

REPUBLIQUE ALGERIENNE DEMOCRATIQUE et POPULAIRE.
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique.

UNIVERSITE MOULOU D MAMMERI, TIZI-OUZOU



Faculté des Sciences
Département de Mathématiques

En vu de l'obtention du diplôme de Master
en
Recherche Opérationnelle
Option : Méthodes et Modèles de décision

Thème :

Etude comparative entre deux méthodes de résolution de
problèmes de commande optimale :
Application : Régulation d'un four thermique.

Présenté par :

TITOUCHE Slimane

Sous la direction de :

Professeur AIDENE MOHAMED.

Devant le jury d'examen composé de :

M.	MERAKRB Abdelkader ;	Maître de conférences B ;	U.M.M.T.O ;	Président
M.	AIDENE Mohamed ;	Professeur	UMMTO	Rapporteur
Mme.	MOUSSOUNI Nacima ;	Maître assistante A ;	U.M.M.T.O ;	Examinatrice.
Mme.	KARA Fadhila ;	Maître assistante A ;	U.M.M.T.O ;	Examinatrice.

Soutenu le 06/10 / 2011

Remerciements

EL HAMDOUNI ALLAH

Tout d'abord je commence par exprimer ma profonde gratitude à mon cher promoteur *M^r* AIDENE Mohamed qui a proposé et accepté de diriger ce travail.

Je le remercie aussi pour sa grande contribution à l'aboutissement de ce travail, et pour son immense disponibilité malgré son emploi de temps chargé.

Mes vives remerciements s'adressent aussi au Président et à l'ensemble des membres du jury pour l'honneur qu'ils me font en acceptant de juger et d'examiner ce travail.

Je tiens également à remercier chaleureusement le professeur SPITERI, P de m'avoir donné vocation lors de sa conférence ici dans notre faculté, ce qui a nourri en moi cet attrait particulier pour le domaine de contrôle optimal.

Je remercie également tous ceux qui ont contribué de près ou de loin à la réalisation de ce modeste travail.

Je suis également très reconnaissant à l'endroit de tous les enseignants qui nous ont formé durant tout notre cursus en général et particulièrement à ceux du Département des Mathématiques de l'Université de Tizi-Ouzou.

Par ailleurs, je remercie aussi les camarades étudiants et amis pour la collaboration et la discussion tout au long de ce travail. Nous avons compris ensemble que c'est du choc des idées que jaillisse la lumière pour l'avancement de la science.

Table des matières

1	Stabilité des équations différentielles ordinaires	6
1.1	Théorèmes d'existence et d'unicité	6
1.2	Stabilité à la Lyapunov pour les EDO	10
1.2.1	Préliminaires	10
1.3	Stabilité et instabilité	12
1.3.1	Stabilité asymptotique	13
1.3.2	Stabilité uniforme	14
1.3.3	Stabilité en temps fini	15
1.4	Stabilité au sens de Lyapunov	16
1.4.1	Position du problème	16
1.4.2	Illustration graphique du concept de stabilité à la Lyapunov	17
2	Principe du Maximum de Pontryaguin	18
2.1	Contrôlabilité	18
2.1.1	Contrôlabilité des systèmes linéaires	19
2.1.2	Contrôlabilité des systèmes linéaires autonomes	20
2.2	Principe de maximum de Pontryaguin	20
2.2.1	Cas sans contrainte sur le contrôle : Principe du maximum faible	21
2.2.2	Le problème de Lagrange	21
2.2.3	Enoncé général	22
2.3	Conditions de transversalité	24
2.3.1	Contraintes sur l'état	26
2.3.2	Méthode de pénalisation.	26
3	Méthodes de résolution	29
3.1	Position du problème et caractérisation de la solution	29
3.2	Méthode de relaxation	32

3.3	Algorithme de relaxation	32
3.4	Algorithme parallèle asynchrone : méthode de relaxation avec liaison asynchrone	33
3.5	Illustration graphique du concept d'algorithmes parallèles	36
3.6	Analyse de convergence	36
3.7	Méthode de gradient conjugué	40
3.7.1	Principe de la méthode	40
3.8	Algorithme de gradient conjugué	41
3.8.1	Point faible de la méthode du gradient conjugué	43
4	Application : Commande optimale pour la régulation d'un processus thermique	44
4.1	But de l'étude	44
4.2	Modélisation et résolution	44
4.3	Forme canonique du problème	45
4.4	Résolution	46
4.4.1	Simulation de l'évolution de la température dans un four thermique	47
4.4.2	Algorithme de relaxation	50
4.4.3	Algorithme de gradient conjugué	53
4.4.4	Résultats d'exécution	55
4.4.5	Comparaison	55
	Bibliographie	62
	Bibliographie	64

Table des figures

1.1	Envelope de A	7
1.2	Stabilité pratique	12
1.3	Stabilité	13
1.4	Stabilité uniforme	15
1.5	Stabilité en temps fini	16
1.6	Concept de stabilité à la Lyapunov	17
2.1	Stabilité pratique	19
3.1	Mode de fonctionnement des algorithmes parallèles	36
4.1	Evolution de la température dans le four	50
4.2	Graphe de la commande	53
4.3	Comparaison entre les deux algorithmes	55

Introduction

Les besoins de commande en ligne de processus dynamiques complexes continus et soumis à des perturbations, exigent parfois une résolution extrêmement rapide de systèmes différentiels avec conditions aux deux bouts, ces systèmes étant natifs des conditions d'optimalité de Pontryaguin. La résolution efficace de ces systèmes algèbro - différentiels permet ainsi de réagir en temps réel aux perturbations éventuelles agissant sur le processus à contrôler de manière optimale. L'intérêt majeur de cette problématique est donc de réduire de façon sensible les temps de calculs et d'autoriser la conduite en temps réel de processus, de grande dimension, interconnectés entre eux. Dans le cas de problèmes à horizon fixe et à état final libre, on présente dans cette étude, un algorithme de relaxation pour la conduite en ligne d'un processus thermique dans lequel la commande est soumise à des contraintes de type inégalité. L'algorithme de relaxation considéré se déduit directement des conditions d'optimalité de Pontryaguin. À cause des contraintes sur la commande, ces dernières conditions d'optimalité conduisent à la résolution d'un problème multivoque, ce qui classiquement revient à projeter les valeurs calculées de la commande sur l'ensemble convexe définissant ces contraintes. L'algorithme de relaxation étant de nature itérative, sous certaines conditions, l'application de point fixe associée au problème à résoudre est contractante dans un espace produit de fonctions continues par morceaux, ce qui permet de mettre en évidence un critère de convergence simple. De plus, il convient de préciser que les conditions de convergence sont indépendantes de la longueur de l'horizon dans lequel les équations d'évolution sont définies. Par ailleurs, les performances de l'algorithme de relaxation sont comparées avec celles de méthodes plus classiques, comme la méthode du gradient conjugué. Les expérimentations numériques montrent clairement que l'algorithme de relaxation est particulièrement intéressant :

- ✓ lorsque le coût du contrôle est important,
- ✓ lorsque les contraintes sur la commande sont saturées où dans ce cas particulier les performances de la méthode du gradient conjugué s'effondrent.

Dans la première partie de ce travail, nous allons tenter d'exposer les fondements théoriques de ces deux grandes approches à travers une étude détaillée. La deuxième partie s'articule sur l'aspect pratique et application de chacune d'elles en utilisant le logiciel Matlab.

Ainsi le premier chapitre est axé sur un rappel de l'essentiel de la théorie de stabilité du fait de rôle crucial qu'elle joue dans les problèmes de commandes, nous allons présenter les différents concepts de stabilité, en particulier celle de **Lyapunov**.

En chapitre deux nous allons aborder le principe de **Pontryaguin** où nous allons détailler les différents aspects théoriques liés à cette large théorie.

Le troisième chapitre quand à lui constitue le cœur battant de présent travail du fait qu'il contient la théorie de résolution de problèmes de commande optimale. Une nouvelle méthode appelée Relaxation est présentée et comparée avec la méthode de gradient conjugué.

Nous terminons par une conclusion résumant l'ensemble des résultats de cette étude dans laquelle nous dégagerons aussi d'éventuelles perspectives de recherche.

Mots clés : Commande optimale, principe de Pontryaguin, méthode de relaxation, méthode du gradient conjugué, systèmes dynamiques.

Notations :

\forall : pour tout.

\exists : il existe.

| ou t.q : tel que.

$A \setminus B$: ensemble A privé de l'ensemble B.

$Conv(A)$: enveloppe convexe de A.

\bar{A} : adhérence de A.

\mathring{A} : intérieur de A.

∂A : frontière de A i.e : $\bar{A} \setminus \mathring{A}$.

$vect$: espace vectoriel engendré par.

∇f : gradient de f .

$Acc(x_0, T)$: ensemble accessible en temps de T depuis x_0 .

$\mathcal{M}_{n,m}(\mathbb{R})$: ensemble des matrices à n lignes et m colonnes à coefficients dans \mathbb{R} .

\mathcal{M}_n : ensemble des matrices carrées d'ordre n.

A' : transposée de la matrice A.

Chapitre 1

Stabilité des équations différentielles ordinaires

Introduction

Certaines méthodes visant à stabiliser les systèmes non linéaires sont basées sur l'utilisation de la linéarisation statique ou dynamique. Elles peuvent conduire à des résultats plus ou moins satisfaisants du point de vue pratique. Cependant, ceux-ci ne s'adressent qu'à une classe relativement restreinte de systèmes physiques : ceux régis par des équations différentielles ordinaires possédant certaines propriétés. L'objectif de ce chapitre est de présenter sommairement la théorie de la stabilité des équations différentielles. Le problème qui nous intéresse ici est celui des systèmes non linéaires. Les premiers résultats sont apparus avec **Lyapunov** à la fin du 19^{ème} siècle et au début du 20^{ème} siècle. Il donne alors une condition suffisante pour la stabilité des systèmes non linéaires. **Chetaev**, quant à lui, montrera un théorème d'instabilité en 1934. Puis, **Massera** démontre une condition nécessaire et suffisante de stabilité. **Hahn** et **Lefschetz** achèveront alors dans les années 1960 cette théorie.

1.1 Théorèmes d'existence et d'unicité

Soit U un ouvert non vide de \mathbb{R}^n ($n \in \mathbb{N}^*$) contenant 0, et I un intervalle non vide de \mathbb{R} , non borné à droite. Soit $f : U \rightarrow \mathbb{R}^n$ une application continue, on lui associe le système :

$$\dot{x} = f(x) \quad (1.1)$$

Soit encore $f : I \times U \rightarrow \mathbb{R}^n$ on lui associe le système :

$$\dot{x} = f(t, x) \quad (1.2)$$

$x(t, t_0, x_0)$ désigne une solution $x(t)$ du système (1.1), ou du système (1.2) telle que $x(t_0) = x_0$ (condition initiale du problème).

Soit $A \subset U$, $\varepsilon > 0$, on définit la boule :

$$B(x, \varepsilon) = \{y \in U; \|y - x\| < \varepsilon\}$$

et $B_\varepsilon(A) = \bigcup_{x \in A} B(x, \varepsilon)$ l'enveloppe de A, comme l'illustre la figue suivante :

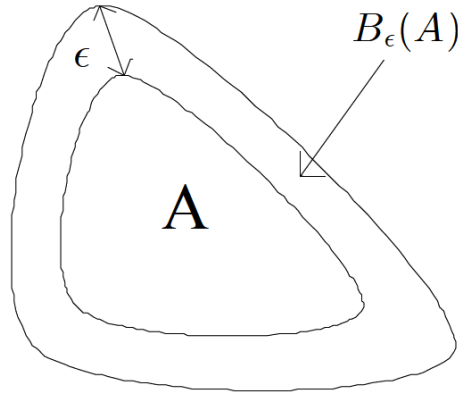


FIG. 1.1: Enveloppe de A

Si $U \in \mathbb{R}^n$, \bar{U} désigne l'adhérence de U dans \mathbb{R}^n et \dot{U} son intérieur.

Soient $a = (a_1, a_2, \dots, a_n) \in U$. On note :

$$Df(a).h = \sum_{i=1}^n \frac{\partial f}{\partial y_i}(a)h_i \text{ et } Df(a) = \sum_{i=1}^n \frac{\partial f}{\partial y_i}(a)dy_i$$

Soient $t_0 \in I$ et $r \in \mathbb{R}_*^+$. On note :

1. $\nu(U)$ un voisinage de 0 dans U .
2. On définit les voisinages cylindriques suivants :

$$c_{t_0, r}(U) = \{(t, y) \in I \times U; t \geq t_0, y \in B(0, r)\}$$

$$\nu_{t_0}(U) = \{(t, y) \in I \times U; t \geq t_0, y \in \nu(U)\}$$

Remarque 1.1.1. On introduit les voisinage cylindriques à base sphérique que l'on utilise pour les démonstrations, afin de simplifier l'écriture. On peut bien le faire de fait que ces derniers définissent bien une base de voisinage.

Théorème 1.1. (*Arzela Péano*)[2] On considère le système (1.2). Alors, par tout point $(t_0, x_0) \in I \times U$ passe au moins une solution maximale $x(t, t_0, x_0)$ définie sur $J \subset I$.

Théorème 1.2. (de Carathéodory)[8] Soit $f : I \times U \rightarrow \mathbb{R}^n$ telle que :

(i) pour tout $x \in U$, $\varphi_y : t \in I \mapsto f(t, x)$ est mesurable.

(ii) pour presque tout t , $\phi_t : x \in U \mapsto f(t, x)$ est continue.

(iii) pour tout compact K de U , il existe $m_k : I \mapsto \mathbb{R}^+$ intégrable sur I telle que :

$$\forall (t, x) \in I \times K, |f(t, x)| \leq m_k(t).$$

On considère le système :

$$\dot{x} = f(t, x), t \in I$$

dit le système de Carathéodory. Alors pour tout $(x_0, t_0) \in \dot{I} \times U$ passe au moins une solution $x(t, t_0, x_0)$ définie pour presque tout t .

preuve. Par hypothèse, on sait que l'on peut trouver $r > 0$ et $h > 0$ tels que :

$$\|x - x_0\| \leq r \Rightarrow \int_{t_0-h}^{t_0+h} m_{\overline{B(x_0, r)}}(s) ds \leq r.$$

On note \mathcal{C} l'ensemble des applications continues de $[t_0 - h, t_0 + h]$ dans $\overline{B(x_0, r)}$.

\mathcal{C} est un convexe fermé de l'espace de banach $C^0([t_0 - h, t_0 + h], \mathbb{R}^n)$.

Si $u \in \mathcal{C}$, $t \in [t_0 - h, t_0 + h]$, on définit :

$$T(u)(t) = x_0 + \int_{t_0}^t f(s, u(s)) ds.$$

Grâce à la définition de h et à (iii), on voit que $T(u)$ est à valeurs dans $\overline{B(x_0, r)}$. De plus

$$|T(u)(t_1) - T(u)(t_2)| = \left| \int_{t_1}^{t_2} f(s, u(s)) ds \right| \leq \int_{t_1}^{t_2} m_{\overline{B(x_0, r)}}(s) ds \xrightarrow{t_2 \rightarrow t_1} 0$$

ce qui montre que $T(u)$ est continue, donc $T(u) \in \mathcal{C}$.

$m_{\overline{B(x_0, r)}}$ étant intégrable, on déduit que :

$$t \rightarrow \int_{t_0}^t m_{\overline{B(x_0, r)}}(s) ds.$$

est uniformément continue. Par conséquent, $T(\mathcal{C})$ est équicontinue. D'après le théorème d'Ascoli, comme $T(\mathcal{C})$ est borné, on a que $T(\mathcal{C})$ est relativement compact dans \mathcal{C} .

Soit $(U_n)_{n \geq 1}$ une suite d'éléments de \mathcal{C} qui tend vers u et $t \in [t_0 - h, t_0 + h]$;

On a :

$$|T(U_n)(t) - T(u)(t)| = \left| \int_{t_0}^t f(s, u_n(s)) - f(s, u(s)) ds \right| \leq \int_{t_0}^t |f(s, u_n(s)) - f(s, u(s))| ds$$

Notons alors J l'ensemble des $t \in [t_0 - h, t_0 + h]$ tels que ϕ_t soit continue.

$I \setminus J$ étant négligable, on a :

$$\|T(U_n) - T(u)\|_\infty \leq \int_{t_0}^t |f(s, u_n(s)) - f(s, u(s))| ds$$

u_n converge uniformément vers u , donc simplement. La continuité de $\phi_s, \forall s \in J$, permet d'affirmer que :

$$|f(s, u_n(s)) - f(s, u(s))| \xrightarrow{n \rightarrow \infty} 0.$$

Puisque on a la domination suivante :

$$|f(s, u_n(s)) - f(s, u(s))| \leq 2m_{\overline{B_{x_0, r}(s)}}$$

on peut dans ce cas licitement appliquer le théorème de la convergence dominée et affirmer que :

$$\|T(u_n) - T(u)\|_\infty \xrightarrow{n \rightarrow \infty} 0$$

Par conséquent T est continue.

En utilisant le théorème du point fixe de **Schauder**, on en déduit que T admet un point fixe $x(t_0, t_0, x_0) \in \mathcal{C}$. De plus cette application vérifie :

$$x(t, t_0, x_0) = x_0 + \int_{t_0}^t f(s, x(s, t_0, x_0)) ds.$$

La fonction $t \rightarrow f(t, x(t))$ est intégrable sur $[t_0 - h, t_0 + h]$, en conséquence le théorème de dérivation des intégrales de **Lebesgue** permet d'affirmer que $x(t, t_0, x_0)$ est presque dérivable partout, et que $\forall t$, on a :

$$\dot{x}(t, t_0, x_0) = f(t, x(t, t_0, x_0)).$$

Définition 1.1.1. Soit $f : I \times U \rightarrow \mathbb{R}^n$, on dit que :

1. f est localement *lipschitzienne* en la première variable, si : $\forall (t_0, x_0) \in I \times U, \exists V : \text{voisinage de } (t_0, x_0) \in I \times U$ et $c > 0$ telle que :

$$\forall ((t, y), (t', y)) \in V^2, \|f(t, y) - f(t', y)\| \leq c \|t - t'\|.$$

L'ensemble des fonction localement *lipschitzienne* en la première variable est noté : $Lip_t(I \times U)$.

2. f est localement *lipschitzienne* en la deuxième variable, si : $\forall (t_0, x_0) \in I \times U, \exists V : \text{voisinage de } (t_0, x_0) \in I \times U$ et $c > 0$ telle que :

$$\forall ((t, y), (t, y')) \in V^2, \|f(t, y) - f(t, y')\| \leq c \|y - y'\|.$$

L'ensemble des fonction localement *lipschitzienne* en la deuxième variable est noté : $Lip_y(I \times U)$.

On note $Lip_{(t,y)}(I \times U)$ l'ensemble des fonctions localement *lipschitziennes* en la première et la seconde variable.

On donne enfin le théorème de **Cauchy-Lipschitz** dont on trouvera la preuve dans [6].

Théorème 1.1.1. (*de Cauchy-Lipschitz*)[2] Soit $f : I \times U \rightarrow \mathbb{R}^n$ continue, $f \in Lip_y(I \times U)$, on considère le système (1.2), alors par tout point $(t_0, x_0) \in I \times U$ passe une seule solution maximale $x(t, t_0, x_0)$ définie sur $J \subset I$.

1.2 Stabilité à la Lyapunov pour les EDO

La stabilité au sens de **Lyapunov** joue un rôle cruciale dans la théorie de la commande, de fait qu'elle vérifié la stabilité temporelle des solutions, cette section sera axée sur les points importants de cette vaste théorie.

1.2.1 Préliminaires

Dans ce qui suit, on donne quelques définitions concernant les points les plus importants qui interviens dans l'illustration cde la théorie de stabilité au sens de **Lyapunov**.

Quelques définitions

Définition 1.2.1. Un point $\bar{x} \in U$ est un **point d'équilibre**, ou point **singulier** du système (1.1) (respectivement (1.2)) si $f(\bar{x}) = 0$ (respectivement $\forall t \in I, f(t, \bar{x}) = 0$).

On considérera toujours l'équilibre en 0. Pour le cas général, il suffit de faire une translation.

Définition 1.2.2. On considère le système (1.1), et

$V : U \rightarrow \mathbb{R}^n$ ayant des dérivées partielles sur U , on définit alors la dérivée totale \dot{V} pour le système (1.2) par :

$$\dot{V}(y) = \sum_{i=1}^n \frac{\partial^i}{\partial x_i} V(y) f_i(y).$$

Définition 1.2.3. Soit le système (1.2) et $V : I \times U \rightarrow \mathbb{R}^n$ ayant les dérivées partielles sur $I \times U$. On définit la dérivée totale \dot{V} pour le système (1.2) par :

$$\dot{V}(t, y) = \frac{\partial V}{\partial t}(t, y) + \sum_{i=1}^n \frac{\partial^i}{\partial x_i} V(t, y) f_i(t, y).$$

Définition 1.2.4. Soit $a \in \mathbb{R}_*^+$ et $\phi : [0, a] \rightarrow \mathbb{R}^+$ une application continue, on dit que ϕ appartient à la classe \mathcal{K} si :

1. ϕ est strictement croissante.
2. $\phi(0) = 0$.

Définition 1.2.5. Soit $\phi : \mathbb{R}^+ \rightarrow \mathbb{R}^+$ une application continue, on dit que ϕ est de classe \mathcal{K}^∞ si :

1. ϕ est strictement croissante.
2. $\phi(0) = 0$.
3. $\lim_{r \rightarrow \infty} \phi(r) = +\infty$.

Exemples 1.2.1. Donnons deux exemples des fonctions de classe \mathcal{K} et de classe \mathcal{K}^∞ :

1. ϕ définie par $\phi(r) = \arctan(r)$ est strictement croissante car :

$$\phi'(r) = \frac{1}{1+r^2} > 0.$$

Ainsi, ϕ appartient à la classe \mathcal{K} , mais $\lim_{r \rightarrow \infty} \phi(r) = \frac{\pi}{2} \neq \infty$ par conséquent ϕ n'est pas de classe \mathcal{K}^∞ .

2. Soit $c > 0$, ϕ définie par $\phi(r) = r^c$ est strictement croissante car :

$$\phi'(r) = cr^{c-1} > 0.$$

De plus, $\lim_{r \rightarrow \infty} \phi(r) = \infty$, donc ϕ appartient à la classe \mathcal{K}^∞ .

1.3 Stabilité et instabilité

Définition 1.3.1. Soit $D_0 \subset D \subset U$ non vides, $(T_s, T_f) \in \mathbb{R}_*^{+2}$ tels que $T_s < T_f$, K un compact non vide de U , on dit que K est (I, T_s, T_f, D_0, D) -pratiquement stable pour le système (1.2), si pour tout $x_0 \in D_0$ et $t_0 \in I$, on a :

1. $x(t, t_0, x_0)$ est définie sur $[t_0, t_0 + T_f]$.
2. $\forall t \in [t_0, t_0 + T_f], x(t, t_0, x_0) \in D$.
3. $\forall t \in [t_0 + T_s, t_0 + T_f], x(t, t_0, x_0) \in K$.

La figure suivante illustre la stabilité pratique :

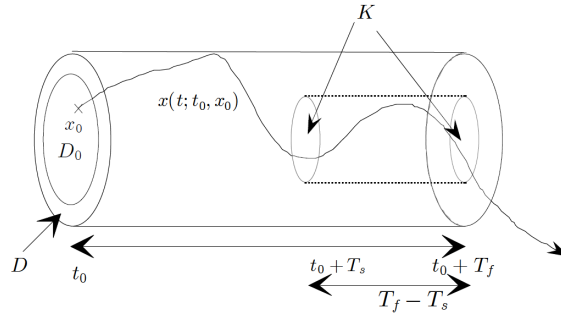


FIG. 1.2: Stabilité pratique

Définition 1.3.2. De plus, K est (I, T_s, T_f, D_0, D) -pratiquement stable en temps fini si K est $(I, T_s, +\infty, D_0, D)$ -pratiquement stable.

Définition 1.3.3. Soit K un compact non vide de U , on dit que K est stable pour le système (1.2) si pour tout $\epsilon > 0$, et $t_0 \in I$, il existe $\delta(\epsilon, t_0) > 0$ tel que :

$$(x_0 \in B_{\delta(\epsilon, t_0)}(K)) \Rightarrow \begin{cases} x(t, t_0, x_0), & \text{est définie pour } t \geq t_0, \\ x(t, t_0, x_0) \in B_\epsilon(K), & \forall t \geq t_0. \end{cases}$$

Exemples 1.3.1. Soit le système :

$$\dot{x} = (6t \sin(t) - 2t)x, t \in \mathbb{R}$$

Il est clair que ce système admet des solutions de la forme :

$$x(t, t_0, x_0) = x_0 \exp\left[\int_{t_0}^t (6s \sin(s) - 2s) ds\right] = x_0 \exp[6 \sin(t) - 6t \cos(t) - t^2 - 6 \sin(t_0) + 6t_0 \cos(t_0) + t_0^2].$$

On voit que le terme en exponentielle est borné pour tout $t \geq t_0$ par une constante $c(t_0)$ qui dépend seulement de t_0 . Ainsi, on a :

$$|x(t, t_0, x_0)| \leq |x_0|c(t_0) \quad \forall t \geq t_0.$$

pour tout $\epsilon > 0$, si l'on choisit $\delta = \frac{\epsilon}{c(t_0)}$, on voit que 0 est stable. La notion de stabilité est bien éllustrée par la figure suivante :

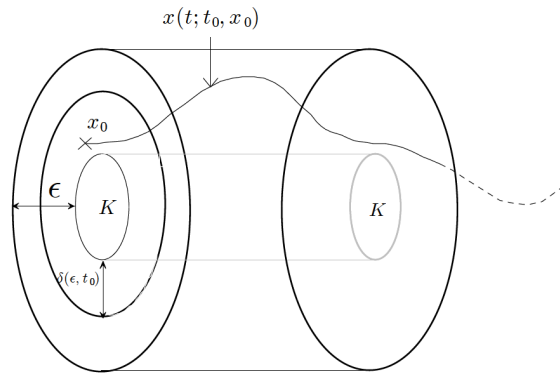


FIG. 1.3: Stabilité

Définition 1.3.4. Soit K un compact non vide de U , stable pour le système (1.2).

Pour tout $\epsilon > 0$, et $t_0 \in I$, il existe un $\delta_{max}(\epsilon, t_0) > 0$ maximum parmi les $\delta(\epsilon, t_0)$. On définit le domaine de stabilité de K par rapport à t_0 par :

$$\mathcal{D}_s(t_0, K) = \bigcup_{\epsilon > 0} D(0, \delta_{max}(\epsilon, t_0))$$

puis le domaine de stabilité de K :

$$\mathcal{D}_s(K) = \bigcup_{t_0 \in I} \mathcal{D}_s(t_0, K).$$

1.3.1 Stabilité asymptotique

Définition 1.3.5. Soit K un compact non vide de U , on dit que K est **attractif** pour le système (1.2) si pour tout $t_0 \in I$, il existe $\delta(t_0) > 0$ tel que :

$$(x_0 \in B_{\delta(t_0)}(k)) \Rightarrow \begin{cases} x(t, t_0, x_0), \\ \lim_{t \rightarrow \infty} d(x(t, t_0, x_0), K) = 0. \end{cases} \quad \text{est définie pour } t \geq t_0,$$

Définition 1.3.6. Soit K un compact non vide de U , on dit que K est **asymptotiquement stable** pour le système (1.2) si :

1. K est stable pour le système (1.2).
2. K est **attractif** pour le système (1.2).

Définition 1.3.7. Soit K un compact non vide de U , on considère le système (1.2). On dit que K est **globalement asymptotiquement stable** pour le système (1.2) si :

1. K est stable pour le système (1.2).
2. Pour tout $t_0 \in I$ et $x_0 \in U$, $x(t, t_0, x_0)$ est définie pour $t \geq t_0$ et $\lim_{t \rightarrow \infty} d(x(t, t_0, x_0), K) = 0$.

Exemples 1.3.2. Soit le système :

$$\dot{x} = \frac{x}{1+t}, t \geq 0.$$

Il admet des solutions de la forme :

$$x(t, t_0, x_0) = x_0 \exp\left(\int_{t_0}^t \frac{-1}{1+s} ds\right) = x_0 \frac{1+t_0}{1+t}$$

. On voit ainsi que 0 est globalement asymptotiquement stable.

1.3.2 Stabilité uniforme

Définition 1.3.8. Soit K un compact non vide de U , on considère le système (1.2). On dit que K est **uniformément stable** pour le système (1.2) si pour tout $\epsilon > 0$, il existe $\delta(\epsilon) > 0$ tel que pour tout $t_0 \in I$,

$$(x_0 \in B_{\delta(\epsilon)}(k)) \Rightarrow \begin{cases} x(t, t_0, x_0), & \text{est définie pour } t \geq t_0, \\ x(t, t_0, x_0) \in B_{\epsilon}(K) \forall t \geq t_0. \end{cases}$$

Remarque 1.3.1. La stabilité uniforme entraîne la stabilité, mais la réciproque est fautive.

Exemples 1.3.3. Reprenons l'exemple précédent :

$\dot{x} = (6t \sin(t) - 2t)x, t \in \mathbb{R}$ On suppose que t_0 prenne les valeurs successives : $t_0 = 2\pi n, n \in \mathbb{N}$, et que $x(t, t_0, x_0)$ soit évalué en $t_0 + \pi$, alors

$$x(t_0 + \pi, t_0, x_0) = x_0 \exp[(4n + 1)(6 - \pi)\pi]$$

Donc si $x_0 \neq 0$ on a :

$$\lim_{n \rightarrow \infty} \frac{x(t_0 + \pi, t_0, x_0)}{x_0} = +\infty.$$

Ainsi, pour tout ϵ donné, il n'existe pas de δ indépendant de t_0 qui permette d'affirmer que 0 est uniformément stable.

La stabilité uniforme est expliquée par la figure suivante :

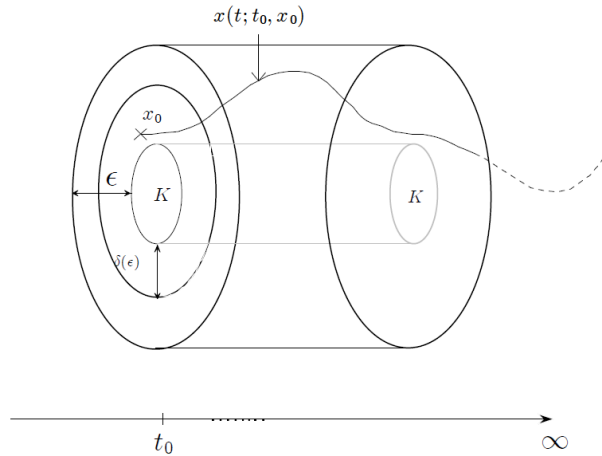


FIG. 1.4: Stabilité uniforme

1.3.3 Stabilité en temps fini

Définition 1.3.9. Soit K un compact non vide de U , on dit que K est **stable en temps fini** pour le système (1.2) si :

1. K est stable pour le système (1.2)
2. Pour tout $t_0 \in I$. il existe $\delta(t_0) > 0$ tel que :

$$(x_0 \in B_{\delta(t_0)}(K)) \Rightarrow \exists 0 < T(t_0, x_0) < +\infty; \forall t \geq T(t_0, x_0), x(t, t_0, x_0) \in K).$$

Avec : $T(t_0, x_0) = \inf T(t_0, x_0) \geq 0; d(x(t, t_0, x_0), K) = 0 \forall t \geq t_0 + T(t_0, x_0)$ qui s'appel le **temps d'établissement** et on dit que $x(t, t_0, x_0)$ converge vers K en temps fini $T_0(t_0, x_0)$.

Cette notion est éllustrée par la figure suivante :

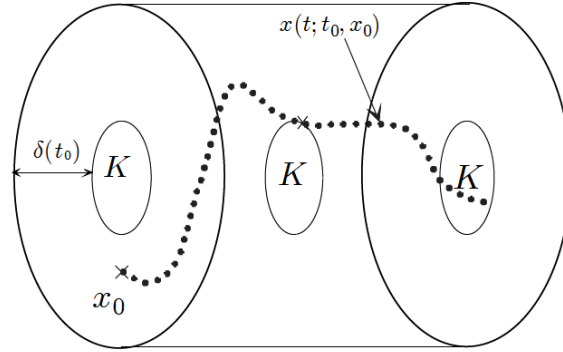


FIG. 1.5: Stabilité en temps fini

Remarque 1.3.2. On remarque que K est stable en temps fini si K est asymptotiquement stable, et si toute solutions qui converge vers K , le fait en tepms fini.

Définition 1.3.10. Soit K un compact non vide de U , et soit le système (1.2).

On dit que K est **globalement stable en temps fini** pour le système (1.2) si :

1. K est stable pour le système (1.2).
2. Pour tout $t_0 \in I$ et $x_0 \in U$:

$$\exists 0 < T(t_0, x_0) < +\infty; \forall t \geq T(t_0, x_0), x(t, t_0, x_0) \in K.$$

Définition 1.3.11. Soit K un compact non vide de U , on dit que K est **uniformément stable en temps fini** pour le système (1.2) si :

1. K est uniformément asymptotiquement stable pour le système (1.2).
2. il existe $\delta > 0$ tel que pour tout $t_0 \in I$:

$$x_0 \in B_\delta(K) \Rightarrow \exists 0 < T(x_0) < +\infty; \forall t \geq T(x_0), x(t, t_0, x_0) \in K.$$

On dit que $x(t, t_0, x_0)$ converge uniformément vers K en temps fini $T(x_0)$.

1.4 Stabilité au sens de Lyapunov

1.4.1 Position du problème

Soit le problème suivant :

$$\begin{cases} \frac{dx}{dt} = f(x, t), t_0 < t \leq T \\ x(t_0) = x_0. \end{cases}$$

Soit (x_0, t_0) une condition initiale du problème et (\bar{x}, t_0) une condition initiale perturbée, on dit que la solution $x(t, t_0, x_0)$ est stable à la **Lyapunov** si :

$$\forall \varepsilon : 0 < \varepsilon < \rho, \forall t \in]\tau, T], \exists \eta(\varepsilon, t) > 0 : \forall \bar{x}_0 : |x_0 - \bar{x}_0| < \eta$$

Définition 1.4.1. On dit que la solution $x(t, t_0, x_0)$ est asymptotiquement stable à la Lyapunov si :

1. $x(t, t_0, x_0)$ est stable.
2. $\lim_{t \rightarrow \infty} (x(t, t_0, \bar{x}_0)) = x(t, t_0, x_0)$.

1.4.2 Illustration graphique du concept de stabilité à la Lyapunov

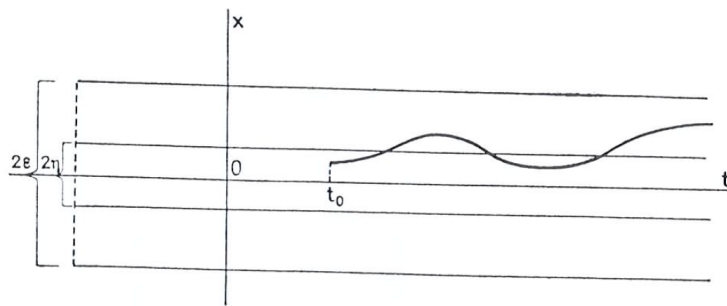


FIG. 1.6: Concept de stabilité à la Lyapunov

Interprétation

Quelque soit la perturbation de la condition initiale x_0 située dans le 'cylindre' de rayon 2η la solution perturbée reste dans le cylindre de rayon 2ε .

Conclusion

Au long de ce chapitre, nous avons abordés les différents points concernant la théorie de stabilité, cette dernière joue un rôle crucial dans la théorie de la commande. Toutefois pour notre cas on ne retiendra que les notions de stabilité à la Lyapunov.

Chapitre 2

Principe du Maximum de Pontryaguin

Introduction

Un système non contrôlable, est systématiquement exclu de cadre d'application de la théorie de control optimal, pour cela on commence toujours par la vérification de la contrôlabilité des système étudiés, on donne dans ce qui suit quelques définitions concernant ce concept.

Ensuite nous allons donner une variante générale de principe du maximum de Pontryaguin et on percevra que dans le cas général ce résultat est partiellement difficile à démontrer. En revanche lorsqu'il n'y a pas de contrainte sur la commande (ce qui n'est pas le cas dans notre étude), la démonstration est simple, et on arrive au principe dit faible. dans ce qui suit on va tenter de présenter les deux cas de figure, en commençant par le plus simple, pour finir par le cas général.

2.1 Contrôlabilité

Un système de contrôle est dit contrôlable si on peut l'amener (en temps fini) d'un état initial quelconque vers un état final déterminé, pour les systèmes de contrôle linéaires en dimension finie, il existe une caractérisation très simple de la contrôlabilité due à **Kalman**.

Définition 2.1. Le système contrôlé $\dot{y}(t) = A(t)y(t) + Bu(t) + r(t)$ est dit contrôlable en temps T si $Acc(y_0, T) = \mathbb{R}^n$, i.e, pour tous $y_0, y_1 \in \mathbb{R}^n$, il existe un contrôle u tel que la trajectoire associée relie y_0 à y_1 en temps T , comme l'explique la figure suivante :



FIG. 2.1: Stabilité pratique

2.1.1 Contrôlabilité des systèmes linéaires

Un système de contrôle linéaire est formulé mathématiquement de la façon suivante :

$$\begin{cases} \dot{y}(t) = A(t)y(t) + B(t)u(t) + r(t). \\ y(0) = y_0 \end{cases} \quad t \in [t_0, T]$$

où : A, B et r sont des applications localement intégrables sur $[t_0, T]$ à valeurs respectives dans : $\mathcal{M}_n(\mathbb{R})$, $\mathcal{M}_{n,m}(\mathbb{R})$, et \mathbb{R}^n , $u \in U_{ad}$: ensemble des commandes admissibles.

Soit : $M(\cdot) : [t_0, T] \rightarrow \mathcal{M}_n(\mathbb{R})$ la résolvante du système linéaire homogène $\dot{y}(t) = A(t)y(t)$ définie par : $\dot{M}(t) = A(t)M(t)$, $M(0) = Id$.

Pour tout contrôle u , le système :

$$\begin{cases} \dot{y}(t) = A(t)y(t) + B(t)u(t) + r(t). \\ y(0) = y_0 \end{cases} \quad t \in [t_0, T]$$

admet une solution unique $y(\cdot) : [t_0, T] \rightarrow \mathbb{R}^n$ donnée par :

$$y(t) = M(t)y_0 + \int_{t_0}^T M(t)M(s)^{-1}(B(s)u(s) + r(s))ds, \forall t \in [t_0, T].$$

Si $r = 0$ et $y_0 = 0$, la solution du système s'écrit :

$$y(t) = M(t) \int_{t_0}^T M(s)^{-1}B(s)u(s)ds.$$

Remarque 2.1.1. On remarque que cette solution est linéaire en u .

Ci-après un théorème donnant une condition générale pour la contrôlabilité des systèmes linéaires.

Théorème 2.1 (2). *Le système $\dot{y}(t) = A(t)y(t) + B(t)u(t) + r(t)$, est contrôlable en temps T si et seulement si la matrice :*

$$C(T) = \int_{t_0}^T M(t)^{-1}B(t)B(t)'M^{-1}(t)dt$$

dite matrice de contrôlabilité, est inversible.

On remarque bien que cette condition ne dépend pas de la condition initiale y_0 , c'est-à-dire si un système linéaire est contrôlable en temps T depuis y_0 , alors il est contrôlable en T depuis n'importe quel point.

2.1.2 Contrôlabilité des systèmes linéaires autonomes

Un système linéaire $\dot{y}(t) = Ay(t) + Bu(t) + r(t)$, est dit **autonome** si les matrices A et B ne dépend pas de t .

Dans ce qui suit, un théorème qui donne une condition nécessaire et suffisante de contrôlabilité dans le cas sans contraintes sur le contrôle.

Théorème 2.2 (2). *On suppose que $U = \mathbb{R}^n$, le système $\dot{y}(t) = Ay(t) + Bu(t) + r(t)$ est contrôlable en temps T , si et seulement si la matrice*

$$C = (B \setminus AB \setminus \dots \setminus A^{n-1}B)$$

est de rang n .

*C : est appelée matrice de **Kalman**, et la condition $\text{rang}C = n$ est appelée condition de **Kalman**.*

Dans le cas où on a une contrainte sur le contrôle u , c'est-à-dire $u \in U_{ad}$, les propriétés de contrôlabilité globale sont reliées aux propriétés de stabilité de la matrice A .

i.e : si $r = 0$, et $0 \in U_{ad}$, si la condition de **Kalman** est vérifiée et si la matrice A est stable (toutes les valeurs propres de A sont de partie réelle strictement négative) alors tout point de \mathbb{R}^n peut être conduit à l'origine en temps fini.

2.2 Principe de maximum de Pontryaguin

On passe maintenant à la présentation de principe de maximum de Pontryaguin selon le cas.

2.2.1 Cas sans contrainte sur le contrôle : Principe du maximum faible

2.2.2 Le problème de Lagrange

Soit à chercher les conditions nécessaire d'optimalité pour le système :

$$\dot{y} = f(t, y(t), u(t)) \quad (2.1)$$

où les contrôles $u(\cdot) \in U_{ad}$ sont définis sur $[0, T]$ et les trajectoires associées doivent vérifier les conditions initiales et finales, à savoir $y(0) = y_0$ et $y(T) = y_1$; le problème est de minimiser un coût de la forme :

$$C(u) = \int_0^T f^0(t, x(t), u(t)) \quad (2.2)$$

où T est fixé.

Associons au système (2.2) le système augmenté suivant :

$$\dot{y} = f(t, y(t), u(t)) \dot{y}^0(t) = f^0(t, y(t), u(t)) \quad (2.3)$$

et notons $\tilde{y} = (y, y^0)$, $\tilde{f} = (f, f^0)$. Le problème se résume donc à la recherche d'une trajectoire solution de (2.3) joignant les points $\tilde{y}_0 = (y_0, 0)$ et $\tilde{y}^1 = (y_1, y^0(T))$, et minimisant la dernière coordonnée $y^0(T)$.

L'ensemble des états accessibles à partir de \tilde{y}^0 pour le système (2.3) est

$$\tilde{Acc}(\tilde{y}_0, T) = \bigcup_{u(\cdot)} \tilde{y}(T, \tilde{y}_0, u).$$

Proposition 2.2.1. *Soit u un contrôle singulier sur $[0, T]$ pour le système de contrôle (2.1), et soit $x(\cdot)$ la trajectoire singulière associée. Alors il existe une application absolument continue $p : [0, T] \rightarrow \mathbb{R}^n$, appelée vecteur adjoint, telle que les équations suivantes sont vérifiées pour presque tout $t \in [0, T]$*

$$\dot{y}(t) = \frac{\partial H}{\partial p}(t, y(t), p(t), u(t)) \quad (2.4)$$

$$\dot{p}(t) = \frac{\partial H}{\partial y}(t, y(t), p(t), u(t)) \quad (2.5)$$

$$\frac{\partial H}{\partial u}(t, y(t), p(t), u(t)) = 0 \quad (2.6)$$

où H est l'hamiltonien du système.

L'équation (2.6) est appelée l'équation de contrainte.

On présente alors le lemme crucial suivant :

Lemme 2.2.1. *Si le contrôle u associé au système de contrôle (2.1) est optimal pour le coût (2.2), alors il est singulier sur $[0, T]$ pour le système augmenté (2.3)*

Démonstration :

Notons \tilde{y} la trajectoire associée, solution du système augmenté (2.3), issue de $\tilde{y}_0 = (y_0, 0)$. Le contrôle u étant optimal pour le coût (2.2), il en résulte que le point $\tilde{y}(T)$ appartient à la frontière de l'ensemble $\tilde{Acc}(\tilde{y}_0, T)$. En effet, s'il existait un voisinage du point $\tilde{y}(T) = (y_1, y^0(T))$ dans $\tilde{Acc}(\tilde{y}_0, T)$ contenant un point $\tilde{y}(T)$ solution du système (2.3) et tel que l'on ait $y_0(T) < y^0(T)$, ce qui contredirait l'optimalité du contrôle u , par conséquent, le contrôle u est singulier pour le système augmenté (2.3) sur $[0, T]$.

La version forte suivante, beaucoup plus difficile à montrer, prend compte les contraintes sur le contrôle, et affirme que cet extremum est un maximum. On a l'énoncé général suivant.

2.2.3 Énoncé général

Théorème 2.2.1 (3). *On considère le système de contrôle dans \mathbb{R}^n*

$$\dot{y}(t) = f(t, y(t), u(t)), \quad (2.7)$$

Où $f : \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^m \longrightarrow \mathbb{R}^n$ est une fonction de classe \mathcal{C}^1 et où les contrôles sont des applications mesurables et bornées définies sur un intervalle $[0, T[$ de \mathbb{R}^+ et à valeur dans $\Omega \subset \mathbb{R}^m$. Soient M_0 et M_1 deux sous ensembles de \mathbb{R}^n . On note U_{ad} l'ensemble des contrôles admissibles u dont les trajectoires associées relient un point initial de M_0 à un point final de M_1 en temps $t < T$.

Par ailleurs on définit le coût d'un contrôle u sur $[0, t]$ où $f^0 : \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^m \longrightarrow \mathbb{R}$ et $g : \mathbb{R} \times \mathbb{R}^n \longrightarrow \mathbb{R}$ sont \mathcal{C}^1 , et $y(\cdot)$ est la trajectoire solution de (2.7) associée au contrôle u .

On considère le problème de contrôle optimal suivant : déterminer une trajectoire reliant M_0 à M_1 et minimisant le coût. Le temps final peut être fixé ou non.

Si le contrôle $u \in U_{ad}$ associé à la trajectoire $y(\cdot)$ est optimal sur $[0, T]$, alors il existe

une application $p(\cdot) : [0, T] \rightarrow \mathbb{R}^n$ absolument continue appelée vecteur adjoint, et un réel $p^0 \leq 0$, tels que le couple $(p(\cdot), p^0)$ est non trivial, et tels que, pour presque tout $t \in [0, T]$,

$$\dot{y}(t) = \frac{\partial H}{\partial p}(t, y(t), p^0, \mu(t)), \quad \dot{p}(t) = -\frac{\partial H}{\partial y}(t, y(t), p(t), p^0, u(t)), \quad (2.8)$$

où $H(t, y, p, p^0, u) = \langle p, f(t, y, u) \rangle$ est le hamiltonien du système, et on a la condition de maximisation presque partout sur $[0, T]$

$$H(t, y(t), p(t), p^0, u(t)) = \max_{v \in \Omega} H(t, y(t), p(t), p^0, v). \quad (2.9)$$

Si de plus le temps final pour joindre la cible M_1 n'est pas fixé, on a la condition au temps final T

$$\max_{v \in \Omega} H(T, x(T), p(T), p^0, v) = -p^0 \frac{\partial g}{\partial t}(T, x(T)), \quad (2.10)$$

Si de plus M_0 et M_1 (ou juste l'un des deux ensembles) sont des variétés de \mathbb{R}^n ayant des espaces tangents en $y(0) \in M_0$ et $y(T) \in M_1$, alors le vecteur adjoint peut être construit de manière à vérifier les conditions de transversalité aux deux extrémités (ou juste l'une des deux)

$$p(0) \perp T_{y(0)}M_0 \quad (2.11)$$

et

$$p(T) - p^0 \frac{\partial g}{\partial y}(T, y(T)) \perp T_{y(T)}M_1. \quad (2.12)$$

Remarque 2.1. Si le contrôle u est continu au temps T , la condition (2.10) peut s'écrire

$$H(T, y(T), p(T), p^0, u(T)) = -p^0 \frac{\partial g}{\partial t}(T, y(T)). \quad (2.13)$$

Remarque 2.2. Si la variété M_1 s'écrit sous la forme

$$M_1 = \{y \in \mathbb{R}^n \mid F_1(y) = \dots = F_p(y) = 0\},$$

où les F_i sont des fonctions de classe C^1 sur \mathbb{R}^n , alors la condition (2.12) se met sous la forme

$$\exists \lambda_1, \dots, \lambda_p \in \mathbb{R} \setminus p(T) = \sum_p^{i=1} \lambda_i \nabla F_i(y(T)) + p^0 \frac{\partial g}{\partial y}(T, y(T)). \quad (2.14)$$

Remarque 2.3. Dans les conditions du théorème, on a de plus pour presque tout $t \in [0, T]$

$$\frac{d}{dt}H(t, y(t), p(t), p^0, u(t)) = \frac{\partial H}{\partial t}(t, y(t), p(t), p^0, u(t)).$$

En particulier si le système augmenté est autonome, i.e. si f et f^0 ne dépendent pas de t , alors H ne dépend pas de t , et on a

$$\forall t \in [0, T] \max_{v \in \Omega} H(y(t), p(t), p^0, v) = \text{cste.}$$

2.3 Conditions de transversalité

Conditions de transversalité sur le vecteur adjoint

Dans ce paragraphe le temps final pour atteindre la cible peut être fixé ou non. Réécrivons les conditions (2.11) et (2.12) dans les deux cas importants suivants.

Problème de lagrange

Dans ce cas le coût s'écrit

$$C(t, u) = \int_0^t f^0(s, y(s), u(s)) ds,$$

i.e. $g=0$. Les conditions de transversalité (2.11) et (2.12) sur le vecteur adjoint s'écrivent alors

$$p(0) \perp T_{y(0)}M_0, p(T) \perp T_{y(T)}M_1. \quad (2.15)$$

Problème de Mayer

. Dans ce cas le coût s'écrit

$$C(t, u) = g(t, y(T)),$$

i.e. $f^0 = 0$. Les conditions de transversalité (2.11) et (2.12)(ou (2.14)) ne se simplifient pas a priori.

Mais dans le cas particulier important où $M_1 = \mathbb{R}^n$, autrement dit le point final $x(T)$ est libre, la condition (2.12) devient

$$p(T) = p^0 \frac{\partial g}{\partial y}(T, y(T)). \quad (2.16)$$

Si de plus g ne dépend pas du temps, on a coutume d'écrire $p(T) = p^0 \nabla g(y(T))$, autrement dit le vecteur adjoint au temps final est égal, à la constante p^0 près, au gradient de g pris

au point final.

Conditions de transversalité sur le Hamiltonien

La condition (2.10) n'est valable que si le temps final pour atteindre la cible n'est pas fixé. Dans ce paragraphe nous nous plaçons donc dans ce cas.

La seule simplification notable de cette condition est le cas où la fonction g ne dépend pas du temps t (ce qui est vrai par exemple pour un problème de Lagrange), et la condition de transversalité (2.13) sur le Hamiltonien devient alors

$$\max_{v \in \omega} H(T, y(T), p(T), p^0, v) = 0, \quad (2.17)$$

ou encore si u est continu au temps T ,

$$H(T, y(T), p(T), p^0, u(T)) = 0. \quad (2.18)$$

Autrement dit le Hamiltonien s'annule au temps final.

Remarque 2.4. Si le système augmenté est de plus autonome, i.e. si f et f^0 ne dépendent pas de t , alors d'après la remarque (2.3) on a le long d'une extrémale

$$\forall t \in [0, T] \max_{v \in \omega} H(y(t), p(t), p^0, v) = 0.$$

Généralisation des conditions de transversalité

Pour écrire les conditions de transversalité associées à un problème de contrôle plus général, il faut écrire les relations adéquates en termes de multiplicateurs de Lagrange.

Par exemple considérons un problème de Lagrange avec des conditions aux limites mélanges, i.e. on cherche une trajectoire solution de

$$\dot{x}(t) = f(t, x(t), u(t)),$$

minimisant le coût

$$C(T, u) = \int_0^T f^0(t, x(t), u(t)) dt,$$

et vérifiant les conditions aux limites

$$(x(0), x(T)) \in M,$$

où M est une sous variété de $\mathbb{R}^n \times \mathbb{R}^n$.

On peut alors montrer (voir [1]) que dans ce cas les conditions de transversalité (2.11)

et (2.12) sur le vecteur adjoint s'écrivent

$$(-p(0), p(T)) \perp T_{y(0), y(T)} M.$$

Un cas important de conditions mélangées est le cas des trajectoires périodiques, i.e.

$x(0) = y(T)$ non fixé. Dans ce cas on a

$$M = (y, y) \setminus y \in \mathbb{R}^n,$$

et la condition de transversalité donne

$$p(0) = p(T).$$

Autrement dit, non seulement la trajectoire est périodique, mais ainsi son relèvement extrémal.

2.3.1 Contraintes sur l'état

Principe du maximum avec contrainte sur l'état.

Le principe du maximum tel qu'il vient d'être énoncé prend en compte des contraintes sur le contrôle, mais ne prend pas en compte d'éventuelles contraintes sur l'état. Ce problème est en effet beaucoup plus difficile. Il existe plusieurs versions du principe du maximum avec contraintes sur l'état. La théorie est cependant beaucoup plus compliquée, et nous ne l'abordons pas dans cette mémoire. Une différence fondamentale avec le principe du maximum classique est que la présence de contraintes sur l'état peut rendre le vecteur adjoint discontinu. On rajoute alors des conditions de saut, ou de jonction.

En fait, lorsqu'il existe des contraintes sur l'état de la forme $c_i(y) \leq 0$, $i = 1, \dots, p$, où les fonctions $c_i : \mathbb{R}^n \rightarrow \mathbb{R}$ sont de classe C^1 , alors le vecteur adjoint $p(\cdot)$ est solution de l'équation intégrale

$$p(t) = p(T) + \int_t^T \frac{\partial H}{\partial x} dt - \sum_{i=1}^p \int_t^T \frac{\partial c_i}{\partial x} du_i,$$

où les u_i sont des mesures positives ou nulles dont le support est contenu dans $t \in [0, T] \setminus c_i(y(t)) = 0$.

2.3.2 Méthode de pénalisation.

Un moyen simple de manipuler des contraintes sur l'état est de résoudre un problème de contrôle optimal modifié, où, dans la théorie LQ, on pondère cette contrainte, de manière à la forcer à être vérifiée. Le principe général de cette méthode est le suivant. Supposons qu'on veuille imposer à l'état d'appartenir à un sous ensemble $C \subset \mathbb{R}^n$. Donnons-nous une fonction g sur \mathbb{R}^n , nulle sur C et strictement positive ailleurs (il faut être capable de construire une telle fonction). Alors, en ajoutant au coût $C(t, u)$ le scalaire $\lambda \int_0^T g(y(t)) dt$,

où $\lambda > 0$ est un point que l'on peut choisir assez grand, on espère que la résolution de ce problème de contrôle optimal modifié va forcer la trajectoire à rester dans l'ensemble C . En effet si $y(t)$ sort de l'ensemble C , et si λ est grand, alors le coût correspondant est grand, et probablement la trajectoire ne sera pas optimale.

La justification théorique de ce procédé réside dans la proposition générale suivante.

Proposition 2.3.1. *Soit (E, d) un espace métrique, C un sous-ensemble de E , et f une fonction k -lipschitzienne sur E . Pour tout $x \in E$, posons $g(y) = d(y, C)$. Supposons que la fonction f restreinte à C atteint son minimum en $x_0 \in C$, i.e.*

$$f(y_0) = \min_{y \in C} f(y).$$

Alors, pour tout réel $\lambda \geq k$, on a

$$f(y_0) + \lambda g(y_0) = \min_{y \in C} f(y) + \lambda g(y),$$

i.e. y_0 est aussi un point où $f + \lambda g$ atteint son minimum sur C . La réciproque est vraie si de plus $\lambda > k$ et si C est fermé.

Démonstration :

Raisonnons par l'absurde, et supposons qu'il existe $y \in E$ et $\epsilon > 0$ tels que $f(y) + \lambda d(y, C) < f(y_0) - \lambda \epsilon$. Soit alors $z \in E$ tel que $d(y, z) \leq d(y, C) + \epsilon$. On a

$$f(z) \leq f(y) + kd(y, z) \leq f(y) + \lambda d(y, C) + \lambda \epsilon < f(y_0),$$

ce qui est une contradiction.

Pour la réciproque, supposons que $\lambda > k$ et que C est fermé. Soit $y_0 \in C$ un point où $f + \lambda g$ atteint son minimum sur C , et soit $\epsilon > 0$. Il existe $z \in C$ tel que $d(y_0, z) < d(y_0, C) + \epsilon / \lambda$. On a

$$\begin{aligned} f(z) &\leq f(y_0) + kd(y_0, z) \\ &\leq f(y_0) + kd(y_0, C) + \frac{k}{\lambda} \epsilon \\ &< f(y_0) + \lambda d(y_0, C) - (\lambda - k)d(y_0, C) + \epsilon \\ &< f(z) - (\lambda - k)d(y_0, C) + \epsilon \end{aligned}$$

et donc $(\lambda - k)d(y_0, C) < \epsilon$. le réel $\epsilon > 0$ étant arbitraire, on en déduit que $d(y_0, C) = 0$, et donc $y_0 \in C$ puisque C est fermé. On conclut que pour tout $z \in C$ on a $f(z) \geq f(y_0)$.

Conclusion

Dans ce chapitre nous avons exposé le principe de maximum de Pontryaguin qui caractérise les solutions optimales de problèmes de contrôle optimal, qui a donné naissance

pour beaucoup de méthodes de résolution telle que la méthode de relaxation. Et à la fin de ce chapitre l'essentiel de la théorie de contrôlabilité.

Chapitre 3

Méthodes de résolution

Introduction

Les nécessités de commande en ligne de processus dynamiques complexes continus et soumis à des perturbations, exigent parfois une résolution extrêmement rapide de systèmes différentiels avec conditions aux deux bouts, ces systèmes étant issus des conditions d'optimalité de Pontryaguin. La résolution efficace de ces systèmes algèbro - différentiels permet ainsi de réagir en temps réel aux perturbations éventuelles agissant sur le processus à contrôler de manière optimale. L'intérêt majeur de cette problématique est donc de réduire de façon sensible les temps de calcul et d'autoriser la conduite en temps réel de processus, de grande dimension, interconnectés entre eux. Dans le cas de problèmes à horizon fixe et à état final libre, on présente dans cet étude, un algorithme de relaxation, et un algorithme de gradient conjugué, et les performances des deux méthodes seront comparées expérimentalement.

3.1 Position du problème et caractérisation de la solution

Soit le système physique dont l'état est décrit par l'équation différentielle ordinaire suivantes :

$$\begin{cases} \frac{dy}{dt} = f(y, u, t), u \in U_{ad}, t_0 \leq t \leq T \\ y(t_0) = y_0 \end{cases} \quad (3.1)$$

où U_{ad} représente l'ensemble de commande admissible, qui est un ensemble convexe fermé des applications bornées, continues par morceaux sur $[t_0, T]$.

On désire conduire le système d'un état initial $y(t_0)$ à un état final $y(T)$ en minimisant

la fonction coût suivante :

$$J = S(y(T), T) + \int_{t_0}^T r(y, u, t) dt \quad (3.2)$$

où $S(y(T), T)$ matérialise le coût terminal.

On ait donc en présence d'un problème de commande optimale à horizon fixe et à état final libre.

On fait l'hypothèse que les applications r et f sont deux fois continûment dérivables par rapport à y et u . La solution d'un tel problème est classiquement caractérisée en utilisant le **principe de pontryaguin**, qui compte tenu des contraintes sur la commande s'écrit ici :

$$\begin{cases} \frac{dy}{dt} = \nabla_p H = f(y, u, t), & y(t_0) = y_0 \\ -\frac{dp}{dt} = \nabla_y H = \nabla_y r(y, u, t) + [\frac{\partial f}{\partial y}]^T \cdot p(t), & p(T) = \nabla_y S(y(T), T) \\ \nabla_u H = \frac{\partial r}{\partial u} + [\frac{\partial f}{\partial u}]^T \cdot p(t) + \partial \Psi_{U_{ad}} \end{cases} \quad (3.3)$$

où H est l'Hamiltonien défini par :

$$H(y(t), p(t), u(t), t) = S((T), T) + r(y, u, t) + p^t \cdot f(y, u, t)$$

p est l'état adjoint, $\Psi_{u_{ad}}$ est la fonction indicatrice du convexe et $\partial \Psi_{u_{ad}}$ est le sous différentiel de la fonction indicatrice du convexe U_{ad} . Rappelons que, classiquement, les conditions d'optimalité précédentes expriment le fait que le Hamiltonien $H(y, p, u, t)$ possède un minimum absolu. Notons qu'en l'absence de coût terminal, i.e. si $S(y(T), T) = 0$, alors $p(T) = 0$. La prise en compte des contraintes sur la commande conduit à la résolution d'un problème multivoque ; ceci provient de la perturbation de la troisième équation par le sous- différentiel de la fonction indicatrice du convexe U_{ad} application multivoque dont nous rappelons ci-dessous la définition ainsi que les principales propriétés.

Définition 3.1.1. Soit E un espace vectoriel normé et E' sont dual topologique ; soit ϕ une application non-différentiable ; l'ensemble des $\xi \in E'$ noté $\partial \phi(\mu)$ tel que :

$$\phi(\mu) - \phi(\sigma) - \langle \xi, \mu - \sigma \rangle \geq 0, \forall \mu \in E, \xi \in \partial \phi(\mu)$$

est le sous différentiel de ϕ en μ et ξ est le sous-gradient.

Remarque 3.1.1. Si ϕ est une application différentiable alors $\partial \phi(\mu)$ se réduit à $\{\phi'(\mu)\}$. Pour voir clairement, la fonction $\phi = |\mu|$ n'est pas différentiable à l'origine ; le sous

différentiel de cette application est donc $\partial\phi(\mu) \equiv \text{sing}(\mu)$, soit

$$\partial\phi(\mu) = \text{sing}(\mu) = \begin{cases} -1, & \text{si } \mu < 0 \\ (-1, +1), & \text{si } \mu = 0 \\ +1, & \text{si } \mu > 0. \end{cases}$$

Proposition 3.1.1. [6] Un élément μ_0 vérifie $\phi(\mu_0) = \min_{\mu \in E} \phi(\mu)$, si et seulement si

$$0 \in \partial\phi(\mu_0)$$

Preuve

On sait que $\mu_0 \in E$ vérifie $\phi(\mu_0) \leq \phi(\mu), \forall \mu \in E$ si et seulement si on a $\phi(\mu) \geq \phi(\mu_0) + \langle 0, \mu - \mu_0 \rangle$, ce qui implique que $0 \in \partial\phi(\mu_0)$.

Proposition 3.1.2. $\partial\phi$ est un opérateur monotone multivoque de E dans E'

En effet :

$$\text{Soit } \xi \in \partial\phi(\mu) \Rightarrow \phi(\sigma) \geq \phi(\mu) + \langle \xi, \sigma - \mu \rangle, \forall \sigma \in E$$

$$\text{Soit } \zeta \in \partial\phi(\pi) \Rightarrow \phi(\sigma) \geq \phi(\pi) + \langle \zeta, \sigma - \pi \rangle, \forall \sigma \in E$$

On pose dans la première inégalité que $\sigma = \pi$, et que dans la deuxième inégalité $\sigma = \mu$, puis on additionne terme à terme pour obtenir : $\langle \xi - \zeta, \mu - \pi \rangle \geq 0$, ce qui établit la monotonie de l'application $\partial\phi$.

Résultat :

Il en résulte que la minimisation de ϕ sur un convexe $U \subset E$ se ramène à la résolution d'une équation multivoque $0 \in \mathbf{A}(\mu)$, avec ($\mathbf{A}(\mu) = \partial(\phi + \Psi_u)$), où Ψ_u est la fonction indicatrice du convexe U , dont on donne la définition ci après

Définition 3.1.2. Soit U un convexe fermé, E un espace vectoriel normé, $U \subset E$; la fonction indicatrice du convexe est alors définie par

$$\Psi_u(z) = \begin{cases} 0 & \text{si } z \in U \\ +\infty & \text{si } z \notin U \end{cases}$$

Propriétés 3.1.1. sous différentiel de Ψ_u

1. $\partial\Psi_u(\mu) = \{\xi \in E' \mid \langle \xi, \mu - \sigma \rangle \geq 0, \forall \sigma \in U\} \Rightarrow \text{Dom}(\partial\Psi_u) = \text{Dom}(\Psi_u) = U$.
2. $\partial\Psi_u(\mu) = \{0\}, \forall \mu \in U$ et si $\mu \in \partial U$, $\partial\Psi_u(\mu)$ coïncide avec le cône des normales à U au point μ .

avec : ∂U est la frontière de U .

Exemples 3.1.1. $U = \{u | u_{max} \geq u_{min}\}$; alors :

$$\partial\Psi_u(v) = \begin{cases}] -\infty, 0] & si & v = u_{min} \\ 0 & si & u_{max} \geq v \geq u_{min} \\ 0, +\infty & si & v = u_{max} \\ \emptyset & sino & \end{cases}$$

application qui est bien monotone.

Remarque 3.1.2. La formule des équations d'optimalité donnée par le système (3.3) traduit la projection de la commande sur le convexe U_{ad} définissant les contraintes; précision que nous cherchons à minimiser $\partial(H + \Psi_{U_{ad}})$, et que, compte tenu de la continuité de H , cela conduit finalement à la résolution de l'équation du système (3.3). De plus, en l'absence de contraintes, il est clair que $\partial\Psi_{U_{ad}} \equiv 0$, et les équations d'optimalité se réduisent aux formules classiques suivantes :

$$\begin{cases} \frac{dy}{dt} = \nabla_p H = f(y, u, t), & y(t_0) = y_0 \\ -\frac{dp}{dt} = \nabla_y H = \nabla_y r(y, u, t) + [\frac{\partial f}{\partial y}]^T \cdot p(t), & p(T) = \nabla_y S(y(T), T) \\ \nabla_u H = \frac{\partial r}{\partial u} + [\frac{\partial f}{\partial u}]^T \cdot p(t) = 0 \end{cases} \quad (3.4)$$

3.2 Méthode de relaxation

En mathématique une technique de relaxation est une méthode d'optimisation qui consiste à remplacer une contrainte stricte en contrainte moins stricte, voir même à la supprimer.

3.3 Algorithme de relaxation

La caractérisation précédente des solutions donne directement une méthodologie de résolution du problème de commande optimale.

Cette méthode consiste à intégrer alternativement par séquence, l'équation d'état dans le sens direct, et l'équation d'état adjoint dans le sens inverse, les étapes de l'algorithme sont résumées comme suit :

Initialisation : On donne une approximation initial de la commande $u^0(t)$.

Etape 1. Détermination de la variable d'état $y^{(q+1)}$ par intégration numérique de l'équation d'état :

$$\frac{\partial y^{(q+1)}}{\partial t} = Ay^{(q+1)} + Bu^{(q)}; y^{(q+1)}(0) = y_0$$

Etape 2. Détermination de l'état adjoint $p^{(q+1)}$ par intégration dans le sens inverse de l'équation d'état adjoint :

$$-\frac{\partial p^{(q+1)}}{\partial t} = A^t p^{(q+1)} + c^t c y^{(q+1)} - c^t z_d; p^{(q+1)}(T) = 0.$$

Etape 3. Détermination de la Commande $u^{(q+1)}(t)$

$$u^{(q+1)}(t) = \underset{U_{ad}}{P}roj(u_d - \frac{\alpha}{\beta} B^t P^{(q+1)}(t)).$$

Les étapes (1), (2) et (3) sont recommencées jusqu'à convergence. On donne ci-après un résultat de convergence.

Proposition 3.3.1. [6] *Si la matrice A est l'opposée d'une M -matrice et si le rapport $\frac{\beta}{\alpha} < k_0$, alors la méthode de relaxation décrite par les étapes précédentes est convergente.*

Remarque 3.3.1. La convergence de la méthode de relaxation est assurée dès que le poids de la commande est plus important que celui de la précision sur l'observation dans le critère $J(u)$ à minimiser.

3.4 Algorithme parallèle asynchrone : méthode de relaxation avec liaison asynchrone

L'approche parallèle asynchrone, consiste en l' intégration simultanée des équations d'état et d'état adjoint , en utilisant deux processeurs ; chaque processeur envoie les résultats de ses calculs à l'autre sous forme de paquets ; les paquets correspondent à un ensemble de valeurs sur un intervalle $[\tau_h, \tau_{h+1}] \subset [0, T]$.

La loi de commande est déterminée par l'un des deux processeurs, et éventuellement par un troisième. Indiquons que l'asynchronisme est dû au fait que chaque processeur fonctionne individuellement à son propre rythme en utilisant les dernières données disponibles (interaction).

Cet algorithme peut être formellement modelé d'une manière mathématique ; et pour cela, nous considérons la notation suivante :

Soit α et v deux entiers avec $\alpha = 3$. Considérons maintenant un ensemble des nombres réels $\{\tau_1, \dots, \tau_v\}$ tel que :

$$\sum_{h=1}^v (\tau_{h+1} - \tau_h) = T$$

avec $\tau_1 = 0$ et $\tau_{v+1} = T$ et on définit les espaces suivants :

$$\begin{cases} \tilde{E}_{1,h} = c(] \tau_h, \tau_{h+1}] ; \mathbb{R}^n) \\ \tilde{E}_{2,h} = c(] \tau_h, \tau_{h+1}] ; \mathbb{R}^n) \\ \tilde{E}_{3,h} = c(] \tau_h, \tau_{h+1}] ; \mathbb{R}^m) \end{cases}$$

où $c()$ représente l'ensemble des fonctions continues, et soit :

$$\tilde{E}_i = \prod_{h=i}^v \tilde{E}_{i,h}; i = 1, \dots, \alpha; E = \prod_{i=1}^v \tilde{E}_i$$

Soit encore

$$\tilde{x} = (\tilde{x}_1, \tilde{x}_2, \tilde{x}_3)^t, \text{ avec } \tilde{x}_1 = y, \tilde{x}_2 = p; \tilde{x}_3 = u$$

et

$$\begin{cases} \tilde{\Lambda}_1(\tilde{x}) = Ay - Bu \\ \tilde{\Lambda}_2(\tilde{x}) = A^t p c^t . c . y + c^t z_d \\ \tilde{\Lambda}_3(\tilde{x}) = B^t + k(u - u_d) \end{cases} \quad (3.5)$$

Enfin, considérons l'ensemble d'opérateurs diagonaux $\tilde{\Lambda}_i^d$ pour $i = 1, \dots, \alpha$ défini par : $\tilde{\Lambda}_1^d = \frac{d}{dt}, D(\tilde{\Lambda}_1^d)$, étant l'ensemble des fonctions n-dimensionnelles continues par morceaux sur $[0, T]$, en ce qui concerne les sous-intervalles $[\tau_h, \tau_{h+1}]$ et qui disparaît quand $t = 0$, $\tilde{\Lambda}_2^d = -\frac{d}{dt}, D(\tilde{\Lambda}_2^d)$, représentant l'ensemble des fonctions n-dimensionnelles continues par morceaux sur $[0, T]$, en ce qui concerne les sous-intervalles $[\tau_h, \tau_{h+1}]$ et qui disparaît quand $t = T$, et $\tilde{\Lambda}_3^d = \partial\psi U_{ad} D(\wedge_3^d)$ étant U_{ad} .

Avec les notations précédentes, le système modélisant le problème physique peut être écrit comme suit :

$$\tilde{\Lambda}_i^d \tilde{x} + \tilde{\Lambda}_i(\tilde{x}) \ni 0, \forall i \in 1, \dots, \alpha \quad (3.6)$$

On peut alors associer l'équation du point fixe au système (3.5) comme suit :

afin de simplifier les notations nous considérons que les fonctions vectorielles $t \rightarrow (y(t), p(t), u(t))$ sont définis sur l'intervalle $[0, T]$. d'ailleurs dans chaque sous-intervalle $]\tau_h, \tau_{h+1}[$ (respectivement $]\tau_k, \tau_{k+1}[$) y (respectivement p) et u vérifient les equations suivante :

$$\begin{cases} \frac{dy}{dt} = Ay + Bu, \forall t \in]\tau_h, \tau_{h+1}[, \forall h \in \{1, \dots, v-1\} \\ -\frac{dp}{dt} = A^t p + C^t . C \xi - C^t z_d, \forall t \in]\tau_k, \tau_{k+1}[, \forall k \in \{v-1, \dots, 1\} \\ B^t \phi + k(u - u_d) + \partial\psi U_{ad} \ni 0 \end{cases} \quad (3.7)$$

où (ξ, ϕ, v) sont des fonctions continues par morceaux sur $[0, T]$, pour la subdivision $[\tau_h, \tau_{h+1}]$ de $[0, t]$ et l'état initial à la frontière de chaque sous-intervalle est donné par le fait qu'on considère y et p sont des fonctions continues sur $[0, T]$

Remarque 3.4.1. Plus précisément, si par exemple $y_h = y_{] \tau_h, \tau_{h+1}]}$, alors nous décrivons $y_{h+1}(\tau_h) = y_h(\tau_h)$ et l'état final analogue pour p_k en temps τ_{k+1} .

Soit $\tilde{W} = \{\xi, \phi, v\}$, où $\xi(t), \phi(t)$ and $v(t)$ sont les valeurs disponibles de $y(t), p(t)$ et $u(t)$ respectivement au temps t ; nous écrivons maintenant le système (3.6) comme suit :

$$\tilde{X}(t) = F(\tilde{W}(t)), \forall t \in [0, T] \quad (*)$$

où F représente la tracée du point fixe.

Nous pouvons à présent définir l'algorithme de relaxation avec communication asynchrone en utilisant les définitions suivantes :

Définition 3.4.1. Soit q_1 et q_2 , deux entiers positifs, et :

$s = q_1 \bmod(v) + 1$, $s' = v - q_2 \bmod(v)$ et $q = q_1 + q_2$

$$\begin{cases} \text{si } s > 1 \text{ alors} \\ \hat{s}(q) = \text{Sup} \bar{s} < s | y_{] \tau_{\bar{s}}, \tau_{\bar{s}+1}]}^{(q)} \text{ soit calculée par le premier processeur.} \\ \text{sinon} \\ \hat{s}(q) = v. \end{cases}$$

$$\begin{cases} \text{si } s' < v \text{ alors} \\ \hat{s}'(q) = \text{inf} \bar{s}' > s' | p_{] \tau_{\bar{s}'}, \tau_{\bar{s}'+1}]}^{(q)} \text{ soit calculée par le second processeur.} \\ \text{sinon} \\ \hat{s}'(q) = 1. \end{cases}$$

Alors la stratégie $s(q)$ est définie par :

Si $\hat{s}(q) \equiv ((q_1 - 1) \bmod(v) + 1) \bmod(v)$ alors $s \in s(q)$ et $q_1 = q_1 + 1$.

Et si

$\hat{s}'(q) \equiv (v - q_2 - 1) \bmod(v)$ alors $s' \in s(q)$ et $q_2 = q_2 + 1$.

Où $\bmod(v)$ représente l'opérateur modulo.

Remarque 3.4.2. la valeur de $s(q)$ est alors la valeur de l'index d'intervalle de temps $] \tau_h, \tau_{h+1}]$, pour laquelle un processeur est en charge de l'évaluation de nouveaux composants dans cette période, en utilisant les valeurs des composants connues de précédent itérations, quand les deux processeurs commencent en même temps leurs calculs sur la même période, $s(q)$ n'est pas nécessairement réduite à un élément. Notons aussi, que la stratégie de balayage de l'intervalle de temps par un des processeurs est séquentielle.

Définition 3.4.2. Soit $X^0 \in D(F)$, et la séquence X^q qui est définie par induction comme suit :

$\forall q \in \mathbb{N}, \forall i \in 1, \dots, \alpha, \forall t \in [0, T], \forall k \in 1, \dots, v :$

$$\tilde{x}_i^{q+1} = \begin{cases} \tilde{x}_i^q(t) & \text{si } k \notin s(q) \\ F_i(\tilde{W}(t)) & \text{si } k \in s(q) \end{cases} \quad (3.8)$$

3.5 Illustration graphique du concept d'algorithmes parallèles

Dans cette section nous allons présenter un schéma illustratifs de concept d'algorithmes parallèles :

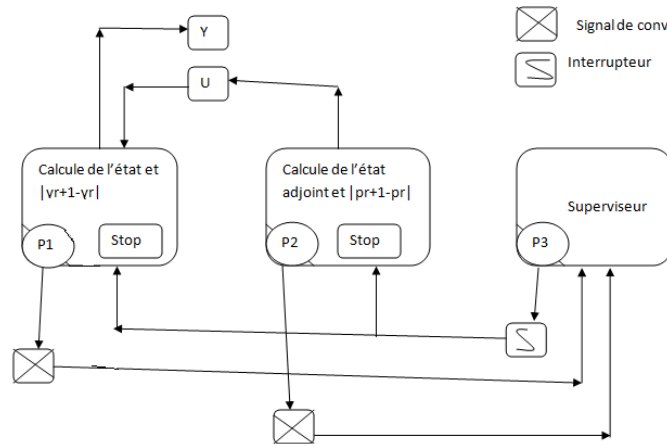


FIG. 3.1: Mode de fonctionnement des algorithmes parallèles

3.6 Analyse de convergence

Notons, dans un premier temps que pour l'algorithme parallèle défini précédemment, les processus des itérations sont continus par morceaux sur $[0, T]$, sous telles conditions, les opérateurs surgissant dans le problème (3.6), sont définis de l'ensemble des fonctions continues par morceaux dans l'ensemble des fonctions continues. Pour analyser la convergence de l'algorithme précédent, nous pouvons considérer des opérateurs $\beta \geq \alpha$, qui représentent les points de décomposition dans la pratique, et nous montrons que la tracée du point fixe liée à cette décomposition se contracte pour une norme scalaire adaptée dans l'ensemble

des fonctions continues par morceaux, puis par l'application des résultats de El TARAZI, les résultats précédents garantissent la convergence des itérations asynchrones pour cette décomposition particulière.

En effet, soit $\beta = 2n + m$ et soient les espaces suivants $E_{\iota,h} = C([\tau_h, \tau_{h+1}])$, $\iota \in 1, \dots, n$

$$|\bar{y}_{\iota,h}| = \sup_{t \in [\tau_h, \tau_{h+1}]} |y(t)|, \bar{y} \in E_{\iota,h}. \quad (3.9)$$

Considérons maintenant les espaces suivants

$$E_{\iota} = \prod_{h=1}^v E_{\iota,h} \quad (3.10)$$

normé par

$$|y|_{\infty} = \max_{h \in 1, \dots, v} |y|_{\iota,h}, y \in E_{\iota} \quad (3.11)$$

Nous pouvons également définir les espaces $E_{\iota,h} = C([\tau_h, \tau_{h+1}])$, $\forall \iota \in n + 1, \dots, 2n$, normés de la même manière que dans 3.7. Pour ϕ élément de $E_{\iota,h}$, on définit E_{ι} , $|\phi|_{\infty}$ comme dans les relations (3.8) et (3.9).

Finalement, pour $\iota \in 2n + 1, \dots, \beta$, nous considérons également :

$$E_{\iota} = \prod_{h=1}^v C([\tau_h, \tau_{h+1}])$$

normé de la même façon que dans (3.9).

Soit encore E_{ι}^* l'espace dual de E_{ι} , et \langle, \rangle_{ι} l'appariement entre E_{ι}^* et E_{ι} . Pour tout $x_{\iota} \in E_{\iota}$, on associe ce qu'on appelle la figure dual de E

$$G_{\iota}(x_{\iota}) = \{g_{\iota} \in E_{\iota}^*; \langle x_{\iota}, g_{\iota} \rangle = |x_{\iota}|_{\infty}^2 = \|g_{\iota}\|_*^2\}$$

où, $\|\cdot\|_{\infty}$ représente la norme uniforme, et $\|\cdot\|_*$ représente la norme de E_{ι}^* .

Soit

$$E = \prod_{\iota=1}^{\beta} E_{\iota} \quad \text{et } X = (y(t), p(t), u(t))^t, X \in E$$

Considérons à présent l'opérateur défini par (3.5), qui peut être écrit comme suit :

$\Lambda(X) = aX + h$, avec

$$a = \begin{pmatrix} A & 0 & -B \\ -C^t.C & A^t & 0 \\ 0 & B^t & kI \end{pmatrix} \quad h = \begin{pmatrix} 0 \\ -C^t z_d \\ -k u_d \end{pmatrix}$$

où : I est la matrice identité de \mathbb{R}^n . Si $a_{\iota,k}$ les entrées générales de la matrice a , et si pour $\iota \in 1, \dots, \beta$, h_ι est le ι^{ieme} composant de vecteur h , on pourra alors définir la relation suivante :

$$\wedge_\iota(X) = \sum_{k=1}^{\beta} a_{\iota k} x_k + h_\iota, \forall \iota \in 1, \dots, \beta \quad (3.12)$$

et soit $\wedge(X) = \{\wedge_1(X), \dots, \wedge_\beta(X)\}$. où la relation précédente correspond aux points de décomposition de l'opérateur $\wedge(X)$. D'une façon équivalente on définit les points de décomposition de l'opérateur \wedge^d , en utilisant la méthode suivante :

Pour chaque $\iota \in 1, \dots, n$, \wedge_ι^d est défini par $\wedge_\iota^d = \frac{d}{dt} D(\wedge_\iota^d)$, étant l'ensemble des fonction continues sur $[0, T]$, continues par morceaux sur $[0, T]$ respectivement aux sous ensemble $]\tau_h, \tau_{h+1}]$ et qui disparaît quant $t = 0$.

Pour chaque $\iota \in n+1, \dots, 2n$, \wedge_ι^d est défini par $\wedge_\iota^d = -\frac{d}{dt} D(\wedge_\iota^d)$ étant l'ensemble des fonction continues sur $[0, T]$, continues par morceaux sur $[0, T]$ respectivement aux sous ensemble $]\tau_h, \tau_{h+1}]$ et qui disparaît quant $t = T$.

Pour chaque $\iota \in 2n+1, \dots, \beta$, considérons l'ensemble convexe U_{ad_ι} , et soit ΨU_{ad_ι} qui est la fonction indicatrice du convexe U_{ad_ι} et $\wedge_\iota^d = \Psi U_{ad_\iota}$, $D(\wedge_\iota^d)$ étant U_{ad_ι} ; plus précisément, \wedge_ι^d est défini par :

$$\wedge_\iota^d(x_\iota) = \begin{cases} 0, si & x_\iota(t) \in]x_\iota^m, x_\iota^M] \\]-\infty, 0], & si & x_\iota(t) = x_\iota^m \\]0, +\infty[, & si & x_\iota(t) = x_\iota^M \\ \emptyset, & Sino. \end{cases}$$

On peut alors écrire :

$$\wedge_\iota^d(X) + \wedge_\iota(X) \ni 0, \forall \iota \in 1, \dots, \beta \quad (3.13)$$

Comme précédemment, on peut associer une équation de point fixe pour le système (3.12) de la même façon que dans la relation (*), nous pouvons également définir un point de la méthode de relaxation asynchrone par une recurrence analogue à celle de la relation (3.8) (mais maintenant $i \in \{1, \dots, \beta\}$).

Lemme 3.6.1 (6). *Considérons l'équation différentielle ordinaire suivante :*

$$\frac{d}{dt} x(t) + \lambda x = z(t), \forall t \in [0, T], \lambda \in \mathbb{R}, x(0) = 0 \quad (3.14)$$

et posons les points suivants :

1. z est une fonction continue par morceaux ayant un nombre fini de point de discontinuité

représenté par : $\tau_2, \dots, \tau_{v-1}$

2. $z(t)$ est continue à la droite de chaque point $\tau_i, \forall i \in \{2, \dots, v-1\}$.

Puis ils existent $g \in G(x(t))$ telle que $\langle \frac{d}{dt}x(t), g \rangle \geq 0, \forall t \in [0, T]$.

Théorème 3.6.1 (6). *Si la matrice A , est une M-matrice, alors il existe une valeur positive k_0 , tel que pour tout k supérieur à k_0 , tout les points d'itérations asynchrones appliqués à la résolution du système (3.1) et débutant par x^0 , convergent vers l'unique solution de problème.*

Preuve

Soit x_ι^* , la solution exacte, et x_ι , la valeur de ι^{ieme} composant obtenue par l'algorithme, et soit également W les variables interaction entre la ι^{ieme} équation et les autres équations du système ; W est une fonction continue par morceaux ayant un nombre fini de points de discontinuité.

Nous pouvons écrire les équations vérifiées par x_ι et x_ι^* respectivement, puis soustrayons leurs relations et nous multiplions la différence par $g_\iota \in G_\iota(x_\iota - x_\iota^*)$.

Nous considérons alors trois cas :

Premier cas :

Pour $\iota \in 1, \dots, n$, on a $\langle \frac{d}{dt}(x_\iota - x_\iota^*), g_\iota \rangle + \langle \wedge_\iota(X) - \wedge_\iota(X^*), g_\iota \rangle = 0$
où $X = \{w_1, w_2, \dots, w_{\iota-1}, w_\iota, w_{\iota+1}, \dots, w_\beta\}$

mais, nous pouvons écrire :

$$\wedge_\iota(X) - \wedge_\iota(X^*) = \sum_{k=1}^{\beta} \{ \wedge_\iota(x_1^*, \dots, x_{k-1}^*, w_k, \dots, w_\beta) - \wedge_\iota(x_1^*, \dots, x_k^*, w_{k+1}, \dots, w_\beta) \}$$

et nous avons $\wedge_\iota(x_1^*, \dots, x_{k-1}^*, w_k, \dots, w_\beta) - \wedge_\iota(x_1^*, \dots, x_k^*, w_{k+1}, \dots, w_\beta) = a_{\iota,k}(w_k - w_k^*)$ Par conséquent

$$\langle \wedge_\iota(X) - \wedge_\iota(X^*), g_\iota \rangle = \sum_{k=1}^{\beta} a_{\iota,k}(w_k - w_k^*)$$

and

$$\langle \frac{d}{dt}(x_\iota - x_\iota^*), g_\iota \rangle + a_{\iota,\iota} \langle (x_\iota - x_\iota^*), g_\iota \rangle = - \sum_{k=1, k \neq \iota}^{\beta} a_{\iota,k} \langle w_k - x_k^*, g_\iota \rangle \quad (3.15)$$

A étant une M-matrice, par d'après le lemme précédent la partie gauche de la relation (3.15) est plus grand que $a_{\iota,\iota}|x_\iota - x_\iota^*|_\infty^2$; en outre, classiquement nous avons :

$$- \sum_{k=1, k \neq \iota}^{\beta} a_{\iota,k} \langle w_k - x_k^*, g_\iota \rangle \leq \sum_{k=1, k \neq \iota}^{\beta} |a_{\iota,k}| \|w_k - w_k^*\|_\infty \cdot \|g_\iota\|_*$$

et dû à $\|g_\iota\|_* = |x_\iota - x_\iota^*|_\infty$.

Alors

$$|x_\iota - x_\iota^*|_\infty \leq \sum_{k=1, k \neq \iota}^{\beta} j_{\iota, k} |w_k - x_k^*|_\infty, \forall \iota \in \{1, \dots, n\}$$

où : $j_{\iota, k} = -\frac{|a_{\iota, k}|}{a_{\iota, k}}$.

Deuxième cas Pour $\iota \in n+1, \dots, 2n$, par une transformation appropriée de la variable t , de type $\theta = T - t$, nous observons que ce cas est identique au cas précédent.

Troisième cas Pour $\iota \in 2n+1, \dots, \beta$ et par application des résultats du lemme, et par répétition de raisonnement précédent, on obtient le même type d'inégalité.

Nous pouvons alors écrire :

$$|x_\iota - x_\iota^*|_\infty \leq \sum_{k=1, k \neq \iota}^{\beta} J_{\iota, k} |w_k - x_k^*|_\infty, \forall \iota \in \{1, \dots, \beta\} \quad (3.16)$$

où J est la matrice jacobéenne associée à la matrice N définie par :

$$\begin{cases} n_{\iota, \iota} = a_{\iota, \iota}, & \forall \iota \in 1, \dots, \beta \\ n_{\iota, k} = -|a_{\iota, k}|, & \forall \iota, k \in 1, \dots, \beta, \iota \neq k. \end{cases} \quad (3.17)$$

3.7 Méthode de gradient conjugué

En analyse mathématique, la méthode de gradient conjugué est un algorithme pour résoudre des systèmes d'équations dans la matrice est symétrique et définie positive. Cette méthode imaginée simultanément par **Cornelius Lanczos** et **Mangus Hestnes**, est une méthode itérative convergente en nombre fini d'itérations, cependant grand intérêt pratique du point de vue du temps de calcul vient de ce qu'une initialisation astucieuse dite **préconditionnement** permet d'accélérer le processus de convergence vers la solution exacte, et c'est pour quoi en pratique on se limite à un nombre d'itérations inférieur au nombre d'inconnus.

3.7.1 Principe de la méthode

L'objectif est de minimiser la fonction $f : x \mapsto \frac{1}{2}(Ax, x) - (b, x)$

où A est une matrice carrée symétrique définie positive de taille n .

Le calcul nous montre qu'une solution du problème est la solution du système $Ax = b$: en effet on a : $\nabla f(x) = Ax - b$.

La méthode consiste donc à construire par récurrence une base de vecteurs de \mathbb{R}^n orthogonaux pour le produit scalaire $(x, y) \rightarrow (Ax, y)$, et exprimer le vecteur solution dans cette base.

Il faut donc bien partir d'une estimation initiale x_0 du vecteur solution à fin d'amorcer la récurrence.

On peut alors dire que " tout les coups sont permis" pour obtenir une estimation initiale qui accélère la convergence ; néanmoins, les méthodes de préconditionnement à la fois sur et générales, c'est-à-dire efficace pour toute sorte de matrice symétrique définie positive pour former un x_0 correct soit aussi elle-même coûteuse en terme de temps de calcul.

En pratique l'intuition physique, guidée par la nature physique du problème à résoudre, suggère parfois une initialisation efficace : ces idées ont donné lieu à une littérature spécialisée abondante.

Théorème 3.7.1. *Soit A , une matrice symétrique définie positive d'ordre n . Tout méthode qui utilise des directions conjuguées, conduit à la solution exacte en au plus n itérations.*

Preuve :

En effet les directions d^1, d^2, \dots, d^{n-1} , forment une base A -orthogonale de \mathbb{R}^n de plus puisque x^k est optimale par rapport à toutes les directions $d^j, j = 1, \dots, k - 1$, le vecteur r^k est orthogonale à l'espace $S_{k-1} = \text{vect}\{d^1, \dots, d^{k-1}\}$, par conséquent $r^k \perp S_{k-1} = \mathbb{R}^n$ et donc $r^n = 0$, ce qui implique que : $x^k = x$, x étant la solution du problème.

3.8 Algorithme de gradient conjugué

Soit $u_0, t \in [0, T]$, une approximation initiale de la commande, la méthode de gradient conjugué se décompose en deux parties : une partie dévolue à l'initialisation de l'algorithme et partie itérative constituée d'un corps de boucle.

Initialisation de l'algorithme

calcul de $q^0(t) = \nabla J(u^0(t)) = \text{grad}J(u^0(t)), t \in [0, T]$ par résolution des équations suivantes

$$\begin{cases} \frac{dy^{(0)}}{dt}(t) = Ay^{(0)}(t) + Bu^{(0)} + f(t), t \in [0, T] \\ y^{(0)}(0) = 0 \end{cases}$$

$$-\frac{dp^{(0)}}{dt}(t) = Ap^{(0)}(t) + C^t.Cy^{(0)} - C^t z_d, t \in [0, T] p^{(0)}(T) = 0$$

puis

$$q^{(0)} = G^{(0)} = u^{(0)} + \frac{\alpha}{\beta} B^t p^{(0)} - u_d, t \in [0, T]$$

ainsi que

$$\begin{aligned} \frac{d\hat{y}^{(0)}}{dt}(t) &= A\hat{y}^{(0)}(t) + Bq^{(0)} + f(t), t \in [0, T] \hat{y}^{(0)}(0) = 0 \\ -\frac{d\hat{p}^{(0)}}{dt}(t) &= A\hat{p}^{(0)}(t) + C^t \cdot C\hat{y}^{(0)} - C^t z_d, t \in [0, T] \hat{p}^{(0)}(T) = 0 \end{aligned}$$

$$Aq^{(0)} = q^{(0)} + \frac{\alpha}{\beta} B^t \hat{p}^{(0)}, t \in [0, T] \text{ et}$$

$$\theta_0 = \frac{\langle G^0, q^0 \rangle}{\langle Aq^0, q^0 \rangle}$$

Corps de boucle

Etape1 Détermination de l'état $y^{r+1}(t)$ par intégration numérique de l'équation d'état :

$$\frac{dy^{(r+1)}}{dt}(t) = Ay^{(r+1)}(t) + Bu^{(r+1)} + f(t), t \in [0, T] y^{(r+1)}(0) = y_0$$

Etape2 Détermination de l'état adjoint $p^{r+1}(t)$ par intégration numérique dans le sens rétrograde de l'équation d'état adjoint :

$$-\frac{dp^{(r+1)}}{dt}(t) = Ap^{(r+1)}(t) + C^t \cdot Cy^{(r+1)} - C^t z_d, t \in [0, T] p^{(r+1)}(T) = 0$$

Soit

$$G^{(r+1)} = u^{(r)} + \frac{\alpha}{\beta} p^{(r+1)} - u_d, t \in [0, T]$$

et

$$\vartheta_r = \frac{\|G^r\|^2}{\|G^{r-1}\|^2}; q^r = G^r - \vartheta_r q^{r-1}$$

Etape3 Détermination de \hat{y}^{r+1} et \hat{p}^{r+1} par intégration de

$$\frac{d\hat{y}^{(r+1)}}{dt}(t) = A\hat{y}^{(r+1)}(t) + Bq^{(r)}, t \in [0, T] \hat{y}^{(r+1)}(0) = 0$$

$$-\frac{d\hat{p}^{(r+1)}}{dt}(t) = A\hat{p}^{(r+1)}(t) + C^t \cdot C\hat{y}^{(r+1)}, t \in [0, T] \hat{p}^{(r+1)}(T) = 0$$

$$Aq^{(r)} = q^{(r)} + \frac{\alpha}{\beta} B^t \hat{p}^{(r+1)}, t \in [0, T] \text{ et}$$

$$\theta_r = \frac{\langle G^r, q^r \rangle}{\langle Aq^r, q^r \rangle}$$

Etape4 Détermination de la commande par la méthode de gradient conjugué

$$u^{r+1} = Proj_{u_{ad}}(u^r - \theta_r q^r), t \in [0, T]$$

Remarque 3.1. L'implantation des algorithmes précédents, exige la résolution d'une équation différentielle ordinaire, la méthode de **Runge-Kutta** d'ordre k , figure parmi les méthodes les plus répandues dans la résolutions des équations différentielles ordinaires. Cette méthode est formulée comme suit :

Soit l'équation différentielle ordinaire suivante :

$$\frac{dy}{dt} = f(y, t), t \in [t_0, T] y(t_0) = y_0 \quad (3.18)$$

On discrétise le temps en les points $t_i = t_0 + i \cdot dt$, où dt est le pas de temps et on approxime $y(t_{i+1})$ par la suite récurrente y_{i+1} calculée comme suit

$$\begin{cases} K_1 = h \cdot f(y_i, t_i) \\ K_2 = h \cdot f(y_i + K_1/2, t_{i+1/2}) \\ k_3 = h \cdot f(y_i + k_2/2, t_i + 1/2) \\ k_4 = h \cdot f(y_i + k_3, t_{i+1}) \end{cases} y_{i+1} = y_i + \frac{1}{6}(K_1 + 2 \cdot K_2 + 2 \cdot k_3 + k_4)$$

3.8.1 Point faible de la méthode du gradient conjugué

-Les gradients en x et \tilde{x} sont orthogonaux.

-A chaque pas on prend une direction orthogonale à la direction précédente cheminement de la méthode en "en zigzag".

-lorsque les contraintes sur la commande sont saturées où dans ce cas particulier les performances de la méthode du gradient conjugué s'effondrent.

Conclusion

Tout au long de ce chapitre, Nous avons exposé les points les plus importants concernant les deux méthodes de résolution développées dans ce travail, et nous avons réalisé à quel point la méthode de relaxation est avantageuse par rapport à la méthode de gradient conjugué.

Chapitre 4

Application : Commande optimale pour la régulation d'un processus thermique

Introduction

Cet étude à pour objet la détermination de loi de commande optimale pour la régulation de processus thermique de grande dimension. Divers algorithmes de résolution de problèmes de commande optimale, tels que l'algorithme de relaxation et du gradient conjugué sont représentés et implémentés sous matlab. En fin une comparaison de ces deux méthodes est proposée.

4.1 But de l'étude

Le problème consiste à amener un four à 3 zones de chauffage à une température voisine de $z_d \in \mathbb{R}^n$ en une période de temps finie T.

4.2 Modélisation et résolution

Le modèle est formalisé par le système suivant :

$$\begin{cases} \dot{y} + Ay = Bu; y_0 = 0, & (1) \\ z = cy, & (2) \end{cases}$$

où :

z : représente le vecteur de sortie.

y : représente le vecteur d'état.

u : représente le contrôle.

A : est une M-matrice.

En outre on suppose que $t \in [0, T]$, $y(t) \in \mathbb{R}^n$, et n entier naturel.

L'objectif principale est alors de minimiser la quantité représenté par la formule quadratique suivante :

$$J = \frac{1}{2} \int_0^T \left(\frac{|z - z_d|_2^2}{|z_d|_2^2} + k \frac{|u - u_d|_2^2}{|u_d|_2^2} \right) dt$$

Telle que le paramètre k contrôle le poids comparatif donné aux deux composantes de la fonction coût : exactitude (précision) et l'énergie dépensée.

Le terme u_d est l'entrée stable qui doit être appliquée pour obtenir la sortie z_d désiré par une durée finie.

D'autre part la composante de vecteur de contrôle doit vérifier les contraintes suivantes :

$$u_i^m \leq u_i \leq u_i^M, \forall i \in 1, \dots, m, t \in [0, T].$$

et dans ce qui suit, u_{ad} représente l'ensemble des contrôles admissibles. Le problème est alors de déterminer $u \in U_{ad}$, telle que :

$$J(u) = \text{Min}_{v \in U_{ad}} J(v) \quad (3)$$

sous les contraintes supplémentaires :

$$\begin{cases} \frac{\partial y}{\partial t} + Ay = Bu, y(0) = 0 \\ z = cy \end{cases}$$

4.3 Forme canonique du problème

Par la méthode de l'état adjoint, on sait classiquement que la solution du problème (3) est caractérisée par :

$$\frac{\partial y}{\partial t} + Ay = Bu, t \in [0, T]; y(0) = 0 \quad (4)$$

$$-\frac{\partial p}{\partial t} + A^t p = c^t cy - c^t z_d; p(T) = 0 \quad (5)$$

$$B^t p + k(u - u_d \partial \psi_{u_{ad}}) \ni 0$$

où $p(t)$: est l'état adjoint.

$\partial \psi_{u_{ad}}$: est le sous différentiel de la fonction indicatrice $\psi_{u_{ad}}$ du convexe U_{ad} .

4.4 Résolution

Plusieurs méthodes peuvent être candidates à résoudre ce type de problèmes, cependant dans notre étude on va se contenter de deux méthodes itératives, qui sont :

- La méthode de relaxation.
- La méthode de gradient conjugué.

application numérique

Les données suivantes sont extraites à partir l'article publié par le professeur SPITERI, P. ; LANG, B. ; et MIELLOU, J.C, N° Originale : Gs 346762, N° INIST : 10973693 voir [6] ;

```
A= [0.030 -0.013 -0.0077 -0.00071 -0.00017 -0.00065;
    -0.00017 0.012 -0.00009 -0.00033 -0.00008 -0.00029;
    -0.0075 0 0.040 -0.016 -0.0077 -0.00073;
    0 -0.0030 -0.0019 0.014 -0.00009 -0.0033;
    0 0 -0.0075 -0.0030 0.029 -0.012;
    0 0 0 0 -0.0014 0.013];
```

```
B'= [0.00125 0 0 0 0 0;
    0 0 0.00125 0 0 0;
    0 0 0 0 0.00125 0];
```

```
C= [0 1 0 0 0 0;
    0 0 0 0 1 0;
    0 0 0 0 0 1];
```

tel que :

A : Matrice carré de taille $2n$ (n étant la dimension du four).

C : Une matrice rectangulaire de taille $n * 2n$ (dite d'observation).

B : Une matrice rectangulaire de taille $2n * n$

4.4.1 Simulation de l'évolution de la température dans un four thermique

```

%*****
% Etude de l'evolution de température *
% dans les parois d'un four *
% Résolution d'un système linéaire par *
% la méthode itérative de Gradient conjugué *
%*****
tic; flops(0); clear all; clc;clf; eps=1e-4;k1=300;
% Données initiales
L1=input ('Introduire la valeur de L1 :\n');

L2=input ('Introduire la valeur de L2 :\n');

L3=input ('Introduire la valeur de L3 :\n'); L4=input ('Introduire
la valeur de L4 :\n');

dx=input ('Introduire la valeur du pas dx :\n');

Tfinale=input ('Introduire la valeur de la température finale
:\n'); Tinit=input ('Introduire la valeur de la temperature
initiale :\n'); gama=input('Introduire la valeur
beta/alpha=gama=')
% Calcul des indices
m=round(L1/dx)+1; n=round(L2/dx)+1; m1=round((L1-L3)/(2*dx))+1;
n1=round((L2-L4)/(2*dx))+1; m2=m1+round(L3/dx);
n2=n1+round(L4/dx);
% Initialisation de la température dans le four
for i=1:n for j=1:m T(i,j)=Tfinale; end end
for i=1:n
T(i,1)=Tinit;

```

```

T(i,m)=Tinit;
end
for j=1:m
T(1,j)=Tinit;
T(n,j)=Tinit;
end
% Température de la paroi interne
for i=n1:n2 T(i,m1)=Tfinale; T(i,m2)=Tfinale; end for j=m1:m2
T(n1,j)=Tfinale; T(n2,j)=Tfinale;
end
% Méthode de Gradient conjugué (Itérations)
for k=1:k1 for i=2:n-1 for j=2:m1-1
T(i,j)=gama*(T(i-1,j)+T(i+1,j)+T(i,j-1)+T(i,j+1));
end
end

for i=2:n-1 for j=m2+1:m-1
T(i,j)=gama*(T(i-1,j)+T(i+1,j)+T(i,j-1)+T(i,j+1));
end
end

for i=2:n1-1 for j=m1:m2
T(i,j)=gama*(T(i-1,j)+T(i+1,j)+T(i,j-1)+T(i,j+1));
end
end

for i=n2+1:n-1 for j=m1:m2
T(i,j)=gama*(T(i-1,j)+T(i+1,j)+T(i,j-1)+T(i,j+1));
end
end

if abs(T(n-1,m-1)-T(2,2))<=eps fprintf('\n \n');
fprintf('Températures après "%d itérations\n',k);
fprintf('\n \n'); break; end end for i=1:n

```

```
fprintf('%5.0f\t',T(i,1:m));
fprintf('\n'); end
% Tracé des courbes de température en 3D
hold off;

if n==m figure(1); i=1:n; j=1:m;
grid on;
[x,y]=meshgrid((i-1)*dx,(j-1)*dx);
mesh(x,y,T);

Title('Evolution de la température dans le four');
xlabel('i');
ylabel('j');

zlabel('T (en °C)');
end
Tracé des courbes isothermes en 2D
figure(2);
i=1:n;j=1:m;
grid on;
contour(i,j,T(i,j),15);
title('Lignes isothermes dans les parois du four');
xlabel('i');ylabel('j');
t_mis=toc
Nb_opt=flops
```

Le résultat d'exécution de l'algorithme qui conduit le système à une température finale de $1200C^{\circ}$ à partir d'une température initiale de $0C^{\circ}$ est donné par :

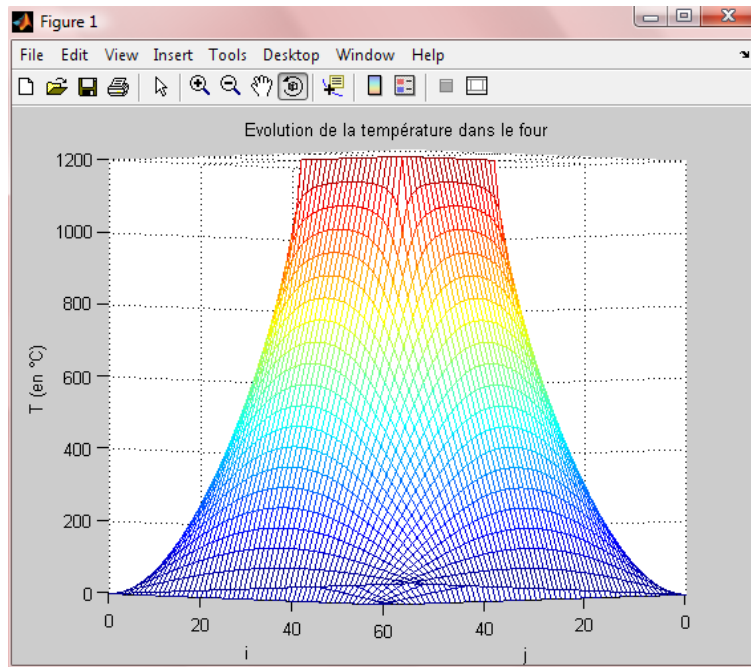


FIG. 4.1: Evolution de la température dans le four

Temps de calcul pour des différentes valeurs de pas de discrétisation dx

dx	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
Temps	23.86	17.84	14.59	11.23	14.54	11.92	12.28	12.98	12.99	18.99

On remarque bien que le bon choix de pas de discrétisation est primordiale, du fait de son répercussion directe sur le temps de calcul.

4.4.2 Algorithme de relaxation

L'algorithme de relaxation codé sous matlab est :

```
function[u]=relax(A,B,F); clear all;clc; clf;
%Résolution de l'equation d'etat par la méthode RK4;
h=0.4; y(:,1)=[0 0 0 0 0 0]';t(1)=0;i=1;
A= [0.030 -0.013 -0.0077 -0.00071 -0.00017 -0.00065;
-0.00017 0.012 -0.00009 -0.00033 -0.00008 -0.00029;
-0.0075 0 0.040 -0.016 -0.0077 -0.00073;
```

```

0 -0.0030 -0.0019 0.014 -0.00009 -0.0033;

0 0 -0.0075 -0.0030 0.029 -0.012;

0 0 0 0 -0.0014 0.013];
B=[0.00125 0 0 0 0 0;
0 0 0.00125 0 0 0; 0
0 0 0 0.00125 0];
u0=[1;1;1]; C= [0 1 0 0 0 0;
0 0 0 0 1 0;

0 0 0 0 0 1];
zd=[200;500;1200];
gama=0.25;
p(:,1)=[-1;-0.0026;-0.12;0.0025;0.36;0.25];
i=1;
while t<=30
U=u0-gama*B*p;
F=B'*U;
k1=h*(A*y(:,i)+F);
k2=h*(A*y(:,i)+k1/2+F);
k3=h*(A*y(:,i)+k2/2+F);
k4=h*(A*y(:,i)+k3+F);
y(:,i+1)=y(:,i)+(1/6)*(k1+2*k2+2*k3+k4);
t(i+1)=i*h;
u=gama*y;
i=i+1;
end
figure;
plot(t,y(1:3,:));
grid on;
text(t(30),y(1,30),'y1');
text(t(30),y(2,30),'y2');
text(t(30),y(3,30),'y3');

```

```

xlabel('Temps
(en s)'); ylabel('Solutions,y1, y2, y3');
figure;
plot(t,u(1:3,:));
grid off;
text(t(30),u(1,30),'u1');
text(t(30),u(2,30),'u2');
text(t(30),u(3,30),'u3'); xlabel('Temps
(en s)'); ylabel('Commandes,u1, u2, u3');

end

function [p]=adj(A,C,zd,y);
%Résolution de l'équation d'état adjoint par la méthode RK4;
clear all; clc; clf;

j=n;
while t<=30
U=u0-gama*B*y; k1=h*(A*p(:,j)+C'*C*y-C'*zd);
k2=h*(A*p(:,j)+k1/2+C'*C*y-C'*zd);
k3=h*(A*p(:,j)+k2/2+C'*C*y-C'*zd);
k4=h*(A*p(:,j)+k3+C'*C*y-C'*zd);
p(:,j+1)=p(:,j)+(1/6)*(k1+2*k2+2*k3+k4);
t(j+1)=j*h;
j=j-1;
p(1:3,30)=0;
end figure;
plot(t,p(1:3,:));
grid on;
text(t(30),p(1,30),'p1');
text(t(30),p(2,30),'p2');
text(t(30),p(3,30),'p3');

```

```

xlabel('Temps (en s)');
ylabel('Solutions,p1, p2, p3');
end
tic
t_mis=toc

```

La figure des commandes est donnée par :

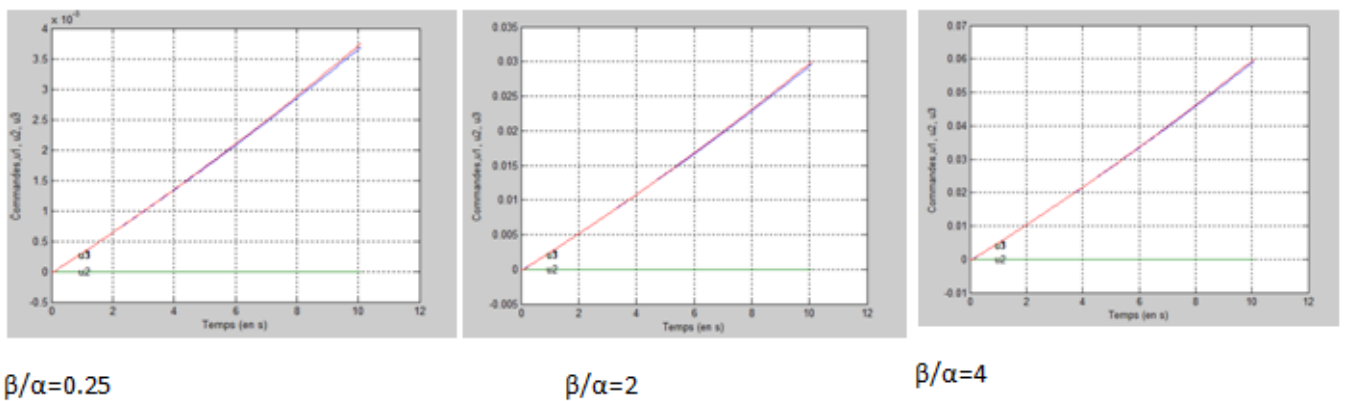


FIG. 4.2: Graphe de la commande

4.4.3 Algorithme de gradient conjugué

Le code de programme matlab de l'algorithme de gradient conjugué :

```

function[u]=gradconj(A,B,F);
clear all; clc; clf; h=0.1;

y(:,1)=[0 0 0 0 0 0]';t(1)=0;i=1;h=0.4;
A= [0.030 -0.013 -0.0077 -0.00071 -0.00017 -0.00065;
-0.00017 0.012 -0.00009 -0.00033 -0.00008 -0.00029;

-0.0075 0 0.040 -0.016 -0.0077 -0.00073;

0 -0.0030 -0.0019 0.014 -0.00009 -0.0033;

```

```

0 0 -0.0075 -0.0030 0.029 -0.012;
0 0 0 0 -0.0014 0.013];
B=[0.00125 0 0 0 0 0;
0 0 0.00125 0 0 0;
0 0 0 0 0.00125 0];
u0=[1;1;1];
C= [0 1 0 0 0 0;
0 0 0 0 1 0;
0 0 0 0 0 1];
zd=[200;500;1200];
gama=input('Donner la valeur gama=');
p(:,1)=[-1;-0.0026;-0.12;0.0025;0.36;0.25];
i=1;
while t<=30
U=u0-gama*B*p;
F=B'*U;
k1=h*(A*y(:,i)+F);
k2=h*(A*y(:,i)+k1/2+F);
k3=h*(A*y(:,i)+k2/2+F);
k4=h*(A*y(:,i)+k3+F);
y(:,i+1)=y(:,i)+(1/6)*(k1+2*k2+2*k3+k4);
t(i+1)=i*h;
i=i+1;
end
u=gama*y;
figure;
plot(t,y(1:3,:));
grid on;
text(t(30),y(1,30),'y1');
text(t(30),y(2,30),'y2');
text(t(30),y(3,30),'y3');
xlabel('Temps (en s)');
ylabel('Solutions,y1, y2, y3');
figure; plot(t,u(1:3,:));

```

```

grid on;
text(t(30),u(1,30),'u1');
text(t(30),u(2,30),'u2');
text(t(30),u(3,30),'u3');
    xlabel('Temps (en s)');
ylabel('Commandes,u1, u2, u3');
tic toc
end

```

4.4.4 Résultats d'exécution

Les résultats d'exécution des deux algorithmes sous matlab sont donnés par le tableau suivant :

	Algorithme de relaxation			Algorithme de gradient conjugué		
Le rapport $\beta/\alpha=\text{gama}$.	0.25	2	4	0.25	2	4
Le temps de calcul en seconde	6.2731	5.7920	4.2685	6.5705	10.0569	18.4397

FIG. 4.3: Comparaison entre les deux algorithmes

4.4.5 Comparaison

Ces résultats montrent clairement que la méthode de relaxation converge en un temps inférieur au temps de convergence de l'algorithme de gradient conjugué, et on remarque aussi que ces résultats sont sensibles aux valeurs de $\frac{\beta}{\alpha}$

Conclusion

La méthode de relaxation est plus performante et moins sensible aux valeurs de rapport $\frac{\beta}{\alpha}$ que celle de gradient conjugué.

Conclusion générale

Nous nous sommes, tout au long de ce travail appesantis sur Les méthodes de résolution de problèmes de commande optimale.

L'objectif ici, à été de présenter deux méthodes différentes de résolution, et de faire une étude comparative entre elles dans le but de comparer leurs performances.

Nous avons exposé une méthode de relaxation séquentiel qui est relativement simple à programmer et nécessite peu de place mémoire; c'est une méthode particulièrement robuste, et plus efficace lorsque les interactions entre les divers composants sont faibles, ce qui ne semble pas être le cas de la méthode de directions conjuguées, qui compte à elle aussi présente une certaine souplesse lors de sa mise en pratique, s'avère efficace dans le cas sans contraintes, et nettement moins performantes lorsque les contraintes sont saturées.

De plus, nous avons exposé un concept d'algorithmes parallèles se basant essentiellement sur le principe d'une exécution simultanée des instances du programme, avec ces deux variantes, synchrone et asynchrone. Cependant faute d'un dispositif matériel permettant la mise en oeuvre pratique d'un tel algorithme, nous nous sommes contentés de définir ces deux structures sans les implémentées.

Ce travail n'est qu'une ébauche d'un sujet qui peut donnée lieu à des perspectives de recherches, tels que les systèmes singulièrement perturbés, correspondants à des problèmes à plusieurs échelles de temps, ainsi qu'à des problèmes multi-critères.

Annexe

On rapelle ci-dessous des définitions et résultats classique voire [6]

Définition 4.4.1. A est une M-matrice si A est inversible, $A^{-1} \geq 0$ et $a_{i,j} \leq 0$ dès que $i \neq j$.

Théorème 4.4.1. \square

Soit A une matrice telle que : $a_{i,j} \leq 0$ dès que $i \neq j$. A est une M-matrice ssi :

- (1). *Les éléments diagonaux $a_{i,i}$ sont strictement positifs.*
- (2). *La matrice de jacobi $J = I - D^{-1}A$, D diagonale de A , est t.q : $\rho(J) < 1$ (rayon spectral de J).*

Théorème 4.4.2. *Soit A une M-matrice et Δ une matrice diagonale non négative. Alors $(A + \Delta)$ est une M-matrice et $(A + \Delta)^{-1} \leq A^{-1}$*

Définition 4.4.2. Une matrice A est diagonale fortement dominante si :

$$|a_{i,i}| \geq \sum_{j \neq i} |a_{i,j}|, \forall i \in \{1, \dots, \dim(A)\}$$

(A est diagonale dominante) et s'il existe au moins un indice K t.q.

$$|a_{k,k}| > \sum_{j \neq k} |a_{k,j}|$$

.

Définition 4.4.3. Soit A une matrice réelle ou complexe. A est réductible, s'il existe une matrice de permutation P , de même dimension, telle que :

$$P^t A P = \begin{pmatrix} B_{1,1} & B_{1,2} \\ 0 & B_{2,2} \end{pmatrix}$$

où $B_{i,i}, i = 1, 2$, sont carrées. Si une matrice n'est pas réductible, elle dite : irréductible.

Lemme 4.4.1. *Une C.N.S pour qu'une matrice A soit irréductible est que pour tout couple d'indices $(i, j), j \neq i$, il existe au moins un ensemble d'indices $i_1, i_2, \dots, i_k = j$, avec $k \geq 1$, tels que les éléments $a_{i,i_1}, a_{i,i_2}, \dots, a_{i_k,j}$ soient tous différents de zéro.*

Définition 4.4.4. Une matrice A est diagonale dominante irréductible, si A est irréductible et diagonale fortement dominante.

Théorème 4.4.3. *Une matrice strictement diagonale dominante ou diagonale dominante irréductible est inversible.*

Théorème 4.4.4. *Si A est une M -matrice alors la partie réelle de ces valeurs propres est positive.*

Topologie des espaces vectoriels normés

Normes sur un espace vectoriel

Soit E un espace vectoriel dans \mathbb{R} , on a les définitions suivantes :

Définition 4.4.5. Une norme est une application définie sur E à valeurs dans \mathbb{R}_+ , notée $\|\cdot\|_E$, et satisfaisant les trois propriétés suivantes :

- (i) $\|e\|_E = 0 \Leftrightarrow e = 0$,
- (ii) $\forall \lambda \in \mathbb{R}, \forall e \in E, \|\lambda e\|_E = |\lambda| \|e\|_E$,
- (iii) $\forall e, w \in E, \|e + w\|_E \leq \|e\|_E + \|w\|_E$ (inégalité triangulaire.)

Exemple

a – Soit $[a, b]$ un intervalle de \mathbb{R} et notons $\mathcal{C}^0([a, b])$ l'espace vectoriel constitué des fonctions continues sur $[a, b]$ et à valeurs dans \mathbb{R} . L'application $\|\cdot\|_{\mathcal{C}^0([a, b])}$ qui à $f \in \mathcal{C}^0([a, b])$ associe

$$\|f\|_{\mathcal{C}^0([a, b])} = \sup_{t \in [a, b]} |f(t)|$$

définit une norme sur $\mathcal{C}^0([a, b])$.

b– Sur $\mathcal{C}^0([a, b])$, on peut aussi définir une autre norme,

$$\|f\|_R = \int_a^b |f(t)| dt$$

On peut donc avoir plusieurs normes sur le même espace.

Quelques rappels

Définition 4.4.6. Soit K une partie de E et soit y un élément de E . On dit que : y adhérent à K si tout voisinage V de y dans E contient un point de K .

Définition 4.4.7. Un ensemble $U \subset E$ est ouvert dans E si $\forall x \in U, \exists \varepsilon$ tel que $B(x, \varepsilon) \subset U$. Avec $B(x, \varepsilon) = \{y \in E; \|x - y\| < \varepsilon\}$ boule ouverte de centre x et de rayon ε .

Définition 4.4.8. Un ensemble $\mathcal{F} \subset E$ est fermé dans E si $E \setminus \mathcal{F}$ est ouvert dans E .

Définition 4.4.9. Soit u_n une suite de E . On dit que u_n converge vers $l \in E$ si

$$\forall \varepsilon > 0, \exists N \in \mathbb{N}, \forall n \geq N \|u_n - l\|_E \leq \varepsilon.$$

Définition 4.4.10. Soit \mathcal{F} un fermé de E . Si une suite $x_n \in \mathcal{F}$ converge vers l , alors $l \in \mathcal{F}$.

Equivalence de normes

On donne la définition suivante

Définition 4.4.11. Soit $\|\cdot\|_{E,1}$ et $\|\cdot\|_{E,2}$ deux normes sur E . On dit que ces deux normes sont équivalentes s'il existe c_1 et c_2 strictement positives telles que

$$c_1 \|e\|_{E,1} \leq \|e\|_{E,2} \leq c_2 \|e\|_{E,1}$$

Compacité

Définition 4.4.12. Un ensemble $A \subset E$ est dit compact si de toute suite d'éléments de A , on peut extraire une sous suite convergente vers un élément de A .

Proposition 4.4.1. Si A est compact, alors A est fermé borné.

Proposition 4.4.2. Si E est de dimension finie, alors A est compact si et seulement si A est fermé borné.

Continuité de la norme

Proposition 4.4.3. *Soit E un espace vectoriel normé. La norme sur E est une application continue, autrement dit la fonction :*

$$\phi : e \in E \mapsto \|e\|_E \in \mathbb{R}$$

est continue.

En effet on a $|\phi(e) - \phi(w)| \leq |w - e|_E$ et ϕ est lipschitzienne donc continue.

Espaces complets

Soit E un espace vectoriel normé, de norme $\|\cdot\|_E$

On définit une suite de **Cauchy** comme suit

Définition 4.4.13. Soit u_n une suite de E . On dit qu' u_n est une suite de Cauchy si

$$\forall \varepsilon > 0, \exists n \geq N, \forall p \geq 0, \|u_{n+p} - u_n\| \leq \varepsilon$$

Proposition 4.4.4. *Toute suite convergente est de Cauchy.*

Espace de Banach

Définition 4.4.14. Un \mathbb{R} -espace vectoriel E est dit complet pour la norme $\|\cdot\|_E$ si toute suite de Cauchy (pour cette norme) est convergente (pour cette norme). Un tel espace est aussi appelé espace de **Banach**

Proposition 4.4.5. *Tout espace vectoriel sur \mathbb{R} normé de dimension finie est complet.*

Théoreme de point fixe de Picard

Définition 4.4.15. Soit A une application de E dans E . On dit que A est contractante s'il existe un réel $\alpha < 1$ tel que

$$\forall u, w \in E \|A(u) - A(w)\|_E \leq \alpha \|u - w\|_E$$

Théorème 4.4.5. *Soit E un espace de Banach et A une application contractante de E dans E . Alors, l'équation*

$$A(u) = u$$

admet une solution unique $u_ \in E$. Cette solution est appelée point fixe de A .*

Dualité

Applications linéaires continues

Dans cette section, E et W désignent deux \mathbb{R} - ev normés.

Proposition 4.4.6. *Soit A une application linéaire de E dans W . Les propositions suivantes sont équivalentes :*

- (1) A est continue.
- (2) A est continue en 0.
- (3) Il existe une constante $c < \infty$ telle que

$$\forall u \in E, \|Au\|_W \leq c\|u\|_E.$$

Définition 4.4.16. L'espace vectoriel des applications linéaires continues de E dans W est noté $\mathcal{L}(E, W)$.

Espace dual

Un cas particulier important d'applications linéaires entre \mathbb{R} - ev normés est celui où l'espace d'arrivée est $W = \mathbb{R}$.

Définition 4.4.17. Soit E un \mathbb{R} - ev normé. $\mathcal{L}(E, \mathbb{R})$ est appelé espace dual de E et est noté E' . Un élément $A \in E'$ est appelé forme linéaire continue et son action sur un élément $e \in E$ est notée à l'aide du crochet de dualité $\langle \cdot, \cdot \rangle_{E, E'}$ de sorte que

$$\langle A, v \rangle = Av \in \mathbb{R}$$

E' est équipé de la norme

$$\|A\|_{E'} = \sup_{u \in E, u \neq 0} \frac{\langle A, u \rangle_{E, E'}}{\|u\|_E}.$$

Bibliographie

Bibliographie

- [1] Abou-Kandil.H, G.Freiling, V. Ionescu, G. Jank, *Matrix Riccati equation, Contrôl and systems theory, Systems and control : Fondation and application*, Birkhäuser Verlag, Basel, 2003.
- [2] C. Zuily and H. Queffelec, *Eléments d'analyse pour l'agrégation*, Masson, 1996.
- [3] Emmanuel. Trélat, *Control optimal : théorie et applications*, Vuibert, collection, "Mathématique concrètes", 2005.

- [4] Francis. N, Dragos. I, *Introduction à la topologie*, Université de Rennes I.
- [5] Jean.L.M, *Analyse numérique avec matlab* Dunod, Paris, 2007.
- [6] Lang. B, Miellou, J.C, Spiteri. P, *Asynchronous algorithms for optimal control problems, Mathematic and computers in simulation, 28, 227-242*, 1986.
- [7] MERAKRB. A, *Optimisation multicritère en contrôle optimal : Application au véhicule électrique*, thèse de doctorat, UMMTO, 2011.
- [8] S. Gonnord and N. Tosel. *Topologie et analyse fonctionnelle : Thèmes d'analyse pour l'agrégation*. Ellipses, 1998.

Bibliographie