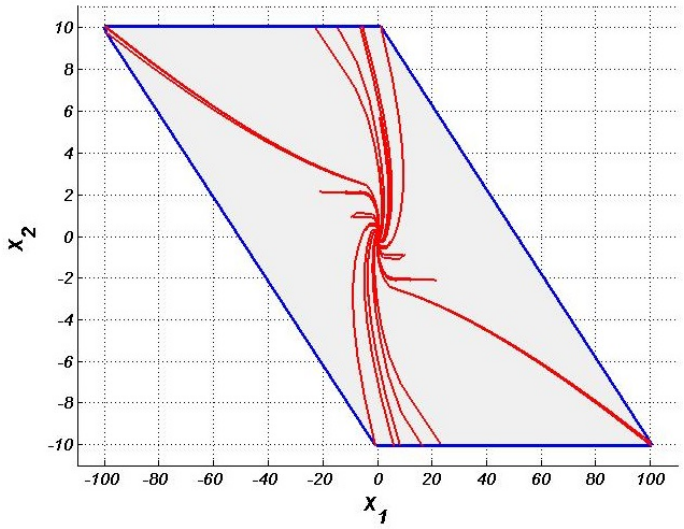


Figure 5.20: The state trajectories for the vertices in the approximated polyhedral partition showing all the trajectories are λ-contractive.

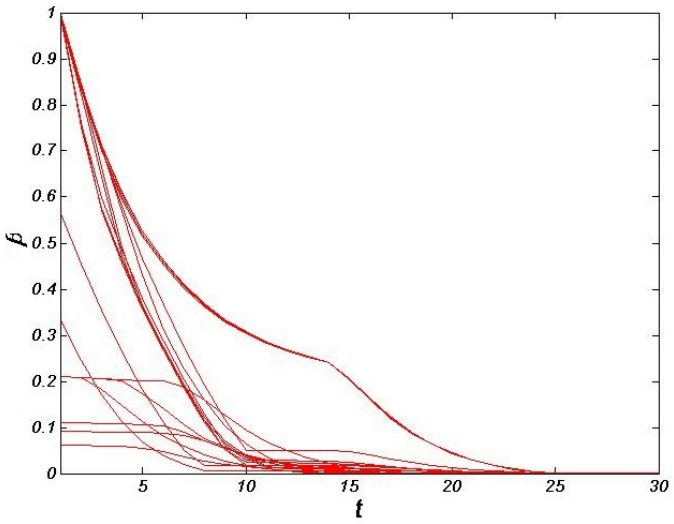


Figure 5.21: The time-simulation presenting the evolution of the β parameters for all the state trajectories depicted in Figure 5.20.

5.8 Conclusion

In this work the analysis on the perturbation of the vertex representation has been presented. The vertex sensitivity that characterizes the admissible perturbation for assuring the non-overlapping properties has been derived. The sensitivity set that preserve the invariance characteristics in the event of perturbation for the PWA control has been computed. It was shown that a perturbed polyhedral partition can be constructed by treating sequentially each vertex with a higher priority on those with a small sensitivity margin.

Chapter 6

PWA function evaluations on CPU and GPU for ADMM and PHA

Contents

6.1 Introduction 101
6.2 Time-splitting approach for Explicit MPC 104
6.2.1 Off-line Computation 106
6.2.2 On-line computation and evaluation 108
6.2.3 Evaluation of PWA functions for Matrix Input 109
6.2.4 Examples 110
6.3 Multi-Stage Stochastic Programming 114
6.3.1 Decomposition of Scenario tree 115
6.3.2 Progressive Hedging Algorithm 115
6.3.3 Example 118
6.4 Conclusions 122

6.1 Introduction

Explicit model predictive control (EMPC) belongs to a class of constrained model based optimization methods whose solution is synthesized in the form of a set of piecewise affine (PWA) feedback laws [AB09, BMDP02]. This kind of control law is easily implemented for real-time systems with small state-space models and relatively fast dynamics. Although explicit control law has advantages over standard MPC, it also comes with high computational cost for higher order systems and/or for large prediction horizon due to the important complexity of the state-space partition.

In this chapter, we aim to enable explicit solutions for the linear discrete-time dynamics with longer prediction horizons and we also treat linear dynamics with probabilistic uncertainties controlled using explicit control laws. Explicit solutions

are simple algebraic functions (i.e), operate within matrix-matrix or matrix-vector multiplication framework. Such expressions are parallelizable by the Central Processing Unit (CPU) and it is also possible to enable the Graphical Processing Unit (GPU) to breakdown large matrices to the available sub processors, to obtain computationally faster operations. In order to empower explicit MPC for longer prediction horizon, the technique we rely on is called operator splitting [SKW13]. The splitting is carried out over the prediction horizon and we have number of subproblems equal to the length of the prediction horizon. Using the Explicit MPC framework, we reduce the $N + 1$ (prediction horizon length) subproblems into three subproblems. We use ADMM (Alternating Direction Method of Multipliers) [SKW13, NLR$^+$15, Lu14, AHW12, DLM15, BPC$^+$11] approach to satisfy the consensus constraints occurred during the decoupling and for convex convergence. This ADMM approach incorporated with the EMPC formulation is solved with CPU. Finally, we compare the computational complexity for the above mentioned formulation for CPU besides the general explicit MPC formulation (without time splitting).

The uncertainties in model parameters, neglected parameter dynamics and disturbances make the plant (real system) differ from the mathematical model. Since the mathematical model is used to make predictions for the future system states, these predicted states will be less accurate as compared to that of the plant. Therefore, to improve the accuracy of prediction, the uncertainties have to be modeled. Recently much research has been devoted to tackle uncertainties in dynamical system for standard MPC [GKMS14, MRFA06]. Robust Min-Max approach [SM98, KM03], Robust tube based MPC [MKF11], and Multi-stage MPC [LFE13], are few of them. In this chapter we consider a discrete-time linear system affected by probabilistic uncertainty. This probabilistic uncertainty is modeled with a scenario tree approach also called multi-stage MPC. The scenario tree is designed with different uncertainties or disturbances realizations. However, the scenario tree approach for the MPC problem comes with huge computational complexity, arising from the modeling of the disturbance realizations. Therefore, we employ a decomposition algorithm for the scenario tree termed as Progressive Hedging Algorithm (PHA) [LFE13, RW91, HW91, BBG14]. The reason behind the adoption of this algorithm is to decompose the scenario tree into different individual scenarios. The decomposition leads to a finite number of low-complexity scenarios which can be solved either sequentially or parallely. The idea is to parallelize all the scenarios with mp-QP cost functions and to explore the advantages of using explicit solutions on GPU over CPU.

The main idea of this chapter is to analyze the complexity of the explicit solutions in terms of computational complexity and memory storage. The objective of this chapter is listed in the following.

1. The comparison of the sequential and parallel evaluations of the PWA functions for the ADMM algorithm.

2. The comparison of the parallel evaluations of the PWA functions for the PHA on CPU and GPU.

Consider a linear discrete-time system given by,

$$x_{t+1} = Ax_t + Bu_t \tag{6.1a}$$
$$y_t = Cx_t, \tag{6.1b}$$

where, $x_t \in \mathbb{R}^n$, $u_t \in \mathbb{R}^m$ and $y_t \in \mathbb{R}^s$ denote the state vector, input variables and output variables respectively at time t. The constraints on the system states and input variables are represented by,

$$\mathcal{X} = \{x : H_x x \leq h_x, \quad H_x \in \mathbb{R}^{n_x \times n}, h_x \in \mathbb{R}^{n_x}\}, \tag{6.2a}$$
$$\mathcal{U} = \{u : H_u u \leq h_u, \quad H_u \in \mathbb{R}^{m_u \times m}, h_u \in \mathbb{R}^{m_u}\}. \tag{6.2b}$$

The state and input constraints are bounded polytopic sets. For explicit MPC, the quadratic cost function with the terminal and stage costs is transformed into a multi-parametric Quadratic Programming (mp-QP) problem as shown in Chapter 2. Considering a discrete-time system given by (6.1) subject to constraints (6.2), the equivalent mp-QP problem is given as

$$J(x, z) = \min_z \frac{1}{2} z^T H z + x^T F^T z + \frac{1}{2} x^T Y x, \tag{6.3a}$$
$$s.t \qquad Gz \leq W + Sx. \tag{6.3b}$$

The optimal solution for (6.3) is represented by a finite set of affine functions defined over the polyhedral partition of the set $\mathcal{R}$, where $\mathcal{P}_M(\mathcal{R}) = [\mathcal{R}_1, \mathcal{R}_2 \cdots \mathcal{R}_M]$.

The explicit control law, corresponding to the optimum argument of (6.3), is synthesized in terms of the piecewise affine function defined over the polyhedral partition of the set $\mathcal{R}$ and it can be described by,

$$u_{pwa}(x_t) = F_i x_t + g_i, \quad \forall x_t \in \mathcal{R}_i. \tag{6.4}$$

Consider the polyhedral set $\mathcal{R}$ and its partition sets $[\mathcal{R}_1, \cdots, \mathcal{R}_M]$, the half-space representation of the set $\mathcal{R}_i, i \in \mathcal{I}_M$ can be written as:

$$\begin{aligned} \mathcal{R}_1 &= \{x : H_1 x \leq h_1, \quad H_1 \in \mathbb{R}^{p_1 \times n_p}, h_1 \in \mathbb{R}^{p_1}\} \\ \mathcal{R}_2 &= \{x : H_2 x \leq h_2, \quad H_2 \in \mathbb{R}^{p_2 \times n_p}, h_2 \in \mathbb{R}^{p_2}\} \\ &\vdots \\ \mathcal{R}_M &= \{x : H_M x \leq h_M, \quad H_M \in \mathbb{R}^{p_M \times n_p}, h_M \in \mathbb{R}^{p_M}\} \end{aligned} \tag{6.5}$$

Definition 6.1.1 *Consider for a given $x \in \mathbb{R}^n$ and the half-space representation of the set $\mathcal{R}_i$ given in* (6.15)*, the point location function of the polyhedral partition of $\mathcal{R}$ is given by,*

$$\textbf{\textit{all}}\ ([H_i\ h_i].[x; -1] < 10^{-\epsilon})^1, \ \forall i \in \mathcal{I}_M, \ \epsilon \in \mathbb{N}_+, \tag{6.6}$$

where i indicates the set which contains x.

[1]For a given vector A, the function *all(A)* returns logical 1 (true) if all the elements are nonzero and returns logical 0 (false) if one or more elements are zero.

6.2 Time-splitting approach for Explicit MPC

We consider the generic finite-time open loop optimal control formulation given in the form,

$$\min \quad \sum_{t=0}^{N} c(x_t, u_t) \tag{6.7a}$$

$$\text{subject to:} \quad x_0 = x(t), t = 0. \tag{6.7b}$$

with the extended-valued cost $c(x_t, u_t)$ given by

$$c(x_t, u_t) = \; x_t^T Q x_t + u_t^T R u_t \tag{6.8a}$$

$$\text{We consider also the constraints::} \quad x_{t+1} = Ax_t + Bu_t, \tag{6.8b}$$

$$x_t \in \mathcal{X}, u_t \in \mathcal{U} \tag{6.8c}$$

We are splitting the problem (6.7) into $N+1$ smaller stage-wise subproblems. In the following, the subscript $_t$ of the decision variables $x_t^{(t)}$, $u_t^{(t)}$ denotes the time and the superscript $^{(t)}$ denotes the index of the subproblems (subproblems for each time instant).

In order to conduct the time splitting, we need to ensure the coupling constraints, that occurred due to the decomposition over time are satisfied. The coupling constraints can be overcome by introducing a copy of each variable that arises from coupling and thereafter imposing the consensus constraints on the global variables and their associated copies. Rewriting the extended-valued cost (6.8) for the time-splitting, we obtain:

$$\min \quad c(x_t^{(t)}, u_t^{(t)}) = \; x_t^{(t)T} Q x_t^{(t)} + u_t^{(t)T} R u_t^{(t)} \tag{6.9a}$$

$$\text{subject to:} \quad x_{t+1}^{(t)} = Ax_t^{(t)} + Bu_t^{(t)}, \quad t = 0, \cdots, N-1 \tag{6.9b}$$

$$\hat{z}_{t+1} = x_{t+1}^{(t+1)}, \quad t = 0, \cdots, N-1 \tag{6.9c}$$

$$\hat{z}_{t+1} = x_{t+1}^{(t)}, \quad t = 0, \cdots, N-1 \tag{6.9d}$$

$$x_t^{(t)} \in \mathcal{X}, u_t^{(t)} \in \mathcal{U} \tag{6.9e}$$

The global variable $\hat{z}$ brings the local subproblem copies $x_t^{(t)}$ and $x_t^{(t+1)}$ in accordance, for example $x_1^{(0)}$ should be equal to $x_1^{(1)}$ and we enforce this by using the global variable $\hat{z}_1 = x_1^{(0)} = x_1^{(1)}$. Therefore each subproblem has two variables, the current state and the input. Now, we introduce and define the dual variables used to deal with consensus equality constraints:

1. $\hat{w}_t$ related with $x_t^{(t)} = \hat{z}_t, \quad \forall t = 1, \cdots, N$,
2. $\hat{v}_t$ related with $x_t^{(t-1)} = \hat{z}_t, \quad \forall t = 1, \cdots, N$.

For the time instant $t = 0$, we need to solve for the coupling constraint for the succeeding state $x_1^{(0)} = x_1^{(1)} = \hat{z}_1$. But for time instant $t = 1 \cdots N-1$, we need to satisfy two coupling constraints: one for the preceding state with the $(t-0)^{th}$

subproblem and one for the succeeding state with the $(t+1)^{th}$ subproblem. By consequence in this case we need two slack variables. For time instant $t = N$, we need to solve only for the preceding state $x_N^{(N)}$ and link it to the $(N-1)^{th}$ subproblem. For further details on time splitting operator for standard MPC, the interested author is referred to [SKW13].

The optimization variables for each subproblem is represented compactly by the vector,

$$\hat{x}_t = (x_t^{(t)}, u_t^{(t)}),\ t = 0, \cdots, N, \tag{6.10}$$

here $\hat{x}_t \in \mathbb{R}^{n+m}$.

Remark 6.2.1 *In [SKW13], the optimization variables of each subproblem are $(x_t^{(t)}, u_t^{(t)}, x_{t+1}^{(t)})$. The state $x_{t+1}^{(t)}$ is only needed for the dual updates. In this work we consider only $(x_t^{(t)}, u_t^{(t)})$ as optimization variables in order to reduce complexity of the explicit solutions. Further the state $x_{t+1}^{(t)}$ is computed using the variables $(x_t^{(t)}, u_t^{(t)})$ during the dual update stage.*

Before pursuing with the developments, let us introduce some matrices obtained from rewriting the finite-time optimal control problem,

$$P_t = \mathbf{diag}(Q_t, R_t) \in \mathbb{R}^{(n+m)\times(n+m)}, \forall t = 0, \cdots, N \tag{6.11a}$$

$$\hat{A} = [A \quad B] \in \mathbb{R}^{n\times(n+m)} \tag{6.11b}$$

$$G_0 = [I_{n\times n} \quad 0] \in \mathbb{R}^{n\times(n+m)} \tag{6.11c}$$

and note that we refer to $\rho > 0$ as the step-size parameter. Now, we write the cost function of the ADMM problem with dual variables for these subproblems. It should be noted that the $N+1$ subproblems can be represented by three different cost functions given in (6.12).
For subproblem at the time instant $t = 0$, the optimization problem is constructed based on the optimization variable $\hat{x}_0^T \in \mathbb{R}^{n+m}$ and the parameters $[\hat{z}_1^T\ \hat{v}_1^T]^T \in \mathbb{R}^{2n}$. For the subproblems at the time instant $t = 1, \cdots, N-1$, we have the same optimization problem given with the optimization variable $\hat{x}_t^T \in \mathbb{R}^{n+m} \in \mathbb{R}^{n+m}$ and the parameters $[\hat{z}_t^T\ \hat{z}_{t+1}^T\ \hat{v}_{t+1}^T\ \hat{w}_t^T]^T \in \mathbb{R}^{4n}$. Similarly for the subproblem at time instant $t = N$, the optimization problem consists of $\hat{x}_N^T \in \mathbb{R}^{n+m}$ as optimization variable and $[\hat{z}_N^T\ \hat{w}_N^T]^T$ as parameters.

for t = 0

$$
\begin{aligned}
J = \min_{\hat{x}_0} & \frac{1}{2}\hat{x}_0^T P_0 \hat{x}_0 - \rho\ \hat{v}_1^T(\hat{A}\hat{x}_0 - \hat{z}_1) + \rho/2 \\
& ||\hat{A}\hat{x}_0 - \hat{z}_1||_2^2 \\
\text{subject to: } & \hat{x}_0 \in \mathcal{X} \times \mathcal{U} \\
& G_0\hat{x}_0 = x_{int} \\
& x_1^{(0)} = Ax_0^{(0)} + Bu_0^{(0)} \\
& \hat{z}_1 = x_1^{(0)} = x_1^{(1)}, \rho > 0 \\
& \text{Optimization variables: } \hat{x}_0^T \\
& \text{Parameters: } [\hat{z}_1^T\ \hat{v}_1^T]^T
\end{aligned}
$$

for t = 1, ⋯, N-1

$$
\begin{aligned}
J = \min_{\hat{x}_t} & \frac{1}{2}\hat{x}_t^T P_t \hat{x}_t - \rho\ \hat{v}_{t+1}^T(\hat{A}\hat{x}_t - \hat{z}_{t+1}) + \rho/2||\hat{A}\hat{x}_t \\
& -\hat{z}_{t+1}||_2^2 - \rho\hat{w}_t^T(G_0\hat{x}_t - \hat{z}_t) + \rho/2||G_0\hat{x}_t - \hat{z}_t||_2^2 \\
\text{subject to: } & \hat{x}_t \in \mathcal{X} \times \mathcal{U} \\
& x_{t+1}^{(t)} = Ax_t^{(t)} + Bu_t^{(t)} \\
& \hat{z}_{t+1} = x_{t+1}^{(t)} = x_{t+1}^{(t+1)} \\
& \hat{z}_t = x_t^{(t-1)} = x_t^{(t)}, \rho > 0 \\
& \text{Optimization variables: } \hat{x}_t^T \\
& \text{Parameters: } [\hat{z}_t^T\ \hat{z}_{t+1}^T\ \hat{v}_{t+1}^T\ \hat{w}_t^T]^T
\end{aligned}
\tag{6.12}
$$

for t = N

$$
\begin{aligned}
J = \min_{\hat{x}_N} & \frac{1}{2}\hat{x}_N^T P_N \hat{x}_N - \rho\ \hat{w}_N^T(G_0\hat{x}_N - \hat{z}_N) + \\
& \rho/2||G_0\hat{x}_N - \hat{z}_N||_2^2 \\
\text{subject to: } & \hat{x}_N \in \mathcal{X} \times \mathcal{U} \\
& x_{N+1}^{(N)} = Ax_N^{(N)} + Bu_N^{(N)} \\
& \hat{z}_N = x_N^{(N-1)} = x_N^{(N)}, \rho > 0 \\
& \text{Optimization variables: } \hat{x}_N^T \\
& \text{Parameters: } [\hat{z}_N^T\ \hat{w}_N^T]^T
\end{aligned}
$$

6.2.1 Off-line Computation

The resulting mp-QP solutions, for the cost functions in (6.12), are given by three finite sets of affine functions defined over the partition of the sets $\mathcal{R}^0, \mathcal{R}^t$ and $\mathcal{R}^N$ whose solutions are associated with the time instants $t = 0,\ t = 1, \cdots, N -$

1 and, $t = N$.

$$\begin{aligned}\mathcal{R}^0 &= [\mathcal{R}_1^0, \cdots, \mathcal{R}_i^0, \cdots, \mathcal{R}_{M_0}^0],\ i \in \mathcal{I}_{M_0} \\ \mathcal{R}^t &= [\mathcal{R}_1^t, \cdots, \mathcal{R}_i^t, \cdots, \mathcal{R}_{M_t}^t],\ i \in \mathcal{I}_{M_t} \\ \mathcal{R}^N &= [\mathcal{R}_1^N, \cdots, \mathcal{R}_i^N, \cdots, \mathcal{R}_{M_N}^N],\ i \in \mathcal{I}_{M_N}\end{aligned} \tag{6.13}$$

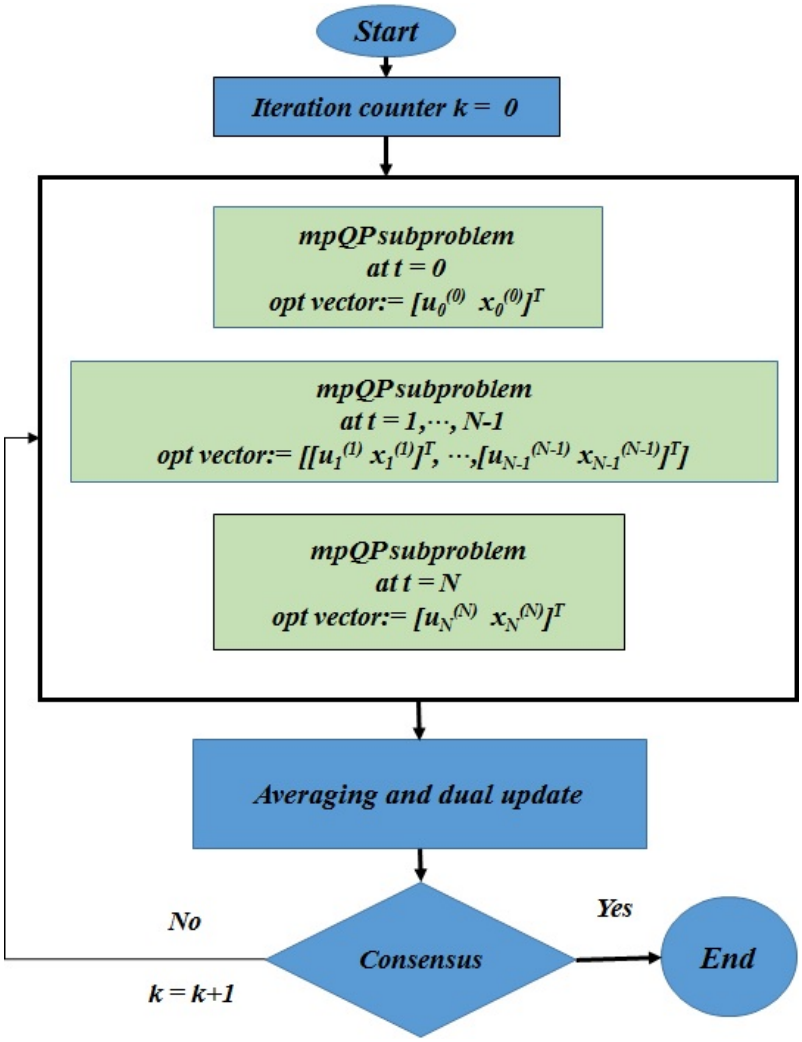


Figure 6.1: Representation of ADMM algorithm

The schematic view of the ADMM algorithm based on the explicit solution of the problems described in (6.12) can be found in Figure 6.1. The associated PWA function matrices are given by,

$$\begin{aligned}F^0 &= [F_1^0, \cdots, F_{M_0}^0] \in \mathbb{R}^{(n+m)\times(2n\times M_0)} \\ G^0 &= [G_1^0, \cdots, G_{M_0}^0] \in \mathbb{R}^{(n+m)\times M_0} \\ F^t &= [F_1^t, \cdots, F_{M_t}^t] \in \mathbb{R}^{(n+m)\times(4n\times M_t)} \\ G^t &= [G_1^t, \cdots, G_{M_t}^t] \in \mathbb{R}^{(n+m)\times M_t} \\ F^N &= [F_1^N, \cdots, F_{M_N}^N] \in \mathbb{R}^{(n+m)\times(2n\times M_N)} \\ G^N &= [G_1^N, \cdots, G_{M_N}^N] \in \mathbb{R}^{(n+m)\times M_N}\end{aligned} \tag{6.14}$$

Consider the polyhedral set $\mathcal{R}^0$ and its partition sets $[\mathcal{R}_1^0, \cdots, \mathcal{R}_i^0, \cdots, \mathcal{R}_{M_0}^0]$, the half-space representation of the set $\mathcal{R}_i^0,\ i \in \mathcal{I}_{M_0}$ can be written by,

$$\begin{aligned} \mathcal{R}_1^0 &= \{x : H_1^0 x \leq h_1^0, \quad H_1^0 \in \mathbb{R}^{p_1^0 \times 2n}, h_1^0 \in \mathbb{R}^{p_1^0}\}, \\ &\vdots \\ \mathcal{R}_{M_0}^0 &= \{x : H_{M_0}^0 x \leq h_{M_0}^0, \quad H_{M_0}^0 \in \mathbb{R}^{p_{M_0}^0 \times 2n}, h_{M_0}^0 \in \mathbb{R}^{p_{M_0}^0}\}. \end{aligned} \tag{6.15}$$

Now we introduce a matrix H_0^{all} that stores all the half-spaces for the polyhedral partition set $\mathcal{R}^0$,

$$H_0^{all} = \begin{bmatrix} H_1^0 & h_1^0 \\ & \vdots \\ H_{M_0}^0 & h_{M_0}^0 \end{bmatrix}$$

here $H_0^{all} \in \mathbb{R}^{H_0^{hyp} \times (2n+1)}$ and $H_0^{hyp} = p_1^0 + p_2^0 + \cdots + p_{M_0}^0$ denotes the total number of half-spaces of the polyhedral partition of the set $\mathcal{R}^0$. Similarly, matrices $H_t^{all} \in \mathbb{R}^{H_t^{hyp} \times (4n+1)}$ and $H_N^{all} \in \mathbb{R}^{H_N^{hyp} \times (2n+1)}$ store all the half-spaces for the polyhedral partition sets $\mathcal{R}^t$ and $\mathcal{R}^N$ respectively. The dimension H_t^{hyp} and H_N^{hyp} denote the total number of half-spaces of the polyhedral partition of the set $\mathcal{R}^t$ and $\mathcal{R}^N$ respectively.

Binary matrices $H_0^{ind} \in \mathbb{R}^{H_0^{hyp} \times M_0}, H_t^{ind} \in \mathbb{R}^{H_t^{hyp} \times M_t}$ and, $H_N^{ind} \in \mathbb{R}^{H_N^{hyp} \times M_N}$ are employed to index the half-spaces for the associated polyhedral partitions of the set $\mathcal{R}^0$, $\mathcal{R}^t$ and $\mathcal{R}^N$ respectively. Where, M_0, M_t and M_N denote the number of polyhedral partition of the sets $\mathcal{R}^0$, $\mathcal{R}^t$ and $\mathcal{R}^N$ respectively.

The integer row vectors, $H_0^{tot} \in \mathbb{R}^{1 \times M_0}, H_t^{tot} \in \mathbb{R}^{1 \times M_t}$and $H_N^{tot} \in \mathbb{R}^{1 \times M_N}$ are used to store the total number of half-spaces for all polyhedral partition of the polyhedral sets $\mathcal{R}^0$, $\mathcal{R}^t$ and $\mathcal{R}^N$ respectively. For example, the vector H_0^{tot} can be represented by:

$$H_0^{tot} = [p_1^0, p_2^0, \cdots, p_{M_0}^0].$$

6.2.2 On-line computation and evaluation

Evaluation of PWA functions for parameters given in vector form

For the mp-QP problems associated with time instant $t = 0$ and $t = N$, we are able to evaluate PWA functions by considering the given parameters in a vector form.

Consider for a given vector $\hat{s}$ the PWA optimal solution is given by the following evaluations,

$$\begin{aligned} &H_t^{tmp} = H_t^{all} * [\hat{s};\ -1],\ t = 0 \text{ or } N, \\ &H_t = H_t^{tmp} < 10^{-\epsilon}, \epsilon \in \mathbb{N}_+,\ t = 0 \text{ or } N, \\ &H_t^{in} = H_t * H_t^{ind},\ t = 0 \text{ or } N, \\ &H_t^{which} = find((H_0^{in} == H_t^{tot}) > 0),\ t = 0 \text{ or } N, \\ &[u_t^{(t)} x_t^{(t)}]^T = F^t(:, :, H_t^{which}(1)) * \hat{s} + G^t(:, H_t^{which}(1)), \\ &\qquad\qquad t = 0 \text{ or } N, \end{aligned} \tag{6.16}$$

Note that for the time instant $t = 0$, the vector $\hat{s} = [\hat{z}_1^T \ \hat{v}_1^T]^T$ while for $t = N$, the vector $\hat{s} = [\hat{z}_N^T \ \hat{w}_N^T]^T$.

6.2.3 Evaluation of PWA functions for Matrix Input

For the mp-QP problems associated with time instant $t = 1, \cdots, N-1$, we can evaluate the PWA function for all the subproblems at the same time.

Consider for a given matrix $\hat{S} = [\hat{S}_1, \cdots, \hat{S}_{N-1}] \in \mathbb{R}^{4n \times (N-2)}$ with
$\hat{S}_t = [\hat{z}_t^T \ \hat{z}_{t+1}^T \ \hat{v}_{t+1}^T \ \hat{w}_t^T]^T, t = 1, \cdots, N-1$, the PWA solution is given by the following steps

$$
\begin{aligned}
&H_t^{tmp} = H_t^{all} * [\hat{S};\ (-1 * \mathbf{I}_{1\times N-2})],\\
&H_t = H_t^{tmp} < 10^{-\epsilon}, \epsilon \in \mathbb{N}_+,\\
&H_t^{in} = H_t * H_t^{ind},\\
&H_t^{which} = (((H_t^{inT} == (H_t^{tot} * \mathbf{I}_{(1\times size(H_t^{in},1))}) > 0))\\
&\begin{bmatrix} u_1^{(1)} & \cdots & u_{N-1}^{(N-1)} \\ x_1^{(1)} & \cdots & x_{N-1}^{(N-1)} \end{bmatrix} =\\
&\qquad\qquad F^t(:,:,H_t^{which}) * \hat{S} + G^t(:, H_t^{which}).
\end{aligned}
\tag{6.17}
$$

Figure 6.1, depicts the functioning of the ADMM algorithm. We first initialize an iteration counter i.e, $k = 0$, and then evaluate the explicit solution for the $N + 1$ subproblems in parallel or sequentially. Once we have all the information necessary, the process proceeds to averaging and dual update. In the next step, the iteration counter is incremented and the explicit solutions of the subproblems are evaluated again if the termination criterion or the consensus is not reached. For the interested readers, the dual update, averaging and the convergence criteria of the ADMM algorithm is presented in [SKW13].

In the following, we briefly discuss about the evaluation of the PWA control law for the ADMM algorithm given in [SKW13] for three different platforms,

1. MTS (MATLAB sequential evaluation of the PWA control law for the $N + 1$ subproblems).
2. MTP (MATLAB parallel evaluation of the PWA control law for the $N + 1$ subproblems).
3. C-S (Sequential evaluation of the PWA control law in C programming for the $N + 1$ subproblems).

MTS- MATLAB sequential: Here, the evaluation of explicit solutionss for the $N+1$ subproblems are done sequentially using the point location function defined in (6.6) with the help of a *for Loop*.

MTP- MATLAB parallel: Here, the evaluation of the explicit solutions, for the $N+1$ subproblems, are carried out in parallel using the functional statements given in (6.16) and (6.17).

C-S- C programming in sequential: Here, we export the sequential controller MATLAB file to C using the following syntax available in MPT 3 toolbox.

"*ctrl.optimizer.toC('primal','file','obj')*"

In the following examples section, we discuss the computational and memory complexity for these platforms.

6.2.4 Examples

Example 1

Consider a linear discrete-time system given by,

$$x_{t+1} = Ax_t + Bu_t \tag{6.18}$$

where,

$$A = \begin{bmatrix} -0.0151 & -60.5651 & 0 & -32.174 \\ -0.0001 & -1.3411 & 0.9929 & 0 \\ 0.00018 & 43.2541 & -0.86939 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} -2.516 & -13.136 \\ -1.689 & -0.2514 \\ -17.251 & -1.5766 \\ 0 & 0 \end{bmatrix}$$

The states and inputs are bounded by the box constraints, $-5 \leq x_i \leq 5$, $i \in \{1, \cdots, 4\}$ and $-5 \leq u_j \leq 5$, $j \in \{1, 2\}$ respectively. The system (6.18) is solved for explicit problem without time-splitting operator using MPT toolbox 3.0 to compute the polyhedral regions and their associated PWA control laws.

The explicit problems solved for different prediction horizons considered in the receding horizon optimization is presented in Table 6.1 with the total number of regions and total number of half-spaces. It can be inferred from Table 6.1 that when the prediction horizon gets longer the explicit controller yields higher number of regions. Moreover, the time taken to solve the explicit problem becomes a long road of misery and not to mention the memory space consumed by the half-spaces and its associated PWA control laws for longer prediction horizon.

These limitations can be overcome by considering time-splitting over the horizon. The explicit solution for the time-splitting operation yields three sets of affine controllers with critical regions and their PWA functions. Table 6.2 shows the total

Table 6.1: EMPC without time splitting

N	$No\ of$ $regions$	$Total\ number of$ $half spaces$	$Time\ taken$ $to\ solve\ in\ (mins)$
5	4,755	39,100	~ 7
7	10,174	85,578	~ 16
10	24,740	205,342	~ 47

Table 6.2: EMPC with time splitting

N	$No\ of\ regions$			$Total\ no\ of$ $half spaces$
	P1	P2	P3	
$\forall$	115	2231	722	86626

number of regions for the three ADMM cost function given in (6.12) and these three affine controllers are denoted as P1, P2 and P3 respectively. The total time taken to solve the mp-QP with primal and dual variables for these subproblems is approximately 16 minutes that is considerably equal to the time taken to solve the explicit problem without time splitting for a horizon 7.

Table 6.3 represents the running time and the iteration numbers for different prediction horizons for different platforms. This table is presented for prediction horizon from 2 to 1000 with iterations and computational time taken for the ADMM convergence. It can be noticed from the table that the C simulations are faster than those of MATLAB. The computation time obtained for the MATLAB parallel approach can be compared with the MATLAB sequential approach but it lags very much behind the C simulations.

Table 6.3: Computational time and the ADMM final iteration counter number for MTS, MTP and C-S for different prediction horizon

N	ρ	$No\ of$ $Iterations$	$Comp\ Time\ in(s)$		
			MTS	MTP	C-S
2	10	38	0.0933	0.127	0.0017
5	10	179	4.495	11.0691	0.0144
10	10	284	14.12	19.86	0.0331
20	10	298	42.01	44.45	0.0597
50	10	356	126.11	89.85	0.1454
100	10	326	241.25	146.73	0.2744
200	10	329	553.47	321.49	0.549
500	10	329	1255.1	798.63	1.1287
1000	10	329	2755.9	1474.5	2.7532

Remark 6.2.2 *The argument that the MATLAB parallel approach could not bring down the running time as expected can be related to the following aspect: The binary matrices H_0^{ind}, H_t^{ind} and H_N^{ind}, and the integer matrices H_0^{tot}, H_t^{tot} and H_N^{tot} are stored as double floating point precision. We recall that for the PWA function evaluation of a matrix input* (6.17)*, most of the evaluation computation is based on the binary and integer matrices. Moreover, it should be noted that approximately 92% of the running time is consumed by evaluating the third line of statements given in* (6.17)*.*

It is possible to scale down the computational time to a very large extent by shifting the software platform from MATLAB to C programming, that we determine to discuss in the coming example. Based on the results obtained through simulations for EMPC problems with and without time splitting for the C files, we can say that the explicit solutions with ADMM cost function is worth implementing real-time for any system that requires larger prediction horizon for on-line control law evaluation.

In the next, we consider an rigorous example with two unstable and two complex eigenvalues.

Example 2

Consider a linear discrete-time system given by,

$$x_{t+1} = Ax_t + Bu_t \tag{6.19}$$

where,

$$A = \begin{bmatrix} 1.1939 & 0 & 0 & 0 & 0 \\ 0 & 1.1148 & 0 & 0 & 0 \\ 0 & 0 & 0.9994 & 0 & 0 \\ 0 & 0 & 0 & 0.4151 & 0.6319 \\ 0 & 0 & 0 & -0.6319 & 0.4151 \end{bmatrix}$$

$$B = \begin{bmatrix} -0.6274 & 0.6224 \\ 0.0530 & -0.1658 \\ 0.0274 & -0.0987 \\ -1.7005 & -0.1149 \\ -0.2190 & 0.0084 \end{bmatrix}$$

The states and inputs are bounded by the box constraints, $-25 \leq x_i \leq 25$, $i \in \{1, \cdots, 5\}$ and $-5 \leq u_j \leq 5$, $j \in \{1, 2\}$ respectively. Table 6.4 presents the number of critical regions, total number of half-spaces and time taken to solve the EMPC problem for different prediction horizons. From the table it can be seen that as the prediction horizon increases, the off-line computational complexity and the system memory requirements are drastically increased. Therefore, implementing such an explicit controller that has 1846254 half-spaces with double precision floating point, for an on-line control application, indeed demands a micro-controller or FPGA hardware with large memory size and capable processor, that can execute a greater number of instructions per clock cycle. Table 6.5 shows the number of regions

Table 6.4: EMPC without time splitting

N	$No\ of$ $regions$	$Total\ number of$ $half spaces$	$Time\ taken$ $to\ solve\ in\ (min)$
2	395	4638	< 1
5	16914	176740	~ 34
10	172161	1846254	~ 1850

obtained from the explicit controller for three subproblems for different ρ values. It can be found from the table that the total number of partitions varies by tunning the ρ constant in the control design formulation.

Table 6.5: EMPC with time splitting for different ρ constants

N	ρ	*No of regions*			*Total no of halfspaces*
		P1	P2	P3	
$\forall$	10	260	7551	3441	417797
	100	260	7557	3409	418086
	1000	260	7564	3405	418388

Table 6.6: Computational time and the ADMM final iteration counter number for MTS and C-S for different prediction horizon and for different ρ constants

N	ρ	*No of Iterations*	*Comp Time in(s)*	
			MTS	C-S
2	10	52	0.5642	0.0032
	100	26	0.2846	0.0014
	1000	18	0.2052	0.0011
5	10	189	22.147	0.0391
	100	392	42.146	0.0809
	1000	486	58.64	0.0784
10	10	276	83.471	0.1251
	100	1125	336.25	0.5051
	1000	1902	587.1	0.6234
20	10	648	560.84	0.6155
	100	1826	1235.5	1.7524
	1000	5548	3662.7	3.758
50	10	1912	3549.5	4.4689
	100	1868	3378.9	4.4048
	1000	9253	16242	15.502
100	10	2387	9876.3	10.9652
	100	2106	7691.1	9.654
	1000	7848	29406	25.96

The simulation results of the MTS and C-S approaches for different ρ constants with computational time and the iterations taken for acceptable error in the convergence criteria is given in Table 6.6. It can be seen from the Table 6.6 that the computational time of the C-S approach makes it suitable and deployable for real-time implementations even for a prediction horizon of 100. On the other hand, the iteration of MTP approach is infeasible because of insufficient memory during the computation of huge matrix structure involved in the PWA evaluation of matrix input (6.17). This problem of insufficient memory is strongly attributed to the MATLAB programming and this is slated to discuss in the follows.

Drawback of using MATLAB programming: In the MATLAB programming, arithmetic operations on binary and integer matrices or vectors are not straightforward to compute. It is only possible to allow MATLAB to perform standard matrix multiplication and convert the result to either binary or integer. For instance, from

Table 6.6, only to store the matrix H_t^{ind} which is of size (294585 $\times$ 7551) as double precision floating point requires 17.7953 GB memory size whereas for the original binary matrix it only takes approximately 278 MB. This drawback can be overcome in the C programming. For information, we have not yet implemented PWA function evaluation of the matrix input on C.

6.3 Multi-Stage Stochastic Programming

Many real-time optimization problems with dynamical systems involve uncertainty. The uncertainty in the system model could arise from the poor approximation of the plant neglecting an important part of the plant dynamics. From the optimization point of view, predicting the future dynamics of the plant with the help of such uncertain model makes the predicted states to differ drastically from the plant. In this section, we address the probabilistic uncertainty in the system model.
Consider a linear discrete-time system affected by probabilistic uncertainty,

$$x_{t+1} = Ax_t + Bu_t + Ed_t, \tag{6.20}$$

where $x_t \in \mathcal{X}$ is the state vector, $u_t \in \mathcal{U}$ is the input vector and $d_t \in \mathcal{D}$ is the probabilistic disturbance vector. The state and input constraint sets are bounded in the form (6.2) and the disturbance/uncertain constraint set $\mathcal{D}$ is bounded and assumed to admit a box constraint. In this work we adopt the scenario tree approach to model the probabilistic uncertainty. Scenario tree approach is one of the popular method for modeling the stochastic problem. The quadratic cost function for the scenario tree [LFE13] approach is given by,

$$\begin{aligned} & min \sum_{s=1}^{S} p_s J_s \\ & J_s = \sum_{t=0}^{N-1} x_{t+1}^{sT} Q x_{t+1}^{s} + u_t^{sT} R u_t^{s} \\ \text{subject to:} \quad & x_{t+1}^{s} = Ax_t^{s} + Bu_t^{s} + Ed_t^{s}, \\ & x_t^{s} \in \mathcal{X}, \\ & u_t^{s} \in \mathcal{U}, \\ & d_t^{s} \in \mathcal{D} \\ & s \in \mathcal{S} \end{aligned} \tag{6.21}$$

where J_s denotes the cost for each scenario, N represents the prediction horizon, S is the number of scenarios and p_s represents the probability of each individual scenario. Finally, $\mathcal{X}$,$\mathcal{U}$ and $\mathcal{D}$ represents state, control and uncertain disturbance constraint sets respectively. Here $\mathcal{S} = \{1, \cdots, s, \cdots, S\}$ denotes the set that contains the indexes of the scenarios.

The goal of the multistage model is to make decisions for different time periods in a sequence, while optimizing the expected objective function value of the current and future stages. For further information for scenario tree MPC, the interested readers are referred to [LFE13].

6.3.1 Decomposition of Scenario tree

Even thought the scenario tree approach is one of robust methods for modeling the uncertainty, the complexity of the problem grows exponentially with the number of uncertain parameters and by increasing the robust horizon. Solving such problems demand huge computational efforts and memory size. To overcome the computational complexities arising from the scenario tree, a decomposition approach is essential by decomposing a huge problem into several small problems also called scenarios.

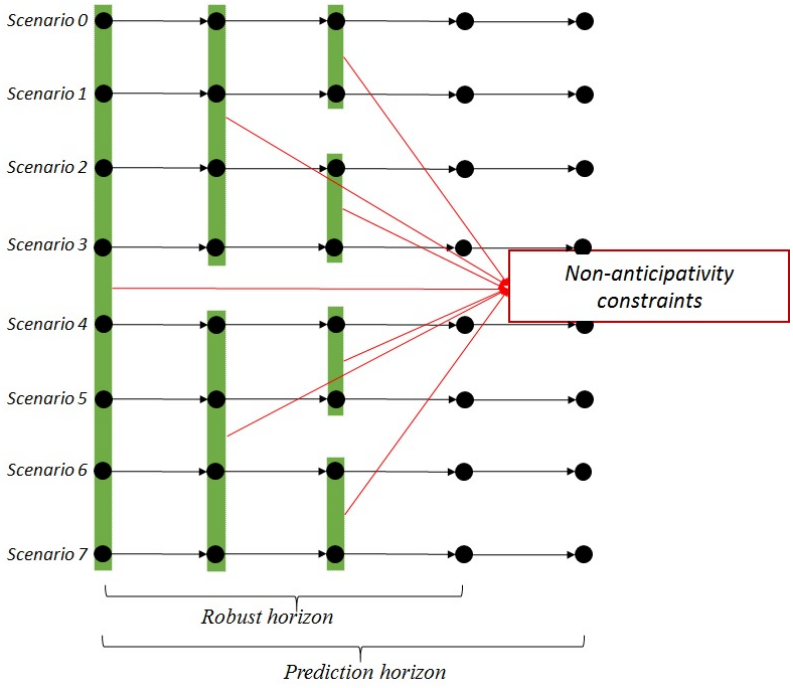


Figure 6.2: Scenario tree decomposition for robust horizon $N_r = 3$.

From Figure 6.2, it can be inferred that the scenario tree is decomposed into S independent scenarios. For this example, the scenario tree is generated for one uncertain parameter with two possible realizations of uncertainty and for robust horizon of 3. In the decomposition approach, the only constraints which have to be satisfied are the non-anticipativity constraints and this can be done by forcing all the inputs originating from the same node to be equal. For further information for scenario tree MPC, the interested reader is referred to [LFE13].

6.3.2 Progressive Hedging Algorithm

Progressive hedging algorithm is one of the popular techniques for solving linear discrete-time stochastic problems. Progressive hedging algorithm was proposed by Rockafellar and Wets [RW91]. The basic idea of the PHA is to place the non-anticipativity constraints to the objective function and this can be achieved by adding the Lagrange multiplier to the objective functions of each individual scenarios. The objective function consists of a quadratic term which ensures the primal

feasibility. The penalty term ρ_w is multiplied with the quadratic term to make the primal feasibility search faster. Then all the scenarios are solved in parallel or in sequence with some convergence criteria. The modified cost function for each individual scenario with an iteration counter denoted by k can be written:

$$\begin{aligned} & \min_{u_k^s} \; J(x_k^s, u_k^s, d_k^s) + \lambda_k^s(u_k^s - u_k^{Avg}) + \tfrac{\rho_w}{2}||u_k^s - u_k^{Avg}||^2, \\ \text{subject to:} \quad & J(x_k^s, u_k^s, d_k^s) = \textstyle\sum_{t=0}^{N} x_t^{sT} Q x_t^s + u_t^{sT} R u_t^s, \\ & x_{t+1}^s = A x_t^s + B u_t^s + E d_t^s, \\ & u_k^s \in \mathcal{U}, x_k^s \in \mathcal{X}, s \in \mathcal{S}. \end{aligned} \tag{6.22}$$

Algorithm 6.3.1 Progressive Hedging Algorithm

Input: $\rho_w > 0$
Output: u_k^s
1: *Initialization*: $\lambda_k^s = [0, 0, \cdots, 0] \in \mathbb{R}^{m \times N_r}, u_k^{Avg} = [0, 0, \cdots, 0] \in \mathbb{R}^{m \times N_r}$, $\epsilon > 0, k = 0$.
2: Solve: $\forall s \in S, \underset{u_k^s}{min}\ J(x_k^s, u_k^s, d_k^s)$.
3: Calculate: $u_k^{Avg} = \frac{1}{N_s} \sum_{s=1}^{N_s} u_k^s$.
4: Check criterion: $\forall s \in S$,
5: **if** $(u_k^s - u_k^{Avg})^2 > \epsilon$ **then**
6: Iteration counter: $k = k + 1$.
7: **else** terminate
8: **end if**
9: Update: $\forall s \in S, \lambda_k^s = \lambda_{k-1}^s + \rho_w(u_{k-1}^s - u_{k-1}^{Avg})$.
10: Solve: $\forall s \in S, \underset{u_k^s}{min}\ J(x_k^s, u_k^s, d_k^s) + \lambda_k^s(u_k^s - u_k^{Avg}) + \frac{\rho_w}{2}||u_k^s - u_k^{Avg}||^2$.
11: Go to step 3.

Algorithm 6.3.1 is the standard Progressive Hedging Algorithm. This algorithm is given for the standard MPC problem with QP cost functions. The only input for the algorithm is the penalty term ρ_w. For the standard MPC problem, the algorithm solves S number of QP cost functions for the first iteration $k = 0$ and this is shown in the second line of the Algorithm 6.3.1. In the third step, the parameter u_k^{Avg} is updated with the aggregated value of the control inputs from all scenarios. In the fourth step, a convergence criterion is established with the accepted level of error by using the scalar term ϵ. If the accepted level of convergence is not reached for the control input, the iteration counter k is incremented by one unit and continue with the "Update" function as given in the ninth line of the Algorithm. The convergence criterion ensures the fulfillment of the non-anticipativity constraints by forcing the control inputs originating from the same root to be identical. Next, the cost function with the Lagrange multiplier and the quadratic term is solved for the input variables for S number of scenarios. The algorithm is continued by jumping to the third line.
Parallelization of the PHA for standard MPC: The PHA can be parallelized by solving the scenarios in parallel. For the Algorithm 6.3.1, the expressions given in lines 2 and 10 are parallelized for all the scenarios.

PHA for Explicit MPC

It is clear from the Algorithm 6.3.1 that for the Explicit MPC, two cost functions (given in lines 2 and 10) should be solved off-line. The first problem denoted by $P0$

is the same for all the scenarios for the iteration counter $k = 0$,

$$\begin{array}{ll} P0 & \underset{u_{t|t},u_{t+1|t},\cdots}{min} J(x_t, u_t, d_t) \\ \text{subject to:} & J(x_t, u_t, d_t) = \sum_{t=0}^{N-1} x_{t+1}^T Q x_{t+1} + u_t^T R u_t, \\ & x_{t+1} = Ax_t + Bu_t + Ed_t, \\ & x_t \in \mathcal{X}, u_t \in \mathcal{U}, d_t \in \mathcal{D}, t = 0, \cdots N-1, \\ \text{Optimization variables:} & [u_{t|t}^T, \cdots, u_{N-1|t}^T]^T, \\ \text{Parameters:} & [x_{t|t}^T, d_{t|t}^T, \cdots, d_{N-1|t}^T]^T. \end{array} \tag{6.23}$$

The second problem with the Lagrange multiplier and the quadratic term denoted by $P1$ is the same for all scenarios for the iteration counter $k > 0$

$$\begin{array}{ll} P1 & \underset{u_{t|t},u_{t+1|t},\cdots}{min} J(x_t, u_t, d_t) + \lambda_{t|t}(u_{t|t} - u_{t|t}^{Avg}) + \frac{\rho_w}{2}||u_{t|t} - u_{t|t}^{Avg}||^2 \\ & + \cdots + \lambda_{N_r-1|t}(u_{N_r-1|t} - u_{N_r-1|t}^{Avg}) + \frac{\rho_w}{2}||u_{N_r-1|t} - u_{N_r-1|t}^{Avg}||^2 \\ \text{subject to:} & J(x_t, u_t, d_t) = \sum_{t=0}^{N-1} x_{t+1}^T Q x_{t+1} + u_t^T R u_t, \\ & x_{t+1} = Ax_t + Bu_t + Ed_t, \\ & x_t \in \mathcal{X}, u_t \in \mathcal{U}, d_t \in \mathcal{D}, t = 0, \cdots N-1, \\ \text{Optimization variables:} & [u_{t|t}^T, \cdots, u_{N-1|t}^T]^T, \\ \text{Parameters:} & [x_{t|t}^T, d_{t|t}^T, \cdots, d_{N-1|t}^T, \lambda_{t|t}^T, \cdots, \lambda_{N_r-1|t}^T, u_{t|t}^{AvgT}, \cdots, u_{N_r-1|t}^{AvgT}]^T. \end{array} \tag{6.24}$$

The explicit controllers for the problems $P0$ and $P1$ in (6.23) and (6.24) are computed. The resulting mp-QP solutions are given by two sets of affine functions defined over the partition of the sets $\mathcal{R}^0$ and $\mathcal{R}^1$ for the problems $P0$ and $P1$ respectively. Next, the evaluation of the PWA functions are done using the functional evaluation for the matrix input given in (6.17). For both $P0$ and $P1$, the parameters of the optimization problem for all the scenarios are given as the matrix input (similar to the one discussed in the previous section). The resulting output from the functional evaluation are the control input sequence for all the scenarios and it can be written in the form,

$$\begin{bmatrix} u_{t|t}^1, u_{t+1|t}^1, \cdots, u_{t+N|t}^1 \\ \vdots \\ u_{t|t}^S, u_{t+1|t}^S, \cdots, u_{t+N|t}^S) \end{bmatrix}. \tag{6.25}$$

Note: In the previous section, the matrix input block is used to compute the PWA control laws for $N-2$ subproblems for the time-splitting approach. Here, the matrix input block is used to evaluate the PWA control laws for all the scenarios in one shot.

6.3.3 Example

Consider a linear double integrator system affected by probabilistic uncertainty. The linear discrete-time state space representation of the system is given by,

$$x_{t+1} = Ax_t + Bu_t + Ed_t, \tag{6.26}$$

where,

$$A = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ 0.5 \end{bmatrix} \text{ and } E = \begin{bmatrix} 1 \\ 1 \end{bmatrix}.$$

The states, input and disturbance are bounded by the constraints, $-50 \leq x_i \leq 50, \quad i \in \{1, 2\}, \ -10 \leq u \leq 10$ and $-2 \leq d \leq 2$ respectively. For both $P0$ and $P1$, the weighting matrices of the states and input are chosen

$$Q = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \text{ and } R = 1$$

in the problem formulation of the open-loop optimal control problem considered for Explicit MPC.

In the following, we conduct some experiments based on different length of robust horizon, for different uncertain realization, but the same system dynamics and weighting matrices are applied for all the cases.

Result I: Solving the EMPC problem for different robust horizons:
Solving the EMPC problem for $P0$ and $P1$ for different robust horizon yields different number of affine controllers and their associated PWA control laws and it is shown in the Table 6.7. Here the term N_r denotes the length of the robust horizon. For this result, the length of the robust horizons are same as that of the prediction horizons. The penalty term $\rho_w = 100$ is chosen.

Table 6.7: EMPC with PHA

N	N_r	$No\ of\ regions(P1)$	$No\ of\ regions(P0)$
2	2	33	17
3	3	89	39
4	4	358	88
5	5	1203	123

Result II: Evaluation of PWA functions for PHA for different robust horizons on CPU and GPU
For this result, we consider that 3^{N_r} scenarios are generated. The different realizations associated to the uncertain parameter d_t are $[-2, 0, 2]^T$. For instance the realizations for the problem $N_r = 2$ are provided with the help of a matrix,

$$\begin{bmatrix} -2 & -2 \\ -2 & 0 \\ -2 & 2 \\ 0 & -2 \\ 0 & 0 \\ 0 & 2 \\ 2 & -2 \\ 2 & 0 \\ 2 & 2 \end{bmatrix}$$

The simulation results for the PHA, executed with the help of evaluation of PWA functions for matrix input, are presented for different robust horizons computed on CPU and GPU in Table 6.8. The computational time of CPU beats that of GPU when we have a smaller number of scenarios to parallelize as it can be inferred from the Table 6.8. When the number of scenarios and the number of regions get bigger, the running time of the GPU eventually beats that of CPU. The average number of iterations taken to achieve a acceptable converge criteria per simulation time is also mentioned in the table.

Figure 6.3 is plotted for the computational time obtained from the CPU and GPU.

Table 6.8: EMPC with PHA

N_r	*No of scenarios*	*Computational Time in seconds*		*Avg iterations*
		MTP-CPU	MTP-GPU	
2	9	0.009	0.1916	39
3	27	0.032	0.4276	47
4	81	0.58	1.102	52
5	243	19.96	11.56	53

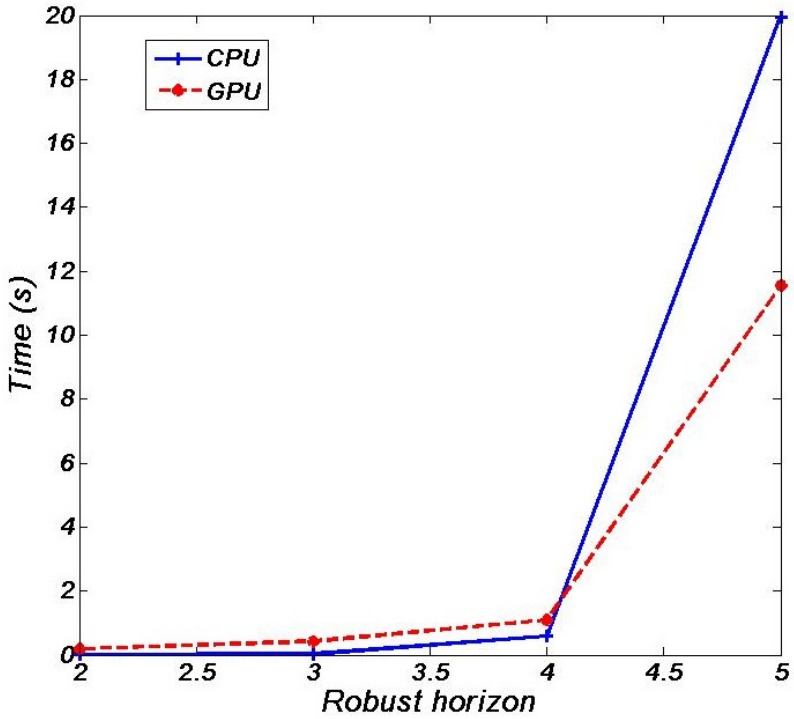


Figure 6.3: Computational time comparison between CPU and GPU

Result III: Comparison of the simulation results between the EMPC problem with PHA and the EMPC problem with standard QP (uncertainty ignored)

This result aims to compare the simulation of the the state trajectories and control input for the EMPC problem with PHA and the EMPC problem with standard QP. The prediction and robust horizon chosen for the EMPC problem with PHA

is $N_r = N_p = 3$. The penalty term $\rho_w = 100$ applied to the optimization problem. For the sake of comparison, the prediction horizon of length 3 is chosen for the standard EMPC problem and the random disturbances for the parameter d_t over the simulation time are also the same for both problems. Figure Figure 6.4 presents

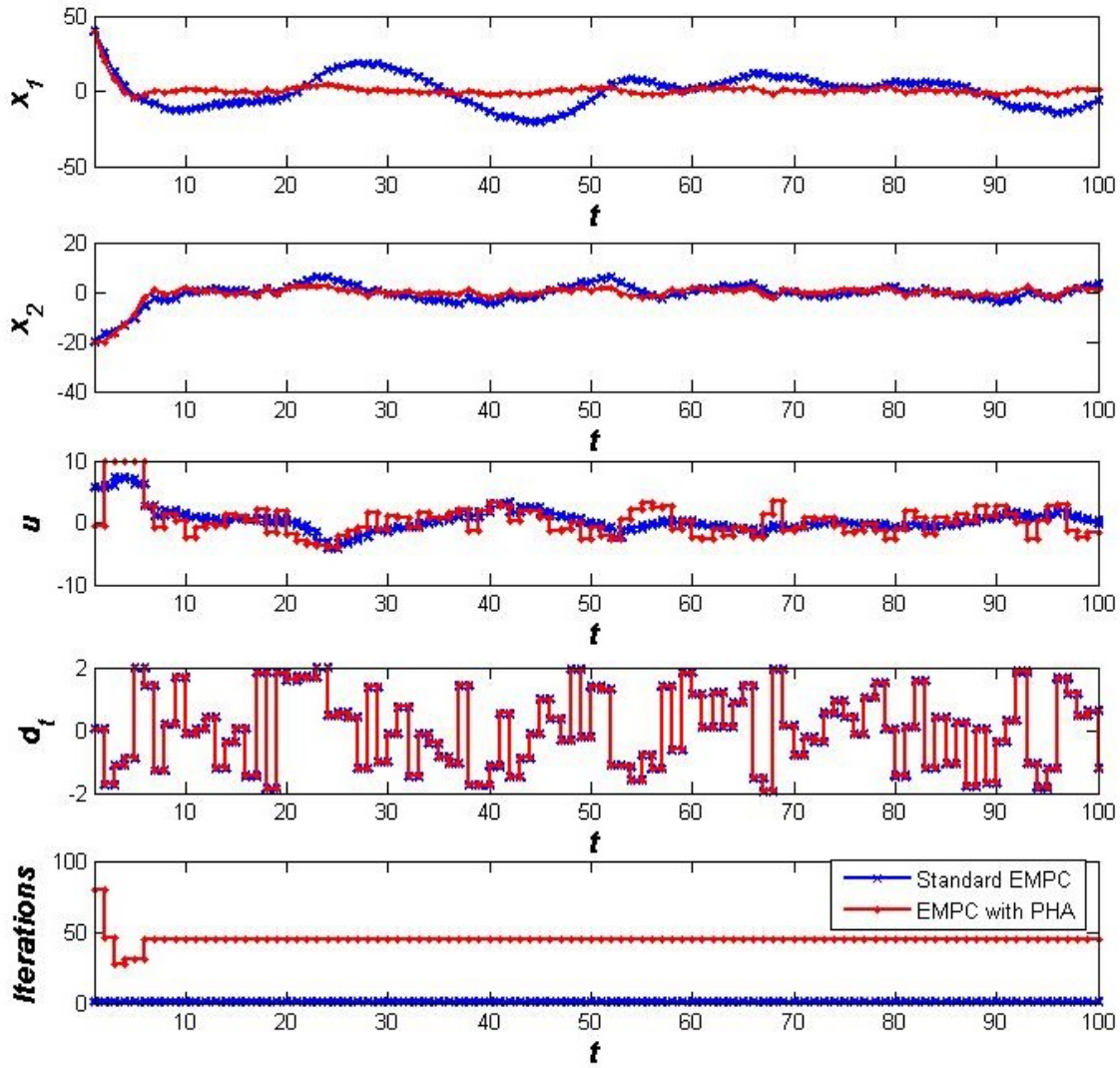


Figure 6.4: Simulation results of the states, input and random disturbance for the EMPC problem with PHA and the standard EMPC. The initial condition $x_0 = [40,\ -20]^T$ is chosen for the simulations.

the simulation results of the state trajectories, control input, random disturbance d_t for the EMPC problem with and without stochastic programming for the initial condition $x_0 = [40, -20]^T$. It can be inferred from the Figure 6.4 that the state trajectories of the state x_1 is oscillating for the standard EMPC problem. The explicit controller based on stochastic method is robust in terms of keeping the states closer to the origin.

6.4 Conclusions

In this work, we enabled the explicit solutions for a finite-time optimal control problem with long prediction horizons and also we use the explicit solution to the synthesis of a robust predictive controller. For control problems with long prediction horizon, this was achieved by incorporating the MPC problem formulation with Alternative Direction Method of Multipliers. The comparison of the computational complexity between the parallel and sequential evaluations of the PWA functions has been discussed. Next, we have employed a scenario tree problem formulation for treating the probabilistic uncertainties arising from the linear dynamical systems. Progressive Hedging Algorithm is used to decompose the scenario tree into a finite number of individual scenarios. Finally, the comparison of the evaluation of the PWA functions on the CPU and GPU for the PHA has been achieved. In conclusion, it is shown that with the increasing of the computation capabilities the explicit solutions are not only useful for small problems, but can be also used for larger problems and robust synthesis.

Chapter 7

Conclusion

This thesis has discussed certain topics related to robustness/fragility analysis, precision in polyhedral partition representation and evaluation of the PWA control laws, the results being primarily intended for the application in relationship with the well known Model Predictive Control paradigm.

7.1 Summary of Thesis Achievements

Robustness Analysis of PWA control laws

- The third chapter of this thesis has discussed the characterization and computation of the gain margin and robustness margin against first order neglected dynamics for a given nominal continuous PWA control law. The gain margin considers the gain variations in the system assuring the invariance property obtained with the PWA control law. A method to compute this gain margin is proposed. The robustness against first order neglected dynamics, consists in characterizing a first order dynamic included in the closed loop preserving the invariance. This robustness margin tries to approach the classical phase margin. In fact, slower the dynamic of the first order system included in the closed loop preserving the invariance is, bigger is phase margin. This robustness margin has been computed using the polyhedral description of the PWA controller. These results can be used for the analysis of constraints fulfillment in MPC whenever the prediction model is over-simplified or gain variations are expected in the actuation channel.

- The fourth chapter has discussed the contractive characterization and computation of the several robustness margin of the continuous PWA control law. A gain margin has been derived guaranteeing the contractive properties of the feedback PWA control law. A robustness margin has been computed for the nominal PWA controller assuring the contractive characteristics for a linear dynamical systems whose system matrices are affected by polytopic uncertainty. In addition, the robustness margin against the first order neglected dynamics has been evaluated preserving the contractive properties of the PWA

controller. All these results can complement the study of MPC feedback by providing stability guarantees.

Perturbation on the vertices of the polyhedral partition

- The fifth chapter of this thesis has discussed the impact of the perturbation on the vertices of the polyhedral partitions. We showed how the polyhedral regions change when a perturbation on the vertex representation of the regions takes place. We characterized the vertex perturbation by computing a set called vertex sensitivity which preserves the non-overlapping property of the polyhedral partition. We also computed a set called vertex-invariant region which preserves the invariant characteristics of the polyhedral partition. The convexity of the polyhedral partitions has been discussed for the vertices that lie on the frontier of the feasible domain. Starting from the vertex-invariant regions for the vertices of the polyhedral partition of a contractive PWA controller, we computed a region called contractive-vertex sensitivity to preserve the contractive properties of the PWA control law. All these results can be seen as a MPC retuning framework, whenever the quantization of reduced representation is expected in real-time implementation.

Sequential and parallel evaluation of the PWA functions

- The sixth chapter of this thesis discussed the complexity of the PWA controller for optimal control problems with longer prediction horizon. We proposed to incorporate the mp-QP problem formulation with the ADMM approach. The comparison of the sequential and parallel evaluation of the PWA functions for the computational complexity has been discussed. We modeled the probabilistic uncertainty of the dynamical systems using the scenario-tree approach. PHA has been adopted to decompose the complex scenario tree into individual scenarios and for the convergence of the coupling constraints. The computational time comparison for the sequential and parallel evaluation of the PWA functions executed on the CPU and GPU for the PHA has been presented.

7.2 Future Works

The procedure and methodology of the robustness analysis, re-tuning and evaluation of the PWA control laws which have been discussed and developed in this thesis work is based on the mp-QP problem whose solution set consists of PWA functions defined over the polyhedral partitions of the state-space. The robustness sets computed in chapters 3 and 4 are based on the vertex representation of the polyhedral partitions. Similarly, the main results developed in chapters 5 and 6 are based on the assumption that the explicit solutions for the open-loop OCP yields the polyhedral partitions. Recently, there are some major developments in the construction of the PWA control laws without the off-line computation of the polyhedral partition [KTHC15]. The construction of the PWA controllers without the polyhedral

partition is briefly discussed in the Appendix. The idea behind bypassing the computation of the critical regions is to scale down the memory of the data structure which is often used to store the off-line solution of the mp-QP problem. In addition to the memory, the computational time required to solve the region free explicit solutions are much faster than that of the solutions with critical regions and this observation is presented in [KTHC15]. From the point of robustness analysis of explicit solution with critical regions, new methodologies have to be developed in order to analyze the stability and robustness of such controllers.

From a different perspective, the authors of this work [SLKC14] have developed a design methodology for the Explicit MPC in the presence of finite precision arithmetic. In order to make explicit solutions viable to the real-time control applications, it is essential to bring down the complexity of the solutions. There is further scope in conducting future research regarding the presence of finite precision arithmetic. The future objective will be based on the stability analysis of the closed-loop system controlled by the explicit solution under finite precision. Starting from the controller design with the finite precision arithmetic, the goal will be to find the minimum number of finite precision digits required in order to obtain a certain degree of optimality of the explicit solutions.

In chapter 6, it has been shown that the evaluation of the PWA solution in C-programming language is very effective in terms of computational time and memory space required. Also, the C language empowers the programmer to store the binary numbers as a single bit. The objective will be to develop the evaluation of the PWA function in parallel based on C-programming.

Bibliography

[AB09] Alessandro Alessio and Alberto Bemporad. *A Survey on Explicit Model Predictive Control*, pages 345–369. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009.

[AHW12] Mariette Annergren, Anders Hansson, and Bo Wahlberg. An ADMM algorithm for solving l_1 regularized MPC. *CoRR*, abs/1203.4070, 2012.

[AMJO16] Parisa Ahmadi-Moshkenani, Tor Arne Johansen, and Sorin Olaru. On degeneracy in exploration of combinatorial tree in multi-parametric quadratic programming. In *Decision and Control (CDC), 2016 IEEE 55th Conference on*, pages 2320–2326. IEEE, 2016.

[BB09] D. Bernardini and A. Bemporad. Scenario-based model predictive control of stochastic constrained linear systems. In *Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*, pages 6333–6338, Dec 2009.

[BBG14] Alberto Bemporad, Leonardo Bellucci, and Tommaso Gabbriellini. Dynamic option hedging via stochastic model predictive control based on scenario simulation. *Quantitative Finance*, 14(10):1739–1751, 2014.

[Bel13] Richard Bellman. *Dynamic programming*. Courier Corporation, 2013.

[Ber95] Dimitri P. Bertsekas. *Dynamic programming and optimal control*, volume 1. Athena scientific Belmont, MA, 1995.

[BF02] Alberto Bemporad and C. Filippi. Suboptimal Explicit RHC via Approximate Multiparametric Quadratic Programming. Technical report, May 2002.

[Bit88a] Georges Bitsoris. On the positive invariance of polyhedral sets for discrete-time systems. *Systems & control letters*, 11(3):243–248, 1988.

[Bit88b] Georges Bitsoris. Positively invariant polyhedral sets of discrete-time linear systems. *International Journal of Control*, 47(6):1713–1726, 1988.

[BJJ12] Farhad Bayat, Tor Arne Johansen, and Ali Akbar Jalali. Flexible piecewise function evaluation methods based on truncated binary search trees and lattice representation in explicit mpc. *IEEE Transactions on Control Systems Technology*, 20(3):632–640, 2012.

[Bla99] F. Blanchini. Set invariance in control. *Automatica*, 35(11):1747 – 1767, 1999.

[BM99] Alberto Bemporad and Manfred Morari. Robust Model Predictive Control: A Survey. *Robustness in Identification and Control*, 245:207–226, 1999.

[BMDP02] Alberto Bemporad, Manfred Morari, Vivek Dua, and Efstratios N. Pistikopoulos. The explicit linear quadratic regulator for constrained systems. *Automatica*, 38(1):3 – 20, 2002.

[BPC+11] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Mach. Learn.*, 3(1):1–122, January 2011.

[BV95] Georges Bitsoris and Marina Vassilaki. Constrained regulation of linear systems. *Automatica*, 31(2):223–227, 1995.

[BV04] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, New York, NY, USA, 2004.

[CB97] Eduardo F. Camacho and Carlos A. Bordons. *Model Predictive Control in the Process Industry*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 1997.

[DH99] C. E. T. Dórea and J. C. Hennet. (a, b)-invariant polyhedral sets of linear discrete-time systems. *Journal of Optimization Theory and Applications*, 103(3):521–542, Dec 1999.

[DLM15] T. V. Dang, K. V. Ling, and J. M. Maciejowski. Embedded admm-based qp solver for mpc with polytopic constraints. In *2015 European Control Conference (ECC)*, pages 3446–3451, July 2015.

[dlPBF04] Munoz de la Pena, A. Bemporad, and C. Filippi. Robust explicit mpc based on approximate multi-parametric convex programming. In *2004 43rd IEEE Conference on Decision and Control (CDC) (IEEE Cat. No.04CH37601)*, volume 3, pages 2491–2496 Vol.3, Dec 2004.

[DO15] V. Dombrovskii and T. Obedko. Portfolio optimization in the financial market with regime switching under constraints and transaction costs using model predictive control. In *2015 European Control Conference (ECC)*, pages 3371–3376, July 2015.

[FJO13] Christian Feller, Tor Arne Johansen, and Sorin Olaru. An improved algorithm for combinatorial multi-parametric quadratic programming. *Automatica*, 49(5):1370–1376, 2013.

[GBN11] Arun Gupta, Sharad Bhartiya, and PSV Nataraj. A novel approach to multiparametric quadratic programming. *Automatica*, 47(9):2112–2117, 2011.

[GC86] P. O. Gutman and M. Cwikel. Admissible sets and feedback control for discrete-time linear dynamical systems with bounded controls and states. *IEEE Transactions on Automatic Control*, 31(4):373–376, Apr 1986.

[GG85] P. O. Gutman and S. Gutman. A note on the control of uncertain linear dynamical systems with constrained control input. *IEEE Transactions on Automatic Control*, 30(5):484–486, May 1985.

[GJ12] A. Grancharova and T.A. Johansen. *Explicit Nonlinear Model Predictive Control: Theory and Applications*. Lecture notes in control and information sciences. Springer, 2012.

[GKMS14] Graham C. Goodwin, He Kong, Galina Mirzaeva, and María M. Seron. Robust model predictive control: reflections and opportunities. *Journal of Control and Decision*, 1(2):115–148, 2014.

[HKD+06] F. Herzog, S. Keel, G. Dondi, L. M. Schumann, and H. P. Geering. Model predictive control for portfolio selection. In *2006 American Control Conference*, pages 8 pp.–, June 2006.

[HKJM13] M. Herceg, M. Kvasnica, C.N. 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.

[HOB14] Morten Hovd, Sorin Olaru, and George Bitsoris. Low complexity constraint control using contractive sets. *IFAC Proceedings Volumes*, 47(3):2933 – 2938, 2014. 19th IFAC World Congress.

[HW91] Thorkell Helgason and Stein W. Wallace. Approximate scenario solutions in the progressive hedging algorithm. *Annals of Operations Research*, 31(1):425–444, Dec 1991.

[IK15] Deepak Ingole and Michal Kvasnica. FPGA implementation of explicit model predictive control for closed loop control of depth of anesthesia. *IFAC-PapersOnLine*, 48(23):483 – 488, 2015. 5th IFAC Conference on Nonlinear Model Predictive Control NMPC 2015.

[JJST07] T. A. Johansen, W. Jackson, R. Schreiber, and P. Tondel. Hardware synthesis of explicit model predictive controllers. *IEEE Transactions on Control Systems Technology*, 15(1):191–197, Jan 2007.

[KBM96] Mayuresh V. Kothare, Venkataramanan Balakrishnan, and Manfred Morari. Robust constrained model predictive control using linear matrix inequalities. *Automatica*, 32(10):1361 – 1379, 1996.

[KM03] E. C. Kerrigan and J. M. Maciejowski. Robustly stable feedback min-max model predictive control. In *Proceedings of the 2003 American Control Conference, 2003.*, volume 4, pages 3490–3495 vol.4, June 2003.

[KTHC15] M. Kvasnica, B. Takács, J. Holaza, and S. Di Cairano. On region-free explicit model predictive control. In *2015 54th IEEE Conference on Decision and Control (CDC)*, pages 3669–3674, Dec 2015.

[KZC15] A. Knyazev, P. Zhu, and S. Di Cairano. Explicit model predictive control accuracy analysis. In *2015 54th IEEE Conference on Decision and Control (CDC)*, pages 2389–2394, Dec 2015.

[LCRM04] W. Langson, I. Chryssochoos, S.V. Raković, and D.Q. Mayne. Robust model predictive control using tubes. *Automatica*, 40(1):125 – 133, 2004.

[Lei81] J. R. Leigh. *Functional Analysis and Linear Control Theory*. Academic Press, Inc., Orlando, FL, USA, 1981.

[LFE13] Sergio Lucia, Tiago Finkler, and Sebastian Engell. Multi-stage nonlinear model predictive control applied to a semi-batch polymerization reactor under uncertainty. *Journal of Process Control*, 23(9):1306 – 1319, 2013.

[Lu14] Liang Lu. Separable model predictive control via alternating direction method of multipliers for large-scale systems. *IFAC Proceedings Volumes*, 47(3):10499 – 10504, 2014. 19th IFAC World Congress.

[Luc14] S. Lucia. Robust multi-stage nonlinear model predictive control, phd thesis, 2014.

[Lue97] David G. Luenberger. *Optimization by Vector Space Methods*. John Wiley & Sons, Inc., New York, NY, USA, 1st edition, 1997.

[LWM08] K.V. Ling, B.F. Wu, and J.M. Maciejowski. Embedded model predictive control (MPC) using a FPGA. *IFAC Proceedings Volumes*, 41(2):15250 – 15255, 2008. 17th IFAC World Congress.

[MDM09] S. Mariethoz, A. Domahidi, and M. Morari. Sensorless explicit model predictive control of permanent magnet synchronous motors. In *2009 IEEE International Electric Machines and Drives Conference*, pages 1250–1257, May 2009.

[MJ12] M. Mönnigmann and M. Jost. Vertex based calculation of explicit mpc laws. In *2012 American Control Conference (ACC)*, pages 423–428, June 2012.

[MKF11] David Q. Mayne, Eric C. Kerrigan, and Paola Falugi. Robust model predictive control: advantages and disadvantages of tube-based methods. *IFAC Proceedings Volumes*, 44(1):191 – 196, 2011. 18th IFAC World Congress.

[MRFA06] D.Q. Mayne, S.V. Raković, R. Findeisen, and F. Allgöwer. Robust output feedback model predictive control of constrained linear systems. *Automatica*, 42(7):1217 – 1222, 2006.

[MRRS00] D.Q. Mayne, J.B. Rawlings, C.V. Rao, and P.O.M. Scokaert. Constrained model predictive control: Stability and optimality. *Automatica*, 36(6):789 – 814, 2000.

[MRTT53] T.S. Motzkin, H. Raiffa, G. L. Thompson, and R. M. Thrall. The double description method. In H. W. Kuhn and A. W. Tucker, editors, *Contributions to the Theory of Games II*. Princeton University Press, 1953.

[NLR+15] Robert Nishihara, Laurent Lessard, Ben Recht, Andrew Packard, and Michael Jordan. A general analysis of the convergence of admm. In David Blei and Francis Bach, editors, *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)*, pages 343–352. JMLR Workshop and Conference Proceedings, 2015.

[NOBRA14] N. A. Nguyen, S. Olaru, G. Bitsoris, and P. Rodriguez-Ayerbe. Explicit fragility margins for PWA control laws of discrete-time linear systems. In *2014 European Control Conference (ECC)*, pages 1450–1455, June 2014.

[NORA+16] Ngoc Anh Nguyen, Sorin Olaru, Pedro Rodríguez-Ayerbe, George Bitsoris, and Morten Hovd. Explicit robustness and fragility margins for linear discrete systems with piecewise affine control law. *Automatica*, 68:334 – 343, 2016.

[OD04] S. B. Olaru and D. Dumur. A parameterized polyhedra approach for explicit constrained predictive control. In *2004 43rd IEEE Conference on Decision and Control (CDC) (IEEE Cat. No.04CH37601)*, volume 2, pages 1580–1585 Vol.2, Dec 2004.

[OD05] S. Olaru and D. Dumur. Avoiding constraints redundancy in predictive control optimization routines. *IEEE Transactions on Automatic Control*, 50(9):1459–1465, Sept 2005.

[Oga01] Katsuhiko Ogata. *Modern Control Engineering*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 4th edition, 2001.

[ONB+13] 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 *2013 17th International Conference on System Theory, Control and Computing (ICSTCC)*, pages 380–385, Oct 2013.

[PFKP09] Efstratios N. Pistikopoulos, Nuno P. Faísca, Konstantinos I. Kouramas, and Christos Panos. Explicit robust model predictive control. *IFAC Proceedings Volumes*, 42(11):243 – 248, 2009. 7th IFAC Symposium on Advanced Control of Chemical Processes.

[PGD07] E Pistikopoulos, M Georgiadis, and Vivek Dua. *Multi-Parametric Programming: Theory, Algorithms and Applications, Volume*. Weinheim: WileyVCH, 2007.

[Pon87] Lev Semenovich Pontryagin. *Mathematical theory of optimal processes*. CRC Press, 1987.

[QB03] S.Joe Qin and Thomas A. Badgwell. A survey of industrial model predictive control technology. *Control Engineering Practice*, 11(7):733 – 764, 2003.

[RAO13] P. Rodríguez-Ayerbe and S. Olaru. On the disturbance model in the robustification of explicit predictive control. *International Journal of Systems Science*, 44(5):853–864, 2013.

[RM12] J. B. Rawlings and D. Q. Mayne. Postface to "model predictive control: Theory and design", 2012.

[RW91] R. T. Rockafellar and Roger J.-B. Wets. Scenarios and policy aggregation in optimization under uncertainty. *Mathematics of Operations Research*, 16(1):119–147, 1991.

[San94] E. De Santis. On positively invariant sets for discrete-time linear systems with disturbance: an application of maximal disturbance sets. *IEEE Transactions on Automatic Control*, 39(1):245–249, Jan 1994.

[SDDG00] Maria M Seron, José A De Doná, and Graham C Goodwin. Global analytical model predictive control with input constraints. In *Decision and Control, 2000. Proceedings of the 39th IEEE Conference on*, volume 1, pages 154–159. IEEE, 2000.

[SGDD03] María M. Seron, Graham C. Goodwin, and José A. De Doná. Characterisation of receding horizon control for constrained linear systems. *Asian Journal of Control*, 5(2):271–286, 2003.

[SKW13] G. Stathopoulos, T. Keviczky, and Y. Wang. A hierarchical time-splitting approach for solving finite-time optimal control problems. In *2013 European Control Conference (ECC)*, pages 3089–3094, July 2013.

[SLKC14] Andrea Suardi, Stefano Longo, Eric C. Kerrigan, and George A. Constantinides. Robust explicit mpc design under finite precision arithmetic. *IFAC Proceedings Volumes*, 47(3):2939 – 2944, 2014. 19th IFAC World Congress.

[SM98] P. O. M. Scokaert and D. Q. Mayne. Min-max feedback model predictive control for constrained linear systems. *IEEE Transactions on Automatic Control*, 43(8):1136–1142, Aug 1998.

[SOH09] F Scibilia, Sorin Olaru, and M Hovd. Approximate explicit linear mpc via delaunay tessellation. In *Control Conference (ECC), 2009 European*, pages 2833–2838. IEEE, 2009.

[SOH11] Francesco Scibilia, Sorin Olaru, and Morten Hovd. On feasible sets for mpc and their approximations. *Automatica*, 47(1):133 – 139, 2011.

[TBGRI16] Gergely Takács, Gabriel Batista, Martin Gulan, and Boris Rohaľ-Ilkiv. Embedded explicit model predictive vibration control. *Mechatronics*, 36:54 – 62, 2016.

[TJB02] Petter Tondel, Tor Arne Johansen, and Alberto Bemporad. Computation and approximation of piecewise affine control laws via binary search trees. In *Decision and Control, 2002, Proceedings of the 41st IEEE Conference on*, volume 3, pages 3144–3149. IEEE, 2002.

[TJB03] P. Tøndel, T.A. Johansen, and A. Bemporad. Evaluation of piecewise affine control via binary search tree. *Automatica*, 39(5):945 – 950, 2003.

[VB89] Marina Vassilaki and Georges Bitsoris. Constrained regulation of linear continuous-time dynamical systems. *Systems & Control Letters*, 13(3):247–252, 1989.

[YBHJ03] Sheng Yunlong, Liu Bin, Su Hongye, and Chu Jian. Robust model predictive control for constrained linear systems based on contractive set and multi-parameter linear programming. In *Systems, Man and Cybernetics, 2003. IEEE International Conference on*, volume 4, pages 3073–3078 vol.4, Oct 2003.

Appendix A

Appendices

A.1 Explicit MPC

Let us begin with the multi-parametric Quadratic Programming (mp-QP) problem of the form:

$$\min_{U} \frac{1}{2} U^T H U + x^T F U \tag{A.1a}$$

$$subject\ to: \quad GU \leq W + Ex \tag{A.1b}$$

The parametric solution to (A.1) is a piecewise affine function of the form,

$$U^* = F_i x + g_i \tag{A.2}$$

A.1.1 Explicit solutions with Critical regions

Now find all the possible combinations of constraints active in (A.1) to construct the PWA solution in (A.2). To do this, we first rewrite (A.1) into the form:

$$\min_{z} \frac{1}{2} z^T H z \tag{A.3a}$$

$$subject\ to: \quad Gz \leq W + Sx \tag{A.3b}$$

with $z = U + H^{-1}F^T x$ and $S = E + GH^{-1}F^T$. Next we regroup (A.3) into active and inactive constraints,

$$\min_{z} \frac{1}{2} z^T H z \tag{A.4a}$$

$$subject\ to: \quad G_{\mathcal{A}} z = W_{\mathcal{A}} + S_{\mathcal{A}} x \tag{A.4b}$$

$$G_{\mathcal{N}} z < W_{\mathcal{N}} + S_{\mathcal{N}} x \tag{A.4c}$$

where, $\mathcal{A} \subset \{1, \cdots, p\}$ is the index set of active constraints and $\mathcal{P} \subset \{1, \cdots, p\}$ is the index set of inactive constraints. The index sets $\mathcal{A}$ and $\mathcal{P}$ are disjoint sets. The

Karush Kuhn Tucker conditions for (A.4) are given by,

$$Hz^* + G_{\mathcal{A}}^T\lambda^* + G_{\mathcal{A}}^T\mu^* = 0 \tag{A.5a}$$
$$G_{\mathcal{A}}z^* = W_{\mathcal{A}} + S_{\mathcal{A}}x \tag{A.5b}$$
$$G_{\mathcal{N}}z^* < W_{\mathcal{N}} + S_{\mathcal{N}}x \tag{A.5c}$$
$$\lambda^{*T}(G_{\mathcal{A}}z^* - W_{\mathcal{A}} - S_{\mathcal{A}}x) = 0 \tag{A.5d}$$
$$\mu^{*T}(G_{\mathcal{N}}z^* - W_{\mathcal{N}} - S_{\mathcal{N}}x) = 0 \tag{A.5e}$$
$$\lambda^* \geq 0 \tag{A.5f}$$
$$\mu^* \geq 0 \tag{A.5g}$$

Since $G_{\mathcal{N}}z^* - W_{\mathcal{N}} - S_{\mathcal{N}}x < 0$ for all inactive constraints from (A.5d), we assume that $\mu^* = 0$. Then we can derive the optimal z^* from (A.5a), we get:

$$z^* = -H^{-1}G_{\mathcal{A}}^T\lambda^* \tag{A.6}$$

Now by substituting (A.6) into (A.5b), we obtain

$$\lambda^* = -(G_{\mathcal{A}}H^{-1}G_{\mathcal{A}}^T)^{-1}(W_{\mathcal{A}} + S_{\mathcal{A}}x) \tag{A.7}$$

To simply the equation (A.7) we can write it as,

$$\lambda^* = Q(\mathcal{A})x + q(\mathcal{A}) \tag{A.8}$$

where,

$$Q(\mathcal{A}) = -(G_{\mathcal{A}}H^{-1}G_{\mathcal{A}}^T)^{-1}S_{\mathcal{A}} \tag{A.9a}$$
$$q(\mathcal{A}) = -(G_{\mathcal{A}}H^{-1}G_{\mathcal{A}}^T)^{-1}W_{\mathcal{A}} \tag{A.9b}$$
$$\tag{A.9c}$$

Substituting (A.7) into (A.6) we finally obtain

$$z^* = -H^{-1}G_{\mathcal{A}}^T(G_{\mathcal{A}}H^{-1}G_{\mathcal{A}}^T)^{-1}(W_{\mathcal{A}} + S_{\mathcal{A}}x) \tag{A.10}$$

which can be written as

$$z^* = F(\mathcal{A})x + g(\mathcal{A}) \tag{A.11}$$

$$F(\mathcal{A}) = -H^{-1}G_{\mathcal{A}}^T(G_{\mathcal{A}}H^{-1}G_{\mathcal{A}}^T)^{-1}S_{\mathcal{A}} \tag{A.12a}$$
$$g(\mathcal{A}) = -H^{-1}G_{\mathcal{A}}^T(G_{\mathcal{A}}H^{-1}G_{\mathcal{A}}^T)^{-1}W_{\mathcal{A}} \tag{A.12b}$$

The subset of the parametric space where z^* and λ^* satisfy primal feasibility and dual feasibility forms the critical region

$$\mathcal{P}(\mathcal{A}) = \{x \mid G_{\mathcal{N}}z^* < W_{\mathcal{N}} + S_{\mathcal{N}}x, \lambda^* \geq 0\} \tag{A.13}$$

which is a polyhedron in half-space representation. We replace the strict inequalities in (A.13) by non-strict:

$$\mathcal{P}(\mathcal{A}) = \{x \mid A(\mathcal{A})x \leq b(\mathcal{A})\} \tag{A.14}$$

with

$$A(\mathcal{A}) = \begin{bmatrix} G_{\mathcal{N}}F(\mathcal{A}) - S_{\mathcal{N}} \\ -Q(\mathcal{A}) \end{bmatrix}, b(\mathcal{A}) = \begin{bmatrix} W_{\mathcal{N}} - G_{\mathcal{N}}g(\mathcal{A}) \\ q(\mathcal{A}) \end{bmatrix} \tag{A.15}$$

A.1.2 Explicit solutions without Critical regions

The other way to obtain the PWA affine solution for the problem (A.1) is based on the extensive enumeration method [GBN11, FJO13, AMJO16, KTHC15], where we can generate all optimal active sets without having to construct the critical regions. The procedure enumerates all possible active combinations of active constraints and group them like a tree structure and giving each node of the tree with a cardinal number. To determine the optimality of a particular candidate $\mathcal{A}$. We solve the linear program with decision variables z, x, λ and t.

$$\underset{t,z,x,\lambda}{max}\ t \tag{A.16a}$$
$$subject\ to:\ Hz + G_{\mathcal{A}}^{T}\lambda = 0, \tag{A.16b}$$
$$G_{\mathcal{A}}z = W_{\mathcal{A}} + S_{\mathcal{A}}x, \tag{A.16c}$$
$$t \leq W_{\mathcal{N}} + S_{\mathcal{N}}x - G_{\mathcal{N}}z, \tag{A.16d}$$
$$\lambda \geq t, \tag{A.16e}$$
$$t \geq 0 \tag{A.16f}$$

– Enumerate all possible combinations of active constraints with cardinality $M = \{0, \cdots, m\}$.
– For each candidate active set solve the LP (A.16). If the LP is feasible, add the active candidate to the list of optimal active set. If the LP is infeasible, drop (A.16b). IF the new LP is infeasible, discard the candidate as well as all other candidates which are its supersets.

To compute the optimal control for a given x, We follow the below Algorithm.

Algorithm: Explicit solutions without critical regions

INPUT: Matrices Q_i, q_i, $i = 1, \cdots, M$ from (A.9),
list of optimal active sets $\{\mathcal{A}_1, \cdots, \mathcal{A}_M\}$ data H^{-1},
G, W and S and the state vector x.
OUTPUT: z* or U* for given x.

1. **for** i =1,$\cdots$,M **do**
2. $\lambda \leftarrow Q_i x + q_i$
3. **if** $\lambda \geq$ **then**
4. $z \leftarrow$ -$H^{-1}G_{\mathcal{A}_i}^{T}\lambda$
5. $\mathcal{N}_i \leftarrow \{1, \cdots, p\}\mathcal{A}_i$
6. **if** $G_{\mathcal{N}_i}z < W_{\mathcal{N}_i} + S_{\mathcal{N}_i}x$ **then**
7. return $z^* \leftarrow z$
8. **end if**
9. **end if**

10. **end for**

The (A.16) solves for the combination of active constraint set. Since there are p constraints (in G matrix) and m decision variables in (A.4), the root node $\mathcal{A} = \emptyset$ (where no constraints would be active) and at the final stage of the tree, m constraints would be active (the dimension of the decision variable). With increase in p and m, the number of combinations of constraints will also increase. However, not all candidates are considered, for example, we know that the 2nd and the 4th constraints cannot be simultaneously active, then all the subsets containing the 2nd and 4th constraints will be infeasible as well. Thus we can eliminate all the supersets that contains this combination.

Title: Robustness of Explicit MPC Solutions
Keywords: Robustness, predictive, control, explicit, robustness/fragility margin
Abstract: The control design techniques for linear or hybrid systems with constraints lead often to off-line state-space partitions with non-overlapping convex polyhedral regions. This corresponds to a piecewise affine (PWA) state feedback control laws associated to polyhedral partition of the state-space. Such control laws can be effectively implemented on hardwares for real-time control applications. However, the robustness of the explicit solutions depends on the accuracy of the mathematical model of the dynamical systems. The uncertainties in the system model poses serious challenges concerning the stability and implementation of the piecewise affine control laws. Motivated by the challenges facing the explicit solutions for the uncertainties in the dynamical systems, this thesis is mostly related to their analysis and re-design. The first part of this thesis aims to compute robustness margins for a given nominal PWA control law obtained for a linear discrete-time system. Classical Robustness margin i.e., gain margin and phase margin, considers the gain variation and phase variation of the model for which the stability of the closed loop is preserved.
The second part of the thesis aims to consider perturbation in the representation of the vertices of the polyhedral regions. The quantized state-space partitions lose some of the important property of the explicit controllers: "non-overlapping", "convexity" and "invariant" characterization. Two different sets called vertex-sensitivity and sensitivity margin are defined and determined to characterize admissible perturbation preserving the non-overlapping and the invariance property of the controller respectively. The third part analyse the complexity of the explicit solutions in terms of computational time and memory storage. Sequential and parallel evaluations of the PWA functions for the Alternating Direction Method of Multiplier (ADMM) algorithm are compared. In addition a comparison of the computational complexity of the parallel evaluations of the PWA functions for the Progressive Hedging Algorithm (PHA) on the Central Processing Unit (CPU) and Graphical Processing Unit (GPU) is made.

Titre: Robustesse de la commande prédictive explicite
Mots-clés: Robustesse, prédictive, commande, explicite, marges de robustesse/ fragilité
Résumé: Les techniques de conception de lois de commande pour les systèmes linéaires ou hybrides avec contraintes conduisent souvent à des partitions de l'espace d'état avec des régions polyédriques convexes. Ceci correspond à des lois de commande par retour d'état affine (PWA) par morceaux associées à une partition polyédrale de l'espace d'état. De telles lois de commande peuvent être effectivement mises en œuvre sur des plateformes matérielles pour des applications de commande en temps réel. Cependant, la robustesse des solutions explicites dépend de la précision du modèle mathématique des systèmes dynamiques. Les incertitudes dans le modèle du système posent de sérieux défis en ce qui concerne la stabilité et la mise en œuvre des lois de commande affines par morceaux. Motivé par les défis auxquels font face les solutions explicites par rapport aux incertitudes dans les modèles des systèmes dynamiques, cette thèse est principalement axée sur leur analyse et à leur retouche. La première partie de cette thèse vise à calculer les marges de robustesse pour une loi de commande PWA nominale donnée obtenue pour un système de temps discret linéaire. Les marges de robustesse classiques, c'est-à-dire la marge de gain et la marge de phase, considèrent la variation de gain et la variation de phase du modèle pour lequel la stabilité de la boucle fermée est préservée.
La deuxième partie de la thèse vise à considérer des perturbations dans la représentat-ion des sommets des régions polyédriques. Les partitions de l'espace d'état quantifiées perdent une partie des propriétés importantes des contrôleurs explicites: "non-chevauchement", "convexité" et/ou " invariance". Deux ensembles différents appelés sensibilité aux sommets et marge de sensibilité sont déterminés pour caractériser les perturbations admissibles, en préservant respectivement la propriété de non-chevauchement et d'invariance du contrôleur. La troisième partie vise à analyser la complexité des solutions explicites en termes de temps de calcul et de mémoire. Une première comparaison entre les évaluations séquentielles et parallèles des fonctions PWA par l'algorithme ADMM (Alternating Direction Method of Multiplier) est faite. Ensuite, la complexité computationnelle des évaluations parallèles des fonctions PWA pour l'algorithme de couverture progressive (PHA) sur l'unité centrale de traitement (CPU) et l'unité de traitement graphique (GPU) est comparée.

Université Paris-Saclay
Espace Technologique / Immeuble Discovery
Route de l'Orme aux Merisiers RD 128 / 91190 Saint-Aubin, France