

LEROY Sébastien



Master 2 Recherche Système Dynamique et Signaux

**COMMANDE OPTIMALE
SELON LE PRINCIPE DU MAXIMUM
PAR ALGORITHME GÉNÉTIQUE**

SOMMAIRE

<u>I Introduction</u>	p.4
<u>II Analyse de la problématique</u>	p.5
<u>2.1 Principe du minimum de Pontryagin</u>	p.5
<u>2.2 Fonctionnement des algorithmes génétiques</u>	p.6
<u>2.3 Le système étudié</u>	p.7
2.3.1 Présentation du système	p.7
2.3.2 Modélisation du système	p.8
2.3.3 Expression de l'Hamiltonien.....	p.9
<u>2.4 Méthodologie générale</u>	p.11
<u>III Recherche d'un critère d'arrêt</u>	p.12
<u>3.1 Idée générale</u>	p.12
<u>3.2 Calcul du temps de parcours en ligne droite</u>	p.13
<u>3.3 Calcul du temps nécessaire pour la rotation</u>	p.13
<u>3.4 Calcul d'un majorant du temps optimal</u>	p.16

<u>IV Optimisation d'une fonction quadratique</u>	p.17
4.1 <i>Le problème</i>	p.17
4.2 <i>Simplification du problème</i>	p.17
4.3 <i>Résolution du problème</i>	p.18
4.4 <i>Retour au problème d'origine</i>	p.18
4.5 <i>Résultats préliminaires sur l'optimisation</i>	p.19
4.6 <i>Recherche du paramètre h optimal</i>	p.22
4.7 <i>Application à la recherche d'optimum sous contrainte linéaire</i>	p.24
<u>V Problème de localisation des adjoints initiaux</u>	p.25
5.1 <i>Cas où seul le temps est optimisé</i>	p.25
5.2 <i>Cas où le critère est une fonction quadratique de la commande</i>	p.25
5.2.1 <i>Idee générale</i>	p.25
5.2.2 <i>Recherche du domaine générateur</i>	p.26
5.2.3 <i>Prise en compte de l'évolution de l'adjoint</i>	p.29
5.2.4 <i>Résultats obtenus</i>	p.31
<u>VI Conclusion</u>	p.32
<u>Bibliographie</u>	p.34
<u>Remerciement</u>	p.34

I Introduction

Le stage se déroule à l'ONERA (Office National d'Etude et de Recherche Aérospatiale), dans le département Prospective et Synthèse (DPRS).

Pour un système dynamique donné par une équation d'état du type $\begin{cases} \dot{x} = f(x, t, u) \\ u \in U \end{cases}$, le problème posé consiste à déterminer la commande u qui rendra la trajectoire de x optimale selon des critères de temps et/ou de consommation.

En effet, la connaissance de la commande optimale fournit des indications précieuses sur les performances que l'on peut attendre du système et permet d'évaluer les résultats obtenus par le système réel.

Par ailleurs, les algorithmes génétiques suscitent beaucoup d'intérêt au DPRS car ils permettent d'obtenir de façon efficace (mais sans garantie de l'atteindre¹) un optimum global, contrairement à d'autres méthodes plus conventionnelles qui ne fournissent généralement qu'un optimum local.

Ainsi, l'objectif de ce projet est de mettre en place une méthodologie permettant d'appliquer les algorithmes génétiques à la recherche de la commande optimale.

Dans ce mémoire, après avoir défini plus en détail la problématique de l'étude, nous verrons les différents problèmes théoriques qui ont été abordés, ou qui restent à aborder, comme la recherche de la commande optimale à adjoint donné, la majoration du temps de trajet optimal, ou encore la localisation d'adjoints initiaux.

¹ La démonstration de la convergence de l'algorithme génétique n'est aujourd'hui pas démontrée, son comportement étant le résultat d'une démarche fortement marquée par une recherche aléatoire de la solution.

II Analyse de la problématique

2.1 Principe du minimum de Pontryagin

Considérons un système défini par les équations d'état $\begin{cases} \dot{x} = f(x, t, u) \\ u \in U \end{cases}$. L'objectif est de trouver la commande u qui nous permet de minimiser une fonction coût

$$J(x, u) = c(x(t_F)) + \int_{t_0}^{t_F} g(x, u, t) dt .$$

On associe à notre problème son Hamiltonien H , défini (pour un problème de minimisation) par $H(x, \lambda, u) = g(x, u, t) + \langle f(x, u), \lambda \rangle$, λ étant un vecteur appelé l'adjoint du système.

On peut démontrer que l'état et l'adjoint vérifient les équations

$$\begin{cases} \dot{x} = \frac{\partial H}{\partial \lambda} \\ \dot{\lambda} = -\frac{\partial H}{\partial x} \end{cases} \quad (1)$$

Par ailleurs, le principe du minimum, mis en place par *Lev Pontryagin* dans les années 1960, nous dit² que la commande optimale u^* , la trajectoire optimale x^* et l'adjoint associé λ^* vérifient

$$H(x^*, \lambda^*, u^*) = \min_{u \in U} H(x^*, \lambda^*, u) . \quad (2)$$

Ainsi, le système d'équations (1) nous permet de trouver une équation différentielle sur λ , tandis que l'équation (2) nous permet de trouver la commande optimale pour un état et un adjoint donnés. Le problème se résume donc à trouver la valeur de l'adjoint à l'instant initial t_0 .

L'objectif de ce stage est d'utiliser un algorithme génétique, afin de trouver cet adjoint initial.

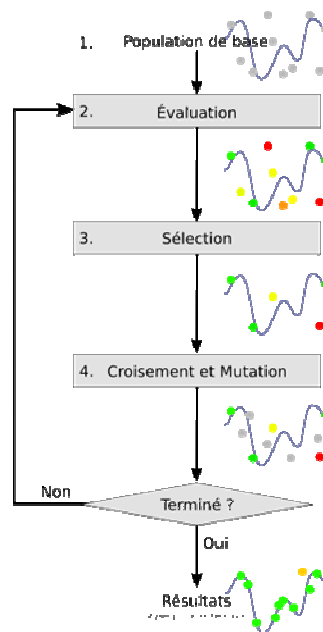
² Le principe du minimum nous donne aussi des égalités (les conditions de transversalité), que nous n'utiliserons pas.

2.2 Fonctionnement des algorithmes génétiques

Les algorithmes génétiques, que l'on utilise pour trouver une solution à un problème d'optimisation, s'inspirent directement du principe de la sélection naturelle de Darwin.

Au départ, l'algorithme prend en entrée une population de base (ici, une population d'adjoints), générée aléatoirement.

- Cette population est alors évaluée selon un critère.
- Les meilleurs éléments, c'est-à-dire les éléments qui ont obtenu la meilleure note selon notre critère, sont sélectionnés, tandis que les moins bons sont mis de côté.
- On obtient alors une nouvelle génération par *croisement* (un élément fils contient une combinaison des données de deux éléments parents) et par *mutation* (une partie de l'élément fils peut être altérée) des éléments sélectionnés.
- Enfin, on réitère l'algorithme sur la nouvelle population.



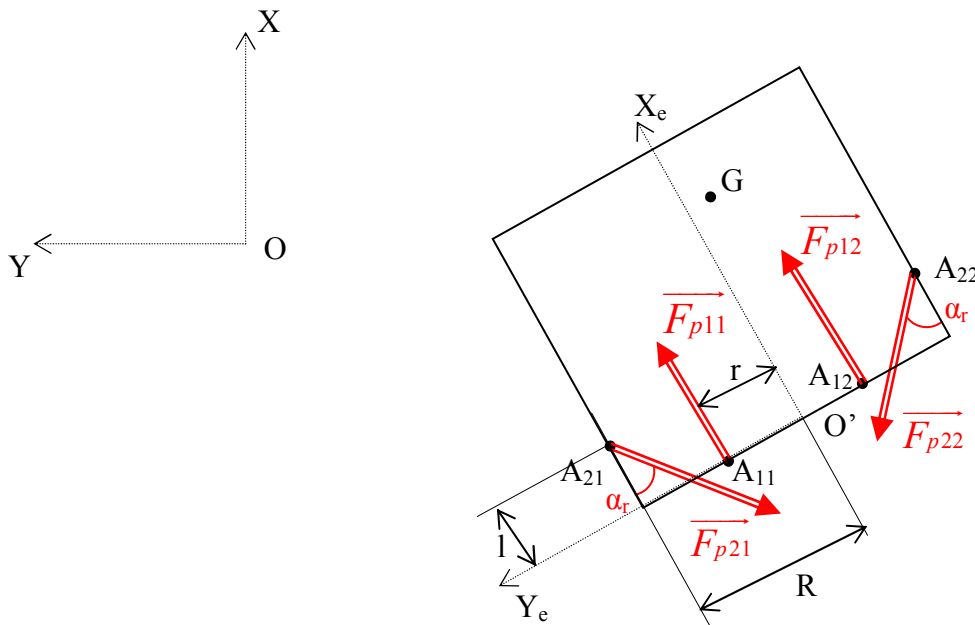
2.3 Le système étudié

Afin d'illustrer la méthodologie que nous allons mettre en place, nous mènerons les calculs et les démonstrations sur un système relativement simple.

2.3.1 Présentation du système

Considérons un système propulsé à l'aide de quatre tuyères alimentées par de l'ergol :

- Deux tuyères, situées sur la base, permettant de propulser le système vers l'avant.
- Deux tuyères latérales, permettant de propulser le système vers l'arrière.



On considère les repères :

- (O, X, Y) , un repère galiléen
- (O', X_e, Y_e) , un repère lié au bus, O' étant le centre du système.

Nous noterons R , le rayon du système et l , la distance selon X_e des tuyères latérales par rapport à la base.

De plus, nous supposons que G , le centre d'inertie, est décalé par rapport à l'axe du bus et nous noterons (x_G, y_G) ses coordonnées.

Les tuyères sont situées en A_{11} , A_{12} , A_{21} et A_{22} . Elles délivrent respectivement un débit μ_{11} , μ_{12} , μ_{21} , μ_{22} en ergol et nous noterons \vec{F}_{p11} , \vec{F}_{p12} , \vec{F}_{p21} et \vec{F}_{p22} les forces de poussée générées.

Nous noterons :

- θ , l'angle entre les vecteurs \vec{X} et \vec{X}_e
- \vec{F}_P , la force de poussée
- I_{sp} , l'impulsion spécifique de l'ergol
- M , la masse du système
- J , le moment d'inertie du système en G dans la base (\vec{X}_e, \vec{Y}_e)
- g_0 , l'accélération de la pesanteur en A.
- q , la vitesse angulaire du repère (A, X_e, Y_e) par rapport à (O, X, Y)

Enfin, nous supposons que le débit total des tuyères en ergol est constant et vaut :

$$\bar{\mu} = \mu_{11} + \mu_{12} + \mu_{21} + \mu_{22} \quad (3)$$

2.3.2 Modélisation du système

Commençons par exprimer les forces de poussée dans le repère lié au bus :

$$\begin{aligned} \vec{F}_{p11} &= \begin{pmatrix} \mu_{11} I_{sp} g_0 \\ 0 \end{pmatrix} & \vec{F}_{p12} &= \begin{pmatrix} \mu_{12} I_{sp} g_0 \\ 0 \end{pmatrix} \\ \vec{F}_{p21} &= -\mu_{21} I_{sp} g_0 \begin{pmatrix} \cos \alpha_r \\ \sin \alpha_r \end{pmatrix} & \vec{F}_{p22} &= \mu_{22} I_{sp} g_0 \begin{pmatrix} -\cos \alpha_r \\ \sin \alpha_r \end{pmatrix} \end{aligned}$$

D'où, la force de poussée totale dans le repère (O', X_e, Y_e)

$$\vec{F}_p = I_{sp} g_0 \begin{pmatrix} \mu_{11} + \mu_{12} - (\mu_{21} + \mu_{22}) \cos \alpha_r \\ (\mu_{22} - \mu_{21}) \sin \alpha_r \end{pmatrix}$$

Donc, dans le repère (O, X, Y) , on a

$$\vec{F}_p = I_{sp} g_0 \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} \begin{pmatrix} \mu_{11} + \mu_{12} - (\mu_{21} + \mu_{22}) \cos \alpha_r \\ (\mu_{22} - \mu_{21}) \sin \alpha_r \end{pmatrix} \quad (4)$$

Exprimons à présent les moments des forces de poussée en G :

$$M_{G11} = (y_G - r) \mu_{11} I_{sp} g_0 \qquad M_{G12} = (y_G + r) \mu_{12} I_{sp} g_0$$

$$M_{G21} = \pm \left\| \overrightarrow{GA_{21}} \wedge \overrightarrow{F_{P21}} \right\| = \mu_{21} I_{sp} g_0 ((R - y_G) \cos \alpha_r + (x_G - l) \sin \alpha_r)$$

$$M_{G22} = \pm \left\| \overrightarrow{GA_{22}} \wedge \overrightarrow{F_{P22}} \right\| = \mu_{22} I_{sp} g_0 ((l - x_G) \sin \alpha_r - (R + y_G) \cos \alpha_r)$$

Ainsi, le principe fondamental de la dynamique nous donne

$$\begin{cases} \ddot{V} = \frac{I_{sp} g_0}{M} \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} \begin{pmatrix} \mu_{11} + \mu_{12} - (\mu_{21} + \mu_{22}) \cos \alpha_r \\ (\mu_{22} - \mu_{21}) \sin \alpha_r \end{pmatrix} \\ \dot{\theta} = q \\ \dot{q} = \frac{I_{sp} g_0}{J} (\mu_{11}(y_G - r) + \mu_{12}(y_G + r) + \mu_{21}(R - y_G) \cos \alpha_r + \mu_{21}(x_G - l) \sin \alpha_r + \\ \mu_{22}(l - x_G) \sin \alpha_r - \mu_{22}(R + y_G) \cos \alpha_r) \end{cases} \quad (5)$$

2.3.3 Expression de l'Hamiltonien

Nous chercherons ici à minimiser le temps de trajet de notre système et/ou l'énergie dépensée, c'est-à-dire $J(x, u) = \int_{t_0}^{t_f} dt$, ou $J(x, u) = \int_{t_0}^{t_f} (1 + u^T Q u) dt$, Q étant une matrice symétrique définie positive.

Ainsi, l'Hamiltonien de notre système vaut

- $H(x, u, \lambda) = 1 + \langle f(x, u), \lambda \rangle$ dans le cas où seul le temps de trajet est optimisé
- $H(x, u, \lambda) = 1 + u^T Q u + \langle f(x, u), \lambda \rangle$ dans le cas où l'on cherche aussi à optimiser l'énergie dépensée.

L'équation (1) nous donne alors

$$\left\{ \begin{array}{l} \dot{\lambda}_1 = -\frac{\partial H}{\partial V_x} = 0 \\ \dot{\lambda}_2 = -\frac{\partial H}{\partial V_y} = 0 \\ \dot{\lambda}_3 = -\frac{\partial H}{\partial \theta} = -(\lambda_1 \quad \lambda_2) A'(\theta)u \\ \dot{\lambda}_4 = -\frac{\partial H}{\partial q} = -\lambda_3 \end{array} \right. \quad (6)$$

2.4 Méthodologie générale

On remarque que le problème est très différent selon que l'on traite le cas où seul le temps de parcours est optimisé, et le cas où l'on prend en compte des critères énergétiques.

En effet, dans le premier cas, l'Hamiltonien est linéaire en u et les contraintes sur la commande sont linéaires. Ainsi, le problème de minimisation de l'Hamiltonien pour un état et un adjoint donné peut se résoudre à l'aide d'un algorithme comme le simplexe.

La linéarité en u de l'Hamiltonien disparaît dans le second cas. Il faudra donc mettre en place une méthode permettant de minimiser l'Hamiltonien pour un état et un adjoint donné.

Comme il a été précisé plus haut, nous allons utiliser un algorithme génétique pour trouver un adjoint initial permettant de déterminer la commande optimale. Les populations d'adjoints testées par l'algorithme génétique pourront être évaluées en :

- résolvant numériquement les équations en x , λ , et u .
- en prenant comme critère une fonction de la distance minimale de notre système par rapport à un état cible³ et du temps mis pour s'approcher au mieux de cet état.

Cette méthodologie soulève quelques problèmes.

En effet, nous devons tout d'abord nous interroger sur l'intervalle de temps sur lequel il faut résoudre les équations différentielles régissant notre système. De plus, il faut définir une zone de recherche de l'adjoint, afin de pouvoir initialiser l'algorithme génétique.

³ Pour la suite, nous supposons que l'état final vaut 0.

III Recherche d'un critère d'arrêt

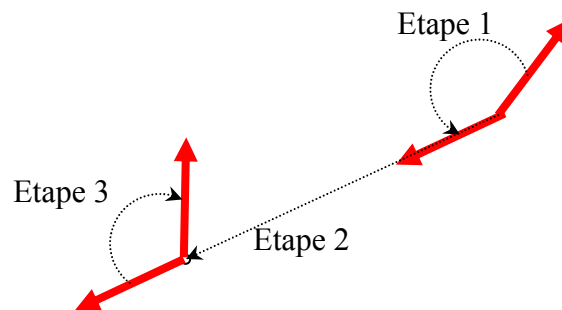
Comme nous l'avons précisé dans le paragraphe précédent, nous devons définir un intervalle de temps sur lequel nous allons intégrer nos équations différentielles. Nous pouvons, par exemple, rechercher un majorant du temps de trajet optimal.

3.1 Idée générale

Par définition de la trajectoire optimale, si une trajectoire T amène notre système dans l'état voulu, alors le temps de trajet de T sera plus long que le temps de trajet de la trajectoire optimale. Ainsi, pour trouver un majorant de t_f , il suffit de calculer le temps de trajet pour une trajectoire particulière.

Considérons la trajectoire T qui se déroule en trois temps :

- Tout d'abord, le système tourne sur lui-même afin de « pointer » vers l'état cible.
- Puis le système se déplace en ligne droite jusqu'à la vitesse nulle
- Enfin, le système tourne à nouveau sur lui-même jusqu'à avoir $\theta=0$.



Calculons le temps nécessaire pour chaque étape.

3.2 Calcul du temps de parcours en ligne droite

On a
$$\|\dot{\vec{V}}\| = \frac{I_{sp}g_0}{M} \left\| \begin{pmatrix} \mu_{11} + \mu_{12} - (\mu_{21} + \mu_{22}) \cos \alpha_r \\ (\mu_{22} - \mu_{21}) \sin \alpha_r \end{pmatrix} \right\|$$

$\|\dot{\vec{V}}\|$ est maximal pour $\mu_{11} + \mu_{12} = \bar{\mu}$. On a alors $\|\dot{\vec{V}}\| = \frac{I_{sp}g_0}{M} \bar{\mu}$.

D'où
$$\|\vec{V} - \vec{V}_0\| = \frac{I_{sp}g_0}{M} \bar{\mu} t$$

Ainsi, pour $t=t_2$ (t_2 étant le temps de parcours en ligne droite), on a

$$\|\vec{V}_0\| = \frac{I_{sp}g_0}{M} \bar{\mu} t_2$$

Donc,
$$t_2 = \frac{M \|\vec{V}_0\|}{I_{sp}g_0 \bar{\mu}} \quad (7)$$

3.3 Calcul du temps nécessaire pour la rotation

Commençons par calculer l'accélération maximale que le système peut obtenir en rotation, tout en restant sur place.

Pour que le système soit en rotation pure, il faut que

$$\begin{cases} \mu_{11} + \mu_{12} - (\mu_{21} + \mu_{22}) \cos \alpha_r = 0 \\ \mu_{21} = \mu_{22} \\ \mu_{11} + \mu_{12} + \mu_{21} + \mu_{22} = \bar{\mu} \end{cases},$$

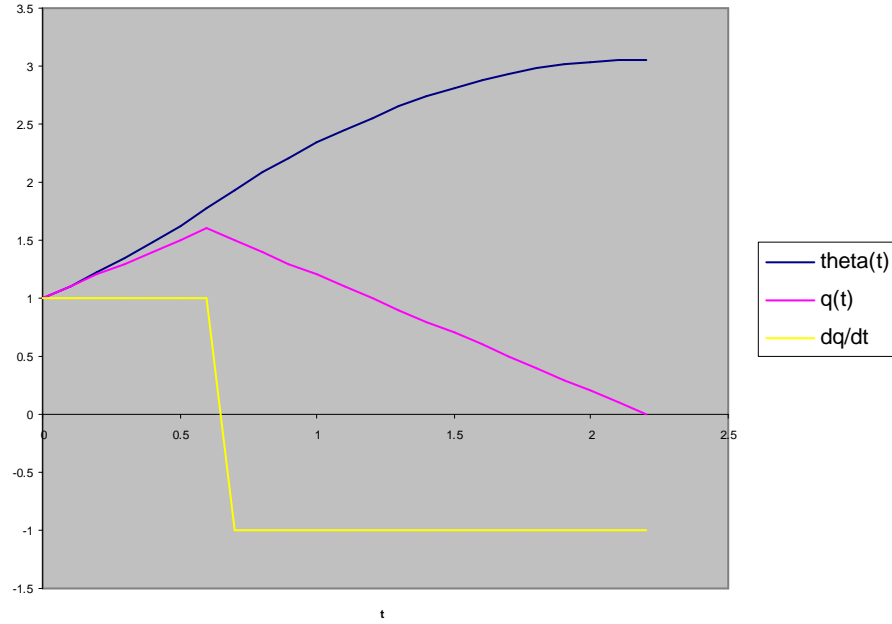
ce qui donne les relations $\mu_{21} = \mu_{22} = \frac{\bar{\mu}}{2(1 + \cos \alpha_r)}$ et $\mu_{12} = \frac{\bar{\mu}}{1 + \cos \alpha_r} - \mu_{11}$.

Dans ce cas, on a alors $\dot{q} = \frac{I_{sp}g_0}{J} (-2r\mu_{11} + \frac{\bar{\mu}r \cos \alpha_r}{\cos \alpha_r + 1})$, avec $\mu_{11} \in [0; \frac{\bar{\mu} \cos \alpha_r}{1 + \cos \alpha_r}]$.

D'où
$$-\frac{I_{sp}g_0 \bar{\mu}r \cos \alpha_r}{J \cos \alpha_r + 1} \leq \dot{q} \leq \frac{I_{sp}g_0 \bar{\mu}r \cos \alpha_r}{J \cos \alpha_r + 1} \quad (8)$$

Posons $\dot{q}_{Max} = \frac{I_{sp}g_0 \bar{\mu}r \cos \alpha_r}{J \cos \alpha_r + 1}$.

Pour déterminer le temps minimal t_1 nécessaire pour passer d'un angle θ_0 à un angle θ_f , nous allons supposer que $\dot{q} = \dot{q}_{Max}$ pour $0 \leq t \leq \tilde{t}_1$, et $\dot{q} = -\dot{q}_{Max}$ pour $\tilde{t}_1 \leq t \leq t_1$ (nous supposons dans un premier temps que $\theta_f \geq \theta_0$).



Si q_0 est la vitesse de rotation du système à $t=0$, pour $t = \tilde{t}_1$, on a

$$\theta(\tilde{t}_1) = \theta_1 = \frac{1}{2} \dot{q}_{Max} \tilde{t}_1^2 + q_0 \tilde{t}_1 + \theta_0$$

et
$$q(\tilde{t}_1) = \dot{q}_{Max} \tilde{t}_1 + q_0$$

Pour $t=t_1$,
$$\theta_f = -\frac{1}{2} \dot{q}_{Max} (t_1 - \tilde{t}_1)^2 + q(\tilde{t}_1)(t_1 - \tilde{t}_1) + \theta_1$$

De plus, pour $t=t_1$, la vitesse de rotation doit être nulle. Ainsi,

$$q(t_1) = 0 = -\dot{q}_{Max}(t_1 - \tilde{t}_1) + q(\tilde{t}_1) \quad (9)$$

On a alors
$$\theta_f = -\frac{1}{2} \dot{q}_{Max} (t_1 - \tilde{t}_1)^2 + \dot{q}_{Max} (t_1 - \tilde{t}_1)^2 + \frac{1}{2} \dot{q}_{Max} \tilde{t}_1^2 + q_0 \tilde{t}_1 + \theta_0$$

$$\Delta\theta = \theta_f - \theta_0 = \frac{1}{2} \dot{q}_{Max} \left(\tilde{t}_1 + \frac{q_0}{\dot{q}_{Max}} \right)^2 + \frac{1}{2} \dot{q}_{Max} \tilde{t}_1^2 + q_0 \tilde{t}_1$$

D'où, l'équation
$$\dot{q}_{Max}\tilde{t}_1^2 + 2q_0\tilde{t}_1 + \frac{1}{2}\frac{q_0^2}{\dot{q}_{Max}} - \Delta\theta = 0 \quad (10)$$

La résolution de l'équation (10) nous donne

$$\tilde{t}_1 = \frac{-2q_0 + \sqrt{2q_0^2 + 4\dot{q}_{Max}\Delta\theta}}{2\dot{q}_{Max}}$$

Donc, d'après la relation (9), $t_1 = 2\tilde{t}_1 + \frac{q_0}{\dot{q}_{Max}} = \frac{-q_0 + \sqrt{2q_0^2 + 4\dot{q}_{Max}\Delta\theta}}{\dot{q}_{Max}}$

Dans le cas, où $\theta_f \leq \theta_0$, on peut supposer que $\dot{q} = -\dot{q}_{Max}$ pour $0 \leq t \leq \tilde{t}_1$ et $\dot{q} = \dot{q}_{Max}$ pour $\tilde{t}_1 \leq t \leq t_1$. Un calcul similaire nous donne alors

$$t_1 = 2\tilde{t}_1 + \frac{q_0}{\dot{q}_{Max}} = \frac{-q_0 + \sqrt{2q_0^2 + 4\dot{q}_{Max}|\Delta\theta|}}{\dot{q}_{Max}} \quad (11)$$

Remarque :

➤ Dans le cas où $q_0=0$ (ce qui arrive nécessairement dans la phase 3 de la

trajectoire), on a
$$t_1 = 2\sqrt{\frac{|\Delta\theta|}{\dot{q}_{Max}}}$$

➤ Si $q_0^2 \geq 2\dot{q}_{Max}|\Delta\theta|$, alors \tilde{t}_1 est négatif. Il faut donc inverser les commandes. Le calcul de t_1 peut alors se résumer dans le tableau

	$q_0^2 \leq 2\dot{q}_{Max} \Delta\theta $	$q_0^2 \geq 2\dot{q}_{Max} \Delta\theta $
$\Delta\theta \geq 0$	$t_1 = \frac{-q_0 + \sqrt{2q_0^2 + 4\dot{q}_{Max} \Delta\theta }}{\dot{q}_{Max}}$	$t_1 = \frac{q_0 - \sqrt{2q_0^2 - 4\dot{q}_{Max} \Delta\theta }}{\dot{q}_{Max}}$
$\Delta\theta \leq 0$	$t_1 = \frac{-q_0 + \sqrt{2q_0^2 + 4\dot{q}_{Max} \Delta\theta }}{\dot{q}_{Max}}$	$t_1 = \frac{q_0 + \sqrt{2q_0^2 - 4\dot{q}_{Max} \Delta\theta }}{\dot{q}_{Max}}$

3.4 Calcul d'un majorant du temps optimal

Finalement, si $\Delta\theta_1$ et $\Delta\theta_2$ sont les angles de rotation que le système doit effectuer dans la première et la dernière phase de la trajectoire, on a

$$t_f \leq \frac{\pm q_0 \pm \sqrt{2q_0^2 \pm 4\dot{q}_{Max}|\Delta\theta_1|}}{\dot{q}_{Max}} + \frac{M \|\vec{v}_0\|}{I_{sp}g_0\bar{\mu}} + 2\sqrt{\frac{|\Delta\theta_2|}{\dot{q}_{Max}}} \quad (12)$$

IV Optimisation d'une fonction quadratique.

4.1 Le problème

On cherche à trouver la commande optimale u permettant de minimiser la fonction $f(u) = u^T Q u$, où Q est une matrice symétrique définie positive. Rappelons que notre modèle est linéaire en u et que, par ailleurs, la commande doit respecter la contrainte

$$\begin{cases} \bar{u} = u_{11} + u_{12} + u_{21} + u_{22} \\ u_{11}, u_{12}, u_{21}, u_{22} \geq 0 \end{cases} \quad (S1)$$

Ainsi, l'Hamiltonien du problème peut se mettre sous la forme :

$$H(u) = u^T Q u - \langle a, u \rangle + K, \quad (13)$$

où a et K sont des vecteurs dépendant de l'état x et de l'adjoint λ . Par ailleurs, la contrainte sur la commande peut se mettre sous la forme

$$Au \leq B \quad (14)$$

4.2 Simplification du problème

Comme Q est symétrique définie positive, la matrice peut être mise sous la forme $Q = T^T T$, où T est une matrice triangulaire inversible (décomposition de Cholesky). Ainsi, on a

$$u^T Q u = \langle u, Q u \rangle = \langle u, T^T T u \rangle = \langle T u, T u \rangle$$

De plus, comme K ne dépend pas de la commande, ce vecteur n'intervient pas dans la recherche de la commande optimale. Ainsi, si on pose

$$v = T u,$$

le problème revient à minimiser la fonction

$$f(v) = \langle v, v \rangle - \langle a, T^{-1} v \rangle = \langle v, v \rangle - \langle (T^{-1})^T a, v \rangle,$$

avec la contrainte

$$A T^{-1} v \leq B$$

Ainsi, dans un premier temps, nous allons tenter de résoudre le problème

$$\langle u^* - a, u^* \rangle = \min_{Au \leq B} \langle u - a, u \rangle \quad (15)$$

4.3 Résolution du problème

Il suffit de remarquer que

$$\langle u - a, u \rangle = \langle (u - \frac{1}{2}a) - \frac{1}{2}a, (u - \frac{1}{2}a) + \frac{1}{2}a \rangle = \left\| u - \frac{1}{2}a \right\|^2 - \frac{1}{4} \|a\|^2 \quad (16)$$

Ainsi, minimiser $\langle u - a, u \rangle$ revient à minimiser $\left\| u - \frac{1}{2}a \right\|^2$ et u^* peut donc être vu comme la projection de $\frac{1}{2}a$ sur l'ensemble U , défini par $U = \{u, Au \leq B\}$.

4.4 Retour au problème d'origine

Avec les notations utilisées plus haut, la commande optimale u^* vérifie :

$$u^* = T^{-1}v^*,$$

où v^* est la projection du vecteur $\frac{1}{2}(T^{-1})^T a$ sur l'ensemble $U = \{u, ATu \leq B\}$ et T vérifie $Q = T^T T$.

Ainsi, pour calculer la commande optimale, il faut pouvoir :

- Déterminer la projection d'un vecteur sur l'ensemble U
- Décomposer la matrice Q en un produit $Q = T^T T$.

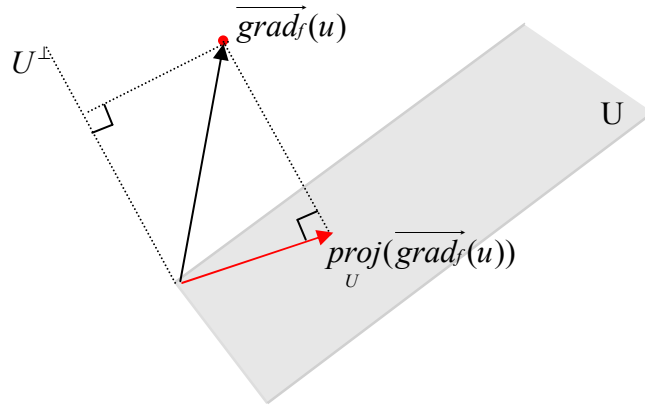
En ce qui concerne la décomposition de Q , la démonstration de la décomposition de Cholesky nous donne un algorithme.

4.5 Résultats préliminaires sur l'optimisation

Soit $f : \mathbb{R}^n \rightarrow \mathbb{R}$ une fonction de classe C^2 . On pose $\text{proj}_U(\overrightarrow{\text{grad}}_f(x))$ la projection du gradient de f en x sur l'ensemble U défini par $U = \{u, Au = 0\}$.

Considérons la suite u_n définie sur \mathbb{R}^n par la relation de récurrence :

$$u_{n+1} = u_n - h \cdot \text{proj}_U(\overrightarrow{\text{grad}}_f(u_n)) \quad (17)$$



Nous allons montrer que, sous certaines conditions sur f , cette suite converge vers un minimum local de f sur U .

On a

$$\begin{aligned} \|u_{n+1} - u_n\| &= \left\| u_n - h \cdot \text{proj}_U(\overrightarrow{\text{grad}}_f(u_n)) - u_{n-1} + h \text{proj}_U(\overrightarrow{\text{grad}}_f(u_{n-1})) \right\| \\ &= \left\| (u_n - u_{n-1}) - h(\text{proj}_U(\overrightarrow{\text{grad}}_f(u_n)) - \text{proj}_U(\overrightarrow{\text{grad}}_f(u_{n-1}))) \right\| \end{aligned} \quad (18)$$

Comme la projection orthogonale sur U est une application linéaire, on peut écrire $\|u_{n+1} - u_n\| = \|u_n - u_{n-1} - hP(\overrightarrow{\text{grad}}_f(u_n) - \overrightarrow{\text{grad}}_f(u_{n-1}))\|$, où P est la matrice représentant la projection sur U .

Posons

$$u(\eta) = u_{n-1} + \eta(u_n - u_{n-1}) = \begin{pmatrix} u^1(\eta) \\ \vdots \\ u^m(\eta) \end{pmatrix}$$

et $g(\eta) = \overrightarrow{\text{grad}}_f(u(\eta)) = \overrightarrow{\text{grad}}_f(u_{n-1} + \eta(u_n - u_{n-1}))$, avec $\eta \in [0, 1]$.

On a

$$g'(\eta) = \frac{\partial(\overline{\text{grad}f})}{\partial u^1} \frac{d(u^1)}{d\eta} + \dots + \frac{\partial(\overline{\text{grad}f})}{\partial u^m} \frac{d(u^m)}{d\eta}$$

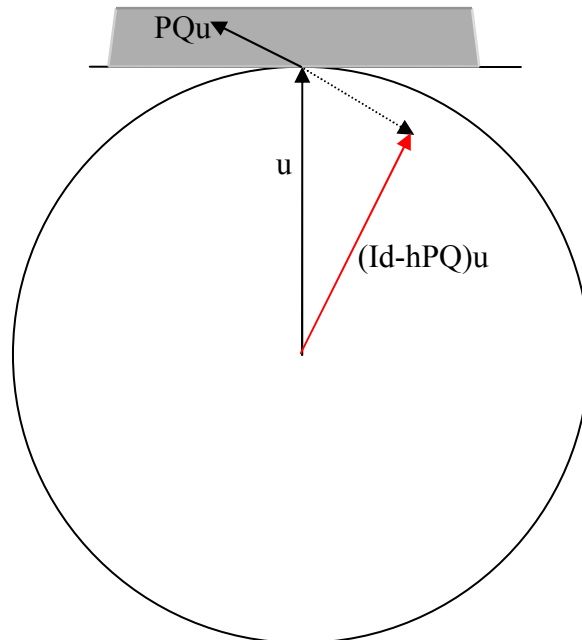
$$g'(\eta) = \begin{pmatrix} \frac{\partial^2 f}{\partial u^1{}^2} \frac{d(u^1)}{d\eta} + \dots + \frac{\partial^2 f}{\partial u^1 \partial u_n} \frac{du^n}{d\eta} \\ \vdots \\ \frac{\partial^2 f}{\partial u^1 \partial u_n} \frac{d(u^1)}{d\eta} + \dots + \frac{\partial^2 f}{\partial u_n^2} \frac{du^n}{d\eta} \end{pmatrix} = \begin{pmatrix} \frac{\partial^2 f}{\partial u^1{}^2} & \dots & \frac{\partial^2 f}{\partial u^1 \partial u_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial u^1 \partial u_n} & \dots & \frac{\partial^2 f}{\partial u_n^2} \end{pmatrix} \frac{du}{d\eta}$$

Ainsi, si $Q = \begin{pmatrix} \frac{\partial^2 f}{\partial u^1 \partial u^1} & \dots & \frac{\partial^2 f}{\partial u^1 \partial u^m} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial u^m \partial u^1} & \dots & \frac{\partial^2 f}{\partial u^m \partial u^m} \end{pmatrix}$ est la matrice hessienne de f , on a $g'(\eta) = Q(u_n - u_{n-1})$.

Or, d'après le théorème des accroissements finis, il existe θ dans $[0, 1]$ tel que $g(1) - g(0) = g'(\theta)$. On peut donc écrire

$$\|u_{n+1} - u_n\| = \|(Id - hPQ)(u_n - u_{n-1})\| \leq \|Id - hPQ\| \cdot \|u_n - u_{n-1}\| \quad (19)$$

Cherchons une condition pour que $\|Id - hPQ\|$ soit strictement inférieur à 1.



La figure ci-dessus nous montre que s'il existe $h > 0$ tel que $\|(Id - hPQ)u\| < \|u\|$, alors, nécessairement,

$$\langle PQu, u \rangle > 0 \quad (20)$$

Or, P représente une projection orthogonale et est donc une matrice symétrique. On a alors,

$$\langle PQu, u \rangle = \langle Qu, Pu \rangle$$

D'où, pour $u \in U$,
$$\langle PQu, u \rangle = \langle Qu, u \rangle > 0 \quad (21)$$

Autrement dit, pour avoir la relation $\|(Id - hPQ)u\| < \|u\|$, il faut que la matrice Q soit définie positive, c'est-à-dire que f soit convexe sur U .

Dans ces conditions, on peut écrire $\|u_{n+1} - u_n\| \leq k \|u_n - u_{n-1}\|$. Ainsi, si p est positif, on a

$$\begin{aligned} \|u_{n+p} - u_n\| &\leq \|u_{n+p} - u_{n+p-1}\| + \|u_{n+p-1} - u_{n+p-2}\| + \dots + \|u_{n+1} - u_n\| \\ \|u_{n+p} - u_n\| &\leq (k^{n+p-1} + k^{n+p-2} + \dots + k^n) \|u_1 - u_0\| \end{aligned}$$

D'où
$$\|u_{n+p} - u_n\| \leq \frac{k^n}{1-k} \|u_1 - u_0\| \quad (22)$$

On a alors
$$\|u_{n+p} - u_n\| \xrightarrow{n \rightarrow +\infty} 0$$

Comme U est complet, la suite u_n converge vers un vecteur \bar{u} , qui vérifie

$$h \cdot \underset{U}{proj}(\overline{grad}_f(\bar{u})) = 0.$$

Donc, comme f est convexe, \bar{u} minimise f sur U .

Remarque :

- La formule (22) nous fournit un majorant de l'erreur $\|\bar{u} - u_n\|$. Ainsi,

$$\|\bar{u} - u_n\| = O(k^n) \text{ sur } n \rightarrow +\infty.$$

- Dans le cas d'une fonction quadratique $u \rightarrow u^T Qu$ où Q est une matrice symétrique définie positive, la matrice hessienne de f vaut $2Q$. f est donc convexe.

- Dans le cas où f est une projection, la matrice hessienne est constante et vaut $2Id_n$. L'équation (19) nous donne alors $\|u_{n+1} - u_n\| = |1 - 2h| \|u_n - u_{n-1}\|$, qui s'annule pour $h = \frac{1}{2}$. Le minimum de f est alors trouvé dès la première itération.

Ainsi, dans notre problème initial (optimiser l'équation (13)), deux méthodes s'offrent à nous : soit on cherche à minimiser l'Hamiltonien tel quel et, dans ce cas, la suite u_n convergera de façon plus ou moins rapide, soit on transforme le problème en un problème de projection (ce qui implique de décomposer la matrice Q sous la forme $Q = T^T T$ et d'inverser T) et, dans ce cas, la suite u_n trouve tout de suite une solution optimale.

4.6 Recherche du paramètre h optimal

Pour que la suite u_n converge le plus vite possible, il faut que $k^2 = \|(Id - hPQ)u\|^2$ soit le plus petit possible.

On a
$$k^2 = \|u\|^2 + h^2 \langle PQu, PQu \rangle - 2h \langle u, PQu \rangle$$

$$k^2 = \|u\|^2 + h^2 \langle u, QP^2Qu \rangle - 2h \langle Pu, Qu \rangle$$

$$k^2 = \|u\|^2 + h^2 \langle u, QPQu \rangle - 2h \langle u, Qu \rangle \quad (23)$$

Ainsi, k^2 est minimal pour $h = \frac{\langle u, Qu \rangle}{\langle u, QPQu \rangle}$. (24)

Si λ_{min} est la plus petite valeur propre de Q et μ_{max} , la plus grande valeur propre de QPQ , alors une valeur possible de h pourrait être $h = \frac{\lambda_{min}}{\mu_{max}}$.

Pour la valeur optimale de h , on a alors $k^2 = \|u\|^2 - \frac{\langle u, Qu \rangle^2}{\langle u, QPQu \rangle}$ (25)

Dans le pire des cas, $\langle u, Qu \rangle = \lambda_{min} \|u\|^2$ et $\langle u, QPQu \rangle = \mu_{max} \|u\|^2$. On a alors,

$$k < \|u\| \sqrt{1 - \frac{\lambda_{min}^2}{\mu_{max}}} \quad (26)$$

Finalement, pour résumer,

- Si f est une fonction de classe C^2 , convexe sur U , la suite u_n définie par la relation de récurrence $u_{n+1} = u_n - h \cdot \underset{U}{\text{proj}}(\overline{\text{grad}_f(u_n)})$ converge vers \bar{u} , le minimum de f sur U .
- De plus, si $h = \frac{\lambda_{\min}}{\mu_{\max}}$, où λ_{\min} est la plus petite valeur propre de Q , la matrice hessienne de f et μ_{\max} est la plus grande valeur propre de QPQ , P étant la matrice de projection orthogonale sur U , alors on a une estimation de l'erreur $\|\bar{u} - u_n\| = O_{n \rightarrow +\infty}(k^n)$, avec $k < \|u\| \sqrt{1 - \frac{\lambda_{\min}^2}{\mu_{\max}}}$.

4.7 Application à la recherche d'optimum sous contrainte linéaire

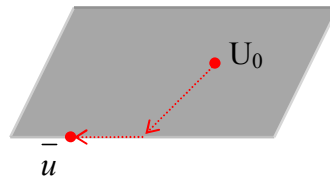
Nous allons tenter de chercher un minimum local d'une fonction f convexe sur un ensemble U défini par $u \in U \Leftrightarrow \begin{cases} Au = B \\ u \geq 0 \end{cases}$ (toute forme de contrainte linéaire peut se mettre sous cette forme).

Un algorithme possible d'optimisation est de partir d'un vecteur u_0 de l'ensemble U et de définir la suite u_n par la formule (17). Il y a alors deux possibilités :

- Soit la suite u_n converge vers un élément de U . Dans ce cas on a trouvé un minimum local.
- Soit il existe un entier j à partir duquel u_j a une composante $u_j(i)$ négative. Dans ce cas, on adapte le paramètre h de telle façon que $u_j(i)$ soit nul. Si une nouvelle itération nous fait encore sortir de U , on ajoute une contrainte $u_j(i)=0$: on a alors

$$u \in U \Leftrightarrow \begin{cases} A'u = B' \\ u \geq 0 \end{cases}, \text{ avec } A' = \begin{pmatrix} & & A & & \\ 0 & \dots & 1 & \dots & 0 \end{pmatrix} \text{ et } B' = \begin{pmatrix} B \\ 0 \end{pmatrix}. \text{ On réitère alors}$$

l'algorithme avec les nouvelles contraintes.



Dans la pratique, l'algorithme a été comparé avec les résultats obtenus par la fonction *fmincon* de Matlab : les résultats sont identiques (aux arrondis numériques près).

De plus, dans le cas d'une fonction quadratique, la fonction n'admet qu'un minimum global. Donc, cet algorithme nous permet d'obtenir le minimum global de la fonction sur U .

Enfin, comme il a été démontré dans le paragraphe précédent, l'algorithme trouve bien le point optimal en une itération lorsque f est une projection orthogonale.

V Problème de localisation des adjoints initiaux

Pour que l'algorithme génétique puisse fonctionner efficacement, nous devons définir une zone dans laquelle l'algorithme doit chercher les adjoints initiaux qui engendreront la trajectoire optimale.

5.1 Cas où seul le temps est optimisé

Dans ce cas, l'Hamiltonien vaut $H(x, \lambda, u) = 1 + \langle \lambda, f(x, u) \rangle$, et donc, si h est un nombre réel,

$$H(x, h\lambda, u) = 1 + h \langle \lambda, f(x, u) \rangle .$$

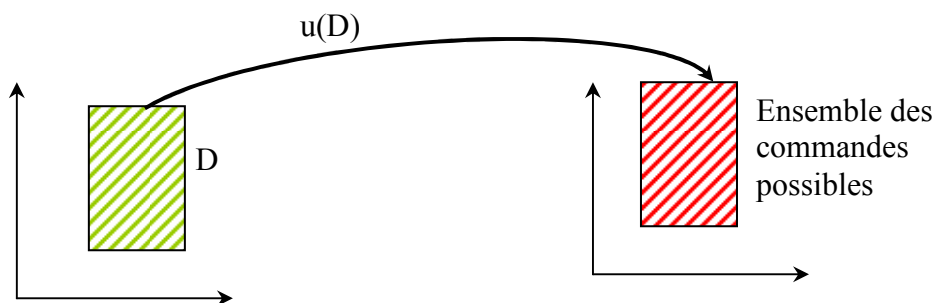
Ainsi, λ et $h\lambda$ donnent la même commande optimale, et donc la même trajectoire optimale. Autrement dit, la norme de l'adjoint initial n'a aucune importance.

On peut alors rechercher l'adjoint initial dans une boule non nulle dont le rayon est choisi arbitrairement.

5.2 Cas où le critère est une fonction quadratique de la commande

5.2.1 Idée générale

Rappelons que, pour un adjoint donné, la commande optimale est une projection orthogonale sur un ensemble borné. Ainsi, il existe un ensemble borné D tel que, pour toute commande u_0 possible, il existe un adjoint dans D dont la commande optimale associée est u_0 .



Ainsi, pour localiser les adjoints initiaux, une méthode possible est de :

- Déterminer un domaine d'adjoints D permettant de générer toutes les commandes possibles.
- Prendre en compte l'évolution de l'adjoint au cours du temps afin que le domaine dans lequel évolue λ contienne D à tout instant.

Remarque : nous mènerons ici, à titre d'exemple, les calculs dans le cas particulier où Q est la matrice identité.

5.2.2 Recherche du domaine générateur

Pour trouver le domaine D , nous allons résoudre formellement l'équation $u(\lambda) = \tilde{u}$, \tilde{u} étant une commande vérifiant les contraintes.

Pour cela, commençons par donner quelques précisions sur l'expression de la commande en fonction de l'adjoint.

Pour x et λ donnés, on cherche à minimiser $u^T Q u + 1 + \langle f(x, u), \lambda \rangle$.

$$\text{De plus, } f(x, u) = \begin{pmatrix} A(\theta) \\ 0 \\ C \end{pmatrix} u + \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix} x = B u + \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix} x, \text{ avec}$$

$$A(\theta) = \frac{I_{sp} g_0}{M} \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} \begin{pmatrix} 1 & 1 & -\cos \alpha_r & -\cos \alpha_r \\ 0 & 0 & -\sin \alpha_r & \sin \alpha_r \end{pmatrix}$$

$$A(\theta) = \frac{I_{sp} g_0}{M} \begin{pmatrix} \cos \theta & \cos \theta & -\cos(\theta + \alpha_r) & -\cos(\theta - \alpha_r) \\ \sin \theta & \sin \theta & -\sin(\theta + \alpha_r) & -\sin(\theta - \alpha_r) \end{pmatrix}$$

et où C est de la forme $C = (c1 \ c2 \ c3 \ c4)$.

Si l'on pose $Q = T^T T$ et $v = T u$, le problème revient à minimiser

$$v^T v + \langle B T^{-1} v, \lambda \rangle = v^T v + \langle T^{-1T} B^T \lambda, v \rangle,$$

avec la contrainte $\bar{\mu} = (1 \ 1 \ 1 \ 1) u = (1 \ 1 \ 1 \ 1) T^{-1} v = a^T v$.

Ainsi, si P_a est la matrice de projection sur l'orthogonale de a , et si $u_0 = \frac{\bar{\mu}}{4} \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$, on a

$$v = -\frac{1}{2} P_a T^{-1T} B^T \lambda + u_0,$$

et
$$u = -\frac{1}{2} T^{-1} P_a T^{-1T} B^T \lambda + T^{-1} u_0 \quad (27)$$

Dans le cas où $Q = \text{Id}$, on a
$$u = -\frac{1}{2} P_a B^T \lambda + u_0 \quad (28)$$

et
$$P_a = \begin{pmatrix} 3/4 & -1/4 & -1/4 & -1/4 \\ -1/4 & 3/4 & -1/4 & -1/4 \\ -1/4 & -1/4 & 3/4 & -1/4 \\ -1/4 & -1/4 & -1/4 & 3/4 \end{pmatrix}.$$

On a alors

$$-\frac{1}{2} P_a B^T = \frac{I_{sp} g_0}{8M} \begin{pmatrix} -2 \cos \theta (1 + \cos \alpha_r) & -2 \sin \theta (1 + \cos \alpha_r) & 0 & d_1 \\ -2 \cos \theta (1 + \cos \alpha_r) & -2 \sin \theta (1 + \cos \alpha_r) & 0 & d_2 \\ 2 \cos \theta (1 + \cos \alpha_r) - 4 \sin \theta \sin \alpha_r & 2 \sin \theta (1 + \cos \alpha_r) + 4 \sin \alpha_r \cos \theta & 0 & d_3 \\ 2 \cos \theta (1 + \cos \alpha_r) + 4 \sin \theta \sin \alpha_r & 2 \sin \theta (1 + \cos \alpha_r) - 4 \sin \alpha_r \cos \theta & 0 & d_4 \end{pmatrix}$$

Où d_1, d_2, d_3 et d_4 sont définis par
$$\begin{pmatrix} d_1 \\ d_2 \\ d_3 \\ d_4 \end{pmatrix} = P_a C^T.$$

Ainsi, si on pose
$$\begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{pmatrix} = \frac{8M}{I_{sp} g_0} (\tilde{u} - u_0),$$
 λ vérifie les équations

$$\begin{cases} -2(1 + \cos \alpha_r)(\lambda_1 \cos \theta + \lambda_2 \sin \theta) + d_1 \lambda_4 = u_1 \\ -2(1 + \cos \alpha_r)(\lambda_1 \cos \theta + \lambda_2 \sin \theta) + d_2 \lambda_4 = u_2 \\ 2(1 + \cos \alpha_r)(\lambda_1 \cos \theta + \lambda_2 \sin \theta) + 4 \sin \alpha_r (\lambda_2 \cos \theta - \lambda_1 \sin \theta) + d_3 \lambda_4 = u_3 \end{cases} \quad (29)$$

On remarque que λ_3 n'intervient pas dans le calcul de la commande. On peut donc fixer arbitrairement la valeur de λ_{30} (par exemple $\lambda_{30}=1$)

Des deux premières équations, on déduit que

$$\lambda_4 = \frac{u_1 - u_2}{d_1 - d_2}, \quad (30)$$

et
$$\lambda_1 \cos \theta + \lambda_2 \sin \theta = \frac{d_1 u_2 - d_2 u_1}{2(\cos \alpha_r + 1)(d_2 - d_1)}.$$

La troisième équation nous permet alors d'obtenir

$$\lambda_2 \cos \theta - \lambda_1 \sin \theta = \frac{1}{4 \sin \alpha_r} \left(u_3 + \frac{u_1(d_2 + d_3) - u_2(d_1 + d_3)}{d_2 - d_1} \right),$$

d'où

$$\lambda_1 + i \lambda_2 = e^{i\theta} \left(\frac{d_1 u_2 - d_2 u_1}{2(\cos \alpha_r + 1)(d_2 - d_1)} + \frac{i}{4 \sin \alpha_r} \left(u_3 + \frac{u_1(d_2 + d_3) - u_2(d_1 + d_3)}{d_2 - d_1} \right) \right) \quad (31)$$

Comme les u_i sont dans l'intervalle $[u] = \frac{8M}{I_{sp} g_0} ([0; \bar{\mu}] - u_0) = \left[-\frac{2M\bar{\mu}}{I_{sp} g_0}; \frac{6M\bar{\mu}}{I_{sp} g_0} \right]$, on a

$$\lambda_4 \in \frac{[u] - [u]}{d_1 - d_2}, \text{ et si on pose}$$

$$[a] = \frac{d_1 [u] - d_2 [u]}{2(\cos \alpha_r + 1)(d_2 - d_1)}$$

$$[b] = \frac{1}{4 \sin \alpha_r} \left([u] + \frac{[u](d_2 + d_3) - [u](d_1 + d_3)}{d_2 - d_1} \right),$$

λ_1 et λ_2 sont dans l'enveloppe de $[a]$ et $[b]$ (c'est-à-dire dans le plus petit intervalle contenant a et b).

Finalement, pour résumer, on a
$$D = \begin{pmatrix} [[a] \cup [b]] \\ [[a] \cup [b]] \\ \{1\} \\ \frac{[u] - [u]}{d_1 - d_2} \end{pmatrix}.$$

5.2.3 Prise en compte de l'évolution de l'adjoint

Considérons D_0 , un pavé de \mathbb{R}^4 , et posons $D_t = \{\lambda(t), \lambda(0) \in D_0\}$. Pour que D_0 permette d'obtenir toutes les trajectoires optimales possibles sur l'intervalle de temps $[0; t_f]$, il faut qu'à chaque instant t , D_t contienne D .



Pour déterminer D_0 , nous allons calculer un majorant de $|\lambda(t_f) - \lambda_0|$.

Rappelons que le système est régi par les équations d'état

$$\begin{cases} \vec{V} = \frac{I_{sp} g_0}{M} \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} \begin{pmatrix} \mu_{11} + \mu_{12} - (\mu_{21} + \mu_{22}) \cos \alpha_r \\ (\mu_{22} - \mu_{21}) \sin \alpha_r \end{pmatrix} \\ \dot{\theta} = q \\ \dot{q} = \frac{I_{sp} g_0}{J} (\mu_{11}(y_G - r) + \mu_{12}(y_G + r) + \mu_{21}(R - y_G) \cos \alpha_r + \mu_{21}(x_G - l) \sin \alpha_r + \mu_{22}(l - x_G) \sin \alpha_r - \mu_{22}(R + y_G) \cos \alpha_r) \end{cases}$$

De plus, l'adjoint vérifie les équations

$$\begin{cases} \dot{\lambda}_1 = \frac{-\partial H}{\partial V_x} = 0 \\ \dot{\lambda}_2 = \frac{-\partial H}{\partial V_y} = 0 \\ \dot{\lambda}_3 = \frac{-\partial H}{\partial \theta} = -\frac{I_{sp} g_0}{M} (\lambda_1 \quad \lambda_2) \begin{pmatrix} -\sin \theta & -\cos \theta \\ \cos \theta & -\sin \theta \end{pmatrix} \begin{pmatrix} \mu_{11} + \mu_{12} - (\mu_{21} + \mu_{22}) \cos \alpha_r \\ (\mu_{22} - \mu_{21}) \sin \alpha_r \end{pmatrix} \\ \dot{\lambda}_4 = \frac{-\partial H}{\partial q} = -\lambda_3 \end{cases}$$

Ainsi, λ_1 et λ_2 sont constant.

Par ailleurs, on remarque que $\begin{pmatrix} -\sin \theta & -\cos \theta \\ \cos \theta & -\sin \theta \end{pmatrix} = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix}$.

Ainsi, on a $\dot{\lambda}_3 = -(\lambda_1 \quad \lambda_2) \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} \vec{V} = (-\lambda_2 \quad \lambda_1) \vec{V}$

$$\dot{\lambda}_3 = \lambda_1 \dot{V}_y - \lambda_2 \dot{V}_x \quad (32)$$

On a alors
$$|\dot{\lambda}_3| \leq \sqrt{\lambda_1^2 + \lambda_2^2} \left\| \dot{\vec{V}} \right\| \leq \sqrt{\lambda_1^2 + \lambda_2^2} \frac{I_{sp} g^0}{M} \bar{\mu}$$

D'où
$$|\lambda_3 - \lambda_{30}| \leq \sqrt{\lambda_1^2 + \lambda_2^2} \frac{I_{sp} g^0}{M} \bar{\mu} t$$

$$\begin{aligned} \lambda_{30} - \sqrt{\lambda_1^2 + \lambda_2^2} \frac{I_{sp} g^0}{M} \bar{\mu} t &\leq \lambda_3 \leq \lambda_{30} + \sqrt{\lambda_1^2 + \lambda_2^2} \frac{I_{sp} g^0}{M} \bar{\mu} t \\ -\lambda_{30} - \sqrt{\lambda_1^2 + \lambda_2^2} \frac{I_{sp} g^0}{M} \bar{\mu} t &\leq \dot{\lambda}_4 \leq -\lambda_{30} + \sqrt{\lambda_1^2 + \lambda_2^2} \frac{I_{sp} g^0}{M} \bar{\mu} t \\ -\lambda_{30} t - \frac{1}{2} \sqrt{\lambda_1^2 + \lambda_2^2} \frac{I_{sp} g^0}{M} \bar{\mu} t^2 &\leq \lambda_4 - \lambda_{40} \leq -\lambda_{30} t + \frac{1}{2} \sqrt{\lambda_1^2 + \lambda_2^2} \frac{I_{sp} g^0}{M} \bar{\mu} t^2 \\ -|\lambda_{30}| t - \frac{1}{2} \sqrt{\lambda_1^2 + \lambda_2^2} \frac{I_{sp} g^0}{M} \bar{\mu} t^2 &\leq \lambda_4 - \lambda_{40} \leq |\lambda_{30}| t + \frac{1}{2} \sqrt{\lambda_1^2 + \lambda_2^2} \frac{I_{sp} g^0}{M} \bar{\mu} t^2 \end{aligned} \quad (33)$$

Ainsi, on obtient le domaine de recherche pour λ_{40} :

$$\lambda_{40} \in \frac{[u] - [u]}{d_1 - d_2} + (|\lambda_{30}| t_f + \frac{1}{2} \sqrt{\lambda_1^2 + \lambda_2^2} \frac{I_{sp} g^0}{M} \bar{\mu} t_f^2) [-1; 1] \quad (34)$$

Comme λ_1 et λ_2 sont constants, ces deux composantes restent dans l'intervalle calculé précédemment.

Par ailleurs, on remarque que l'intervalle de recherche de λ_{40} fait intervenir le temps de trajet pour la trajectoire optimale. Or nous ne connaissons pas cette valeur. Nous pouvons alors utiliser le majorant de t_f que nous avons calculé dans le paragraphe III.

5.2.4 Résultats obtenus

La méthode a été testée en Visual C++, avec les valeurs :

➤ $\bar{\mu} = 2.5$

➤ $I_{sp} = 300$

➤ $M = 2500$

➤ $J = 2000$

➤ $l = 1.5$

➤ $R = 1$

➤ $r = 0.5$

➤ $\alpha_r = \frac{\pi}{4}$

➤ $x_0 = \begin{pmatrix} -10 \\ 15 \\ 0 \\ 0 \end{pmatrix}$

Nous obtenons $D = \begin{pmatrix} [-6.01; 6.01] \\ [-6.01; 6.01] \\ \{1\} \\ [-3.40; 3.40] \end{pmatrix}$ et $D_0 = \begin{pmatrix} [-6.01; 6.01] \\ [-6.01; 6.01] \\ \{1\} \\ [-1436.82; 1436.82] \end{pmatrix}$.

VI Conclusion

Finalement, comme nous avons pu le voir, l'utilisation d'un algorithme génétique pour la recherche d'adjoints initiaux peut soulever quelques problèmes, auxquels on peut d'ores et déjà donner une méthode pour les résoudre :

La recherche d'un intervalle de temps sur lequel on intègre les équations différentielles peut se faire en majorant le temps de trajet optimal, ce majorant se calculant en évaluant le temps de parcours nécessaire pour une trajectoire particulière (si une trajectoire nous permet de nous retrouver dans un état cible, son temps de parcours est forcément plus long que celui de la trajectoire optimale).

La méthode de recherche de la commande optimale, lorsque l'état et l'adjoint sont donnés est différente selon que l'on cherche à minimiser l'énergie nécessaire ou non :

- Lorsque l'on n'optimise que le temps de trajet, un simple algorithme du simplexe peut suffire.
- Si l'on cherche aussi à optimiser l'énergie consommée, on peut appliquer la méthode mise en place dans le paragraphe IV.

Enfin, la localisation de l'adjoint initial dans une zone de recherche est aussi différente selon les cas :

- Lorsque l'énergie consommée n'est pas prise en compte, on peut localiser l'adjoint initial dans une boule dont le rayon est choisi arbitrairement.
- Si l'énergie consommée est prise en compte, alors l'adjoint initial se situe dans un pavé permettant de générer toutes les trajectoires possibles sur un intervalle de temps fini.

Au moment où ce mémoire est rédigé, le stage est encore loin d'être fini. Ainsi, nous n'avons pas encore pu tester l'algorithme génétique (en effet, en parallèle, je mets en place une plateforme permettant de tester les résultats obtenus).

Par ailleurs, il reste à définir le critère de sélection, c'est-à-dire la fonction de la distance minimale et du temps mis pour approcher au mieux l'état final.

Enfin, nous pourrions par la suite réfléchir sur une méthode mixte de recherche de l'adjoint optimal. En effet, si les algorithmes génétiques convergent très rapidement lors des premières itérations, la vitesse de convergence ralentit très vite par la suite. Ainsi, il pourrait être intéressant de combiner un algorithme génétique (sur les premières itérations, afin de localiser un optimum global dans une zone restreinte) avec une méthode d'optimisation plus classique (et plus rapide, mais qui ne donne généralement qu'un optimum local).

Bibliographie

- Boudarel, Delmas, Guichet, « *Commande Optimale des processus* »
- P.Borne, G.Dauphin-Tanguy, J-P. Richard, F.Rotella, I.Zambettakis, « *Commande et optimisation des processus* », 1990

Remerciements

- Frédéric Damongeot et Bernard Sacleux pour m’avoir encadré pendant ce stage
- Toute l’équipe au sein de laquelle je travaille pour son accueil.