N°d'ordre: ....

République Algerienne Démocratique et Populaire
Ministère de l'enseignement supérieur et de la Recherche scientifique
Université Mouloud Mammeri de Tizi Ouzou
Faculté des Sciences
Département de Mathématiques
Laboratoire Laromad

Mémoire de Master

Filière : Mathématiques
Spécialité : Recherche Opérationelle

Présenté par:

Ait amara Naima

Yahiaoui Sarah

Résolution d'un problème de programmation quadratique en contrôle optimal

Soutenue le Septembre 2022 devant le jury :

Aidene Mohamed	Professeur	UMMTO	Président
Nouri Naima	M.Assistante.A	UMMTO	Examinatrice
Oukacha Brahim	Professeur	UMMTO	Encadreur

Année Universitaire : 2021/2022

Dédicace

J'ai le grand plaisir de dédier ce modeste travail accompagné d'un profond amour :

À ma très chère **MÉRE**, celle qui a attendu les fruits de sa bonne éducation et de ses dévouements, qui me donne toujours l'espoir de vivre et qui n'a jamais cessé de prier pour moi, que dieu te préserve et t'accorde santé, longue vie et bonheur.

À mes chères sœurs **MOUNIA**, **RADIA**, **DALILA**, **SAIDA**, et **RABIAA** , ainsi que leurs enfants.

À mes chers fréres **LOUNIS**, **RACHID**, et **SOFIANE**, ainsi que leurs enfants.

À l'homme de ma vie **TAKFARINAS**, pour ses encouragements, son soutien, son infinie patience et sa compréhension.

À ma belle mére **TOUNSIA** qui m'a toujours encouragée, et qui n'a jamais cesse de prier pour moi, que dieu te préserve et t'accorde santé, longue vie, et bonheur.

À toute ma famille ainsi que ma belle famille **BELAIDI**, pour leurs constants soutiens.

À mes meilleures amies **NAIMA**, **NOUARA**, **SONIA**,**SALMA**, et **BAHIA**.

À tout mes **PROFESSEURS** du département mathématiques qui m'ont transmis le savoir, l'esprit de recherche et de perseverance dans mes études durant mon cursus.

À ma chère binôme **SARAH YAHIAOUI** , qui a contribuée à la réalisation de ce travail.

AIT AMARA NAIMA

J'ai l'honneur et le plaisir de dédier ce travail :

À **MES PARENTS** pour leurs soutien et l'encouragement durant toutes ces années d'études , qu'ils trouvent ici le témoignage de ma profonde reconnaissance .

À mes chers frères pour leurs soutiens et leurs présence dans les bons moments comme dans les pires.

À tout mes professeurs du département mathématiques qui m'ont transmis le savoir, l'esprit de recherche et de perseverance dans mes études durant mon cursus.

À mes chères amies sans exception

À ma chère binôme **NAIMA AIT AMARA** pour les moments passés ensemble pour la réalisaton de ce travail.

YAHIAOUI SARAH

Remerciements

Tout travail de recherche n'est jamais l'oeuvre d'une seule personne , à cet effet je tiens à exprimer ma reconnaissance à tous ceux qui ont contribué de près ou de loin à l'élaboration de ce travail.

Nous tenons à exprimer notre gratitude et nos plus sincères remerciements à notre promoteur **Pr OUKACHA BRAHIM** pour avoir bien voulu nous encadrer pour la réalisation de ce travail.Nous devons également le remercier pour sa disponibilité et ses précieux conseils tout au long de notre travail.

Il m'est très agréable de remercier **Mr AIDENE MOHAMED** qui nous a fait l'honneur de présider le jury de soutenance. Nous tenons également à remercier **Mme NOURI NAIMA** qui a acceptée d'être l'examinatrice de ce mémoire.

Sans oublier **ALI DJEBID** pour son aide précieuse dans l'utilisation du logiciel Latex ainsi que ses collègues et les responsables et les enseignants du département Mathématiques.

Nous remercions du fond du coeur toutes **NOS FAMILLES** pour leurs soutiens et leurs encouragements.

Tizi-Ouzou, le 2 octobre 2022.

Table des matières

Table des matières v

Introduction 1

1 Formulation du problème 2
1.1 Introduction . . . 2
1.2 Position du problème de contrôle optimal . . . 4
1.2.1 Problème de contrôle optimal à temps fixe . . . 6
1.2.2 Problème de contrôle optimal à temps libre . . . 7
1.3 Forme canonique du problème de contrôle optimal discret 8
1.3.1 Problème de contrôle optimal canonique . . . 9
1.4 Problème de programmation mathématique . . . 10
1.4.1 Problème de base . . . 10
1.5 équivalence des problèmes d'optimisation . . . 11

2 Contrôle optimal et programmation quadratique 14
2.1 Formulation du problème général de contrôle avec coût quadratique et transformation en problème de programmation quadratique . . . 14
2.1.1 Problème de contrôle quadratique (PCQ) . . . 14
2.2 L'existence d'une solution optimale au problème de programmation quadratique . . . 16
2.2.1 Problème de programmation quadratique (PQ) . . . 16
2.3 Condition suffisante pour une solution optimale unique d'un problème de programmation quadratique . . . 21
2.4 Les conditions nécessaires et suffisantes d'optimalités pour le problème de programmation quadratique . . . 23
2.5 Applications aux problèmes de contrôle illimités . . . 24
2.5.1 Le problème d'énergie minimale : forme de limite à deux points 24
2.5.2 Le problème du coût minimum quadratique . . . 30
2.6 Problèmes de contrôle quadratique avec contraintes d'inégalités . . . 34
2.6.1 Problème avec des contraintes d'énergie minimale . . . 35
2.6.2 Le problème du coût minimum à contraintes quadratiques. . . 36
2.7 Conditions d'optimalité pour le problème de programmation quadratique canonique . . . 37
2.7.1 Problème de programmation quadratique canonique(PQC) . . 37
2.8 Le problème de minimisation dérivée . . . 38
2.8.1 Le problème dérivé . . . 38

3 Algorithme de résolution 40
3.1 Algorithme du simplexe . . . 40

3.1.1 L'algorithme du simplexe pour la programmation quadratique 41
3.2 Généralisation 41
3.3 Convergence 46
3.3.1 Hypothèse de non-dégénérence 46

Conclusion générale **48**

Bibliographie **49**

Notations **50**

Introduction générale

En mathématiques, le contrôle désigne la théorie qui vise à comprendre la façon dont une commande permet aux humains d'agir sur un système qu'ils souhaitent maitriser.

La théorie du contrôle optimal est une généralisation du calcul des variations qui a été développé au milieu du dix-huitième siècle.
Le but de la théorie du contrôle optimal est de conduire un système dynamique contrôlé d'une configuration donnée à une configuration visitée à atteindre, tout en minimisant (ou maximisant) un certain coût et en respectant certaines contraintes.
Généralement, la théorie du contrôle optimal traite des systèmes dont la dynamique est décrite par une équation différentielle ordinaire. Les différents types de modèles mathématiques résultant dépendent fortement de la nature du problème à traiter, et il peut être exprimé sous différentes formes : linéaire, non linéaire, convexe ou non convexe, etc
Selon la nature du problème, plusieurs méthodes ou approches existent en littérature qui permet d'obtenir des décisions optimales ou sous-optimales en temps de calcul qui dépend de la complexité du modèle mathématique.
Afin de bien mener notre travail de recherche, le mémoire est subdivisé en trois chapitres :

Le 1^{er} chapitre est consacré à la formulation du problème de contrôle optimal. Nous aborderons ce problème à temps fixe et ainsi qu'à temps libre, ensuite on a transformé le problème de contrôle optimal discret sous forme canonique, enfin on a déduit l'équivalence de ces problèmes d'optimisation.
Les conditions d'optimalité pour le problème de programmation quadratique canonique seront aussi énoncées. La dernière section de ce chapitre est consacrée au problème de minimisation dérivée.

Le 2 ème chapitre portera sur " Le Contrôle Optimal et La Programmation Quadratique" dont lequel on traitera le problème de contrôle quadratique suivi des conditions nécéssaires et suffisantes d'optimalités en citant quelques applications aux problèmes de contrôle illimités et quelques problèmes de contrôle quadratique avec contraintes d'inégalités

Pour finir un dernier chapitre intitulé "Algorithme de résolution" qui contiendra l'algorithme du simplexe pour la résolution du problème de programmation quadratique sera présenté ainsi que la condition de convergence qui sera traité.

Formulation du problème

1

1.1 Introduction

Les problèmes de contrôle optimal étudiés dans ce chapitre sont des problèmes de contôle optimal discrets, de plus, les problèmes de contrôle optimal sont traités comme cas spécial de problèmes de programmation mathématique. Il y a essentiellement deux raisons pour s'intéresser à l'optimisation de contrôle discret.
La première raison est d'ordre pédagogique ; la seconde est plus importante est d'ordre technique.
Les problèmes de contrôle optimal discrets sont des problèmes définies dans des espaces de dimensions finies et, nécessitent beaucoup moins d'outils mathématiques dans leur traitement que des problèmes de contrôle continus. En outre, l'extension des résultats des espaces de dimensions finies à des espaces de dimensions infinies, est conceptuellement simple.

Il semble donc pédagogiquement efficace d'étudier le contrôle optimal discret en détail et ensuite pour en apprendre d'avantage sur le contrôle optimal continu à travers des extensions.

Notre principale raison d'attacher autant d'importance au contrôle optimal discret est d'ordre technique, découle de l'utilisation sans cesse croissante des calculateurs numériques dans le contrôle des systèmes dynamiques.
Dans tout calcul effectué sur un calculateur numérique, on ne peut faire mieux que d'obtenir un ensemble fini de nombres réels. Donc, pour résoudre un problème de contrôle optimal continu de la forme :

$$\text{Minimiser} \quad \int_0^{t_f} f^0(x(t), u(t), t), \qquad t \in [0, t_f]; \tag{1.1}$$

sous les contraintes

$$\frac{d}{dt}x(t) = f(x(t), u(t), t) \quad x(t) \in E^n, u(t) \in E^m; \tag{1.2}$$

$$x(0) \in X_0, \quad x(t_f) \in X_f; \tag{1.3}$$

$$u(t) \in U. \tag{1.4}$$

Nous sommes obligés de recourir à une certaine forme de discrétisation. Si la discrétisation n'est régie que par la nature des formules d'intégration à utiliser pour résoudre (1.2) et calculer (1.1), puis, dans tout schéma itératif,

nous devons calculer et stocker un très grand nombre de points $x(t_i)$ et $u(t_i)$, avec $t_i \in [0, t_f]$ à chaque itération.
Ainsi, une discrétisation simple nécessite une grande capacité de mémoire et entraine généralement de longs temps de calcul, ce qui est inacceptable si des ordinateurs relativement moins puissant doivent être utilisés pour contrôler une dynamique relativement rapide.

La discrétisation privilégiée est celle qui résulte de la restriction des entrées $u(.)$ à une classe de fonction représentable par un ensemble fini de paramètres. Le choix de la classe de fonctions auxquelles $u(.)$ doit appartenir et du nombre de paramètres à calculer peut-être utilisé par le concepteur pour gagner une grande quantité de liberté dans le contrôle de la dimension, et donc aussi du calcul-complexité dimensionnelle du problème d'optimisation qui en résulte.
Le coût d'une telle simplification est une réduction des performances.

Cependant, sans restriction supplémentaires, le problème (1.1) peut ne pas être résolu (dans un délai raisonnable sur ordinateur), et là il n'y a peut-être pas d'autre choix que de le restreindre davantage pour le résoudre .
Nous allons maintenant considérer quelques exemples simples de discrétisations des entrées couramment utilisé $u(.)$. Supposons qu'en plus de (1.4) nous exigeons que les entrées soient constantes par morceaux.
En particulier, nous pouvons exiger que :

$$u(t) = u_i, \qquad i \in [iT, (i+1)T] \tag{1.5}$$

$$u_i \in U, \qquad i = 0, 1, \cdots, k-1,$$

$$\text{où} \quad t_f = kT \quad \text{et} \quad k \quad \text{un paramètre entier.}$$

Le problème de contrôle optimal en temps discret prendra alors la forme suivante :

$$\text{Minimiser} \sum_{i=0}^{k-1} f_i^0(x_i, u_i); \tag{1.6}$$

$$x_{i+1} - x_i = f_i(x_i, u_i); \qquad i = 0, 1, 2, \cdots, k-1; \tag{1.7}$$

$$x_0 \in X_0, \qquad x_k \in X_f, \qquad u_i \in U. \tag{1.8}$$

Où $f_i(x_i, u_i)$ et $f_i^0(x_i, u_i)$ sont calculés comme suit :
Pour $t \in [iT, (i+1)T)$ soit $x_i(t)$ la solution de (1.2) correspondant à $u(t) \equiv u_i$ pour $t \in [iT, (i+1)T]$ et satisfaisant $x_i(iT) = x_i$.
En plus nous devons avoir cela pour $i = 0, 1, 2, \cdots, k-1, x_{i+1} = x_i((i+1)T)$.
Par conséquent,

$$x_{i+1} - x_i = \int_{iT}^{(i+1)T} f(x_i(t), u_i, t) dt, \tag{1.9}$$

Et puisque x_i est uniquement déterminé par x_i et u_i , les fonctions dans (1.6) et (1.7) sont correctement définis, comme suit :

$$f_i^0(x_i, u_i) = \int_{iT}^{(i+1)T} f^0(x_i(t), u_i, t) dt; \tag{1.10}$$

$$f_i(x_i, u_i) = \int_{iT}^{(i+1)T} f(x_i(t), u_i, t)dt. \tag{1.11}$$

D'autres discrétisations de la commande peuvent être considérés :

$$u(t) = \sum_{j=0}^{s} u_i^j t_j \tag{1.12}$$

$$t \in I_i, u_i^j \in A_j \subset E^m,$$

$$\text{où} \quad \cup I_i = [0, t_f]$$

et les A_i sont convenablement liés à l'original ensemble des contraintes U. Notez que lorsque cette discrétisation particulière est interprétée en termes d'un système dynamique discret tel que (1.7), on constate que le vecteur $u_i = (u_i^0, u_i^1, \ldots, u_i^s) \in E^m$ et donc de dimension différente du vecteur $u(t)$.

Une fois qu'une discrétisation du type ci-dessus a été effectuée, le problème de contrôle optimal d'origine devient un problème de programmation mathématique de dimension finie.

Remarque 1.1 *Un problème de contrôle optimal continu peut générer un problème de programmation mathématique de dimension finie sans donner lieu simultanément à un problème de contrôle optimal à commande discrète.*

Par exemple, ce serait le cas si nous trouvions nécessaire de restreindre les entrées $u(.)$ comme suit :

$$\text{pour} \quad t \in [0, t_f] \qquad u(t) = \sum_{i=0}^{s} u_i \varphi(t), \qquad u_i \in U_i \subset E^1, \tag{1.13}$$

$$\text{où } \varphi_i : E^1 \to E^m.$$

Le fait que le contrôle optimal et la programmation mathématique sont essentiellement une seule branche, a conduit les auteurs Canon et al. [1966] ainsi que Neustadt [1966] , Neustadt [1967] et Halkin et Neustadt [1966] à la construction d'une théorie unifiée de l'optimisation.

Comme nous le verrons plus tard, cette théorie aboutit à des simplifications conceptuelles substantielles et facilite la transcription d'algorithmes de programmation non linéaire hautement avancés pour une utilisation dans le contrôle optimal discret.

1.2 Position du problème de contrôle optimal

Pour définir un problème de contrôle optimal, nous devons spécifier la dynamique du système, les contraintes sur les commandes et sur la trajectoire, en plus, nous devons spécifier une fonction de coût.

Ici nous considérerons des systèmes de dimensions finies dont la dynamique satisfait une équation de différence de la forme :

$$x_{i+1} - x_i = f_i(x_i, u_i) \qquad i = 0, 1, 2, \ldots, \tag{1.14}$$

Où $x_i \in E^n$ est l'état du système à l'instant $i = 0, 1, 2, \ldots$; $u_i \in E^m$ est l'entrée du système à l'instant $i = 0, 1, 2, \ldots$; et les $f_i(\cdot, \cdot)$ pour $i = 0, 1, 2, \ldots$ des fonctions définis sur $E^n \times E^m$ à E^n Supposé être continuellement différentiable toujours en x, mais pas toujours en u.

La durée d'un processus de contrôle optimal peut être soit pré-attribué ou non ; c'est-à-dire qu'il est fixe ou estimé.
Il est nécessaire de soulignez cette distinction dès le départ, car on a plus de résultats, et d'algorithmes, pour les problèmes à temps fixe que pour les problèmes de temps libre.

Dans un premier temps on va se limiter à des problèmes de contrôle optimal à temps fixe, et nous verrons plus tard comment les problèmes à temps libre peuvent souvent être résolu en résolvant une séquence de problèmes à temps fixe avec des durées croissantes.

En tout état de cause, nous supposerons toujours que la durée du processus optimal est finie, car sinon l'optimisation pourrait être effectuées dans un espace de dimension infinie, qui sort du cadre de ce modèle.

On s'interessera là au type de contraintes que nous rencontrerons dans des processus à temps fixe, que nous supposons avoir une durée de k étapes.

Nous supposons que l'on nous donne k sous-ensembles de E^m, qu'on désignera par U, pour $i = 0, 1, \ldots, k-1$; $k+1$ sous ensembles de E^n qu'on désignera par X_i pour $i = 0, 1, \ldots, k$; une fonction $h(.)$ de $E^{n(k+1)} \times E^{km}$ à E^s où s est un entier positif ; et un sous-ensemble $D \subset E^s$.

En fonction du résultat souhaité, par la suite on peut imposer plusieurs conditions aux ensembles U_i, X_i, et D ainsi qu'à la fonction $h(.)$.

Considèrons u=$(u_0, u_1, \ldots, _{k-1})$ une séquence de contrôle et soit x=$(x_0, x_1, \ldots, x_k)$ la trajectoire correspondante déterminée par le système 1.1.

Définition 1.1 *On dira qu'une séquence de contrôle u et une trajectoire correspondante x sont admissibles si elles satisfaisaient les contraintes de contrôle 1.1,*

$$u_i \in U_i, \qquad i = 0, 1, \ldots, k-1; \tag{1.15}$$

Les contraintes sur l'état du système

$$x_i \in X_i, \qquad i = 0, 1, \ldots, k; \tag{1.16}$$

et les contraintes sur la trajectoire

$$h(x, u) \in D. \tag{1.17}$$

Dans un problème à valeur bornée, les ensembles U_i peuvent être des intervalles de la forme $U_i = \{u : |u| \leq 1\}$ pour $i = 0, 1, \ldots, k-1$; les ensembles X_0 et X_k peuvent être des variétés de la forme $X_i = \{x : g_i(x) = 0\}$ pour $i = 0, k$ où $g_i : E^n \to E^{l_i}$ pour $i = 0, k$; et les autres contraintes peuvent être inexistantes ; c'est-à-dire $X_i = E^n$ pour $i = 1, 2, \ldots, k-1$ et $h \equiv 0$.
La fonction h est couramment utilisée pour exprimer des limitations imposées par la quantité totale de ressources disponibles, telles que l'énergie ou le carburant, auquel cas (1.17) peut prendre la forme :

$$\sum_{i=0}^{k-1} \|u_i\|^2 \leq d \quad \text{ou} \quad \sum_{i=0}^{k-1}\sum_{j=0}^{m} |u_i^j| \leq d, \tag{1.18}$$

où $\|.\|$ est la norme euclidienne.

Enfin, nous supposons que l'on a une fonction de coût a valeur réelle

$$f(\cdot,\cdot) \tag{1.19}$$

défini sur $E^{n(k+1)} \times E^{mk}$. Comme exemple typique de la fonction $f(\cdot,\cdot)$ considérons celui défini par :

$$f(x,u) = \|x_k - x_d\|^2 + \sum_{i=0}^{k-1} \|u_i\|^2; \tag{1.20}$$

Qui exprime l'écart entre l'état terminal x_k et un point donné $x_d \notin X_k$, additionné de l'énergie dépensée par le système (1.1) de l'état initial $x_0 \in X_0$ à l'état terminal $x_k \in X_k$.

Nous pouvons maintenant combiner le système dynamique (1.14) avec les contraintes (1.15) à (1.17) et la fonction de coût (1.19) pour obtenir une formulation générale du problème de contrôle optimal à temps fixe.

1.2.1 Problème de contrôle optimal à temps fixe

Étant donné un système dynamique :

$$x_{i+1} - x_i = f_i(x_i, u_i), \qquad i = 0, 1, \ldots, k-1; \tag{1.21}$$

Où les états $x_i \in E^n$ pour $i = 0, 1, \ldots, k$, les commandes $u_i \in E^m$ pour $i = 0, 1, \ldots, k-1$, et k est la durée spécifiée pour le processus,avec les sous-ensembles $X_i \subset E^n$ pour $i = 0, 1, \ldots, k$, les sous-ensembles $U_i \subset E^m$ pour $i = 0, 1, \ldots, k-1$, un sous-ensemble $D \subset E^*$, une fonction de contrainte $h(.)$ de $E^{n(k+1)} \times E^{mk}$, et une fonction de coût à valeur réelle $f(\cdot,\cdot)$ définie sur $E^{n(k+1)} \times E^{mk}$.

Cherchons une séquence de contrôle $\hat{u}$ =$(\hat{u}_0, \hat{u}_1, \dots, \hat{u}_{k-1})$ et une trajectoire correspondante $\hat{x}$ =$(\hat{x}_0, \hat{x}_1, \dots, \hat{x}_k)$ satisfaisant (1.21),avec

$$\hat{u}_i \in U_i, \qquad i = 0, 1, \dots, k-1; \tag{1.22}$$

$$\hat{x}_i \in X_i, \qquad i = 0, 1, \dots, k; \tag{1.23}$$

$$h(\hat{x}, \hat{u}) \in D. \tag{1.24}$$

Telles que pour chaque séquence de contrôle u= $(u_0, u_1, \dots, u_{k-1})$ et chaque trajectoire correspondante x= $(x_0, x_1, \dots, x_k)$ satisfaisant (1.22) à (1.24)

$$f(\hat{x}, \hat{u}) \leq f(x, u) \tag{1.25}$$

Remarque 1.2 *Un problème à temps libre diffère d'un problème à temps fixe en un détail important : la durée du processus k n'est pas spécifiée à l'avance.*

La technique la plus simple d'étendre la définition du problème de contrôle optimal à temps fixe à des problèmes à temps libre en supposant qu'au lieu d'avoir une fonction de coût fixe f et une fonction de contrainte h, nous avons une séquence de telles fonctions, $f_{(k)}$ et $h_{(k)}$ paramétrées par la durée k.

De même, nous devons supposer que nous avons une séquence de sous-ensembles D_k. Ainsi, nous considérerons le problème à temps libre comme une séquence de problèmes à temps fixe.

1.2.2 Problème de contrôle optimal à temps libre

Étant donné un système dynamique :

$$x_{i+1} - x_i = f_i(x_i, u_i), \qquad i = 0, 1, 2, \dots \tag{1.26}$$

Avec les sous-ensembles $X_i \subset E^n$ pour $i = 0, 1, 2, \dots$, les sous-ensembles $U_i \subset E^n$ pour $i = 0, 1, 2, \dots$, les sous-ensembles $D_k \subset E^n$ pour $k = 0, 1, 2, \dots$, une séquence de fonctions de contrainte $h_{(k)}(\cdot, \cdot)$ défini de $E^{n(k+1)} \times E^{mk}$ dans E^s pour $k = 0, 1, 2, \dots$, et une séquence de fonctions de coût à valeur réelle $f_{(k)}(\cdot, \cdot)$ défini sur $E^{n(k+1)} \times E^{mk}$ pour $k = 0, 1, 2, \dots$.

Cherchons un entier k, une séquence de contrôle $\hat{u}_{\hat{k}} = (\hat{u}_0, \hat{u}_1, \dots, \hat{u}_{\hat{k}-1})$, et une trajectoire correspondante $\hat{x}_{\hat{k}} = (\hat{x}_0, \hat{x}_1, \dots, \hat{x}_{\hat{k}})$ satisfaisant (1.26), avec

$$\hat{u}_i, \in U_i \qquad i = 0, 1, \cdots, \hat{k}-1; \tag{1.27}$$

$$\hat{x}_i \in X_i, \qquad i = 0, 1, \cdots, \hat{k}; \tag{1.28}$$

$$h_{\hat{k}}(\hat{x}_{\hat{k}}, \hat{u}_{\hat{k}}) \in D_{(\hat{k})}. \tag{1.29}$$

Telles que pour chaque $k = 0, 1, \dots$, chaque séquence de contrôle $u_k = (u_0, u_1, \dots, u_{k-1})$, et pour chaque trajectoire correspondante $x_k = (x_0, x_1, \dots, x_{k-1})$ satisfaisant (1.26) à (1.29), avec k prenant la place de $\hat{k}$, on a :

$$f_{\hat{k}}(\hat{x}_{\hat{k}}, \hat{u}_{\hat{k}}) \leq f_k(x_k, u_k) \tag{1.30}$$

1.3 Forme canonique du problème de contrôle optimal discret

Il existe différentes manières d'associer un coût à un processus de contrôle à temps fixe (x, u), c'est-à-dire une séquence de contrôle u et une trajectoire correspondante x.
La plus courante consiste à attribuer un coût à chaque état de transition, auquel cas le coût total est la somme de ces coûts individuels ,
et nous avons donc :

$$f(x,u) = \sum_{i=0}^{k-1} f_i^0(x_i, u_i), \tag{1.31}$$

Où $f_i^0(\cdot,\cdot)$ pour $i = 0, 1, \ldots, k-1$ sont des fonctions à valeur réelle défini sur $E^n \times E^m$.

Exemple 1.1 *a) Considérons un problème d'énergie minimale pour lequel*

$$f(x,u) = \sum_{i=0}^{k-1} \|u_i\|^2, \tag{1.32}$$

Où $\| \, . \, \|$ désigne la norme euclidienne.

b) Supposons que l'on souhaite minimiser la valeur $\varphi(x_k)$ d'une fonction à valeur réelle de l'état terminal, c'est-à-dire :

$$f(x,u) = \varphi(x_k). \tag{1.33}$$

Si nous examinons (1.33) *et* (1.31), *nous constatons qu'on supposons $f_i^0(\cdot,\cdot) \equiv 0$ pour $i = 0, 1, \ldots, k-2$ et $f_{k-1}^0(x_{k-1}, u_{k-1}) = \varphi(x_{k-1} + f_{k-1}(x_{k-1}, u_{k-1}))$, nous pouvons convertir* (1.33) *sous la forme de* (1.31).

Par ces exemples il est clair que de nombreux problèmes de contrôle optimal ont des fonctions de coût qui peuvent être écrites sous la forme (1.31).
Cependant, il existe également un certain nombre de problèmes de contrôle optimal dont le coût associé au procédé de contrôle (x, u) ne peut être décomposé en une somme.

Exemple 1.2 *lorsque la fonction de coût $f(\cdot,\cdot)$ est sous la forme :*

$$f(x,u) = \max_{i=0,1,\ldots,k-1} \varphi^i(x_i, u_i) \tag{1.34}$$

C'est-à-dire, lorsque nous avons comme objectif minimiser la "déviation" maximale d'un processus de contrôle .

Remarque 1.3 *Les problèmes de contrôle optimal à temps fixe avec des coûts de la forme* (1.31) *sont les plus courants.*

Il est parfois commode de combiner la dynamique d'équations (1.14) *avec l'expression de coût* (1.31) *en une seule variation du système dynamique comme suit :*
Soit les scalaires x_i^0 *pour* $i = 0, 1, \ldots, k$ *déterminé par l'équation :*

$$x_{i+1}^0 - x_i^0 = f_i^0(x_i, u_i), \qquad i = 0, 1, \ldots, k-1, \tag{1.35}$$

avec $x_0^0 = 0$*, où* $((x_0, x_1, \ldots, x_k), (u_0, u_1, \ldots, u_{k-1}))$ *le processus de contrôle du système* (1.14).

On suppose $x = (x^0, x) \in E^{n+1}$ *, où* $x^0 \in E^1$ *et* $x \in E^n$ *[c'est-à-dire* $x = (x^0, x^1, \ldots, x^n)$*], et pour* $i = 0, 1, \ldots, k-1$ *nous définissons les fonctions*

$f_i : E^{n+1} \times E^m \to E^{n+1}$ *par* $f_i(x, u) = (f_i^0(x, u), f_i(x, u))$.

Enfin, nous combinons les équations (1.14) *et* (1.35) *suivantes :*

$$x_{i+1}^0 - x_i^0 = f_i^0(x_i, u_i), \qquad i = 0, 1, \ldots, k-1 \tag{1.36}$$

Avec l'introduction du système augmenté (1.36)*, on est conduit au cas particulier suivant du problème de contrôle optimal à temps fixe.*

1.3.1 Problème de contrôle optimal canonique

Étant donné le système dynamique suivant :

$$x_{i+1} - x_i = f_i^0(x_i, u_i), \qquad i = 0, 1, \ldots, k-1; \tag{1.37}$$

où les états $x_i \in E^{n+1}$ pour $i = 0, 1, \ldots, k$, les commandes $u_i \in E^m$ pour $i = 0, 1, \ldots, k-1$ et k est la durée spécifié du processus, ainsi que les sous-ensembles $U_i \subset E^m$ pour $i = 0, 1, \ldots, k$, les sous-ensembles $U_i \subset E^m$ pour $i = 0, 1, \ldots, k-1$, un sous-ensemble $D \subset E^s$, et une fonction de contrainte $h(\cdot ; \cdot)$ défini de $E^{(n+1)(k+1)} \times E^{km}$ dans E^s.

Cherchons la séquence de contrôle $\hat{u} = (\hat{u}_0, \hat{u}_1, \cdots, \hat{u}_k)$ et une trajectoire correspondante $\hat{x} = (\hat{x}_0, \hat{x}_1, \ldots, \hat{x}_{k-1})$, satisfaisant (1.37), avec

$$\hat{u}_i \in U_i, \qquad i = 0, 1, \ldots, k-1; \tag{1.38}$$

$$\hat{x}_i \in X_i, \qquad i = 0, 1, \ldots, k-1; \tag{1.39}$$

$$h(\hat{x}, ; \hat{u}) \in D, \tag{1.40}$$

Telles que pour chaque séquence de contrôle u et toute trajectoire correspondante x satisfaisant (1.37) à (1.40),on a

$$\hat{x}_k^0 \leq x_k^0, \tag{1.41}$$

où x_k^0 est la première composante de x_k .

Remarque 1.4 *Dans la définition ci-dessus, nous n'avons pas supposé l'hypothèse implicite par* (1.37) *que les fonctions f_i pour $i = 0, 1, \ldots, k-1$ et h ne dépend pas de la variable de coût x^0.*

Comme on le verra dans l'exemple suivant , la suppression de cette hypothèse nous permet de traiter une classe importante de problèmes sans augmenter la dimension du système dynamique.

Lorsque le système (1.37) est bien un système augmenté, alors les fonctions f_i et h ne dépendent pas de x^0, c'est-à-dire $f_i : E^n \times E^m \to E^{n+1}$ et $h : E^{n(k+1)} \times E^m \to E^s$ et les ensembles X_i prennent la forme

$$X_0 = \{0\} \times X_0, \tag{1.42}$$

$$X_i = E^1 \times X_i, \qquad i = 1, 2, \ldots, k.$$

L'objectif de la transformation d'un problème de contrôle optimal sous la forme canonique est de rendre le problème complètement géométrique.

Exemple 1.3 *Il n'est pas toujours nécessaire d'augmenter le système dynamique* (1.14) *afin de transformer le problème sous une forme canonique.*
C'est clairement le cas lorsque le coût est de la forme :

$$f(x, u) = x_k^1 \tag{1.43}$$

Considérons donc le cas où l'on a comme objectifs un objet à se déplacer sur une ligne d'une position initiale x_0 à une position aussi éloignée possible (par exemple, un problème balistique).

Dans ce cas, nous pouvons nous soucier peu des valeurs terminales des autres variables d'état.
Alors si pour $i = 0, 1, \ldots, k$ nous laissons $x_i^0 = -(x_i^1 - x_0^1)$ et renuméroter les autres variables d'état $x_i^2, x_i^3, \ldots, x_i^n$ comme $x_i^1, x_i^2, \ldots, x_i^{n-1}$ nous constatons que le problème initial a été transcrit sous forme canonique sans que nous ayons augmenté l'espace d'état.

1.4 Problème de programmation mathématique

Il sera commode d'adopter une forme canonique pour le problème de programmation mathématique. Nous l'appellerons problème de base.

1.4.1 Problème de base

Compte tenu d'une valeur réelle continuellement différentiable une fonction $f(.)$ définie sur E^n, une fonction continuellement différentiable $r(.)$ de E^n à E^m, et un sous-ensemble $\Omega \subset E^n$,
Cherchons un vecteur $\hat{z} \in E^n$ satisfaisant

$$\hat{z} \in \Omega \tag{1.44}$$

$$r(\hat{z}) = 0 \tag{1.45}$$

de telle sorte que

$$f(\hat{z}) \leq f(z) \tag{1.46}$$

pour tous les $z \in E^n$ satisfaisant (1.44) et (1.45).

Notez que nous sommes abstenus de dire comment l'ensemble Ω est caractérisé et que nous avons choisi de souligner la présence des contraintes d'égalité sur la minimisation de $f(z)$, la raison de la décomposition des contraintes est qu'il est souvent impossible d'obtenir des conditions nécessaires significatives d'optimalité sans imposer cette structure supplémentaire.

Remarque 1.5 *La formulation du problème de base n'implique pas que les contraintes d'égalité ne puissent entrer dans la caractérisation de l'ensemble Ω.*
L'ensemble Ω peut-être de la forme $\Omega = \{z : s(z) = 0, q(z) \leq 0\}$, où $s : E^n \to E^l$ et $q : E^n \to E^m$.
Nous choisissons simplement de mettre en évidence certaines des contraintes d'égalité.

Définition 1.2 *Le vecteur $z \in E^n$ sera dit réalisable (ou solution réalisable) pour le problème de base s'il vérifie* (1.44) *et* (1.45).
Un vecteur $\hat{z} \in E^n$ sera dit optimal (ou solution optimale) pour le problème de base s'il est réalisable et vérifie (1.46) *pour tous les vecteurs possibles $z \in E^n$*

1.5 Équivalence des problèmes d'optimisation

Nous allons maintenant montrer que les problèmes de contrôle optimal à temps fixe ou du problème de contrôle optimal discret sous forme canonique peuvent être transcrit sous la forme du problème de base et transcrit en un problème de contrôle optimal à deux niveaux . Le fait que ces transformations soient possibles est important, car il permet d'interpréter les résultats obtenus pour l'un ou l'autre de ces problèmes à la lumière de l'autre.

Considérons donc à nouveau le problème de contrôle optimal à temps fixe, et soit $z = (x_0, x_1, ..., x_k, u_0, u_1, ..., u_{k-1})$,
où $u = (u_0, u_1, \ldots, u_{k-1})$ et $x = (x_0, x_1, \ldots, x_k)$ sont une séquence de contrôle et une trajectoire correspondante du système (1.21) respectivement.

Supposons que pour $i = 0, 1, ..., k$ les sous-ensembles X_i de E^n [voir (1.16)] sont de la forme :

$$X_i = [x : g_i(x) = 0; x \in X_i'], \tag{1.47}$$

où $g_i(.)$ est une fonction continuellement différentiable allant de E^n à E^{l_i} et X_i' est un sous-ensemble de E^n qui ne peut être décrit par un système d'équations.
Supposons également que pour $i = 0, 1, ..., k$ les sous-ensembles U_i [voir (1.15)] de E^m sont de la forme :

$$U_i = \{u_i : \psi_i(u_i) = 0, \quad u_i \in U_i'\}, \tag{1.48}$$

où ψ_i est une fonction continuellement différentiable défini sur E^m dans E_i^s et U_i' est un sous-ensemble de E^m qui, encore une fois, ne peut être décrit par un système d'équations.
Enfin, soit $V = h^{-1}(D)$ [voir (1.17)].
Delà, pour convertir les problèmes de contrôle optimal à temps fixe à un problème de contrôle optimal discret canonique, nous pouvons laisser

$$\Omega = (X_0' \times X_1' \times ... \times X_K' \times U_0' \times U_1' \times ... \times U_{k-1}') \cap V, \tag{1.49}$$

et on défini la fonction $r(.)$ par

$$r(z) = \begin{bmatrix} x_1 - x_0 - f_0(x_0, u_0) \\ x_2 - x_1 - f_1(x_1, u_1) \\ . \\ . \\ . \\ x_k - x_{k-1} - f_{k-1}(x_{k-1}, u_{k-1}) \\ g_0(x_0) \\ g_1(x_1) \\ . \\ . \\ . \\ g_k(x_k) \\ \psi_0(u_0) \\ \psi_1(u_1) \\ . \\ . \\ . \\ \psi_{k-1}(u_{k-1}) \end{bmatrix} \tag{1.50}$$

La fonction de coût reste bien entendu la même, $f(z) = f(x, u)$

Nous transcrivons le problème de base en un problème de contrôle optimal en une étape de la forme (1.2.1) en traitant la variable z comme un contrôle et en construisant le système dynamique fictif donné par l'équation :

$$x_{i+1} - x_i = f_i(x_i, z), \qquad i = 0, \tag{1.51}$$

où $x_i = (x_i^1, x_i^2, ..., x_i^m) \in E^m$, pour $i = 0, 1$ et $f_0(x_0, z) \triangleq r(z)$.

Les contraintes deviennent alors $X_0 = \{0\}, X_1 = \{0\}$ et $U_0 = \Omega$, le coût de cette transition en une étape est donné par $f_0^0(x_0, z) \triangleq f(z)$
Pour obtenir (1.3.1), nous procédons simplement comme dans la section (1.3).
Nous avons ainsi vu que pour les besoins de l'analyse, le problème de contrôle

optimal du temps fixe et le problème de base sont équivalent, et nous choisirons donc la forme à utiliser dans l'analyse de tout problème particulier simplement sur la base de la commodité.

Contrôle optimal et programmation quadratique 2

2.1 Formulation du problème général de contrôle avec coût quadratique et transformation en problème de programmation quadratique

Nous allons maintenant considérer une classe de problèmes de contrôle optimal discrets pour lequel le problème de base correspondant est un problème de programmation quadratique de la forme :

$$\text{Minimiser } 1/2\langle z, Q_z\rangle + \langle d, z\rangle \tag{2.1}$$

sous les contraintes

$$Rz = c, \qquad Pz \leq v, \qquad \alpha \leq z \leq \beta.$$

Où d, α et β sont dans E^n, Q est une matrice $(n \times n)$ symétrique (généralement définie semi-positive), R est une matrice $(m \times n)$, P est une matrice $(l \times n)$, et $c \in E^m$ et $v \in E^l$ (on supposera toujours que $\alpha^i < \beta^i$ pour $i = 1, 2, ..., n$ et qu'il est possible d'avoir $\alpha^i = -\infty$ et $\beta^i = +\infty$ pour tout i et j dans $\{1, 2, ..., n\}$).

La formulation de cette classe de problèmes est destinée à être suffisamment général pour inclure tous les problèmes de contrôle optimal discrets avec des coûts quadratiques.

Pourtant, les variations possibles sur de tels problèmes sont si étendues qu'il est pratiquement impossible de formuler une classe suffisamment générale pour inclure tous les cas possibles.
La formulation suivante est la plus simple qui comprendra tous les problèmes standard.

2.1.1 Problème de contrôle quadratique (PCQ)

Considérons un système dynamique décrit par l'équation linéaire suivante :

$$x_{i+1} - x_i = Ax_i + Bu_i. \tag{2.2}$$

Où $x_i \in E^n$ est l'état du système à l'instant i, $u_i \in E^m$ est l'entrée du système à l'instant i, A est une matrice $n \times n$, et B est une matrice $n \times m$.

Cherchons une séquence de contrôle $\hat{u} = (\hat{u}_0, \hat{u}_1, ..., \hat{u}_{k-1})$ et une trajectoire correspondante $\hat{x} = (\hat{x}_0, \hat{x}_1, \ldots, \hat{x}_k)$ déterminé par (2.2), qui minimise la fonction de coût quadratique :

$$f(z) = 1/2\langle z, Q_z\rangle + \langle d, z\rangle, \tag{2.3}$$

où $z = (x, u) = (x_0, x_1, ..., x_k, u_0, u_1, ..., u_{k-1})$, Q est une matrice de dimension $[(k+1)n + km] \times [(k+1)n + km]$ symétrique définie semi-positive, et $d \in E^{(k+1)n+km}$.

La minimisation est soumise aux contraintes :

$$u_i \in U_i = \{u : F_i u_i \leq w_i\}, \qquad i = 0, 1, ..., k-1; \tag{2.4}$$

$$x_i \in X_i^{'} = \{x : H_i x \leq v_i\}, \qquad i = 0, 1, ..., k \tag{2.5}$$

.

Et en outre, x_0 et x_k doivent satisfaire :

$$G_0 x_0 = c_0, \qquad G_k x_k = c_k, \tag{2.6}$$

où les F_i et w_i pour $i = 0, 1, ..., k$, les H_i et v_i pour $i = 0, 1, ..., k-1$, et G_0,G_k,c_0, et c_k sont des matrices appropriées et vecteurs

Remarque 2.1 *Certaines des matrices de* (2.3) *à* (2.6) *peuvent être nulles.*

Nous réduisons maintenant ce modèle à un problème de base en utilisant l'identification vu au paragraphe (1.5)

Soit z tel que défini (2.3) et soit f et $\dot{r}$ défini par :

$$f(z) = 1/2\langle z, Q_z\rangle + \langle d, z\rangle \tag{2.7}$$

$$\dot{r}(z) = \begin{bmatrix} x_1 - x_0 - Ax_0 - Bu_0 \\ . \\ . \\ . \\ x_k - x_{k-1} - Ax_{k-1} - Bu_{k-1} \\ G_0 x_0 - c_0 \\ G_0 x_0 - c_0 \\ G_k x_k - c_k \end{bmatrix} = Rz - c. \tag{2.8}$$

Où c est de la forme $c = (0, ..., 0, c_0, c_k)$.
Ω est défini par :

$$\Omega = X_0' \times X_1' \times X_2' \times ...X_k' \times U_0 \times U_1 \times ... \times U_{k-1}; \tag{2.9}$$

Donc Ω est de la forme :

$$\Omega = \{z : Pz \leq v, \alpha \leq z \leq \beta\}, \tag{2.10}$$

Où P,v,α et β sont dérivés de (2.4) et (2.5).

Remarque 2.2 *Il existe une méthode alternative pour transcrire les problèmes de contrôle en un problème de base qui conduit à un nombre réduit de variables.*
Cette méthode implique la résolution de l'équation (2.2) *pour chacun des états* x_i *avec* $i = 1, ..., k$*, en fonction de l'état initial* x_0 *et de la séquence de contrôle* $u = (u_0, u_1, ..., u_{k-1})$ *puis en éliminant ces états du problème en les substituant dans la fonction de coût et des contraintes.*

2.2 L'existence d'une solution optimale au problème de programmation quadratique

On va étudier à présent la question de savoir quand une solution optimale d'un problème de programmation quadratique existe.
Dans cette section nous verrons que nous n'avons besoin que d'un minimum d'hypothèses pour garantir l'existence d'une solution optimale.
Premièrement, nous introduisons une forme du problème de programmation quadratique (2.1) ce qui facilitera notre tâche, et une terminologie :

2.2.1 Problème de programmation quadratique (PQ)

$$\begin{array}{l} \text{Minimiser} \quad f(z) = 1/2\langle z, Q_z\rangle + \langle d, z\rangle \\ \qquad \text{sous les contraintes} \qquad\qquad\qquad (PQ) \\ Rz = c \quad \text{et} \quad \alpha \leq z \leq \beta, \end{array}$$

Où Q est une matrice $n \times n$ symétrique semi-définie positive, $d \in E^n$, R est une matrice $m \times n$, et α,β sont des vecteurs dans E^n dont les composantes peuvent prendre les valeurs $\pm\infty$

Théorème 2.1 *Si* $\langle d, z\rangle = 0$ *pour chaque* $z \in E^n$ *satisfaisant* $Qz = 0$ *et* $Rz = 0$ *et il existe une solution réalisable à (PQ), alors ce dernier admet une solution optimale au problème.*

Démonstration. Soit $\Omega\prime = \{z : Rz = c, \alpha \leq z \leq \beta\}$; c'est-à-dire que $\Omega\prime$ est l'ensemble de solutions réalisables de (PQ). $\Omega\prime$ est non vide, par hypothèse.
La démonstration du théorème sera faite en deux parties.
Nous montrerons d'abord que $f(z)$ est borné par le bas pour $z \in \Omega\prime$, et donc qu'il existe un nombre fini γ qui est le minimum de $f(z)$ pour $z \in \Omega\prime$.

Ensuite, nous allons montrer qu'il existe un $z^* \in \Omega\prime$, tel que $f(z^*) = \gamma$.
Pour commencer, désignons par $\pi(R)$ et $\pi(Q)$ les espaces nuls de R et Q, respectivement.

Par hypothèse , $d \in [\pi(R) \bigcap \pi(Q)]^{\perp}$, le complément orthogonal de $\pi(R) \cap \pi(Q)$.
Cela équivaut à dire que :

$$d \in [\pi(\begin{bmatrix} Q \\ \cdots \\ R \end{bmatrix})]^{\perp}$$

Où $\begin{bmatrix} Q \\ \cdots \\ R \end{bmatrix}$ une matrice $(n+m) \times n$, avec Q et R comme sous-matrices , comme indiqué.

Puisque d est orthogonal à l'espace nul de $\begin{bmatrix} Q \\ \cdots \\ R \end{bmatrix}$, il doit être contenu dans la gamme de $\begin{bmatrix} Q \\ \cdots \\ R \end{bmatrix}^T = \left[Q \vdots R^T \right]$.
Par conséquent, nous devons avoir

$$d = Qu + R^T v, \tag{2.11}$$

où u et v sont des vecteurs dans E^n et E^m, respectivement. Substituons (2.11) dans l'expression de $f(z)$, et on obtient :

$$f(z) = \langle z, Q_z \rangle + \langle Qu + R^T v, z \rangle = \langle z, Q_z \rangle + \langle u, Q_z \rangle + \langle v, Rz \rangle. \tag{2.12}$$

Maintenant, nous ne nous intéressons qu'aux vecteurs $z \in E^n$ satisfaisant $Rz = c$. Il s'ensuit donc que

$$f(z) = \langle z + u, Q_z \rangle + \langle v, c \rangle \quad , \quad \text{pour tout } z \in \Omega\prime. \tag{2.13}$$

Comme Q est symétrique et semi-défini positive, on obtient pour Q l'expansion spectrale :

$$Q = \sum_{i=1}^{n} \lambda^i \xi_i \rangle \langle \xi_i, \tag{2.14}$$

Où le $\lambda^i \geq 0$ pour $i = 1, 2, ..., n$ sont les valeurs propres de Q, $\xi_i = (\xi_i^1, \xi_i^2, ..., \xi_i^n)$ est un ensemble correspondant de valeurs propres orthonormées vecteurs de Q, $\xi_i \rangle \langle \xi_i$ est une matrice $n \times n$ dont les l^{ieme} et k^{ieme} éléments donnent $\xi_i^l \ \xi_i^k$.
Donc

$$\langle z + u, Qz \rangle = \sum_{i=1}^{n} \lambda^i (\langle \xi_i, z \rangle^2 + \langle \xi_i, \mu \rangle \langle \xi_i, z \rangle) \tag{2.15}$$

Où les quantités entre parenthèses,

$$\langle \xi_i, z \rangle^2 + \langle \xi_i, \mu \rangle \langle \xi_i, z \rangle, \qquad i = 1, 2, ..., n,$$

sont de la forme $x^2 + ax$, qui admet un minimum fini pour chaque valeur finie de a.
Puisque $\lambda^i \geq 0$, pour $i = 1, 2, ..., n$, nous pouvons conclure que $\langle z + \mu, Qz \rangle$ doit être délimité par le bas pour tout z.

Nous avons déjà vu que $f(z) = \langle z + \mu, Qz \rangle + \langle v, c \rangle$ pour tout $z \in \Omega\prime$.
Donc $f(z)$ est délimité par le bas sur $\Omega\prime$. Soit γ la plus grande borne inférieure de $f(z)$ pour $z \in \Omega\prime$:

$$\gamma = inf_{z \in \Omega\prime} \quad f(z) > -\infty. \tag{2.16}$$

Ceci complète la première partie de la preuve.

Puisque γ est l'infimum de $\Omega\prime$, il s'ensuit qu'il existe une séquence de points $\{z_i\}_i^\infty$ dans $\Omega\prime$, telles que $f(z_i) \to \gamma$.

Si $\{z_i\}_i^\infty$ est convergent, ou même s'il a une sous-séquence convergente, c'est ce qu'il fallait démontrer, puisque le point limite de cette séquence serait en $\Omega\prime$ et donnerait la valeur requise pour f.

Cependant, cette séquence n'a pas besoin d'être bornée, puisque $\Omega\prime$ n'a pas besoin d'être borné, et donc il n'a pas besoin d'avoir une sous-séquence convergente.

À présent il suffit de projeter $\Omega\prime$ et la séquence $\{z_i\}_i^\infty$ sur un sous-espace de manières à obtenir une séquence bornée avec certaines propriétés intéressantes.

Notez que puisque Q est symétrique, le complément orthogonal de $\pi(Q)$, l'espace nul de Q, est $\Re(Q)$, la gamme de Q.
Soit P une matrice $n \times n$ telle que $Px = 0$, pour tous $x \in \pi(Q)$, et $Px = x$, pour tout $x \in \Re(Q)$; c'est-à-dire que P est une matrice de projection orthogonal.

Soit $\{z_i\}_i^\infty$ une suite dans $\Omega\prime$, telle que $f(z_i) \to \gamma$ avec

$$y_i = Pz_i \qquad i = 1, 2, \cdots \tag{2.17}$$

Et

$$\hat{\Omega}\prime = \{y \in E^n : y = Qz, z \in \Omega\prime\}. \tag{2.18}$$

$$y_i \in \hat{\Omega} \quad \text{pour chaque} \quad i.$$

En outre, si nous définissons $h : E^n \to E^1$

$$h(z) = \langle z + \mu, Qz \rangle, \tag{2.19}$$

Alors

$$h(y_i) = h(z_i) = f(z_i) - \langle v, c \rangle, \quad \text{pour tout} \quad i, \tag{2.20}$$

puisque par définition de y_i ,

$$z_i = y_i + x_i, \tag{2.21}$$

où $x_i \in \pi(Q)$. Il s'ensuit que

$$h(y_i) \to \gamma - \langle v, c \rangle. \tag{2.22}$$

Delà, la suite $\{h(y_i)\}$ est bornée.
De plus, Q restreint à $\Re(Q)$ est défini positive, et donc $\{h(y_i)\}_{i=1}^{\infty}$ est borné si et seulement si $\{y_i\}_{i=1}^{\infty}$ est borné.

On conclut donc de (2.21) qu'il existe au moins un point limite y^* vers lequel converge une sous-suite de $\{y_i\}_{i=1}^{\infty}$.
Si $\Omega\prime$ est fermé, la démonstration s'arrête, puisque y^* appartient à $\Omega\prime$, et tout vecteur $z^* \in \Omega\prime$ correspondant pour lequel $y^* = Pz^*$ satisfait $f(z^*) = \gamma$.

Puisqu'il n'est pas vrai en général que l'image d'un ensemble fermé sous une application linéaire soit fermée, il faut faire appel à la nature particulière de l'ensemble $\Omega\prime$, à savoir au fait qu'il s'agit d'un ensemble polyèdre convexe.
Le théorème de représentation suivant est énoncé : □

Théorème 2.2 *Tout polyédre convexe, non vide, fermé S dans E^n de forme*

$$S = \{x : Ax \leq b\}; \tag{2.23}$$

peut-être exprimé comme la somme vectorielle

$$C^{\triangle} + D^L = \{x = x_1 + x_2 : x_1 \in C^{\triangle}, x_2 \in D^L\}; \tag{2.24}$$

d'un polyèdre convexe fermé et borné $C^{\triangle}$ de la forme

$$C^{\triangle} = \{x : x = Cu, u \geq 0, \quad \sum_i u_i = 1\}; \tag{2.25}$$

et un cône polyédrale convexe fermé D^L de la forme

$$D^L = \{x : x = Dw, w \geq 0\}. \tag{2.26}$$

Inversement, tout ensemble non vide de la forme $C^{\triangle} + D^{L}$ *est un ensemble polyédrale fermé convexe .*
Si nous représentons S comme $C^{\triangle} + D^{L}$*, il est facile de voir que l'image de* $C^{\triangle} + D^{L}$ *sous une transformation linéaire P est un ensemble de la même forme,*
Autrement dit , $P[C^{\triangle} + D^{L}] = \hat{C}^{\triangle} + \hat{D}^{L}$*, où*

$$\hat{C}^{\triangle} = \{y : y = (PC)u, u \geq 0, \quad \sum_i u_i = 1\} \tag{2.27}$$

$$\hat{D}^{L} = \{y : y = (PD)w, \quad w \geq 0\}. \tag{2.28}$$

De l'inverse du théorème (2.2)*, nous concluons que* $\hat{C}^{\triangle} + \hat{D}^{L}$ *est un ensemble polyédrale convexe fermé.*

Bien que l'ensemble $\Omega^{'}$ *n'ait pas la forme spécifique de S dans le théorème, on peut facilement vérifier que ces formes sont équivalentes, et donc c'est un polyèdre convexe fermé.*

Corollaire 2.1 *Si* $d = 0$ *et qu'il existe une solution réalisable à (PQ), il existe alors une solution optimale à (PQ).*

Corollaire 2.2 *Si Q est définie positive et qu'il existe une solution réalisable à (PQ), il existe alors une solution optimale à (PQ).*

Corollaire 2.3 *Si R est non singulière et qu'il existe une solution réalisable pour (PQ), alors il existe une solution optimale à (PQ).*

Corollaire 2.4 *Si* $\pi(R) \cap \pi(Q) = \{0\}$ *et qu'il existe une solution réalisable à (PQ), il existe alors une solution optimale à (PQ).*

Ce dernier corollaire est d'un intérêt particulier, puisque, comme nous le verrons dans la section suivante, la condition $\pi(R) \cap \pi(Q) = \{0\}$ est également une condition suffisante pour que la solution optimale à (PQ) soit unique.

Remarque 2.3 *Il convient de souligner que les conditions du théorème* (2.1) *ne sont que des conditions suffisantes pour l'existence d'une solution optimale.*
Des solutions optimales peuvent, en effet, exister même si les conditions celles-ci ne sont pas remplies.
En effet, en se référant à la preuve du théorème (2.1)*, nous voyons que l'hypothèse*

$$d \in [\pi(\begin{bmatrix} Q \\ .. \\ R \end{bmatrix})]^{\perp}$$

N'était nécessaire que pour montrer que $f(z)$ *est borné par le bas pour* $z \in \Omega\prime$*.*
Par conséquent, nous pouvons substituer le théorème (2.1) *par un résultat plus général.*

Théorème 2.3 *Si* $f(z)$ *est borné par le bas pour* $z \in \Omega\prime$ *et* $\Omega\prime$ *n'est pas vide.*
Alors il existe une solution optimale à (PQ).

2.3 Condition suffisante pour une solution optimale unique d'un problème de programmation quadratique

Dans cette section, nous supposerons toujours qu'une solution optimale au problème de programmation quadratique existe, et nous chercherons à établir des conditions qui garantiront qu'il y aura une solution optimale unique.

Comme nous le verrons plus loin, le problème de programmation quadratique est plus facile à résoudre lorsqu'il existe une solution optimale unique.

Il ne semble pas y avoir d'ensemble de conditions qui sont à la fois nécessaires et suffisantes pour que le problème de programmation quadratique ait une solution unique.

De nombreux ensembles de conditions suffisantes sont possibles, nous examinerons dans ce cas un ensemble qui présente non seulement l'avantage d'être relativement simple à vérifier, mais aussi qui implique la non singularité de certaines matrices qui apparaissent dans les algorithmes de résolution du problème de programmation quadratique.

Nous énonçons maintenant ces conditions sous forme d'un théorème.
La terminologie est la même que dans la section (2.2).

Théorème 2.4 *Si $\pi(R) \cap \pi(Q) = \{0\}$ et que (PQ) admet une solution réalisable, alors il a une solution optimale unique.*

Démonstration. Notez d'abord que si une solution réalisable de (PQ) existe, alors une solution optimale existe d'après le corollaire (2.3).
Par conséquent, nous avons besoin de montrer seulement qu'il ne peut y avoir deux solutions optimales distinctes.

Nous le prouvons par contradiction.
Supposons que z_1 et z_2 ou $z_1 \neq z_2$, sont les deux solutions optimales de (PQ), alors $z_2 = z_1 + z_0$ pour certains $z_0 \neq 0$, et $z_0 \in \pi(R)$. Désignant à nouveau l'ensemble de tous solutions réalisables par $\Omega\prime$, on constate que

$$z(\lambda) = \lambda z_1 + (1-\lambda) z_2 \in \Omega\prime, \text{ pour chaque } \lambda \in [0,1]$$

Car $\Omega\prime$, est un ensemble convexe.
Ainsi $z(\lambda)$ est une solution réalisable de (PQ) pour chaque $\lambda \in [0,1]$.

Nous montrerons que pour tout $\lambda \in [0,1]$, $f(z(\lambda)) < f(z_1) = f(z_2)$ ce qui est en contradiction avec l'hypothèse selon laquelle z_1 et z_2 sont tous deux optimaux. Notons que

$$f(z_1) = f(z_2) = f(z_1 + z_0) = f(z_1) + f(z_0) + \langle z_1, Q z_0 \rangle \tag{2.29}$$

Et par conséquent

$$f(z_0) = -\langle z_1, Q z_0 \rangle \tag{2.30}$$

Après quelques transformations algébriques , on peut montrer que

$$f(z(\lambda)) = f(z_1) + (1-\lambda)[\langle z_1, Qz_0\rangle + 1/2(1-\lambda)\langle z_0, Qz_0\rangle + \langle z_0, d\rangle]. \quad (2.31)$$

Delà

$$f(z(\lambda)) = f(z_1) - (\lambda(1-\lambda)/2)\langle z_0, Qz_0\rangle \quad (2.32)$$

Puisque $\pi(R) \cap \pi(Q) = \{0\}$ et $z_0 \in \pi(R)$,

Nous concluons que $z_0 \notin \pi(Q)$, c'est-à-dire que $\langle z_0, Qz_0\rangle > 0$ (ceci fait suite à l'écriture de Q sous forme d'expansion spectrale).

Donc $f(z(\lambda)) < f(z_1)$ pour tous $\lambda \in (0,1)$.

La condition que les espaces nuls de Q et R n'aient que le vecteur zéro en commun est relativement facile à vérifier, car il est équivalent à vérifier que la matrice $\begin{bmatrix} Q \\ R \end{bmatrix}$ $(n+m) \times n$ est de rang n .

Si nous faisons la restriction que R est de rang m, c'est à dire que les contraintes d'égalité forment un ensemble d'équations linéairement indépendantes, nous obtenons un résultat encore plus fort, sous forme du théorème suivant : □

Théorème 2.5 *Supposons que la matrice R est de rang m; alors Si $\pi(Q) \cap \pi(R) = \{0\}$ si et seulement si la matrice $(n+m) \times (n+m)$*

$$D = \begin{bmatrix} -Q & R^T \\ R & 0 \end{bmatrix} \quad (2.33)$$

est non singulière.

Démonstration. Chaque vecteur $u \in E^n \times E^m$ peut-être partitionné en deux composantes, $u = (x, v)$ où $x \in E^n$ et $v \in E^m$. Si D est non-singulière, alors pour chaque vecteur u de la forme $u = (x, 0)$, avec $x \neq 0$,
nous avons

$$Du = \begin{bmatrix} -Qx \\ Rx \end{bmatrix} \neq 0 \quad (2.34)$$

Ce qui implique que $\pi(Q) \cap \pi(R) = \{0\}$.

Inversement, supposons que $\pi(Q) \cap \pi(R) = \{0\}$ et que $u^* = (x^*, v^*)$
satisfait $Du^* = 0$.

Ensuite, à partir de (2.33),

$$R^T v^* = Qx^* \quad (2.35)$$

$$Rx^* = 0. \tag{2.36}$$

En utilisant les équations (2.35) et (2.36), on obtient

$$\langle x^*, Qx^* \rangle = \langle x^*, R^T v^* \rangle = \langle Rx^*, v^* \rangle = 0, \tag{2.37}$$

Et donc $x^* \in \Re(Q)$. D'aprés (2.36), du fait que $x^* = 0$,
donc $\pi(Q) \cap \pi(R) = \{0\}$.
En partons de (2.35) et que

$$R^T v^* = 0. \tag{2.38}$$

Donc R est de rang m, $R^T v^* = 0$ implique que $v^* = 0$. Par conséquent, la solution correspondante à $Du^* = 0$ est $u^* = 0$, et donc D est non singulière.

Remarque 2.4 *À première vue, ce résultat peut ne pas sembler plus significatif que le fait noté précédemment que $\begin{bmatrix} Q \\ R \end{bmatrix}$ doit être de rang n. Cependant, la matrice D dans* (2.33) *joue un grand rôle lorsque nous nous appliquons au problème de programmation quadratique, et le fait qu'elle soit non singulière sous les hypothèses énoncées ci-dessus est très utile*

□

2.4 Les conditions nécessaires et suffisantes d'optimalités pour le problème de programmation quadratique

Dans le but d'énoncer des algorithmes, on va restreindre le problème à l'une des deux formes canoniques de problème de programmation quadratique (2.1) :

$$\text{Minimiser } 1/2\langle z, Qz \rangle + \langle d, z \rangle, \qquad \text{sous les contraintes} \quad Rz = c \quad \text{et} \quad \alpha \leq z \leq \beta. \tag{2.39}$$

$$\text{Minimiser } 1/2\langle z, Qz \rangle + \langle d, z \rangle, \qquad \text{sous les contraintes} \quad Rz = c \quad \text{et} \quad z \geq 0. \tag{2.40}$$

Théorème 2.6 *Si $\hat{z}$ est une solution de recherche au problème* (2.39)*, alors $\hat{z}$ est une option avec la solution si et seulement s'il existe un vecteur $\psi = (\psi^1, ..., \psi^m) \in E^m$, telle que pour $i = 1, 2, ..., n$,*

$$\begin{aligned} \langle r_i, \psi \rangle - \langle q_i, \hat{z} \rangle - d^i &= 0 \qquad si \quad \alpha^i < \hat{z}^i < \beta^i \\ \langle r_i, \psi \rangle - \langle q_i, \hat{z} \rangle - d^i &\leq 0 \qquad si \quad \hat{z}^i = \alpha^i \\ \langle r_i, \psi \rangle - \langle q_i, \hat{z} \rangle - d^i &\geq 0 \qquad si \quad \hat{z}^i = \beta^i, \end{aligned} \tag{2.41}$$

où pour $i = 1, 2, ..., n$, r_i est la i^{eme} colonne de R, q_i est la i^{eme} colonne de Q, et d^i est la i^{eme} composante de d.

2.5 Applications aux problèmes de contrôle illimités

Nous allons maintenant considérer plusieurs cas particuliers du problème de contrôle quadratique (2.1) qui n'ont pas de contraintes d'inégalité.

Ces problèmes n'ont pas été choisis à cause de leur grande signification pratique, mais parce qu'ils sont parmi les rares qui peuvent avoir des solutions sous forme fermée. Cela nous permet de comparer et d'évaluer diverses méthodes d'attaque.
Les notions qui seront traitées dans cette section sont également pertinente pour les formes les plus générales du problème de contrôle quadratique.

Comme premier exemple, nous considérerons le cas particulier suivant :

2.5.1 Le problème d'énergie minimale : forme de limite à deux points

Considérons un système dynamique décrit par le système linéaire :

$$x_{i+1} - x_i = Ax_i + Bu_i, \qquad i = 0, 1, ..., k-1; \tag{2.42}$$

Où $x_i \in E^n$ est l'état du système à l'instant $i, u_i \in E^m$ est l'entrée du système à l'instant i, A est une matrice $n \times n$, et B est une matrice $n \times m$.

Cherchons une séquence de contrôle $\hat{u} = (\hat{u}_0, \hat{u}_1, ..., \hat{u}_{k-1})$ qui transfère le système (2.42) de l'état initial $\hat{x}_0 = c_0$ à l'état terminal $\hat{x}_k = c_k$ et minimiser

$$f(u) = \int_{i=0}^{k-1} \langle u_i, u_i \rangle = \int_{i=0}^{k-1} \|u_i\|^2 \tag{2.43}$$

Hypothèse 2.1 $k \geq n$

Hypothèse 2.2 *Le système* (2.42) *est entièrement contrôlable ; c'est-à-dire que la matrice dimensionnelle* $(n \times nm)$ $[(B : (I+A)B \vdots \cdots \vdots (I+A)^{n-1}B)]$ *est de rang n.*

Résolution du problème 2.5.1 par une transcription directe :
Évidemment, le problème (2.5.1) est un cas particulier du problème de contrôle quadratique, et on peut donc le réduire à un problème de programmation quadratique sous la forme :

$$\text{Minimiser} \quad 1/2\langle z, Q_z \rangle \tag{2.44}$$

sous la contrainte
$Rz = c$;
Où $z = (x_0, x_1, ..., x_k, u_0, u_1, \cdots, u_{k1})$

$$Q = \begin{bmatrix} 0 & 0 \\ 0 & 2I \end{bmatrix} \tag{2.45}$$

est une matrice $[(k+1) + n + km] \times [(k+1)n + km]$ avec I une matrice identité $km \times km$.

$$R = \begin{bmatrix} -(I+A) & I & 0 & ... & 0 & -B & 0 & ... & ... & 0 \\ 0 & -(I+A) & I & 0 & 0 & 0 & -B & 0 & ... & 0 \\ ... & ... & ... & ... & ... & ... & ... & ... & ... & ... \\ 0 & ... & 0 & -(I+A) & I & 0 & 0 & ... & 0 & -B \\ I & 0 & ... & ... & 0 & 0 & ... & ... & ... & 0 \\ 0 & ... & ... & ... & I & 0 & ... & ... & ... & 0 \end{bmatrix} \tag{2.46}$$

est une matrice $(k+2)n \times [(k+1)n + km]$ et

$$c = (0, 0, ..., c_0, c_k) \in E^{(k+2)n} \tag{2.47}$$

Notez qu'il n'y a pas de terme linéaire dans la fonction de coût du problème (2.44), et donc une solution optimale à ce problème existe chaque fois qu'une solution réalisable existe.
Puisqu'il n'y a pas de contraintes dans (2.44) de la forme $\alpha_i \leq z_i \leq \beta_i$, une solution réalisable existe si R est de rang $(k+2)n$.

Lemme 2.1 *Le problème* (2.44) *possède une solution optimale unique.*

Démonstration. Selon le théorème (2.4), il suffit de montrer que $\pi(Q) \cap \pi(R) = \{0\}$.

Clairement, l'espace nul de Q est l'ensemble des vecteurs $\tilde{z}$ de la forme $\tilde{z} = (x_0, x_1, ..., x_k, 0, ..., 0)$. Nous devons donc examiner le système :

$$R\tilde{z} = \begin{bmatrix} -(I+A) & I & 0 & ... & ... & 0 \\ 0 & -(I+A) & I & 0 & ... & 0 \\ ... & ... & ... & ... & ... & ... \\ 0 & ... & ... & ... & -(I+A) & I \\ I & 0 & ... & ... & ... & 0 \\ 0 & ... & ... & ... & 0 & I \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ . \\ . \\ . \\ x_k \end{bmatrix} = 0 \tag{2.48}$$

En posons :

$$x_0 = 0 \tag{2.49}$$

Nous pouvons commencer par le bloc de lignes supérieur et déterminer successivement que

$$\begin{aligned} x_1 &= (I+A)x_0 = 0 \\ x_2 &= (I+A)x_1 = 0 \\ &.................... \\ x_k &= (I+A)x_{k-1} = 0 \end{aligned} \tag{2.50}$$

Par conséquent, $\pi(Q) \cap \pi(R) = \{0\}$, ce qui est suffisant pour garantir que (2.44) a une solution optimale unique. □

Nous appliquons maintenant le théorème (2.6) pour obtenir un système d'équations à partir duquel la solution optimale $\hat{z}$ du problème (2.44) peut être calculée.

Puisque pour ce cas $\alpha^i = -\infty$ et $\beta^i = +\infty$ pour $i = 1, 2, ..., (k+1)n + km$, on obtient alors le vecteur $\tilde{z}$ satisfaisant

$$R\tilde{z} = c \tag{2.51}$$

qui est optimal si et seulement s'il existe un vecteur $\psi \in E^{(k+1)n+km}$ tel que

$$\langle r_i, \psi \rangle - \langle q_i, \hat{z} \rangle = 0, \quad i = 1, 2, ..., (k+1)n + km \tag{2.52}$$

On peut écrire (2.52) sous forme matricielle :

$$R^T\psi - Q\hat{z} = 0. \tag{2.53}$$

En combinant (2.51) et (2.53) , on voit que la solution $(\hat{z}, \psi)$ de l'équation matricielle

$$\begin{bmatrix} -Q & R^T \\ R & 0 \end{bmatrix} \begin{bmatrix} \hat{z} \\ \psi \end{bmatrix} = \begin{bmatrix} 0 \\ c \end{bmatrix}, \tag{2.54}$$

donnera la solution optimale souhaitée $\hat{z}$.

On devrait maintenant comprendre l'intérêt d'établir le rang de R et l'unicité de la solution optimale $\hat{z}$ du problème (2.44), car il résulte immédiatement du théorème (2.5) que la matrice du premier membre du système (2.54) est non singuliere.
On obtient donc

$$\begin{bmatrix} \hat{z} \\ \psi \end{bmatrix} = \begin{bmatrix} -Q & R^T \\ R & 0 \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ c \end{bmatrix} \tag{2.55}$$

Le calcul de $\hat{z}$ nécessite l'inversion d'une matrice dont les dimensions peuvent être assez grandes même pour un problème relativement simple.

Cependant, cette matrice contient de nombreux zéros, ce qui aide à l'inversion numérique. L'exemple suivant devrait donner une idée du degré de difficulté impliqué dans la résolution du problème par cette méthode.

Exemple 2.1 *Soit $n = 2, m = 1$ et $k = 4$, et soit le système (2.42) de la forme spécifique*

$$x_{i+1} - x_i = \begin{bmatrix} -1 & 1 \\ -2 & -2 \end{bmatrix} x_i + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u_i \tag{2.56}$$

L'état initial est $c_0 = (2, 2)$, et l'état terminal $c_4 = (0, 0)$. On vérifie facilement que (2.58) est un système complètement contrôlable.

Dans ce cas R est la matrice dimensionnelle (12×14)

(2.57)

$$R = \begin{bmatrix}
0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
2 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\
0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 2 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 2 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & -1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 2 & 1 & 0 & 1 & 0 & 0 & 0 & -1 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0
\end{bmatrix}$$

La matrice

$$\begin{bmatrix} -Q & R^T \\ R & 0 \end{bmatrix} \tag{2.58}$$

est de dimension 26×26 et nécessiterait un espace si grand pour l'écrire.
On vérifier que la séquence de contrôle optimale est

$$\hat{u} = (-14/23, 10/23, 22/23, 28/23). \tag{2.59}$$

Il ressort clairement de cet exemple que même les problèmes simples deviennent assez lourds avec cette approche.
L'une des raisons est qu'il n'a pas été tenu compte de la nature particulière des contraintes d'égalité.

Nous avons déjà souligné dans la remarque (2.2) qu'il peut parfois être plus pratique d'utiliser une approche alternative pour convertir un problème de contrôle en un problème de base.
Nous allons maintenant voir quelles sont les simplifications qui peuvent être obtenues en tenant compte de la structure de l'équation du système.

Résolution du problème (2.5.1) **par transcription alternative** :

En résolvant le problème (2.42) pour x_k, avec $x_0 = c_0$, on obtient

$$x_k = (I + A)^k c_0 + \sum_{i=0}^{k-1} (I + A)^{k-i-1} B u_i, \tag{2.60}$$

et le problème (2.5.1) est maintenant considéré comme équivalent au problème :

$$\text{Minimiser} \quad 1/2\langle u, Qu \rangle \tag{2.61}$$

sous la contrainte

$$Ru = c,$$

Où $Q = 2I$, I étant la matrice d'identité $km \times km$

$$R = [(I + A)^{k-1}B|(I + A)^{k-2}B|...|(I + A)B|B], \tag{2.62}$$

Et

$$c = c_k - (I + A)^k c_0 \tag{2.63}$$

Le problème (2.61) est clairement de la forme de (PQ).
Les remarques significatives issues de cette reformulation sont :

D'abord (puisque $(k \geq n)$, la matrice apparaissant dans l'hypothèse de contrôlabilité complète de l'hypothèse (2.2) est une sous-matrice de R, telle que donnée par (2.62).
Il s'ensuit immédiatement que R est de rang n. Ainsi une solution réalisable à ce problème existe toujours.
Deuxièmement, Q est une matrice définie positive, et donc la solution optimale existe et unique.
Évidemment, les énoncés sont maintenant beaucoup plus faciles à établir qu'avant la reformulation.
formnula Considérons maintenant la solution de ce problème. On peut facilement vérifier à partir du théorème (2.6) qu'une condition nécessaire et suffisante pour qu'une séquence de contrôle $\hat{u}$ soit optimale est qu'il existe un vecteur ψ tel que

$$\begin{bmatrix} -Q & R^T \\ R & 0 \end{bmatrix} \begin{bmatrix} \hat{u} \\ \psi \end{bmatrix} = \begin{bmatrix} 0 \\ c \end{bmatrix} \tag{2.64}$$

Comme dans (2.54), la matrice du système (2.64) est non singulière, et donc elle peut être inversée pour trouver $\hat{u}$.
Notez, cependant, la dimension de la matrice est $(km + n) \times (km + n)$, ce qui est nettement inférieur à $[km + (2k + 3)n] \times [km + (2k + 3)n]$, la dimension de la matrice du système (2.54).
Dans l'exemple précédent cela est équivaut à 6×6 par rapport à 26×26 dans (**??**).

Cependant, nous n'avons pas trop utilisé la structure du problème. Puisque $Q = 2I$, nous pouvons résoudre le système (2.64), ce qui donne

$$\hat{u} = 1/2R^T\psi. \tag{2.65}$$

Cette expression, peut être substitué dans le deuxième ensemble d'équations dans (2.64) pour obtenir :

$$R\hat{u} = 1/2RR^T\psi = c. \tag{2.66}$$

Puisque R est de rang complet, la matrice RR^T est définie positive et donc non singulière,donc

$$\psi = 2(RR^T)^{-1}c, \tag{2.67}$$

et la substitution dans (2.65) donne

$$\hat{u} = R^T(RR^T)^{-1}c \tag{2.68}$$

Ainsi à partir de la structure de ce problème, il suffit d'inverser la matrice $n \times n$ RR^T pour obtenir la solution optimale $\hat{u}$.
C'est en effet, une amélioration significative par rapport à l'approche précédente, puisque la dimension de RR^T est indépendante de k.

Pour illustrer la différence, résolvons à nouveau le problème de l'exemple précédent.

Exemple 2.2 *Soit $n = 2, m = 1$ et $k = 4$ et que le système* (2.42) *soit de la forme spécifique*

$$x_{i+1} - x_i = \begin{bmatrix} -1 & 1 \\ -2 & -2 \end{bmatrix} x_i \begin{bmatrix} 0 \\ 1 \end{bmatrix} u_i,$$

avec $c_0 = (2, 2)$ et $c_4 = (0, 0)$

En évaluant R et c, on obtient

$$R = \begin{bmatrix} 7 & 3 & 1 & 0 \\ 15 & 7 & 3 & 1 \end{bmatrix} \qquad c = (-2, -2) \tag{2.69}$$

par conséquent,

$$RR^T = \begin{bmatrix} 59 & 129 \\ 129 & 284 \end{bmatrix} \tag{2.70}$$

et

$$\hat{u} = (-14/23, 10/23, 22/23, 28/23) \tag{2.71}$$

Ainsi, bien que la transcription alternative d'un problème de contrôle optimal sous la forme d'un problème de base ne soit pas toujours pratique pour les problèmes non linéaires, elle offre des avantages significatifs pour de nombreux problèmes de dynamique linéaire.
Avant de quitter le problème de l'énergie minimale, cependant, nous devons envisager une autre approche de sa solution.

L'approche de contrôle du problème d'énergie minimale

Par « approche de contrôle » du problème minimum d'énergie (2.5.1), nous entendrons l'examen du problème au moyen de la nécessité Pour tout $x \neq 0$ la forme quadratique $\langle x, RR^T x\rangle = \|R^T x\|^2 > 0$ puisque $R^T x \neq 0$, par l'hypothèse du rang.

2.5.2 Le problème du coût minimum quadratique

Considérons le système dynamique régi par le système

$$x_{i+1} - x_i = Ax_i + Bu_i, \qquad i = 0, 1, \cdots, k-1, \tag{2.72}$$

où la notation est la même que précédemment. L'état initial x_0 doit appartenir à une variété linéaire décrite par :

$$G_0 x_0 = c_0, \tag{2.73}$$

où G_0 est une matrice $l_0 \times n$ de rang l_0 et $c_0 \in E^{l_0}$ est un vecteur fixe, et l'état terminal x_k doit appartenir à la variété linéaire suivante :

$$G_k x_k = c_k, \tag{2.74}$$

où G_k est une matrice $l_k \times n$ de rang l_k et $c_k \in E^{l_k}$ est un vecteur fixe.
Cherchons une séquence de contrôle $\hat{u} = (\hat{u}_0, \hat{u}_1, ..., \hat{u}_k)$ et une trajectoire correspondante $\hat{x} = (\hat{x}_0, \hat{x}_1, ..., \hat{x}_k)$ qui minimise

$$1/2\langle x, Q_x, x\rangle + 1/2\langle u, Q_u, u\rangle \tag{2.75}$$

sous les contraintes (2.72) à (2.74) , où Q_x et Q_u sont deux matrices symétriques définies semi-positives.

Comme dans le problème d'énergie minimale (2.5.1) , nous ferons les hypothèses suivantes :

Hypothèse 2.3 $k \geq n$

Hypothèse 2.4 *Le système* (2.72) *est entièrement contrôlable .*

Évidemment, le problème (2.5.2) est un cas particulier du problème de contrôle quadratique (PQ). La matrice Q dans (2.3) prend dans ce cas la forme

$$Q = \begin{bmatrix} Q_x & 0 \\ 0 & Q_u \end{bmatrix} \tag{2.76}$$

La conversion du problème (2.5.2) en un problème de programmation quadratique devrait maintenant être simple. Ainsi la fonction de coût devient

$$f(z) = 1/2\langle z, Q_z \rangle, \tag{2.77}$$

où $z = (x_0, x_1, ..., u_0, u_1, ..., u_{k-1})$ et Q est donné en (2.76).
Les contraintes d'égalité deviennent

$$Rz = c, \tag{2.78}$$

$$R = \begin{bmatrix} -(I+A) & I & 0 & \cdots & \cdots & 0 \\ 0 & -(I+A) & I & 0 & \cdots & 0 \\ 0 & \cdots & \cdots & \cdots & -(I+A) & I \\ G_0 & 0 & \cdots & \cdots & \cdots & 0 \\ 0 & \cdots & \cdots & \cdots & 0 & G_k \end{bmatrix} \tag{2.79}$$

Et $c = (0, 0, ..., 0, c_0, c_k)$

Puisqu'il n'y a pas de terme linéaire dans la fonction de coût, il découle immédiatement du corollaire (2.1) qu'une solution optimale à ce problème existe chaque fois qu'une solution réalisable existe.

De plus, les hypothèses (2.3) et (2.4), ainsi que le fait qu'il n'y a pas de contraintes d'inégalité, garantie qu'une solution réalisable existe.
Donc une solution optimale à ce problème existe toujours.

Au sujet de l'unicité des solutions optimales du problème (2.5.2) nous pouvons vérifier les conditions du théorème (2.4), c'est-à-dire que nous devons montrer que $\Re(Q) \cap \Re(R) = \{0\}$ Cela peut être fait en vérifiant la matrice

$$\begin{bmatrix} Q \\ \cdots \\ R \end{bmatrix} \tag{2.80}$$

Pour voir si elle a le rang complet. Alternativement, puisque R est de rang complet [en raison des hypothèses (2.3) et (2.4), ainsi que l'hypothèse de rang complet sur G_0 et G_k], nous pouvons, en conséquence du théorème Canon et al. [1970] , déterminer si la matrice

$$\begin{bmatrix} Q & R^T \\ R & 0 \end{bmatrix}$$

est non singulière.

À cause de la forme spéciale de Q [voir (2.76)], le rang de (2.80) devient plus facile à vérifier lorsque Q_x ou Q_u est défini positive. Par exemple, si Q_u est définie positive, alors le rang de la matrice dans (2.80) est le même que le rang de la matrice

$$\begin{bmatrix} & & Q_x & & \\ -(I+A) & I & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & \cdots & -(I+A) & I \\ G_0 & \cdots & \cdots & \cdots & 0 \\ 0 & \cdots & \cdots & \cdots & G_k \end{bmatrix} \tag{2.81}$$

Solution de (2.5.2) **par transcription directe**

La solution du problème de programmation quadratique formulé ci-dessus est similaire dans sa procédure à la solution correspondante du problème d'énergie minimale (2.5.1) .
Il suffit de dire que nous arrivons finalement au point où nous devons résoudre l'équation

$$\begin{bmatrix} -Q & R^T \\ R & 0 \end{bmatrix} \begin{bmatrix} \hat{z} \\ \psi \end{bmatrix} = \begin{bmatrix} 0 \\ c \end{bmatrix} \tag{2.82}$$

Par conséquent, chaque fois que la solution optimale du problème (2.5.2) est unique, on peut résoudre (2.82) en inversant la matrice du membre de gauche.

Il convient de noter que (2.82) peut être résolu pour un $\hat{z}$ optimal même lorsque la matrice apparaissant à gauche est singulière.

L'existence d'une solution optimale au problème (2.5.2), ainsi que le fait que (2.82) exprime une condition nécessaire et suffisante d'optimalité,garantissent que (2.82) est également une solution optimale à (2.5.2)

Solution de (2.5.2) **par transcription alternative**

Dans la transcription alternative du problème (2.5.2) en un problème de programmation quadratique, on élimine tous les vecteurs x_i autres que x_0 en le résolvant au moyen de (2.72).

Dans le problème d'énergie minimale (2.5.1), cela était très facile à faire car seule la contrainte terminale impliquait l'un de ces x_i.
Maintenant, cependant que les x_i apparaissent également dans la fonction de coût, ce qui rend la conversion en un problème de programmation quadratique beaucoup plus complexe.

Comme précédemment, nous résolvons (2.72) pour les vecteurs x_i, on obtient :

$$x_i = (I+A)^i x_0 + \sum_{j=0}^{i-1} (I+A)^{i-j-1} B u_j, \qquad i = 1, 2, ..., k. \tag{2.83}$$

Si on défini le vecteur z dans E^{km+n} comme étant

$$z = (x_0, u_0, u_1, ..., u_{k-1}), \tag{2.84}$$

Alors l'expression (2.83) devient

$$x_i = M_i z \qquad i = 1, 2, ..., k, \tag{2.85}$$

Où M_i est une matrice $n \times (km + n)$

$$M_i = [(I+A)^i \vdots (I+A)^{i-1} B \vdots \cdots \vdots (I+A)B \vdots B \vdots 0 \cdots \vdots 0], \qquad i = 1, 2, ..., k. \tag{2.86}$$

Donc la trajectoire $x = (x_0, x_1, ..., x_k)$ est donnée par

$$x = Mz, \tag{2.87}$$

où M est une matrice $(k+1)n \times (km + n)$

$$M = \begin{bmatrix} I & 0 & \cdots & 0 \\ & & M_1 & \\ & & M_2 & \\ \cdots & \cdots & \cdots & \cdots \\ & & M_k & \end{bmatrix} = \begin{bmatrix} I & 0 & \cdots & \cdots & \cdots & 0 \\ I+A & B & 0 & \cdots & \cdots & 0 \\ (I+A)^2 & (I+A)B & B & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ (I+A)^k & (I+A)^{k-1}B & \cdots & \cdots & \cdots & B \end{bmatrix} \tag{2.88}$$

La fonction de coût (2.75) devient

$$\langle x, Q_x x \rangle + \langle u, Q_u u \rangle = \langle z, P_z \rangle, \tag{2.89}$$

où

$$P = M^T Q_x M + N, \tag{2.90}$$

et

$$N = \begin{bmatrix} 0 & 0 \\ 0 & Q_u . \end{bmatrix}$$

On vérifie facilement que P est une matrice symétrique définie semi-positive .

Les contraintes aux limites peuvent maintenant être reformulées en termes de z. La variété de contraintes initiale (2.73) devient :

$$G_0 M_0 z = c_0, \tag{2.91}$$

où M_0 est une matrice $n \times (km + n)$

$$M_0 = [I \vdots 0 \vdots \cdots \vdots 0] \tag{2.92}$$

La variété de contraintes terminales (2.74), conjointement avec (2.85), donne

$$G_k M_k z = c_k, \tag{2.93}$$

où M est telle que définie en (2.86).

La matrice M_0 est clairement de rang n, et les hypothèses (2.3) et (2.4) garantissent que M_k a le rang n.

On peut vérifier que $G_0 M_0$ et $G_k M_k$ ont respectivement par conséquent le rang l_0 et l_k . À partir de ce point, nous procédons à la minimisation de (2.89) sous les contraintes d'égalité (2.91) et (2.93).
Il devrait être clair que nous arriverons à nouveau à l'expression

$$\begin{bmatrix} -P & R^T \\ R & 0 \end{bmatrix} \begin{bmatrix} \hat{z} \\ \psi \end{bmatrix} = \begin{bmatrix} 0 \\ c \end{bmatrix}, \tag{2.94}$$

où $\hat{z}$ est la solution optimale, ψ est le multiplicateur correspondant, R est la matrice $(l_0 + l_k) \times (km + n)$ donné par :

$$R = \begin{bmatrix} G_0 M_0 \\ \cdots\cdots \\ G_k M_k \end{bmatrix}, \tag{2.95}$$

et $c = (c_0, c_k)$. Notez qu'il n'est pas immédiatement évident qu'une solution unique au problème d'optimisation (2.5.2) implique la non singularité de la matrice dans (2.94), puisque R peut ne pas être de rang complet.

2.6 Problèmes de contrôle quadratique avec contraintes d'inégalités

Il a été montré que la forme la plus générale du problème de contrôle quadratique peut être réduite à un problème de programmation quadratique.

La suite sera consacrée à la présentation d'un algorithme de résolution du problème de programmation quadratique (sous des hypothèses convenables, qui seront énoncées plus loin) dans le cas où des contraintes d'inégalité sont présentes.

Considérons dans un premier temps la nature des problèmes de programmation quadratique résultant de différents cas particuliers du problème de contrôle quadratique.

Les deux classes de problèmes considérés dans la section (2.5), avec quelques contraintes d'inégalité supplémentaires, nous seront bien utiles.

2.6.1 Problème avec des contraintes d'énergie minimale .

Considérons un système dynamique décrit par l'équation suivante :

$$x_{i+1} - x_i = Ax_i + Bu_i, \qquad i = 0, 1, ..., k-1, \tag{2.96}$$

avec les contraintes initiales et terminales

$$x_0 = c_0 \quad \text{et} \quad x_k = c_k, \tag{2.97}$$

et avec les contraintes de contrôle

$$|u_i^j| \leq 1; \qquad j = 1, 2, ..., m; \quad i = 0, 1, ..., k-1, \tag{2.98}$$

où $u_i = (u_i^1, u_i^2, ..., u_i^m); \quad i = 0, 1, ..., k-1$
On cherche une suite de contrôle $\hat{u} = (\hat{u}_0, \hat{u}_1, ..., \hat{u}_{k-1})$ qui minimise

$$\sum_{i=0}^{k-1} \langle u_i, u_i \rangle \tag{2.99}$$

Sous réserve de (2.96) ,(2.97) et (2.99)

Hypothèse 2.5 $k \geq n$.

Hypothèse 2.6 *Le système* (2.96) *est entièrement contrôlable.*

La transformation du problème (2.6.1) en un problème de programmation quadratique peut à nouveau être accomplie en éliminant d'abord les x_i pour $i = 1, 2, ..., k$ au moyen de l'équation aux différences. Le problème de programmation quadratique équivalent résultant est le suivant :

$$\text{Minimiser } 1/2\langle u, Qu \rangle \tag{2.100}$$

sous les contraintes

$$Ru = c \tag{2.101}$$

$$-1 \leq u^j \leq 1 \qquad j = 1, 2, ..., km, \tag{2.102}$$

où $Q = 2I$,

$$R = [(I+A)^{k-1}B \vdots (I+A)^{k-2}B \vdots \cdots \vdots (I+A)B \vdots B] \tag{2.103}$$

et

$$c = c_k - (I+A)^k c_0. \tag{2.104}$$

Notons que (2.100) ne diffère de (2.44) que par les contraintes d'inégalité (2.102).Il y a deux points à propos de ce problème qui sont significatifs en ce qui concerne les algorithmes de résolution des problèmes de programmation quadratique.
Tout d'abord, la matrice R apparaissant en (2.101) est de rang n, comme on le déduit facilement de (2.6). C'est une hypothèse qui est requise par les algorithmes discutés plus tard.Considérons maintenant ce qui arrive au problème de coût quadratique minimum (2.5.2) lorsque nous ajoutons des bornes pour les variables de contrôle.

2.6.2 Le problème du coût minimum à contraintes quadratiques.

Considérons un système dynamique régi par l'équation suivante :

$$x_{i+1} - x_i = Ax_i + Bu_i \qquad i = 0, 1, ..., k-1, \tag{2.105}$$

avec les contraintes initiales et terminales

$$G_0 x_0 = c_0, \qquad G_k x_k = c_k, \tag{2.106}$$

où G_0 est une matrice $l_0 \times n$ de rang l_0 et G_k est une matrice $l_k \times n$ de rang l_k ; avec les contraintes de contrôle

$$|u_i^j| \leq 1, \qquad j = 1, 2, ..., m, \quad i = 0, 1, ..., k-1, \tag{2.107}$$

où $u_i = (u_i^1, u_i^2, ..., u_i^m)$, et avec la fonction de coût

$$\langle x, Q_x x \rangle + \langle u, Q_u u \rangle, \tag{2.108}$$

où Q_x et Q_u sont deux matrices symétriques défini semi-positives,
$x = (x_0, x_1, ..., x_k)$ et $u = (u_0, u_1, ..., u_{k-1})$.
Cherchons une séquence de contrôle u qui minimise (2.108) sous réserve de (2.105), (2.106) et (2.107).

Hypothèse 2.7 $k \geq n$.

Hypothèse 2.8 *Le système* (2.105) *est complètement contrôlable.*

Nous convertissons ce problème en un problème de programmation quadratique par la méthode de transcription alternative décrite à la section (2.5), en commençant par l'équation (2.83) .On vérifié facilement que si l'on définit

$$z \triangleq (x_0, u_1, ..., u_{k-1}), \tag{2.109}$$

Alors le problème de programmation quadratique résultant est :

$$\text{Minimiser } \langle z, Pz \rangle \tag{2.110}$$

sous les contraintes

$$Rz = c \tag{2.111}$$

$$|z^i| \leq 1, \qquad i = n+1, n+2, ..., n+km, \tag{2.112}$$

où P ,R et c sont défini dans (2.90) et (2.95)

Notez qu'il n'est plus vrai que toutes les variables soient bornées, même si les contrôles sont bornés.

Plus précisément, les n premières composantes de z correspondent à x_0 sont illimités. En général, donc, tous les problèmes de programmation quadratique qui découlent d'un problème de contrôle n'ont pas toutes les variables bornées.

L'algorithme du simplexe est un algorithme de nature combinatoire, et il détermine soit une solution optimale, soit il détermine que la solution optimale n'existe pas, qui peut être utiliser pour ce problème.

2.7 Conditions d'optimalité pour le problème de programmation quadratique canonique

Les algorithmes dont nous allons parlé s'appliquent à des problèmes de programmation quadratique de la forme suivante.

2.7.1 Problème de programmation quadratique canonique(PQC)

Étant donné une matrice $m \times n (m \leq n)$ de rang complet R, une matrice Q symétrique définie semi-positive $n \times n$, et les vecteurs $c \in E^m$ et $d \in E^n$. Trouvons un $\hat{z} \in E^n$ qui minimise

$$1/2\langle z, Qz \rangle + \langle z, d \rangle \tag{2.113}$$

sous les contraintes

$$Rz = c, \qquad z \geq 0. \tag{2.114}$$

Théorème 2.7 *Un vecteur $z \in E^n$ est une solution optimale à (PQC) si et seulement s'il existe des multiplicateurs $\psi = (\psi^1, \psi^2, ..., \psi^m)$ et $\xi = (\xi^1, \xi^2, .., \xi^n)$ tels que*

$$Qz + R^T - \xi + d = 0 \qquad Rz - c = 0 \tag{2.115}$$

$$z \geq 0 \qquad \xi \geq 0 \qquad \langle z, \xi \rangle = 0.$$

Le système d'équations et d'inégalités ci-dessus peut être écrit sous une forme plus symétrique en décomposant le vecteur ψ en une partie positive, que nous notons par ψ_+, et une partie négative, que nous notons par $-\psi_-$, soit

$$\psi = \psi_+ - \psi_- \tag{2.116}$$

Avec la restriction que

$$\psi_+ \geq 0 \qquad \psi_- \geq 0, \tag{2.117}$$

$$\psi_+^i \psi_-^i = 0 \qquad i = 1, 2, ..., m. \tag{2.118}$$

Notez que (2.118) permet à ψ_+^i ou ψ_-^i d'être strictement positif, mais pas les deux en même temps. À cause de (2.117),(2.118) l'expression peut être écrit de manière plus compacte comme

$$\langle \psi_+, \psi_- \rangle = 0. \tag{2.119}$$

la substitution de 2.116 ,2.117,et2.119 par 2.115 conduit au système d'équations et d'inégalités suivant :

$$\begin{gathered} Qz - R^T - \xi + R^T\psi_+ = -d \qquad Rz = c \\ z \geq 0 \qquad \psi_- \geq 0 \qquad \xi \geq 0 \qquad \psi_+ \geq 0 \\ \langle z, \xi \rangle = \langle \psi_+, \psi_- \rangle = 0. \end{gathered} \tag{2.120}$$

2.8 Le problème de minimisation dérivée

L'effet du théorème (2.7) a été de réduire la résolution du problème de programmation quadratique canonique (2.7.1) à celui de la résolution du système (2.120).
À son tour, une solution à ce système peut être obtenue en résolvant le problème de minimisation dérivée énoncé ci-dessous.
On peut noter à ce stade qu'il est courant en programmation mathématique de trouver des solutions à des systèmes d'équations et d'inégalités en résolvant un problème d'optimisation convexe pour lequel il existe des algorithmes efficaces.

A titre d'exemple, nous verrons plus loin qu'une légère modification de l'algorithme du simplexe permet de résoudre le problème que nous allons définir, et donc d'obtenir une solution au PQC (2.7.1)

2.8.1 Le problème dérivé

$$\text{Minimiser} \quad \langle l, y \rangle$$

sous les contraintes

$$Ax = g, \qquad x \geq 0; \tag{2.121}$$

$$\langle v, w \rangle = 0, \tag{2.122}$$

Où

$$x = (z, \psi_-, \xi, \psi_+, y) = (v, w, y) \in E^{2(n+m)+n} \tag{2.123}$$
$$v = (z, \psi_-) \in E^{n+m}, \qquad w = (\xi, \psi_+) \in E^{n+m}$$

$$A = \begin{bmatrix} Q & -R^T & -I & R^T & K \\ R & 0 & 0 & 0 & 0 \end{bmatrix}, \quad \text{de dimension} \quad (n+m) \times [2(n+m)+n] \tag{2.124}$$

$$g = (-d, c) \in E^{n+m}, \qquad l = (1, 1, ..., 1) \in E^n$$

Les matrices Q, R, d et b sont telles que définies en (2.7.1) ; K une matrice $n \times n$.

Dans la suite de ce chapitre, nous utiliserons les termes solution de base, solution de base non dégénérée, etc, qui sont tous définis et discutés dans les sections (2.5) et (2.6).

Définition 2.1 *Un vecteur x est dit solution de base réalisable au problème dérivé* (2.8.1) *si x est une base de* (2.121) *et, en plus, satisfait* (2.122).
Une solution de base réalisable qui est une solution de base non dégénérée de (2.121) *sera appelée une solution de base non dégénérée réalisable.*

Remarque 2.5 *Clairement, toute solution de base réalisable x au problème* (2.8.1) *est une solution réalisable ; C'est,*

$$x \in \Omega\prime = \Omega \cap \{x : \langle v, w \rangle = 0\},$$

Où

$$\Omega = \{x : Ax = g, x \geq 0\}. \tag{2.125}$$

Lemme 2.2 *Supposons que le système d'équations et d'inégalités* (2.120) *ait une solution.*
Alors $\hat{x} = (\hat{v}, \hat{w}, \hat{y})$ est une solution optimale pour le problème dérivé (2.8.1) *si et seulement si $(\hat{v}, \hat{w})$ satisfait le système* (2.120).

Démonstration. Pour tout $x \in \Omega\prime$, $\langle l, y \rangle \geq 0$ si et seulement si $y = 0$.
le résultat recherché suit alors . □

3 Algorithme de résolution

3.1 Algorithme du simplexe

Considérons maintenant un algorithme dû à Wolfe [1959] pour résoudre le problème de programmation quadratique canonique (2.7.1).
Le succés de cet algorithme est incontestablement dû à la facilité avec laquelle il converge (de manière démontrable) que si les matrices apparaissant dans l'énoncé du PQC (2.7.1) satisfont la condition supplémentaire suivante :

Hypothèse 3.1 $\langle d, \eta \rangle = 0$ *pour tout* $\eta \in E^n$ *satisfaisant* $Q\eta = 0$ *et* $R\eta = 0$
Plusieurs conditions suffisantes pour la satisfaction de l'hypothèse (3.1) *ont été donnés dans les sections* (2.2) *et* (2.3).
Rappelons que d'après le théorème (2.1), *c'est une condition suffisante pour que le PQC* (2.7.1) *ait une solution optimale.*

Initialisation de l'algorithme.
Nous devons d'abord obtenir une solution de base réalisable x_0 au problème dérivé (2.8.1), et ceci en deux étapes.
Nous utilisons l'algorithme du simplexe pour obtenir une solution de base réalisable au système
$Rz = c, \qquad z \geq 0.$

Soit z_0 la solution de base ainsi obtenue, et soit $\bar{J}(z_0) \subset 1, 2, ..., n$ l'ensemble des indices de base associé.
On peut maintenant définir les composantes restantes x_0, ainsi que la matrice diagonale $n \times n$ $K = [k_{ii}]$ apparaissant en (2.124), comme suit :

Pour $i = 1, 2, \cdots, n, \quad \xi_0^i = 0$

Pour $i = 1, 2, \cdots, m, \quad \psi_{-0}^i = \psi_{+0}^i = 0$

Pour $i = 1, 2, \cdots, n$

$$k_{ii} = \begin{cases} \delta^i \triangleq (-d - Q - z_0 + R^T\psi_{-0} + \xi_0 - R^T\psi_{+0}) & , si \quad \delta^i \neq 0 \\ 1 & , si \quad \delta^i = 0 \end{cases}$$

$$y_0^i = \begin{cases} 1 & , si \quad \delta^i \neq 0 \\ 0 & , si \quad \delta^i = 0 \end{cases}$$

3.1.1 L'algorithme du simplexe pour la programmation quadratique

Supposons que x_N est une solution de base réalisable au problème dérivé (2.8.1) Soit $\bar{J}(x_N) \subset 1, 2, ..., 2(n+m)+n$ l'ensemble des indices de base associé avec x_N.

Étape 1. Utiliser la solution de base x_N dans l'algorithme du simplexe pour calculer une solution de base améliorée x_{N+1} au problème :

$$\text{Minimiser} \quad \langle l, y \rangle \tag{3.1}$$

sous les contraintes

$$Ax = g \qquad x \geq 0 \,,$$

Avec la condition secondaire suivante :

Si $j \in \bar{J}(x_N) \cap \{1, 2, \cdots, n+m\}$ n'admet pas $j+n+m$ dans l'ensemble des indices de base

$\bar{J}(x_{N+1})$ pour x_{N+1} et si $j \in \bar{J}(x_N) \cap \{n+m+1, ..., 2(n+m)\}$ n'admet pas

$$j-(n+m) \quad \text{dans l'ensemble d'indices de base} \quad \bar{J} = (x_{N+1}). \tag{3.2}$$

Si une solution de base réalisable x_{N+1} ne peut pas être construite conformément à (3.2), alors arrêtez ; sinon, passez à l'étape 2. Notez que x_{N+1} est une solution de base réalisable.

Étape 2. Si $\langle l, y_{N+1} \rangle = 0$. Alors arrêtez. Si $\langle l, y_{N+1} \rangle > 0$, définir $x_N = x_{N+1}$ et revenir a l'étape 1
Pour résumer, l'algorithme (3.1.1) peut atteindre critère d'arrêt avec l'une de deux manières
Arrêt 1. L'algorithme, sous la condition (3.1), ne donne pas une solution de base améliorée au problème 3.1.1.

Remarque 3.1 *Nous verrons plus loin que chaque fois que l'hypothèse* (3.1) *est vérifiée, la condition d'arrêt 1 coïncide avec la condition d'arrêt suivante.*

Arrêt 2. À une certaine itération $\langle l, y_N \rangle = 0$. Dans ce cas, x_N est une solution optimale au problème dérivé (2.8.1). Les n premières composantes de x_N autrement dit, z_N^i pour $i = 1, 2, \cdots, n$, sont une solution optimale au problème de programmation quadratique canonique.

Théorème 3.1 *Supposons que l'hypothèse* (3.1) *soit satisfaite. Soit $\hat{x} = (\hat{z}, \hat{\Psi}_-, \hat{\xi}, \hat{\Psi}_+, \hat{y})$ la dernière solution de base réalisable générée par l'algorithme* (3.1.1) *à laquelle la condition* (3.2) *est satisfaite. Alors $\hat{y} = 0$, et $\hat{z}$ est une solution optimale de PQC* (2.7.1).

3.2 Généralisation

Le but de cette section est d'indiquer comment l'algorithme peut être généralisé de sorte qu'il s'applique au problème de programmation quadratique canonique (2.7.1) indépendamment de l'hypothèse (3.1) soit satisfait ou non.

En outre, une solution initiale réalisable ne sera pas nécessaire.
La première étape vers notre objectif est de faire une observation trompeusement simple. Supposons qu'au lieu d'introduire n variables y^i pour $i = 1, 2, ..., n$, comme nous l'avons fait pour le problème dérivé (2.8.1) , nous introduisons une seule variable, que nous désignerons à nouveau par y.
Ainsi, à la place du problème dérivé d'origine, considérons le nouveau problème dérivé :

$$\text{Minimiser} \quad y \quad \text{sous les contraintes} \tag{3.3}$$

$$Ax = g, \qquad x \geq 0, \qquad \langle v, w \rangle = 0 \tag{3.4}$$

Où maintenant

$$A = \begin{bmatrix} Q & -R^T & -I & R^T & k_1 \\ R & 0 & 0 & 0 & k_2 \end{bmatrix}, \quad \text{de dimension} \quad (n+m) \times [2(n+m)+1] \tag{3.5}$$

de plus,

$$x = (z, \psi_-, \xi, \psi_+, y) = (v, x, y) \in E^{2(n+m)+1} \tag{3.6}$$

$$v = (z, \psi_-) \qquad w = (\xi, \psi_+) \qquad g = (-d, c).$$

Dans la formulation ci-dessus, k_1 est une matrice $n \times 1$ et k_2 est une matrice $m \times 1$. Si, pour le moment, on suppose que k_2 est identiquement nul, ensuite, la seule différence de formulation entre le problème dérivé (2.8.1) et le problème (3.3) est que nous avons réduit la matrice diagonale (2.124) en une matrice k_1 de dimension $n \times 1$.
Finalement, nous allons attribuez des valeurs spécifiques à k_1 et k_2 pour faciliter l'initialisation de l'algorithme.
Puisque la restriction $\langle v, w \rangle = 0$ dans (3.4) nous oblige à examiner les composantes de v et w par paires, nous constatons que la discussion est considérablement simplifié par la notation suivante :

Définition 3.1 *Étant donné une solution de base réalisable $x = (v, w, y)$ à* (3.4)*, avec un ensemble des indices de base associé $J(x)$, on dira que v^i pour $i \in \{1, 2, ..., n+m\}$ est une variable de base si $i \in \bar{J}(x)$ et que w^i pour $i \in \{1, 2, ..., n+m\}$ est une variable de base si $(i + n + m) \in \bar{J}(x)$.*

Il est clair que l'algorithme (3.3.1) *peut être appliqué au nouveau problème dérivé* (2.8.1) *sans modification. Cependant, une simplification importante se produit dans la procédure, qui, avec un nouvel ensemble de règles d'arrêt, entraîne une généralisation ou une extension importante de l'algorithme* (3.1.1)*.*
Supposons donc que x_N est une solution de base réalisable pour (3.4)*, avec ; $y_N > 0$, tel que*

$$v_N^i \quad et \quad w_N^i \quad pour \quad i = 1, 2, \cdots, n+m \quad ne\ sont\ pas\ les\ deux\ variables\ de\ base. \tag{3.7}$$

Autrement dit, si v_N^i est basique, alors w_N^i ne l'est pas, et vice versa. Notez qu'en raison de la condition (3.2), l'algorithme (3.1.1) génère une séquence de solutions de base réalisables pour le nouveau problème dérivé (2.8.1) qui satisfont la condition (3.7). Puisque, par hypothèse, $y_N > 0$ et x_N est une de solution de base réalisable, il y a exactement $n+m+1$ variables de base parmi les $2(n+m)$ variables de y_N et w_N. En outre, compte tenu de (3.1.1), il y a un seul indice $j \in \{1, 2, \ldots, n+m\}$ avec la propriété qui ni v_N^j ni w_N^j n'est une variable de base.
Maintenant, pour construire une solution de base réalisable amélioré x_{N+1} selon l'étape (1) de l'algorithme (3.1.1), nous utilisons l'algorithme du simplexe avec la condition secondaire (3.2)

Pour $i = 1, 2, \cdots, n+m$, si v_N^i est une variable de base, alors w_{N+1}^i peut ne pas être une variable de base, et si w_N^i est une variable de base, alors v_{N+1}^i peut ne pas être une variable de base.

Par conséquent, la construction de x_{N+1} nous oblige à faire soit v_{N+1}^j ou w_{N+1}^j est la variable de base [c'est-à-dire j ou $j+n+m$ sont les seuls éléments de $J(x_N)$ qui peuvent être transférés dans $\bar{J}(x_{N+1})$; il n'y a pas d'autres possibilités]. Nous pourrions déterminer laquelle de ces variables devraient être rendu de bases en suivant régulièrement les étapes 1 et 2 de l'algorithme du simplexe dans Canon et al. [1970]. Cependant, une petite réflexion montre que ces calculs ne sont pas nécessaires, car dans la séquence des solutions de base réalisables $x_0, x_1, ..., x_{N-1}, x_N$ que nous avons implicitement supposé avoir été construit par l'algorithme (3.1.1) , l'une des deux situations suivantes doivent se produire :

a. v_{N-1}^j était une variable de base.

b. w_{N-1}^j était une variable de base.

La conclusion devrait maintenant être évidente. Si $v_{N-1}^j(v_{N+1}^j)$ était une variable de base, puis du faite de $w_{N-1}^j(v_{N-})^j)$ une variable de base. Ainsi, nous voyons qu'en appliquant l'algorithme 3.1.1 au problème (3.3) nous pouvons éliminer les calculs impliqués dans les étapes 1 et 2 de l'algorithme du simplexe dans Canon et al. [1970].
Notez cependant que les conditions d'arrêt de l'algorithme (3.1.1) deviennent inopérantes dans ce processus. Ainsi, l'algorithme (3.1.1) a été conçu à l'origine avec l'idée de réduire la valeur d'une forme linéaire à chaque étape, la modification indiquée ci-dessus aboutit à une procédure purement combinatoire qui n'utilise aucunement les valeurs de cette forme linéaire. Notre description du nouvel algorithme : encore incomplet, puisque nous n'avons pas encore indiqué les conditions d'arrêt qui doivent remplacer celles de (3.1.1) .
Comme nous le verrons, c'est le changement des conditions d'arrêt qui augmentent le champ d'application du nouvel algorithme. Avant d'énoncer notre nouvel algorithme avec ses conditions d'arrêts, voyons comment l'initialiser.

Une méthode qui facilement vient à l'esprit est de placer une borne supérieure sur la variable y dont la valeur optimale devrait être zéro. Si nous prenons cela comme l, le problème (3.3) prend la forme suivante :

$$\text{Minimiser } y \quad \text{sous les contraintes} \tag{3.8}$$

$$Ax = g, \qquad x \geq 0, \qquad \langle v, w \rangle = 0. \tag{3.9}$$

Où x a été augmenté d'une composante au-dessus de x dans (3.4) et A et g ont également été convenablement modifiés. Donc

$$x = (z, \psi_-, \xi, \psi_+, y, s) = (v, x, y, s) \in E^{2(n+m)+2}$$

$$v = (z, \psi_-) \qquad w = (\xi, \psi_+),$$

$$g = (-d, b, 1) \in E^{n+m+1} \tag{3.10}$$

$$A = \begin{bmatrix} Q & -R^T & -I & R^T & k_1 & 10 \\ R & 0 & 0 & 0 & k_2 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}, \text{de dimension} \qquad (n+m+1) \times [2(n+m)+2]$$

Notez que, par (3.9) et (3.10), $y + s = 1, s \geq 0$ et $y \geq 0$, et nous aurons $0 \leq y \leq 1$.Pour des raisons qui deviendront claires plus tard [voir théorème (3.4)] nous supposerons désormais qu'aucune colonne de la matrice $\begin{bmatrix} Q \\ R \\ 0 \end{bmatrix}$ est nulle. Ce n'est pas une restriction, car disons que si la i^{ieme} colonne était nulle, alors $r_i = 0$ et $q_i = 0$, et par conséquent, sans perte de généralité, nous pouvons supposer que $d^i = 0$ et reformuler PQC 2.7.1 avec une variable en moins [notez que si $d^i < 0$, alors PQC 2.7.1 n'a pas de solution, puisque $d^i z^i \to -\infty$ comme $z^i \to +\infty$].

Initialisation.Nous devons obtenir une solution de base réalisable au système (3.9). Puisque R est de rang m, par hypothèse, on peut supposer que les colonnes de R sont disposées de sorte que ses m premières colonnes soient linéairement indépendantes.
On définit

$$x_0 = (v_0, w_0, y_0, s_0) \tag{3.11}$$

Avec

$$z_0^i = \begin{cases} 1 & , \quad i = 1, 2, ..., m \\ 0 & , \quad i = m+1, ..., n \end{cases}$$

$$\xi_0^i = \begin{cases} 0 & , \quad i = 1, 2, ..., m \\ 1 & , \quad i = m+1, ..., n \end{cases}$$

$\psi_0^i = 0 \quad , \quad i = 1, 2, ..., m$

$\psi_0^i = 1 \quad , \quad i = 1, 2, ..., m$

$y_0 = 1 \quad , \quad s_0 = 0$

Et que les vecteurs $k_1 \in E^n$ et $k_2 \in E^m$ soient définis par :

$$k_1 = -Qz_0 + R^T\psi_{0-} + \xi_0 - R^T\xi_{0+} - d, \qquad k_2 = -Rz_0 + c. \tag{3.12}$$

Algorithme 1
Supposons que $x_0, x_1, ..., x_n$, avec x_0 et x_1 comme construits en initialisation , sont des solutions de base réalisables générées par l'algorithme tel que pour $j = 1, 2, ..., N$

$$0 < y_i < 1; \tag{3.13}$$

Pour $i = 1, 2, ..., n$ si v_i^j est une variable de base, alors w_j^i ne l'est pas, et si w_j^i est une variable de base, alors v_j^i ne l'est pas.

Étape 1
Notez qu'il existe un indice unique $h \in \{1, 2, ..., n+m\}$ de telle sorte que ni w_N^h ni v_N^h ne sont des variables de base. Depuis une situation similaire doit également tenir pour x_{N-1} avec $N = 2, 3, ...$, nous sommes partis avec deux alternatives :
a. Si v_{N-1} était une variable de base, utilisez la procédure de Canon et al. [1970] pour calculer les points extrêmes adjacents pour rendre w_{N+1}^h de base ; i.e., faire un échange de l'indice $h + n + m \in J(x_N)$ pour certains indices dans l'ensemble des indices de base $\bar{J}(x_N)$. Passez à l'étape 2.
b. Si w_{N-1}^h était une variable de base, utilisez la procédure de Canon et al. [1970] pour rendre v_{N+1}^h de base ; c'est-à-dire, tenter d'échanger l'indice $h \in J(x_N)$ contre un indice en $\bar{J}(x_N)$. Passez à l'étape 2.

Étape 2
Il y a deux résultats possibles pour l'étape 1 :
a. Une nouvelle solution de base réalisable x_{N+1} a été obtenue à l'étape 1. Si $y_{N+1} = 0$ ou $y_{N+1} = 1$, arrêtez. Sinon $0 < y_{N+1} < 1$; définissez $N = N + 1$ et revenez à l'étape 1.
b. Une nouvelle solution de base utilisable n'a pas été obtenue à l'étape 1, c'est-à-dire, un rayon infini $\{x(\theta) : \theta \in [0, \infty)\}$ du polyèdre $\Omega\prime = \{x : Ax = g, x \geq 0\}$ a été rencontré. Dans ce cas, arrêtez.

Condition d'arrêt.
L'algorithme (1) intègre trois conditions d'arrêt.

Arrêt 1.

$$x_{N+1}^{2(n+m)+1} = y_{N+1} = 0.$$

Dans ce cas, les n premières composantes de x_{N+1}, c'est-à-dire le vecteur z_{N+1}, est une solution optimale pour PQC (2.7.1).

Arrêt 2.
À une certaine étape N, un rayon infini du polyèdre a été obtenu, dans ce cas PQC (2.7.1) n'a pas de solution, soit parce qu'il n'a pas de solution réalisable ou parce que l'infimum contraint de la fonction de coût est $-\infty$.

Arrêt 3.
À une certaine étape $N \geq 1$, $x_{N+1}^{2(n+m)+1} = 1$, c'est-à-dire $y_{N+1} = 1$.
Tant que la fonction de coût de PQC (2.7.1) est une forme quadratique définie semi-positive, cette condition d'arrêt ne peut jamais se produire. Ainsi $0 \leq y_N < 1$ pour chaque $N \leq 1$. Cette commande d'arrêt est incluse pour éliminer la nécessité d'établir si la matrice Q est définie semi-positive ou non (généralement une tâche difficile), et l'arrêt si $y_N = 1$ est conçu pour garantir que les calculs seront arrêtés après un nombre fini d'itérations.
Notez que lorsque Q n'est pas définie semi-positive, une solution optimale au nouveau problème dérivé (2.8.1) ne donne qu'une solution réalisable au problème de programmation quadratique, satisfaisant les conditions nécessaires d'optimalité.

3.3 Convergence

Dans cette section, nous allons prouver que l'algorithme (1) doit atteindre une des conditions d'arrêt (1) à (3) et doit donc se terminer après un nombre fini d'itérations. Nous prouverons que cette condition d'arrêt (3) ne se produit jamais lors de la résolution du problème de programmation quadratique canonique (2.7.1) ; c'est-à-dire pour chaque $N \geq 1$, $0 \leq y_N < 1$. Enfin, nous prouverons que si l'algorithme (1) se termine à la condition d'arrêt (2), puis PQC (2.7.1). n'a pas de solution.
Il est intéressant de noter que dans la preuve de convergence des algorithmes que nous sommes fortement appuyés sur le fait que la fonction de coût est réduite à chaque itération. Cet avantage n'est pas disponible pour nous maintenant. En effet, il n'est pas difficile de générer des exemples pour lesquels le scalaire y_N, formellement la fonction de coût pour l'algorithme (3.1.1) , augmente à certaines itérations et diminue à d'autres, en raison du fait que les conditions d'arrêt de l'algorithme (1) sont différentes de celles de (3.1.1). Pour cette raison, la preuve de convergence devient complexe. Cependant, notre tâche peut être quelque peu simplifiée en invoquant une non-dégénérence de l'hypothèse

3.3.1 Hypothèse de non-dégénérence

Chaque solution de base réalisable du système

$$Ax = g, \qquad x \geq 0, \qquad \langle v, w \rangle = 0 \tag{3.14}$$

Est une solution de base réalisable non dégénérée.
En conséquence de cette hypothèse, rappelons que si x est une solution de base à (3.4), alors $\bar{I} = \{i : x^i > 0\}$ est l'ensemble des indices de base défini pour x.

Théorème 3.2 *Supposons que l'hypothèse* (3.3.1) *soit satisfaite pour le problème* (3.3). *Soit* $x_0, x_1, x_2, \ldots$, *une séquence de solutions de base réalisables à* (3.9) *généré par l'algorithme (1). Alors* $x_i \neq x_j$ *pour* $i \neq j$, *avec* $i, j = 0, 1, 2, \ldots$

Corollaire 3.1 *L'algorithme (1) se termine par un nombre fini d'itérations ; c'est-à-dire que l'une des conditions d'arrêt (1) à (3) est rencontrée après un nombre fini d'itérations.*

Théorème 3.3 *Soient* k_1 *et* k_2 *tels que définie en* (3.12), *avec* x_0 *comme donné par* (3.11). *Si nous définissons* $s = 0$, *alors* x_0 *est l'unique solution du système d'équations et d'inégalités* (3.4).

Corollaire 3.2 *Soit* x_N *; pour* $N = 0, 1, 2, \ldots$ *la séquence de solutions de base réalisables générées par l'algorithme (1). Puis pour* $N \geq 1, x_N^{2(n+m)+1} = y_N < 1$ *; ainsi, la condition d'arrêt (3) n'est jamais atteinte.*

Lemme 3.1 *Si le système* $Rz = c$ *et* $z \geq 0$ *a une solution, alors chaque vecteur* $\eta \in E^m$, *satisfaisant le système d'inégalités* $R^T\eta \geq 0$ *satisfait aussi* $\langle \eta, c \rangle \geq 0$.

Lemme 3.2 *Supposons que PQC* (2.7.1) *ait une solution réalisable, c'est-à-dire que le système* $Rz = c$ *et* $z \geq 0$ *a une solution. S'il y a un vecteur* $\eta \in E^n$ *tel que*

$$R\eta = 0, \qquad Q\eta = 0, \qquad \eta \geq 0, \qquad \langle d, \eta \rangle < 0,$$

Ensuite, l'infimum contraint de la fonction de coût dans PQC (2.7.1) *est* $-\infty$.

Théorème 3.4 *Supposons que l'algorithme (1) se termine à l'arrêt de commande (2). Alors soit PQC* (2.7.1) *n'a pas de solution réalisable, ou bien l'infimum de sa fonction de coût est* $-\infty$

Corollaire 3.3 *Si PQC* (2.7.1) *a une solution réalisable et que la contrainte infimum de sa fonction de coût est fini, alors il a une solution optimale [voir. théorème* (2.3)*].*

Conclusion générale

Notre objectif dans ce mémoire est la résolution du problème de programmation quadratique en contrôle optimal. En premier lieu on a montrer que les problèmes de contrôle optimal du temps fixe et le problème de base sont équivalent.
Ensuite, nous nous somme intéréssés à l'étude du problème de programmation quadratique et définir la condition d'existence d'une solution optimal qui se suivera des conditions suffisantes qui garantirons l'unicité de la solution optimal, pour en déduire à la fin les conditions nécéssaires et suffisantes d'optimalités.
On a étudier quelques applications au problème de contrôle illimités sans containtes d'inégalités comme 1^{er} cas , et avec contraintes d'inégalités pour le 2^{eme} cas, dans la section suivante on a traité les conditions d'optimalités pour le problème de programmation canonique, puis on a définit le problème de minimisation dérivée afin de lui appliqué l'algorithme du simplexe en lui apportant une légère modification afin d'obtenir une solution au problème de programmation canonique.
La dernière partie est consacré à la résolution de ce problème et la présentation de l'algorithme du simplexe pour à la fin étudier sa convergence.

Bibliographie

EML Beale. On quadratic proramming. *Naval Research Logistics Quarterly*, 6(3) : 227–243, 1959.

M Canon, C Cullum, et Elijah Polak. Constrained minimization problems in finite-dimensional spaces. *SIAM Journal on Control*, 4(3) :528–547, 1966.

Michael D Canon, Clifton D Cullum Jr, et Elijah Polak. Theory of optimal control and mathematical programming. 1970.

Alan J Goldman. Resolution and separation theorems for polyhedral convex sets. *Linear inequalities and related systems*, 41 :51, 1956.

Hubert Halkin et Lucien W Neustadt. General necessary conditions for optimization problems. *Proceedings of the National Academy of Sciences*, 56(4) : 1066–1071, 1966.

Lucien W Neustadt. An abstract variational theory with applications to a broad class of optimization problems. i. general theory. *SIAM Journal on Control*, 4 (3) :505–527, 1966.

Lucien W Neustadt. An abstract variational theory with applications to a broad class of optimization problems. ii. applications. *SIAM Journal on Control*, 5(1) : 90–137, 1967.

Philip Wolfe. The simplex method for quadratic programming. *Econometrica : Journal of the Econometric Society*, pages 382–398, 1959.

Notations

$A \triangleq B$	A égal à B par définition
$A \supset B$	A contient B
$A \subset B$	A est contenu dans B ; A est un sous-ensemble de B
$A \cup B$	union de A et B
$A \cap B$	intersection de A et B
$A \times B$	produit cartésien de A et B
$\{x : P\}$	ensemble de x ayant la propriété P
$x \in A$	x est un élément de A
$x \notin A$	x n'appartient pas à A
$A/B = \{x : x \in A, x \notin B\}$	(différence d'ensembles)
$\langle .,. \rangle$	produit scalaire (les points représentent des variables non désignées)
$.\rangle\langle.$	pour $x \in E^n$ et $y \in E^m$, $x\rangle\langle y$ est la matrice $n \times m$ xy^T
$\|.\|$	norme euclidienne
$\max_i$	maximum sur i
$x \geq y$	pour $x, y \in E^n, x \geq y$ si $x^i \geq y^i$ pour $i = 1, 2, ..., n$
$F = (f, r)$	
f	fonction de coût
r	fonction de contrainte d'égalité
q	fonction de contrainte d'inégalité
$\Omega, \Omega\prime$	jeux de contraintes
I	ensemble d'indices de contrainte d'inégalité active
J	ensemble d'indices d'indicateur de base
x_i	état du système dynamique à l'instant i
u_i	entrée système à l'instant i
f_i	fonction dynamique
f_i^0	fonction de coût différentiel
x	trajectoire (des états)
u	séquence de contrôle
X_i, X_i', X_i''	contraintes d'espace d'état à l'instant i
U_i	contrainte de contrôle à l'instant i
X_i	phase (état augmenté) à l'instant i
X	trajectoire (des phases)
q_i	fonction de contrainte d'inégalité (espace d'états)
g_i	fonction de contrainte d'égalité (espace d'états)
k	durée du processus de contrôle optimal discret
$A + B = \{x : x = y + z, y \in A, z \in B\}$	(combinaison linéaire d'ensembles)

Résumé

On s'intéresse dans ce mémoire à l'étude d'un problème de programmation quadratique en contrôle optimal, qui consiste à établir les conditions nécéssaires et suffisantes pour qu'une solution optimal existe.

La résolution de ce problème utilise l'algorithme du simplexe en lui apportant quelques modifications pour le généraliser pour tout les problèmes de programmation quadratique sans qu'une solution initiale réalisable ne sera nécéssaire.

Abstract

In this thesis, we are interested in the study of a programming problem quadratic in optimal control, which consists in establishing the necessary conditions sary and sufficient for an optimal solution to exist. The resolution of this problem uses the simplex algorithm by providing it some modifications to generalize it for all pro- quadratic grammation without an initial feasible solution being necessary

Beale [1959] Goldman [1956]