

Contrôle robuste inspiré par l'être humain de mouvements d'atteinte dans un champ de force

Mémoire présenté par
Francesco SICURELLA

en vue de l'obtention du grade de Master
Ingénieur civil électromécanicien

Promoteur(s)
Frédéric CREVECŒUR, Philippe LEFÈVRE

Lecteur(s)
Julien HENDRICKX, François HENROTTE

Année académique 2016-2017



Table des matières

Remerciements	1
Abstract	2
1 Introduction	3
2 Modélisation mathématique d'un système dynamique	8
2.1 Représentation d'état	8
2.1.1 Cas général	8
2.1.2 Cas discret	9
2.1.3 Cas linéaire	10
2.1.4 Cas linéaire avec état augmenté	11
2.2 Signal de mesure	12
2.3 Bruits de traitement et de mesure	13
2.4 Fonction de coût	14
2.5 Discussion	15
3 Contrôle et estimation optimaux	17
3.1 Estimation optimale	18
3.1.1 Critère du minimum de variance	18
3.1.2 Filtre de Kalman	18
3.2 Contrôle optimal	19
3.2.1 Information complète	19
3.2.2 Information incomplète	20
3.3 Discussion	21
4 Modélisation linéaire du bras	23
4.1 Masse ponctuelle à une dimension	24
4.1.1 Équation dynamique	24
4.1.2 Équation musculaire	25
4.1.3 Représentation d'état	26
4.2 Masse ponctuelle à deux dimensions	27
4.3 Discussion	28

5	Valeur des paramètres biomécaniques	30
5.1	Paramètres viscoélastiques k_e et k_v	31
5.2	Constante de temps des muscles τ	33
5.3	Taux de corrélation λ	33
5.4	Discussion	34
6	Récapitulatif	36
7	Calibration de la simulation	38
7.1	Paramètres de la simulation	38
7.1.1	Paramètres des bruits	39
7.1.2	Paramètres de la fonction de coût	39
7.2	Structure du programme	41
7.3	Simulation	42
7.3.1	Paramètres initiaux	42
7.3.2	Variation des paramètres	43
7.4	Discussion	46
8	Modélisation de la perturbation	49
8.1	Modèle de la perturbation	49
8.2	Simulation	51
8.3	Variation des paramètres	53
8.4	Discussion	54
9	Exploration de la fonction de coût	57
9.1	Fonction de coût croissante	57
9.2	Simulation	58
9.3	Variation des paramètres	61
9.4	Discussion	62
10	Modélisation du délai	63
10.1	Contrôle avec délai de mesure	63
10.2	Simulation	65
10.3	Contrôle avec feedback sur les erreurs d'estimation	66
10.4	Discussion	68
11	Modélisation découplée du bras	70
11.1	Modèle cinématique inverse	70
11.2	Équations dynamiques et paramètres utilisés	71
11.3	Simulation	72
11.4	Discussion	74
12	Perspectives et pistes d'amélioration	75
12.1	Perspectives	75
12.2	Pistes d'amélioration	76
13	Conclusion	77

Annexes	81
A Programmation dynamique	81
A.1 Équation de Bellman	81
A.1.1 Cas déterministe	81
A.1.2 Cas stochastique	82
B Modélisation non-linéaire du bras	84
C Paramètres utilisés dans la littérature	86
C.1 Modèles utilisés	86
C.2 Paramètres biomécaniques	86
C.3 Paramètres de simulation	87
D Résultats des simulations	88
E Robustesse du contrôleur aux variations de paramètres	90
F Code GNU Octave	91
Bibliographie	97

Remerciements

La réalisation d'un mémoire est un travail de longue haleine qui n'aurait pas été possible sans le concours d'autres personnes. Je tenais par conséquent à leur exprimer ici mes remerciements.

Je remercie évidemment chaleureusement M. Crevecœur, qui m'a fait partager le sujet de ses recherches et qui m'a guidé dans mon travail tout au long de cette année, et ce avec beaucoup de disponibilité et de motivation.

Je remercie également M. Lefèvre pour avoir supervisé le déroulement de ce mémoire ainsi que les membres du jury MM. Henrotte et Hendrickx qui ont accepté de consacrer du temps à la lecture et à l'évaluation de celui-ci.

Mais un mémoire est également l'aboutissement de tout un parcours académique, et à ce titre je remercie mes parents, Angela et Salvatore, qui m'ont permis de réaliser des études universitaires.

Par ailleurs j'aimerais profiter de l'occasion afin d'adresser une pensée à mes professeurs de secondaire : M. Jamar, qui m'a poussé à continuer à étudier les mathématiques, ainsi que M. Bernimont, pour m'en avoir donné le goût. Sans eux, ce mémoire aurait probablement eu pour sujet la littérature française plutôt que l'ingénierie!

Enfin je remercie mes amis ainsi que mes cokoteurs pour m'avoir soutenu tout au long de l'élaboration de ce mémoire et je vous souhaite à tous une excellente lecture!

Abstract

Le système nerveux est capable de contrôler de manière robuste des mouvements d'atteinte du bras en présence d'un champ de force perturbateur, et ce malgré les bruits et les délais inhérents au système sensorimoteur. La théorie du contrôle permet de fournir un cadre mathématique rendant possible l'étude et la simulation de tels mouvements asservis à une stratégie de contrôle optimal en boucle fermée, donnant lieu à l'hypothèse selon laquelle le cerveau utilise effectivement un mécanisme similaire lors de la planification de mouvements [Shadmehr, 1994]. Cependant l'introduction de délai dans un système asservi le rend rapidement instable. Par ailleurs, des études récentes [Crevecœur, 2014] suggèrent que les paramètres viscoélastiques du bras sont en réalité bien moins élevés que ceux habituellement utilisés dans la littérature, invalidant le rôle prépondérant de la raideur des muscles dans la robustesse du mouvement. Dans ce mémoire nous suivons une approche empirique afin de synthétiser un contrôleur robuste incluant les contraintes du système moteur pour des mouvements d'atteinte dans un champ de force. À l'aide d'un modèle simplifié du bras, nous sommes parvenus à obtenir un contrôle performant et stable en modifiant la fonction de coût utilisée et en implémentant une action corrective tenant compte du délai. Ces résultats soulignent la vraisemblance de l'utilisation d'une telle stratégie par le système nerveux et apportent une solution au problème du contrôle optimal dans les systèmes avec délai.

The nervous system is able to perform robust reaching movements of the upper-limb in presence of a force field perturbation despite the noises and delays inherent in the sensorimotor system. Control theory gives us a mathematical framework allowing us to study and simulate such movements using an optimal feedback control law strategy, which raises the hypothesis stating that the brain actually uses a similar mechanism to plan movements [Shadmehr, 1994]. However, the introduction of delay in a feedback system quickly leads to instability. Besides, recent studies [Crevecœur, 2014] have suggested that the viscoelastic parameters of the arm are in fact smaller than the ones used in literature, invalidating the leading role of the stiffness of the muscles in the robustness of the movement. In this thesis we follow an empirical approach in order to synthesize a robust controller including the constraints of the motor system for reaching movements in a force field. With the help of a simplified model of the arm we managed to obtain an efficient and stable control by modifying the cost function used and by implementing a corrective action taking delay into account. The results stress the plausibility of use of such a strategy by the nervous system and give a solution to the problem of optimal control in time-delay systems.

En 1994, Shadmehr et Mussa-Ivaldi publiaient un article intitulé « *Adaptive Representation of Dynamics during Learning of a Motor Task* » [Shadmehr, 1994] investiguant la manière dont le système nerveux contrôle des mouvements d’atteinte (« reaching movements ») du bras lorsque celui-ci est soumis à un champ de force initialement inconnu. Les sujets participant à l’expérience devaient, par des mouvements du bras, atteindre avec la main des cibles apparaissant dans leur champ visuel à distance fixe mais avec une orientation variable. Le bras des participants était solidaire d’un manipulateur robotique, dépeint à la figure 1.1, contraignant le mouvement dans un plan et appliquant lors de certains essais, des forces perturbatrices de magnitude proportionnelle à la vitesse du bras.

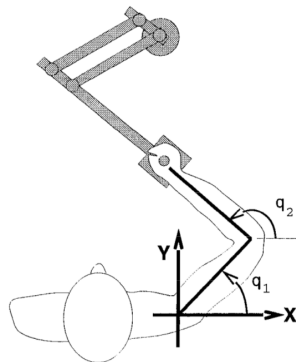


FIGURE 1.1 – Dessin représentant le bras d’un sujet attaché à un manipulateur robotique lors d’expériences sur les mouvements d’atteinte. Source : [Shadmehr, 1994].

Ainsi que l’illustre la gauche de la figure 1.2, les résultats de cette étude ont montré que lorsque aucune perturbation n’est présente les trajectoires du bras consistent essentiellement en des lignes droites à la cible. Cependant, lorsque le champ de force est actif, les trajectoires se trouvent sévèrement perturbées, nécessitant alors un mouvement correctif de la part du bras afin d’atteindre la cible. Cela conduit aux trajectoires de la droite de la figure 1.2 qui exhibent un profil caractéristique dit « en crochet ».

Cependant après plusieurs essais dans lesquels un champ de force est présent, les trajectoires du bras tendent à se rapprocher de nouveau d’une ligne droite. Nous observons là un phénomène d’apprentissage : les participants apprivoisent progressivement la nouvelle dynamique introduite par la perturbation et sont finalement capables d’atteindre la cible avec le trajet direct optimal.

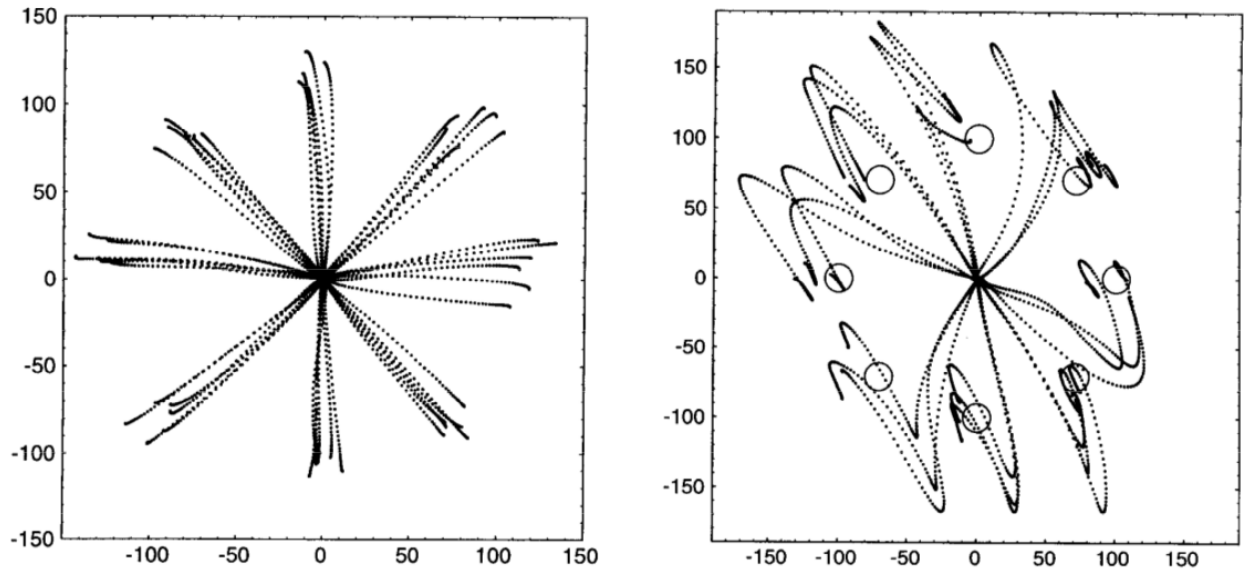


FIGURE 1.2 – Trajectoires typiques lors de mouvements d’atteinte pour des cibles situées à une distance de 10 cm et dont l’orientation est un multiple de 45° . Gauche : en l’absence de perturbation, les trajectoires sont des lignes droites à la cible. Droite : lorsqu’un champ de force est présent, les trajectoires ont un profil en crochet. Les déplacements sont indiqués en mm. Source : [Shadmehr, 1994].

Enfin, si après cette phase d’apprentissage le champ de force est alors retiré, nous observons des trajectoires en crochet mais avec des crochets cette fois dirigés dans le sens opposé. Ces derniers résultats prouvent l’existence d’un effet rémanent lié à l’apprentissage : le système nerveux, qui a intégré le modèle de la perturbation, continue à compenser ses mouvements comme si celle-ci était toujours présente.

Dans ce mémoire, nous allons laisser de côté la question de l’apprentissage pour nous concentrer exclusivement sur ce qui se produit lorsque les participant sont pour la première fois confrontés au champ de force : comment le système nerveux réussit-il, malgré une perturbation présente et inconnue, à coordonner la dynamique du bras afin d’atteindre tout de même la cible ?

Comme nous l’avons vu, le bras effectue en cours de route un mouvement correctif qui lui permet de rectifier sa course. Une interprétation simple pour expliquer ce mouvement en crochet consiste à dire que celui-ci est le produit de la succession de deux mouvements distincts. D’après ce point de vue, le système nerveux, n’ayant initialement pas conscience du champ de force, planifierait un premier mouvement du bras visant à atteindre la cible en ligne droite. Le bras entame alors son déplacement mais est bientôt dévié sous l’effet du champ de force : le système nerveux, changeant complètement sa planification initiale, envisagerait alors un second mouvement permettant de corriger la trajectoire afin d’atteindre la cible.

Mais une autre explication est également possible. L’excitante hypothèse est que ce mouvement en crochet ne serait pas la conséquence d’une reprogrammation explicite du mouvement mais plutôt le résultat d’une tactique de contrôle robuste. Dans ce scénario, la trajectoire résultante de la main ne proviendrait que d’une seule manœuvre du bras suivant une planification motrice initiale, cette planification intégrant dès le départ un mécanisme permettant de réagir aux perturbations rencontrés.

Ce nouveau point de vue a en quelque sorte joué le rôle d’un fer de lance dans le domaine de la recherche sur le contrôle moteur en neurosciences. Par la suite de nombreux chercheurs se sont intéressés à cette problématique, les dispositifs permettant d’investiguer les mouvements d’atteinte du bras se sont perfectionnés

(voir par exemple le dispositif « *KINARM* » de la figure 1.3) et les études réalisées ultérieurement ont donné de nouveaux indices de l'emploi par le système nerveux de stratégies de contrôle robuste.



FIGURE 1.3 – Photographie du « *KINARM End-Point Lab* » mis au point par l'entreprise *BKIN Technologies* et permettant d'étudier les mouvements d'atteinte du bras. Source : <http://www.bkintechologies.com/bkin-products/kinarm-end-point-lab/>.

Toutefois, bien que Shadmehr et Mussa-Ivaldi aient joué le rôle de pionniers dans le domaine et soient parvenus à effectuer des simulations numériques encourageantes, de nombreuses contraintes liées à la physiologie du corps humain n'ont pas été prises en compte dans leur modèle. Dans un article de 2014, Crevecœur et Scott [Crevecœur, 2014] ont notamment mis en évidence que les paramètres viscoélastiques communément admis dans la littérature, et utilisés par Shadmehr et Mussa-Ivaldi dans leur modèle, surestiment en réalité largement la raideur et la viscosité réelles des muscles. D'autre part, leur modèle n'a pas tenu compte des délais inhérents au système sensorimoteur : le temps nécessaire à ce que le système nerveux reçoive l'information sensorielle, la traite et renvoie une commande motrice aux muscles induit en effet un écart d'environ 50 ms entre une mesure effectuée et la réaction correspondante.

Il s'agit de problématiques importantes concernant la validité de l'hypothèse d'un contrôle robuste. Le modèle de Shadmehr et Mussa-Ivaldi se base sur une raideur importante du bras afin de compenser les perturbations introduites par le champ de force. Mais si cette raideur est nettement diminuée, le bras ne peut plus compter dessus afin d'assurer la fiabilité du mouvement. Quant aux délais, ceux-ci sont tristement célèbres en théorie du contrôle pour leur capacité à rendre les systèmes asservis instables : la problématique du contrôle optimal en présence de délais est toujours actuellement une question ouverte.

Et effectivement une fois les paramètres viscoélastiques ajustés et le délai introduit dans le modèle du bras, les simulations deviennent instables. Le contrôleur ne parvient pas à amener la main à la cible et les mouvements du bras évoluent erratiquement. Or, un tel comportement chaotique n'est en pratique jamais observé chez des sujets ne présentant pas de déficit moteur. Lors d'expériences par exemple, le bras des participants ne se met pas à tourner dans tous les sens dès qu'une perturbation est présente.

Le système nerveux est donc bien capable d'effectuer un contrôle moteur correct malgré la présence de perturbation et de délais, et ce malgré la faible raideur des muscles en mouvement. Dès lors est-il possible de réconcilier les résultats donnés par la théorie du contrôle optimal avec la réalité biologique du système moteur ?

Dans ce mémoire, je tenterai d'approfondir la voie ouverte par Shadmehr et Mussa-Ivaldi, et en me basant sur la littérature en ayant découlé, d'apporter une réponse à cette problématique en adoptant une méthodologie empirique. Le but étant de parvenir à synthétiser un contrôleur robuste et stable compatible avec les contraintes biologiques liées au corps humain et appliqué à des mouvements d'atteinte du bras.

Outre le renforcement de l'hypothèse de Shadmehr, la mise au point d'un tel contrôleur constituerait une avancée mathématique intéressante en théorie du contrôle, permettrait une meilleure compréhension du contrôle moteur dans le domaine des neurosciences, et trouverait peut-être un jour, je l'espère, des applications en robotique ou en génie biomédical.

La difficulté d'une telle entreprise, mais c'est aussi ce qui constitue tout son intérêt, est qu'elle nécessite l'intégration de nombreuses disciplines, dont quelques-unes sont esquissées à la figure 1.4. La première partie de ce mémoire sera donc consacrée à la mise au point des bases théoriques nécessaires à la modélisation du bras, tandis que la seconde consistera en l'application et l'exploration de ces outils à travers une simulation numérique.

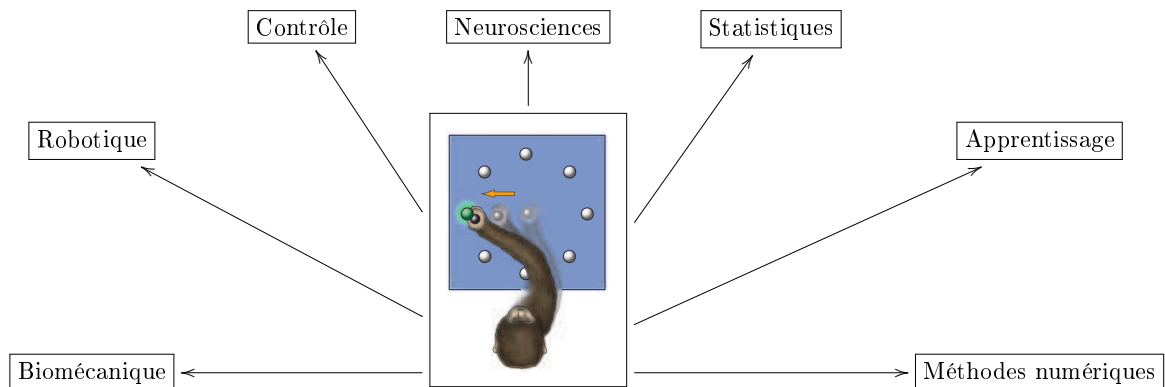


FIGURE 1.4 – La modélisation de mouvements d'atteinte requiert l'intégration de nombreuses disciplines. Source de l'image centrale : [Purves, 2015].

Le chapitre 2 introduira le formalisme mathématique permettant de décrire l'évolution d'un système dynamique soumis à une loi de contrôle. Dans le chapitre 3 nous utiliserons les résultats mathématiques connus en théorie du contrôle afin de dégager une stratégie optimale de contrôle sur notre système.

L'étape suivante consistera alors à mettre au point un modèle mathématique du bras : ce sera l'objet du chapitre 4, où nous réduirons la dynamique de la main à celle du masse ponctuelle en mouvement. Le chapitre 5 discutera quant à lui la valeur des paramètres biomécaniques à appliquer à notre modèle.

Avant d'attaquer la suite du mémoire, les principaux résultats de ces quatre premiers chapitres, constituant le socle sur lequel notre simulation numérique sera bâtie, seront résumés en un bref chapitre, le chapitre 6.

Une fois tous ces outils mis en place, nous arriverons alors au cœur de ce mémoire. Dans le chapitre 7, nous déciderons des paramètres utilisés par notre programme et effectuerons nos premières simulations en l'absence de champ de force. Cela nous permettra notamment d'étudier l'influence de ces différents paramètres sur le comportement de notre modèle. Le chapitre 8 étudiera alors l'ajout d'un champ de force perturbateur sur les trajectoires simulées.

Ensuite, nous essayerons dans le chapitre 9 de modifier les critères de contrôle utilisés par notre contrôleur afin de rendre celui-ci plus performant. Concrètement, cela se traduira par un changement de la fonction de coût utilisée par le contrôleur. Il ne nous restera alors plus qu'à introduire du délai dans le système au chapitre 10 pour tenter de mettre au point une loi de contrôle à la fois stable et robuste.

Par après, le chapitre 11 nous permettra de valider une dernière fois les résultats des chapitres précédents en les testant sur un modèle à deux articulations du bras ignorant les effets de couplage entre ces articulations.

Pour terminer, nous prendrons un peu de recul au chapitre 12 afin d'envisager les perspectives suscitées par les résultats obtenus ainsi que les pistes d'amélioration pouvant être apportées. Ce mémoire se clôturera alors par la conclusion du chapitre 13.

Modélisation mathématique d'un système dynamique

En général, la modélisation d'un système dynamique requiert la résolution d'un système d'équations différentielles. Par exemple, le modèle du bras que nous développerons au chapitre 4 sera décrit par un système de 4 équations différentielles ordinaires (EDO) couplées entre elles :

$$\begin{cases} m \cdot \ddot{x} + k_v \cdot \dot{x} + k_e \cdot x & = F_x + \lambda \cdot F_y \end{cases} \quad (2.1)$$

$$\begin{cases} m \cdot \ddot{y} + k_v \cdot \dot{y} + k_e \cdot y & = F_y + \lambda \cdot F_x \end{cases} \quad (2.2)$$

$$\begin{cases} \tau \cdot \dot{F}_x + F_x & = u_x \end{cases} \quad (2.3)$$

$$\begin{cases} \tau \cdot \dot{F}_y + F_y & = u_y. \end{cases} \quad (2.4)$$

Puisque notre but est de modéliser la dynamique d'un bras en mouvement, nous allons dresser dans ce chapitre la formulation et les outils mathématiques nous permettant d'y arriver en nous inspirant du livre de référence d'Åström et Murray sur les systèmes en boucle fermée [Aström, 2009], du syllabus d'Evans sur la théorie du contrôle optimal [Evans, 1983] ainsi que des notes du cours de modélisation des systèmes biologiques donné à l'UCL [Crevecoeur, 2016].

La majeure partie de ce chapitre est dédiée à la description via une représentation d'état d'un système dynamique contrôlé, avec discrétisation, linéarisation et état augmenté. Puisque la position du bras n'est connu par le système nerveux qu'à travers les informations sensorielles que celui-ci reçoit du système visuel et du système proprioceptif, nous introduirons ensuite un signal de mesure dans le système. Comme ces mesures sont toujours entachées d'un certain niveau de bruit et que c'est aussi le cas du signal moteur envoyé par le cerveau par le muscle, il nous faudra alors ajouter également ces bruits dans le système. Nous mettrons ensuite en place la fonction de coût qui évaluera la qualité de l'état du système et dictera au contrôleur, dans le chapitre 3, les critères devant influencer l'évolution du système. Nous concluerons ensuite en explicitant et en discutant le formalisme Linéaire-Quadratique-Gaussien (LQG) ainsi mis en place.

2.1 Représentation d'état

2.1.1 Cas général

Nous considérons un système dynamique dont l'évolution en fonction du temps t dépend de l'état actuel du système, dont l'information est contenue dans le vecteur d'état $\mathbf{x}$, et soumis à une loi de contrôle $\mathbf{u}$. L'évolution d'un tel système peut être décrit par un système d'EDO de la forme

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)) \quad (2.5)$$

avec

$$\mathbf{x} \in \mathbb{R}^a \quad : \quad \text{le vecteur d'état,} \quad (2.6)$$

$$\mathbf{u} \in \mathbb{R}^b \quad : \quad \text{le vecteur de contrôle,} \quad (2.7)$$

et où $\mathbf{f}$ est une fonction compatible avec les dimensions des différents vecteurs ($\mathbf{f} : \mathbb{R}^a \times \mathbb{R}^b \rightarrow \mathbb{R}^a$). Une telle représentation est appelée modèle ou *représentation d'état*.

Nous posons que le système évolue sur l'intervalle de temps fini $t \in [0, t_f]$ et que sa condition initiale

$$\mathbf{x}(0) = \mathbf{x}_0 \quad (2.8)$$

est connue.

Résoudre l'équation 2.5 consiste à déterminer $\forall t \in]0, t_f]$ la valeur de $x(t)$ à partir de la condition initiale $\mathbf{x}(0)$, du contrôle $\mathbf{u}(t)$ connu à tout instant et de la fonction $\mathbf{f}(\cdot, \cdot)$ connue. Une manière d'envisager un tel système dynamique est de considérer le contrôle $\mathbf{u}$ comme étant l'entrée du système et l'état $\mathbf{x}$ comme étant sa sortie, le contrôle appliqué déterminant la réponse du système ainsi que l'illustre le schéma-bloc de la figure 2.1.

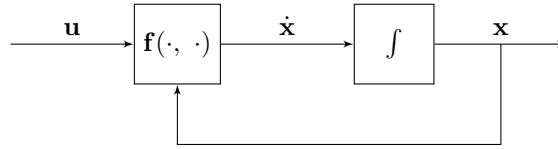


FIGURE 2.1 – Représentation en schéma-bloc d'un système d'EDO général.

2.1.2 Cas discret

Si une solution analytique de l'équation 2.5 n'existe pas ou est trop difficile à calculer, il est possible de recourir à des méthodes de résolution numériques. Diverses méthodes existent, comme par exemple les méthodes de Taylor, de Runge-Kutta, d'Adams-Bashforth-Moulton ou encore de Gear [Legat, 2012]. Ici, nous utiliserons la méthode d'Euler explicite¹ (méthode de Taylor d'ordre 1) qui consiste à utiliser l'approximation suivante :

$$\mathbf{x}(t_k + \Delta t) = \mathbf{x}(t_k) + \Delta t \cdot \dot{\mathbf{x}}(t_k) + \mathcal{O}(\Delta t^2) \quad (2.9)$$

$$\approx \mathbf{x}(t_k) + \Delta t \cdot \mathbf{f}(\mathbf{x}(t_k), \mathbf{u}(t_k)) \quad (2.10)$$

où $\mathcal{O}(\Delta t^2)$ représente le terme d'erreur.

Ce faisant, il est possible de discrétiser notre équation. Décidant d'utiliser un pas de temps fixe, nous pouvons diviser l'intervalle temporel $[0, t_f]$ en n intervalles de temps égaux

$$\Delta t = \frac{t_f}{n} \quad (2.11)$$

et discrétiser ensuite l'intervalle $[0, t_f]$ en $N = n + 1$ points discrets k tels que

$$\begin{aligned} [k] &= (t_{k-1}) \\ &= (\{k - 1\} \cdot \Delta t) \end{aligned} \quad (2.12)$$

1. Celle-ci consiste à écrire le développement en série de Taylor de $\mathbf{x}(t)$ autour d'un point connu $\mathbf{x}(t_k)$ et à ne garder que les termes d'ordre 0 et 1. Le développement faisant apparaître la dérivée de $\mathbf{x}$, et celle-ci étant connue, il est possible d'estimer la solution à un temps $t_{k+1} = t_k + \Delta t$. En répétant l'opération, on peut alors propager la solution à partir d'une condition initiale.

avec $k \in \{1, 2, \dots, N\}$, où nous insistons sur le fait que l'indice k ne démarre pas à 0 mais bien à 1².

Exemple 2.1. Nous pouvons discrétiser l'intervalle de temps $[0, 0.5]$ s avec un pas de temps $\Delta t = 0.01$ s ce qui nous donne $N = 501$ points discrets. Alors $\mathbf{x}[1] = \mathbf{x}(0)$, $\mathbf{x}[2] = \mathbf{x}(0.01 \text{ s})$, $\dots$, $\mathbf{x}[501] = \mathbf{x}(0.5 \text{ s})$.

Appliquant la méthode d'Euler explicite à notre système d'équations temporelles 2.5, nous trouvons alors une formulation discrète pour un système d'EDO :

$$\mathbf{x}_\Delta[k+1] = \mathbf{x}_\Delta[k] + \Delta t \cdot \mathbf{f}(\mathbf{x}_\Delta[k], \mathbf{u}_\Delta[k]). \quad (2.13)$$

Grâce à cette équation, il est maintenant possible de trouver une solution (approchée) de $\mathbf{x}$ en itérant jusque $k = N$ à partir de la condition initiale

$$\mathbf{x}_\Delta[1] = \mathbf{x}_0. \quad (2.14)$$

2.1.3 Cas linéaire

Parmi les systèmes d'EDO, nous allons nous intéresser plus particulièrement à ceux qui sont dits *linéaires*. En effet, les systèmes d'EDO linéaires ont été abondamment étudiés par les théoriciens du contrôle : de nombreux résultats les concernant existent dans la littérature et ceux-ci présentent des propriétés très particulières comme nous le verrons dans le chapitre suivant lorsque nous aborderons les questions de contrôle et d'estimation optimaux.

En toute généralité, un système linéaire d'EDO s'écrit de la manière suivante :

$$\dot{\mathbf{x}}(t) = \mathbf{A}_t \cdot \mathbf{x}(t) + \mathbf{B}_t \cdot \mathbf{u}(t) \quad (2.15)$$

avec

$$\mathbf{x} \in \mathbb{R}^a : \text{ le vecteur d'état,} \quad (2.16)$$

$$\mathbf{u} \in \mathbb{R}^b : \text{ le vecteur de contrôle,} \quad (2.17)$$

$$\mathbf{A}_t \in \mathbb{R}^{a \times a} : \text{ la matrice dynamique,} \quad (2.18)$$

$$\mathbf{B}_t \in \mathbb{R}^{a \times b} : \text{ la matrice de contrôle,} \quad (2.19)$$

et toujours soumis à une condition initiale

$$\mathbf{x}(0) = \mathbf{x}_0. \quad (2.20)$$

La représentation par schéma-bloc d'un tel système est illustré à la figure 2.2.

Comme dans le cas général, nous pouvons utiliser la méthode d'Euler d'explicite pour discrétiser un système linéaire. En appliquant l'équation 2.13, il vient :

$$\mathbf{x}_\Delta[k+1] = \mathbf{A}_\Delta \cdot \mathbf{x}_\Delta[k] + \mathbf{B}_\Delta \cdot \mathbf{u}[k] \quad (2.21)$$

où $k \in \{1, 2, \dots, N\}$ et

$$\mathbf{A}_\Delta = \mathbf{I}_{a \times a} + \Delta t \cdot \mathbf{A}_t, \quad (2.22)$$

$$\mathbf{B}_\Delta = \Delta t \cdot \mathbf{B}_t, \quad (2.23)$$

2. Ce choix permet d'avoir la même convention que les langages `Matlab` ou `Octave` qui démarrent la numérotation des vecteurs à l'indice 1.

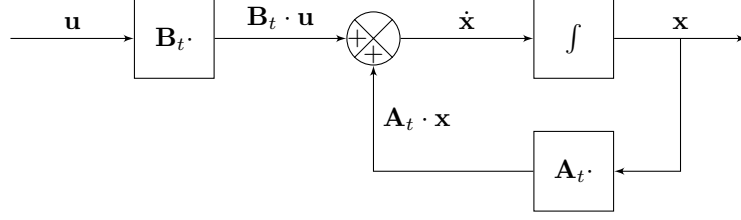


FIGURE 2.2 – Représentation en schéma-bloc d'un système d'EDO linéaire.

avec $\mathbf{I}$ la matrice identité, et soumis à la condition initiale

$$\mathbf{x}_\Delta[1] = \mathbf{x}_0. \quad (2.24)$$

Notons que dans la suite de ce mémoire nous ne nous occuperons plus de la formulation temporelle mais nous nous concentrerons uniquement sur la formulation discrète obtenue par la méthode d'Euler explicite.

Exemple 2.2. Une système masse-ressort-amortisseur soumis à une force F peut être représentée par l'EDO linéaire du second ordre

$$m \cdot \ddot{x}(t) + k_v \cdot \dot{x}(t) + k_e \cdot x(t) = F(t). \quad (2.25)$$

L'état de ce système peut être décrit par le vecteur $\mathbf{x}(t) = \begin{pmatrix} x(t) & \dot{x}(t) \end{pmatrix}^T$ à partir par exemple d'une condition initiale $\mathbf{x}(0) = \begin{pmatrix} 0 & 0 \end{pmatrix}^T$. Dans ce cas nous trouvons la représentation d'état linéaire :

$$\underbrace{\begin{pmatrix} \dot{x} \\ \ddot{x} \end{pmatrix}}_{\dot{\mathbf{x}}} = \underbrace{\begin{pmatrix} 0 & 1 \\ -k_e/m & -k_v/m \end{pmatrix}}_{\mathbf{A}_t} \cdot \underbrace{\begin{pmatrix} x \\ \dot{x} \end{pmatrix}}_{\mathbf{x}} + \underbrace{\begin{pmatrix} 0 \\ 1/m \end{pmatrix}}_{\mathbf{B}_t} \cdot \underbrace{F}_{\mathbf{u}}. \quad (2.26)$$

Les matrices après discrétisation deviennent alors

$$\mathbf{A}_\Delta = \begin{pmatrix} 1 & \Delta t \\ -\Delta t \cdot k_e/m & 1 - \Delta t \cdot k_v/m \end{pmatrix}, \quad (2.27)$$

$$\mathbf{B}_\Delta = \begin{pmatrix} 0 \\ \Delta t/m \end{pmatrix}. \quad (2.28)$$

2.1.4 Cas linéaire avec état augmenté

Le but du contrôle $\mathbf{u}$ va être d'amener notre système de son état initial $\mathbf{x}_0$ à un état final souhaité que nous noterons $\mathbf{x}^*$, l'erreur finale sur l'état étant donnée par la valeur de $\mathbf{x}[N] - \mathbf{x}^*$. Dans la section 2.4, nous allons utiliser une fonction de coût afin de caractériser la performance de notre contrôle. Pour permettre à cette fonction de coût de dépendre à la fois de l'état courant $\mathbf{x}[k]$ et de l'état final souhaité $\mathbf{x}^*$, il existe une manière élégante qui consiste à combiner ces deux vecteurs en un seul pour former ce que nous allons appeler un *vecteur d'état augmenté*.

Nous formons par conséquent le vecteur d'état augmenté

$$\mathbf{x}[k] = \begin{pmatrix} \mathbf{x}_\Delta[k] \\ \mathbf{x}^* \end{pmatrix}. \quad (2.29)$$

L'équation différentielle discrète 2.21 se réécrit alors en fonction du vecteur d'état augmenté

$$\mathbf{x}[k+1] = \mathbf{A} \cdot \mathbf{x}[k] + \mathbf{B} \cdot \mathbf{u}[k] \quad (2.30)$$

où

$$\mathbf{A}_{2a \times 2a} = \begin{pmatrix} \mathbf{A}_\Delta & \mathbf{0}_{a \times a} \\ \mathbf{0}_{a \times a} & \mathbf{I}_{a \times a} \end{pmatrix}, \quad (2.31)$$

$$\mathbf{B}_{2a \times b} = \begin{pmatrix} \mathbf{B}_\Delta \\ \mathbf{0}_{a \times b} \end{pmatrix}, \quad (2.32)$$

avec $\mathbf{0}$ la matrice d'éléments nuls.

Exemple 2.3. Pour le système masse-ressort-amortisseur, si nous souhaitons atteindre une cible située à 10 cm avec une vitesse nulle, le vecteur d'état final souhaité est $\mathbf{x}^* = (0.1 \text{ m} \quad 0 \text{ m/s})^T$. Nous pouvons former alors le vecteur d'état augmenté

$$\mathbf{x}[k] = (x[k] \quad \dot{x}[k] \quad 0.1 \text{ m} \quad 0 \text{ m/s})^T. \quad (2.33)$$

Les matrices discrètes augmentées deviennent alors

$$\mathbf{A} = \begin{pmatrix} 1 & \Delta t & 0 & 0 \\ -\Delta t \cdot k_e/m & 1 - \Delta t \cdot k_v/m & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad (2.34)$$

$$\mathbf{B} = \begin{pmatrix} 0 \\ \Delta t/m \\ 0 \\ 0 \end{pmatrix}. \quad (2.35)$$

2.2 Signal de mesure

L'état d'un système n'est bien souvent connu que via des mesures qui vont rendre compte en tout ou en partie de cet état. Dans le cas de mouvement d'atteinte du bras par exemple, le système nerveux ne connaît l'état de celui-ci (position, vitesse, accélération, force, etc.) que via les informations qu'il reçoit du système sensoriel, et notamment celles en provenance du système visuel et du système proprioceptif.

Afin de modéliser mathématiquement ces mesures effectuées sur le système, nous introduisons un vecteur de mesure $\mathbf{y}$ et considérons que l'observation aussi est linéaire par rapport à l'état³, soit

$$\mathbf{y}[k] = \mathbf{H} \cdot \mathbf{x}[k] \quad (2.36)$$

avec

$$\mathbf{y} \in \mathbb{R}^a : \text{ le vecteur de mesure,} \quad (2.37)$$

$$\mathbf{H} \in \mathbb{R}^{a \times 2a} : \text{ la matrice d'observation.} \quad (2.38)$$

3. Notons que l'observation pourrait aussi être potentiellement influencée par le contrôle appliqué via un terme direct $\mathbf{D}$: $\mathbf{y}[k] = \mathbf{H} \cdot \mathbf{x}[k] + \mathbf{D} \cdot \mathbf{u}[k]$.

2.3 Bruits de traitement et de mesure

Pour terminer, nous allons introduire des composantes stochastiques dans notre système dynamique, qui jusqu'à présent était déterministe, afin de modéliser les fluctuations aléatoires inhérentes à certains systèmes.

Physiologiquement, ces bruits se traduisent dans le système moteur par le fait que les commandes motrices envoyées aux muscles présentent toujours une certaine variabilité (bruit dynamique) et qu'il en va de même pour les informations sensorielles reçues (bruit de mesure).

Par conséquent, nous considérons que du bruit entache non seulement la mesure du système mais également sa dynamique, faisant de $\mathbf{x}$ et de $\mathbf{y}$ des processus stochastiques. Nous considérons de plus que ces bruits sont des bruits additifs :

$$\begin{cases} \mathbf{x}[k+1] &= \mathbf{A} \cdot \mathbf{x}[k] + \mathbf{B} \cdot \mathbf{u}[k] + \xi[k] \\ \mathbf{y}[k] &= \mathbf{H} \cdot \mathbf{x}[k] + \omega[k], \end{cases} \quad (2.39)$$

avec

$$\xi \in \mathbb{R}^a : \text{ le bruit de traitement,} \quad (2.41)$$

$$\omega \in \mathbb{R}^a : \text{ le bruit de mesure.} \quad (2.42)$$

Afin d'illustrer ce système d'équations, un système temporel correspondant sous forme de schéma-bloc est représenté à la figure 2.3.

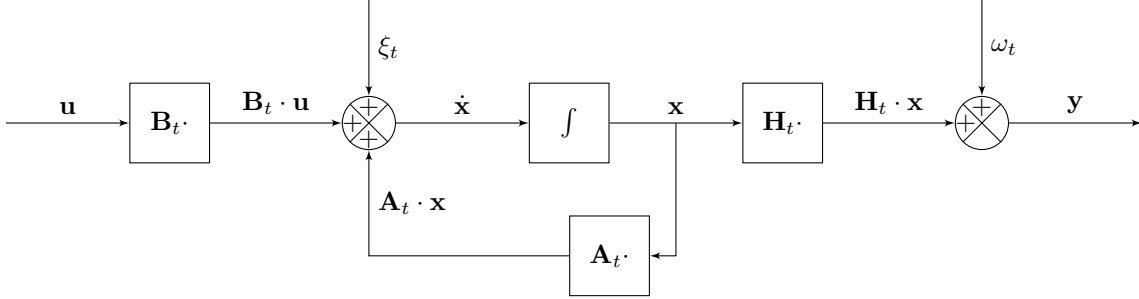


FIGURE 2.3 – Représentation en schéma-bloc d'un système linéaire à bruit gaussien avec mesure de l'état.

Par ailleurs, nous allons poser comme hypothèse simplificatrice que ces deux bruits sont des bruits blancs gaussiens à moyenne nulle et de variance respective $\mathbf{R}_\xi$ et $\mathbf{R}_\omega$:

$$\xi_k \sim \mathcal{N}(0, \mathbf{R}_\xi), \quad (2.43)$$

$$\omega_k \sim \mathcal{N}(0, \mathbf{R}_\omega). \quad (2.44)$$

Pointons le fait que les bruits de traitement et de mesure sont considérés comme décorrélés, et entre eux, et avec l'état $\mathbf{x}$ ($\mathbf{R}_{\xi\omega}, \mathbf{R}_{\xi x}, \mathbf{R}_{\omega x} = 0$). D'autre part leur variance est donnée par

$$\mathbf{R}_\xi[j, k] = R_\xi \cdot \delta_{jk}, \quad (2.45)$$

$$\mathbf{R}_\omega[j, k] = R_\omega \cdot \delta_{jk}, \quad (2.46)$$

où δ_{jk} représente le symbole de Kronecker.

2.4 Fonction de coût

Comme nous l'avons vu, l'évolution d'un système dynamique asservi dépend du contrôle appliqué à celui-ci. La question se pose alors de savoir quel vecteur de contrôle $\mathbf{u}(t)$ nous allons utiliser afin d'atteindre un état final souhaité $\mathbf{x}^*$. Cependant, le contrôle permettant d'arriver à un état donné n'est pas unique. Dans un mouvement d'atteinte par exemple, je peux choisir de rejoindre la cible en ligne droite, mais si un obstacle est présent entre ma main et la cible, je devrais alors réaliser un détour. Les deux contrôles conduisent bien à la cible mais le choix de l'un par rapport à l'autre sera dicté par la nature de la tâche à effectuer.

Par conséquent la stratégie motrice adoptée par le système nerveux ne consiste pas à simplement vouloir atteindre un état final $\mathbf{x}^*$, mais dépend aussi de critères intrinsèques au contexte du mouvement. Pour pouvoir déterminer l'optimalité d'un contrôle, il nous faut donc intégrer une mesure de ces critères dans notre système.

Une manière de définir l'efficacité d'un contrôle consiste à utiliser une fonction de coût J . Plus un contrôle donné minimisera cette fonction et plus celui-ci sera efficace. Dans ce mémoire, nous considérons une fonction de coût courant quadratique :

$$J[k] = E \{ \mathbf{x}^T[k] \cdot \mathbf{Q}_x[k] \cdot \mathbf{x}[k] + \mathbf{u}^T[k] \cdot \mathbf{Q}_u[k] \cdot \mathbf{u}[k] \}, \quad (2.47)$$

avec

$$\mathbf{Q}_x \in \mathbb{R}^{a \times a} : \text{ la matrice de coût dynamique,} \quad (2.48)$$

$$\mathbf{Q}_u \in \mathbb{R}^{b \times b} : \text{ la matrice de coût de contrôle,} \quad (2.49)$$

et $E\{\cdot\}$ l'espérance mathématique. Un tel coût nous permet par conséquent de spécifier des contraintes sur l'état et l'objectif à atteindre via la matrice $\mathbf{Q}_x$ mais également sur le contrôle via la matrice $\mathbf{Q}_u$.

La fonction de coût J est alors donnée par la somme des coûts courants $J[k]$, soit

$$J = \sum_k J[k] \quad (2.50)$$

$$= E \left\{ \sum_{k=1}^{N-1} (\mathbf{x}^T[k] \cdot \mathbf{Q}_x[k] \cdot \mathbf{x}[k] + \mathbf{u}^T[k] \cdot \mathbf{Q}_u[k] \cdot \mathbf{u}[k]) + \mathbf{x}^T[N] \cdot \mathbf{Q}_N \cdot \mathbf{x}[N] \right\} \quad (2.51)$$

où nous avons mis en évidence un coût terminal $J[N] = \mathbf{x}^T[N] \cdot \mathbf{Q}_N \cdot \mathbf{x}[N]$ qui ne dépend plus du contrôle ($u[N]$ étant indéterminé).

Exemple 2.4. Revenant à notre exemple d'un système masse-ressort-amortisseur, il est possible de spécifier comme condition finale que le carré de la distance entre l'état final $\mathbf{x}_\Delta[N] = (x[N] \quad \dot{x}[N])^T$ et l'état souhaité $\mathbf{x}^* = (x^* \quad \dot{x}^*)^T$ doit être minimisé en utilisant la fonction de coût terminal suivante :

$$J[N] = \mathbf{x}^T \cdot \underbrace{\begin{pmatrix} w_x & 0 & -w_x & 0 \\ 0 & w_v & 0 & -w_v \\ -w_x & 0 & w_x & 0 \\ 0 & -w_v & 0 & w_v \end{pmatrix}}_{\mathbf{Q}_N} \cdot \mathbf{x} = w_x \cdot (x[N] - x^*)^2 + w_v \cdot (\dot{x}[N] - \dot{x}^*)^2, \quad (2.52)$$

où w_x et w_v sont des poids.

En théorie du contrôle, le problème d'optimalité consiste alors à trouver un contrôle particulier $\mathbf{u}^*$ minimisant autant que possible la somme des coûts $J[k]$. Autrement dit, nous recherchons $\mathbf{u}^*$ tel que

$$\sum_k J[k](\mathbf{u}^*[k]) \leq \min_{\mathbf{u}} \left\{ \sum_k J[k](\mathbf{u}[k]) \right\}. \quad (2.53)$$

Nous aborderons la résolution du problème d'optimalité dans le chapitre 3.

2.5 Discussion

À travers les différentes sections de ce chapitre, nous avons en mis en place un formalisme connu sous le nom de *Linéaire-Quadratique-Gaussien* (LQG) qui consiste à utiliser :

- a) Une dynamique linéaire pour les variables de contrôle et d'état.
- b) Une fonction de coût quadratique pour les variables de contrôle et d'état.
- c) Des bruits de mesure et de traitement gaussiens.

Le système formés par les équations 2.39 et 2.40 ainsi que la fonction de coût décrite à l'équation 2.51 constituent par conséquent la base mathématique avec laquelle nous allons étudier les mouvements d'atteinte du bras.

Dans le chapitre suivant, nous allons nous attaquer à deux problèmes traditionnels en théorie du contrôle :

- ▷ Comment obtenir une estimation optimale de l'état $\mathbf{x}$ à partir de la mesure $\mathbf{y}$?
- ▷ Quel contrôle optimal $\mathbf{u}$ faut-il appliquer afin de minimiser la somme des coûts J ?

Comme nous allons le voir, le fait de travailler dans le contexte d'une formulation LQG nous permettra d'utiliser des résultats mathématiques efficaces et abondamment étudiés afin de résoudre ces deux problèmes. De plus, comme le mentionne Todorov [Todorov, 2005], il s'agit à l'heure actuelle du seul cadre mathématique connu fournissant des méthodes adaptées au contrôle optimal par feedback ⁴.

Cependant, Todorov, dans le même article, nous met également en garde sur le fait que travailler avec la formulation LQG se fait au prix d'un réalisme biologique réduit. En effet :

- ▷ La dynamique musculosquelettique est en général non-linéaire. Mais il est cependant possible de réaliser des approximations locales de telles dynamiques.
- ▷ Les critères de performance comportementaux ont peu de chance d'être globalement quadratiques. Il faudrait donc soit utiliser une fonction de coût non quadratique et l'approximer, soit utiliser une fonction de coût de la famille des exponentielles de forme quadratique telle que

$$J = E \left\{ \exp \left[\mu \cdot \sum_{k=1}^N (\mathbf{x}^T[k] \cdot \mathbf{Q}_x[k] \cdot \mathbf{x}[k] + \mathbf{u}^T[k] \cdot \mathbf{Q}_u[k] \cdot \mathbf{u}[k]) \right] \right\}. \quad (2.54)$$

4. Pour des systèmes stochastiques et partiellement observables tel que le système musculosquelettique, seules les approches par feedback permettent d'obtenir une performance optimale, l'optimisation en boucle ouverte donnant lieu à un résultat sous-optimal pour les systèmes stochastiques, partiellement observables, continus, non-stationnaires et de grandes dimensions [Todorov, 2005].

Ce contexte Linéaire-Exponentiel-Quadratique-Gaussien (LEQG) permet notamment de modéliser l'aversion ($\mu > 0$) et la prise de risque ($\mu < 0$) et présente des solutions optimales similaires à celles de la formulation LQG ⁵.

- ▷ Dans le système sensorimoteur, le bruit affectant un signal dépend de ce signal. Par exemple, diverses expériences effectuées sur les muscles démontrent qu'un bruit multiplicatif (et non additif!) modélise correctement la relation entre le signal et le bruit moteurs [Jones, 2002]. Il serait plus réaliste d'ajouter du bruit multiplicatif dans notre modèle, qui prendrait alors la forme générale

$$\begin{cases} \mathbf{x}[k+1] &= \mathbf{A} \cdot \mathbf{x}[k] + \mathbf{B} \cdot \mathbf{u}[k] + \xi[k] + \sum_i \varepsilon_i \cdot \mathbf{C}_i \mathbf{u}[k] \\ \mathbf{y}[k] &= \mathbf{H} \cdot \mathbf{x}[k] + \omega[k] + \sum_i \epsilon_i \cdot \mathbf{D}_i \mathbf{x}[k] \end{cases} \quad (2.55)$$

$$\quad (2.56)$$

avec

$$\mathbf{C}_i \in \mathbb{R}^{a \times b}, \mathbf{D}_i \in \mathbb{R}^{a \times a} \quad : \quad \text{les matrices d'étalonnage,} \quad (2.57)$$

$$\varepsilon_i, \epsilon_i \sim \mathcal{N}(0, 1) \quad : \quad \text{les bruits blancs gaussiens.} \quad (2.58)$$

Le bruit multiplicatif permettrait ainsi de capturer certains aspects du compromis entre vitesse et précision lors d'un mouvement [Harris, 1998].

Pour terminer, nous aimerions nous attarder quelque peu sur la matrice d'observation $\mathbf{H}$ utilisé pour former le signal de mesure $\mathbf{y}$. Dans ce mémoire, nous n'allons pas nous inquiéter de la manière dont le système nerveux combine les informations en provenance des divers organes sensoriels. De plus nous allons considérer que notre système est *observable* et qu'il n'est pas nécessaire de reconstruire les informations données par le vecteur $\mathbf{H} \cdot \mathbf{x}$, ce qui revient à utiliser une matrice d'observation unitaire ($\mathbf{H} = \mathbf{I}$). En conséquence, notre signal mesure $\mathbf{y}$ sera simplement une copie de l'état $\mathbf{x}$ entaché par le bruit gaussien ω .

5. Voir [James, 2004].

Contrôle et estimation optimaux

Après avoir développé les bases du formalisme LQG dans le chapitre précédent, nous savons maintenant comment, disposant d'un modèle dynamique linéaire du bras, nous pouvons simuler l'évolution et la mesure de celui-ci à partir d'une condition initiale en tenant compte des bruits sensorimoteurs. Nous savons également comment formuler un critère de performance de l'évolution de ce système via une fonction de coût.

Ce que nous ne savons pas encore en revanche c'est quel contrôle $\mathbf{u}(t)$ il est nécessaire d'utiliser afin de minimiser les contraintes de la tâche imposées par la fonction de coût J . Le contrôle optimal à appliquer va vraisemblablement varier en fonction de l'état dans lequel se trouve le système. Cependant nous ne savons pas non plus comment exploiter de manière efficace l'information donnée par la mesure $\mathbf{y}$ et la connaissance du modèle direct donnée par les matrices $\mathbf{A}$, $\mathbf{B}$ et $\mathbf{H}$ afin de savoir dans lequel état se trouve le système et pouvoir utiliser cette information pour notre contrôle en feedback.

Motivé par ces interrogations, nous allons dans ce chapitre nous attaquer à deux problèmes classiques de la théorie du contrôle, à savoir la question de l'estimation optimale et ensuite celle du contrôle optimal, en suivant les explications données par Mäkilä¹ sur le filtre de Kalman et le contrôle LQG [Mäkilä, 2004], explications elles-mêmes basées sur l'ouvrage de référence d'Åström sur la théorie du contrôle stochastique [Åström, 1970]. Nous clôturerons ensuite ce chapitre par une discussion du principe de séparation suivi des implications d'une stratégie de contrôle robuste sur le système moteur.

Notations

Afin d'alléger les notations mathématiques, nous allons à partir de maintenant considérer comme implicite le fait que

▷ à l'état k , l'état $k - 1$ soit connu :

$$\hat{\mathbf{x}}[k + 1] \iff \hat{\mathbf{x}}[k + 1 | k]. \quad (3.1)$$

▷ l'état d'une matrice $\mathbf{K}$ soit égal à celui du vecteur qu'elle multiplie, idem pour une forme quadratique :

$$\mathbf{K}\mathbf{y}[k] \iff \mathbf{K}[k] \cdot \mathbf{y}[k], \quad (3.2)$$

$$\mathbf{x}^T \mathbf{S} \mathbf{x}[k] \iff \mathbf{x}^T[k] \cdot \mathbf{S}[k] \cdot \mathbf{x}[k]. \quad (3.3)$$

1. N'introduisant dans ce chapitre que les résultats intéressants pour le reste de ce mémoire, nous renverrons le lecteur qui serait intéressé par les démonstrations des formules présentées à l'ouvrage de Mäkilä.

3.1 Estimation optimale

Le problème de l'estimation optimale consiste à trouver, à partir d'une connaissance du modèle direct du système (c'est-à-dire des matrices $\mathbf{A}$, $\mathbf{B}$ et $\mathbf{H}$) et de certaines propriétés des bruits (leur variance respective $\mathbf{R}_\xi$ et $\mathbf{R}_\omega$) le meilleur estimateur $\hat{\mathbf{x}}$ possible pour $\mathbf{x}$ sachant que nous observons une mesure $\mathbf{y}$ du système.

3.1.1 Critère du minimum de variance

Dans un cadre très général, nous pouvons commencer par nous demander quel est, pour deux variables aléatoires dépendantes $\mathbf{X}$ et $\mathbf{Y}$, le meilleur estimateur $\hat{\mathbf{x}}$ de $\mathbf{X}$ connaissant une réalisation $\mathbf{y}$ de $\mathbf{Y}$.

Toutefois, pour pouvoir juger de la notion de « meilleur » estimateur, il nous faut avant tout un critère de jugement, que nous choisirons être celui-ci du *minimum de variance* (“Minimum Mean Square”). Un estimateur $\hat{\mathbf{x}}$ est donc optimal si celui-ci minimise l'espérance de la moyenne du carré de l'erreur par rapport à $\mathbf{X}$ soit l'expression

$$\mathbf{E} \{ (\mathbf{X} - \hat{\mathbf{x}})^T \cdot (\mathbf{X} - \hat{\mathbf{x}}) \mid \mathbf{Y} = \mathbf{y} \}. \quad (3.4)$$

Il est alors possible de montrer² que l'estimateur optimal au sens du minimum de variance est donné par

$$\hat{\mathbf{x}}(\mathbf{y}) = \mathbf{E}\{\mathbf{X} \mid \mathbf{Y} = \mathbf{y}\}. \quad (3.5)$$

3.1.2 Filtre de Kalman

Revenant à notre représentation d'état d'un système, nous nous posons la question de savoir comment estimer de manière optimale l'état $k + 1$ (c'est-à-dire $\mathbf{x}[k + 1]$), lorsque le contrôle et la mesure jusque l'état k sont connus (c'est-à-dire $\mathbf{y}[1], \dots, \mathbf{y}[k]$ et $\mathbf{u}[1], \dots, \mathbf{u}[k]$).

Par l'équation 3.5, nous savons que l'estimateur optimal au sens du minimum de variance est donné par l'expression

$$\hat{\mathbf{x}}[k + 1 \mid k] = \mathbf{E}\{\mathbf{x}[k + 1 \mid k]\}. \quad (3.6)$$

Or, les dimensions de $\mathbf{y}[1], \dots, \mathbf{y}[k]$ et $\mathbf{u}[1], \dots, \mathbf{u}[k]$ grandissant avec le temps, il n'est pas efficace d'appliquer cette équation directement. Dès lors, est-il possible de trouver une expression récursive permettant de calculer $\hat{\mathbf{x}}[k + 1 \mid k]$ à partir de $\hat{\mathbf{x}}[k \mid k - 1]$?

Suite à un énorme travail sur le sujet, le mathématicien Rudolf Kalman³ a démontré par une série de publications dans les années 60 que cela était possible!

Partant de

$$\hat{\mathbf{x}}[1] = \mathbf{E}\{\mathbf{x}[1]\} \quad (3.7)$$

$$= \mathbf{m}_x[1], \quad (3.8)$$

on peut montrer⁴ que l'estimateur optimal dans le cas d'un formalisme LQG est donné récursivement par l'expression suivante :

$$\hat{\mathbf{x}}[k + 1] = \mathbf{A}\hat{\mathbf{x}}[k] + \mathbf{B}\mathbf{u}[k] + \mathbf{K}(\mathbf{y}[k] - \mathbf{H}\hat{\mathbf{x}}[k]). \quad (3.9)$$

Cet estimateur linéaire est appelé *filtre de Kalman* et est composé comme nous pouvons le voir de deux termes :

2. Voir [Mäkilä, 2004].

3. R. E. Kalman (1930 - 2016), ingénieur électricien, mathématicien et inventeur américain d'origine hongroise.

4. Voir [Mäkilä, 2004].

- ▷ Un terme de prédiction $\mathbf{A}\hat{\mathbf{x}}[k] + \mathbf{B}\mathbf{u}[k]$, qui utilise le modèle direct du système afin de prédire son comportement futur.
- ▷ Un terme de mise à jour $\mathbf{K}(\mathbf{y}[k] - \mathbf{H}\hat{\mathbf{x}}[k])$, qui utilise les nouvelles informations (le terme $\mathbf{y}[k] - \mathbf{H}\hat{\mathbf{x}}[k]$ étant appelé à juste titre l'*innovation*) en provenance des mesures afin de corriger cette prédiction.

Le poids accordé au terme de mise à jour va dépendre de la confiance accordée au modèle direct (entaché par le bruit de traitement) par rapport à celle accordée aux mesures (entachées par le bruit de mesure) et s'exprime à travers le gain du filtre $\mathbf{K}$ dont l'expression est donnée par

$$\mathbf{K}[k] = \mathbf{A} \cdot \boldsymbol{\Sigma}_x[k] \cdot \mathbf{H}^T \cdot (\mathbf{H} \cdot \boldsymbol{\Sigma}_x[k] \cdot \mathbf{H}^T + \mathbf{R}_\omega)^{-1} \quad (3.10)$$

où le terme $\boldsymbol{\Sigma}_x$ représente la matrice de variance de l'erreur d'estimation calculée récursivement :

$$\begin{cases} \boldsymbol{\Sigma}_x[1] &= \mathbf{R}_x[1, 1] \\ \boldsymbol{\Sigma}_x[k+1] &= \mathbf{A} \cdot \boldsymbol{\Sigma}_x[k] \cdot \mathbf{A}^T - \mathbf{K}[k] \cdot \mathbf{H} \cdot \boldsymbol{\Sigma}_x[k] \cdot \mathbf{A}^T + \mathbf{R}_\xi. \end{cases} \quad (3.11)$$

$$(3.12)$$

Notons que $\boldsymbol{\Sigma}_x[k]$ ne dépend ni des observations $\mathbf{y}[k]$ ni des contrôles $\mathbf{u}[k]$. Par conséquent les gains $\mathbf{K}[k]$ du filtre peuvent être calculés récursivement a priori !

Quelques remarques :

- ▷ Le filtre de Kalman est également valide lorsque les matrices $\mathbf{A}$, $\mathbf{B}$ et $\mathbf{H}$ varient dans le temps et lorsque $\xi[k]$ et $\omega[k]$ ne sont pas stationnaires.
- ▷ Si $\xi[k]$ et $\omega[k]$ ne sont pas normalement distribués, le filtre de Kalman est seulement le filtre linéaire optimal.
- ▷ Dans son article de 2005, Todorov suggère également d'utiliser un bruit η interne au filtre, modélisant les fluctuations des représentations neuronales. Soit

$$\hat{\mathbf{x}}[k+1] = \mathbf{A}\hat{\mathbf{x}}[k] + \mathbf{B}\mathbf{u}[k] + \mathbf{K}(\mathbf{y}[k] - \mathbf{H}\hat{\mathbf{x}}[k]) + \eta[k] \quad (3.13)$$

avec $\eta \in \mathbb{R}^a$ [Todorov, 2005].

3.2 Contrôle optimal

Une fois le problème de l'estimation optimale résolue, le problème du contrôle optimal consiste maintenant à trouver, à partir d'une connaissance du modèle direct du système et de la variance connue des bruits le contrôle $\mathbf{u}$ permettant de minimiser une fonction de coût J donnée.

3.2.1 Information complète

Dans le cas idéal d'une information complète, le vecteur d'état $\mathbf{x}$ est connu et le contrôleur peut dès lors se servir directement de ce dernier.

Il est possible de montrer⁵ que ce problème d'optimisation peut être résolu de manière séquentielle par des techniques de programmation dynamique⁶. En appliquant une formulation par équation de Bellman dans le cas stochastique à notre représentation d'état, nous trouvons comme résultat que la fonction de coût est minimisée en appliquant la stratégie de contrôle suivante :

$$\mathbf{u}[k] = -\mathbf{L}\mathbf{x}[k]. \quad (3.14)$$

5. Voir [Mäkilä, 2004].

6. Nous renvoyons le lecteur à l'annexe A pour plus d'explications concernant la programmation dynamique ainsi que l'équation de Bellman.

Autrement dit, le contrôle optimal est donné par feedback linéaire sur l'état courant, avec un gain de contrôle $\mathbf{L}$ dépendant de la fonction de coût définie et donné par

$$\mathbf{L}[k] = (\mathbf{B}^T \mathbf{S}[k+1] \mathbf{B} + \mathbf{Q}_u)^{-1} \cdot \mathbf{B}^T \mathbf{S}[k+1] \mathbf{A} \quad (3.15)$$

où $\mathbf{S}[k]$ est donné récursivement par

$$\begin{cases} \mathbf{S}[N] &= \mathbf{Q}_N \\ \mathbf{S}[k] &= \mathbf{A}^T \mathbf{S}[k+1] \mathbf{A} - \mathbf{A}^T \mathbf{S}[k+1] \mathbf{B} \cdot \mathbf{L}[k] + \mathbf{Q}[k]. \end{cases} \quad (3.16)$$

$$(3.17)$$

Nous constatons ici aussi qu'il est possible de calculer ces gains a priori et de manière récursive en itérant de $k = N - 1$ jusque $k = 0$!

Utilisant cette stratégie de contrôle, la valeur minimale de la fonction de coût est alors donnée par

$$\min_{u[1], \dots, u[N]} J = E \{ \mathbf{x}^T \mathbf{S} \mathbf{x}[1] + s[1] \} \quad (3.18)$$

$$= \mathbf{m}_x^T \mathbf{S} \mathbf{m}_x[1] + tr(\mathbf{S}[1] \cdot \boldsymbol{\Sigma}_x[1]) + s[1] \quad (3.19)$$

où la valeur moyenne et la variance initiales de $\mathbf{x}$ sont donnés comme dans le filtre de Kalman par

$$\begin{cases} \mathbf{m}_x[1] &= \mathbf{E} \{ \mathbf{x}[1] \} \\ \boldsymbol{\Sigma}_x[1] &= \mathbf{R}_x[1, 1] \end{cases} \quad (3.20)$$

$$(3.21)$$

et avec $s[k]$ donné par

$$\begin{cases} s[1] &= 0 \\ s[k] &= s[k+1] + tr(\mathbf{R}_\xi \cdot \mathbf{S}[k+1]). \end{cases} \quad (3.22)$$

$$(3.23)$$

3.2.2 Information incomplète

Comme nous en avons discuté au chapitre 2, l'état $\mathbf{x}$ n'est souvent pas accessible directement mais uniquement connu via une mesure $\mathbf{y}$ disponible. Dans ce cas plus réaliste d'une information incomplète, le contrôleur ne connaît plus l'état $\mathbf{x}$ mais doit recourir à une estimation $\hat{\mathbf{x}}$ afin d'effectuer son contrôle.

La stratégie de contrôle optimale sera exactement la même, excepté que nous utiliserons l'estimateur $\hat{\mathbf{x}}[k | k-1]$ en lieu et place de $\mathbf{x}[k]$:

$$\mathbf{u}[k] = -\mathbf{L}[k] \cdot \hat{\mathbf{x}}[k | k-1]. \quad (3.24)$$

Dans la section précédente nous avons trouvé que l'estimateur optimal de $\mathbf{x}$ était donné par le filtre de Kalman : un choix optimal consiste par conséquent à utiliser l'estimateur donné par ce dernier dans la loi de contrôle. Dans ce cas, il faut uniquement changer la formule de $s[k]$ intervenant dans le calcul du coût qui devient :

$$s[k] = s[k+1] + tr(\mathbf{Q} \cdot \boldsymbol{\Sigma}_x[k]) + tr(\mathbf{S}[k+1] \cdot \mathbf{K}[k] \cdot \{ \mathbf{H} \cdot \boldsymbol{\Sigma}_x[k] \cdot \mathbf{H}^T + \mathbf{R}_\omega \} \cdot \mathbf{K}[k]^T). \quad (3.25)$$

Quelques remarques :

- ▷ Cette stratégie de contrôle est également optimale dans le cas déterministe.
- ▷ Si les bruits ne sont pas normalement distribués, le contrôle est seulement le contrôle linéaire optimal car le filtre de Kalman est alors seulement le filtre linéaire optimal.
- ▷ Si le contrôleur ne connaît au temps k que les informations jusqu'au temps $k-d$, il faut utiliser $\hat{\mathbf{x}}[k | k-d]$. Intuitivement, cela revient à dire que le contrôleur a toujours intérêt à utiliser le meilleur estimateur de $\mathbf{x}$ dont il dispose.

3.3 Discussion

Nous savons maintenant comment contrôler de manière optimale notre système. Si nous prenons un peu de recul par rapport aux formules d'estimation et de contrôle que nous avons développé dans ce chapitre, nous pouvons nous rendre compte que celles-ci présentent quelques caractéristiques très particulières.

Premièrement, nous pouvons remarquer que l'estimation optimale et le contrôle optimal sont totalement indépendants l'un de l'autre. Il est donc tout à fait possible de calculer les deux séparément. Cette propriété importante est appelée *principe de séparation* et provient du fait que nous travaillons dans le cadre d'un système LQG.

Toutefois, dès que l'on s'écarte de certaines hypothèses du modèle, cette propriété s'effrite rapidement. Ce sera notamment le cas si les bruits additifs sont corrélés entre eux ($\mathbf{R}_{\xi\omega} \neq 0$) ou si le système contient du bruit multiplicatif (voir section 2.5). Il sera alors nécessaire d'utiliser des gains de filtres adaptatifs ainsi que des approximations numériques de la solution optimale [Todorov, 2005].

Deuxièmement, aussi bien les gains du filtre de Kalman $\mathbf{K}$ que les gains de contrôle $\mathbf{L}$ peuvent être calculés a priori, moyennant la connaissance d'un modèle direct du système. Le lecteur attentif aura également remarqué qu'il existe d'ailleurs une dualité entre l'estimation optimale et le contrôle optimal puisque la procédure de calcul est extrêmement similaire pour les deux⁷. Appliqué au mouvement moteur du bras, nous pouvons interpréter ces résultats pour arriver à une conclusion assez surprenante, à savoir qu'il mathématiquement possible de planifier à l'avance une stratégie optimale de contrôle et d'estimation en fonction d'une tâche. En d'autres mots, il pourrait suffire à notre cerveau d'avoir une connaissance de son environnement (modèle direct) et de la tâche à effectuer (fonction de coût) pour que celui-ci soit capable de mettre en place un contrôle optimal!

Nous pouvons maintenant mieux comprendre l'hypothèse selon laquelle le mouvement correctif d'un bras soumis à un champ de force ne serait pas la conséquence d'une reprogrammation explicite, mais plutôt la conséquence d'une seule stratégie robuste. Une stratégie préprogrammée présenterait de plus un autre avantage : elle ne nécessiterait pas du système nerveux que celui-ci ait à effectuer de nombreux calculs en cours de mouvement, lui permettant ainsi de réagir très rapidement.

Enfin, nous avons considéré dans ce chapitre que le modèle direct, les matrices $\mathbf{A}$, $\mathbf{B}$ et $\mathbf{H}$ étaient parfaitement connues. Dans la pratique, celui-ci se sert plutôt de représentations construites de ces matrices soit des matrices $\hat{\mathbf{A}}$, $\hat{\mathbf{B}}$ et $\hat{\mathbf{H}}$. Il peut en pratique arriver que ce modèle direct interne ne correspondent pas au modèle réel du système.

C'est par exemple le cas lorsqu'un champ de force est subitement activé lors d'une expérience sur les mouvements d'atteinte : le modèle dynamique réel de l'environnement est donné par $\mathbf{A} + \Delta\mathbf{A}$, où $\Delta\mathbf{A}$ est la perturbation introduite par le champ de force, mais le système nerveux a quant à lui intégré un modèle de l'environnement $\hat{\mathbf{A}} = \mathbf{A}$ et va par conséquent commettre des erreurs lors du contrôle. Savoir comment, après plusieurs essais dans un champ de force, le cerveau est capable de diminuer l'erreur de modèle est là toute la question posée par l'apprentissage. Cette vaste problématique sort cependant du cadre de ce mémoire.

Mais outre le cas de franches disparités entre le modèle interne au contrôleur et le modèle réel, nous pourrions également imaginer que dans le système nerveux, ce modèle direct est également entaché d'un certain niveau bruit.

L'introduction de perturbations dans le modèle sera examinée plus en détails dans le chapitre 8. Dans le prochain chapitre, nous mettrons au point un modèle linéaire du bras, ce qui nous permettra de nous lancer

7. Pour plus d'informations sur le sujet voir par exemple [Aström, 1970].

dans les simulations.

Modélisation linéaire du bras

Après ces deux premiers chapitres, nous sommes maintenant capables de calculer l'évolution d'un système à partir d'une condition initiale mais également de contrôler celui-ci de manière optimale afin de satisfaire des contraintes imposées. Il nous manque cependant encore un élément important avant de pouvoir simuler numériquement des mouvements d'atteinte, à savoir un modèle du bras humain. Et la principale difficulté de cette tâche provient du fait que le système musculosquelettique exhibe une dynamique non-linéaire.

Intuitivement, on pourra modéliser le bras et l'avant-bras comme un système plan formé par deux corps rigides et actionné par deux articulations rotoïdes, celle de l'épaule et celle du coude, ainsi que l'illustre la figure 4.1.

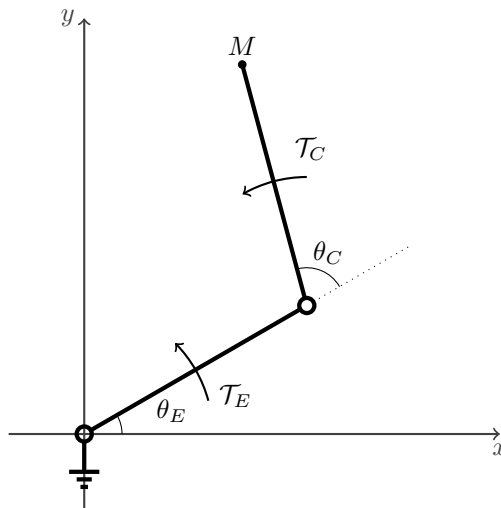


FIGURE 4.1 – Représentation de l'avant-bras et du bras comme un système plan.

En général, l'évolution dynamique d'un tel système mécanique rigide peut être décrite par l'équation d'Euler-Lagrange¹ dont la forme matricielle est donnée par

$$\mathbf{M}(\mathbf{q}) \cdot \ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \cdot \dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = \mathbf{\Gamma} \quad (4.1)$$

1. Pour une dérivation de l'équation d'Euler-Lagrange, voir par exemple [Spong, 2005] ou [Lynch, 2017].

avec

$$\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{n \times n} : \text{ la matrice de masse du système,} \quad (4.2)$$

$$\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^{n \times n} : \text{ la matrice regroupant les effets centripètes et de Coriolis,} \quad (4.3)$$

$$\mathbf{g}(\mathbf{q}) \in \mathbb{R}^n : \text{ le vecteur des forces résultant d'un potentiel,} \quad (4.4)$$

et avec

$$\mathbf{q} \in \mathbb{R}^n : \text{ le vecteur des coordonnées généralisées (indépendantes les unes des autres),} \quad (4.5)$$

$$\mathbf{\Gamma} \in \mathbb{R}^n : \text{ le vecteur des forces externes généralisées correspondant à } \mathbf{q}, \quad (4.6)$$

de manière à ce que $\mathbf{\Gamma}^T \cdot \mathbf{q}$ ait pour unité une puissance.

Dans notre cas, un choix de coordonnées généralisées pourrait être données par les angles $q_1 = \theta_E$ et $q_2 = \theta_C$ et les forces généralisées seraient alors choisies comme étant $\Gamma_1 = \mathcal{T}_E$ et $\Gamma_2 = \mathcal{T}_C$.

Bien que ce modèle soit déjà une simplification de la réalité biologique du corps humain, il conduit, comme nous le développons dans l'annexe B et même en ignorant les effets viscoélastiques des muscles, à des équations dynamiques non-linéaires assez compliquées. Une manière courante dans la littérature de traiter cette non-linéarité consiste à effectuer une approximation linéaire locale afin de pouvoir continuer de bénéficier des techniques de calcul linéaires.

Dans ce chapitre, nous allons avoir recours à une approximation encore plus brute, puisque nous n'allons considérer que l'extrémité du bras, la main, et la traiter comme une masse ponctuelle soumise à la dynamique des muscles. Démarrant avec une simple masse effectuant un mouvement de translation unidimensionnel, nous mettrons en place les modèles biomécaniques et musculaires avant de passer à leur représentation d'état. Il nous suffira ensuite d'étendre simplement ces conclusions à un modèle bidimensionnel.

4.1 Masse ponctuelle à une dimension

Dans une première approche, nous allons analyser un modèle de masse ponctuelle à une dimension spatiale. Bien que rudimentaire, cet exemple nous permettra d'acquérir une bonne intelligibilité du modèle à deux dimensions utilisé par la suite.

4.1.1 Équation dynamique

Considérons une masse ponctuelle m effectuant un mouvement de translation, dont la position en fonction du temps t est dénotée par la coordonnée $x(t)$. Si les muscles du bras délivrent une force $F(t)$ à la main, la dynamique de cette dernière est facilement décrite par la seconde loi de Newton

$$m \cdot \ddot{x}(t) = F(t). \quad (4.7)$$

Bien que commode, cette relation néglige pourtant les propriétés viscoélastiques des muscles. Afin de tenir compte de celles-ci, nous pouvons utiliser un modèle biomécanique analogique qui consiste en la combinaison série ou parallèle d'éléments élastiques, les *ressorts*, avec des éléments visqueux, les *amortisseurs*.

Par exemple, un modèle analogique assez répandu du comportement des muscles est le modèle « à trois éléments » proposé par Hill en 1950. Celui-ci associe un élément contractile (CC), comprenant une composante visqueuse et une composante génératrice de force, à un élément série élastique (CES), le tout en parallèle avec un autre élément élastique (CEP) [Henrotte, 2016]. Ce modèle est illustré à la figure 4.2.

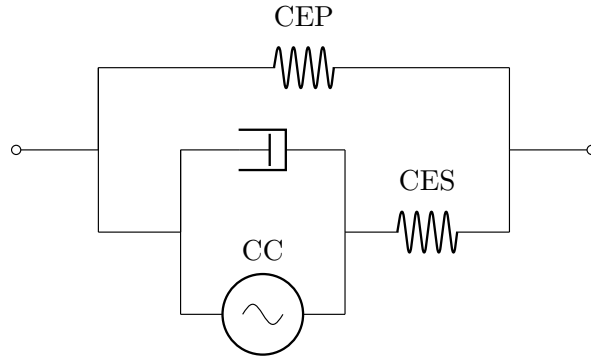


FIGURE 4.2 – Modèle musculaire à trois éléments de Hill.

Dans ce mémoire, nous allons utiliser un modèle plus simple que le modèle Hill : le modèle analogique de Maxwell. Celui-ci se compose en effet d'un ressort en série avec un amortisseur, auquel nous ajoutons une masse, ainsi que le dépeint la figure 4.3.

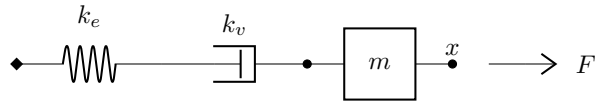


FIGURE 4.3 – Modèle analogique de Maxwell relié à une masse.

L'équation dynamique 4.7 augmentée d'un terme élastique et d'un terme visqueux devient alors

$$m \cdot \ddot{x}(t) + k_v \cdot \dot{x}(t) + k_e \cdot x(t) = F(t) \quad (4.8)$$

où k_v représente le coefficient de viscosité et k_e le coefficient de raideur des muscles².

4.1.2 Équation musculaire

Il nous reste maintenant à modéliser la manière dont les muscles convertissent le signal moteur $u(t)$ reçu en force $F(t)$. Ce phénomène est habituellement simulé par l'application du signal en moteur en entrée d'un filtre passe-bas, la force étant donnée en sortie du filtre.

Certains auteurs utilisent un filtre passe-bas du second ordre :

$$\tau_1 \tau_2 \cdot \ddot{F}(t) + (\tau_1 + \tau_2) \cdot \dot{F}(t) + F(t) = u(t). \quad (4.9)$$

Pour rappel, un tel filtre présente une fréquence de coupure f_c telle que

$$2\pi f_c = \frac{1}{\sqrt{\tau_1 \tau_2}} \quad (4.10)$$

et peut présenter une résonance autour de cette fréquence si la valeur du facteur de qualité

$$Q = \frac{\sqrt{\tau_1 \tau_2}}{(\tau_1 + \tau_2)} \quad (4.11)$$

2. Le lecteur attentif pourra remarquer qu'il s'agit de la même équation que celle utilisée dans les exemples du chapitre 2 !

est supérieure à $\sqrt{2}/2 \approx 0.707$.

Si les constantes de temps sont choisies comme étant égales ($\tau_1 = \tau_2 = \tau$), cela donne lieu à une fréquence de coupure $f_c = 1/\tau$ et à un facteur de qualité $Q = 1/2$. Au-delà de la fréquence de coupure, le gain est de -40 db/dec .

Nous allons quant à nous nous contenter, comme d'autres auteurs également, d'un filtre passe-bas du premier ordre, et ce par souci de simplicité. En effet, l'extension à un filtre du second ordre est assez aisée³ et il nous est de plus tout à fait possible de garder la même fréquence de coupure, l'ordre du filtre déterminant la pente du gain. L'équation modélisant l'impédance des muscles devient alors

$$\tau \cdot \dot{F}(t) + F(t) = u(t) \quad (4.12)$$

où la fréquence de coupure du filtre passe-bas est donnée par $2\pi f_c = 1/\tau$. Au-delà de la fréquence de coupure, le gain est de -20 db/dec .

4.1.3 Représentation d'état

Nous pouvons maintenant appliquer la représentation d'état développée au chapitre 2 à notre modèle formé des équations 4.8 et 4.12 :

$$\begin{cases} m \cdot \ddot{x}(t) + k_v \cdot \dot{x}(t) + k_e \cdot x(t) & = F(t) \\ \tau \cdot \dot{F}(t) + F(t) & = u(t). \end{cases} \quad (4.13)$$

$$(4.14)$$

Choisissant comme vecteur d'état le vecteur

$$\mathbf{x}(t) = \begin{pmatrix} x & \dot{x} & F \end{pmatrix}^T, \quad (4.15)$$

nous reformulons notre système d'équations en vue de l'écrire sous la forme canonique $\dot{\mathbf{x}}(t) = \mathbf{A}_t \mathbf{x}(t) + \mathbf{B}_t \mathbf{u}(t)$, soit

$$\begin{cases} \dot{x} & = \dot{x} \\ \ddot{x} & = -\frac{k_e}{m} \cdot x - \frac{k_v}{m} \cdot \dot{x} + \frac{1}{m} \cdot F_x \\ \dot{F}_x & = -\frac{1}{\tau} \cdot F_x + \frac{1}{\tau} \cdot u_x. \end{cases} \quad (4.16)$$

$$(4.17)$$

$$(4.18)$$

Il est alors aisé d'écrire la représentation d'état temporelle de ce système :

$$\underbrace{\begin{pmatrix} \dot{x} \\ \ddot{x} \\ \dot{F} \end{pmatrix}}_{\dot{\mathbf{x}}} = \underbrace{\begin{pmatrix} 0 & 1 & 0 \\ -k_e/m & -k_v/m & 1/m \\ 0 & 0 & -1/\tau \end{pmatrix}}_{\mathbf{A}_t} \cdot \underbrace{\begin{pmatrix} x \\ \dot{x} \\ F \end{pmatrix}}_{\mathbf{x}} + \underbrace{\begin{pmatrix} 0 \\ 0 \\ 1/\tau \end{pmatrix}}_{\mathbf{B}_t} \cdot \underbrace{\begin{pmatrix} u \end{pmatrix}}_{\mathbf{u}}. \quad (4.19)$$

Cette représentation pourra alors être discrétisée, et le vecteur d'état augmenté, suivant les techniques décrites à la section 2.1 du chapitre 2 afin d'en calculer une solution numérique.

3. Un filtre du second-ordre peut aussi être écrit sous la forme d'une paire de filtres du premier ordre couplés :

$$\begin{cases} \tau_1 \cdot \dot{G}(t) + G(t) & = u(t) \\ \tau_2 \cdot \dot{F}(t) + F(t) & = G(t). \end{cases}$$

4.2 Masse ponctuelle à deux dimensions

Nous allons maintenant étendre le modèle unidimensionnel afin de modéliser le bras comme une masse ponctuelle à deux dimensions spatiales, modèle qui nous servira à effectuer des simulations numériques. Le choix d'un modèle bidimensionnel se justifie par le fait que durant les expériences étudiant les mouvements d'atteinte, les bras des participants sont classiquement solidaires d'un manipulateur rigide contraignant les mouvements à évoluer dans un plan, comme l'esquisse la figure 4.4.

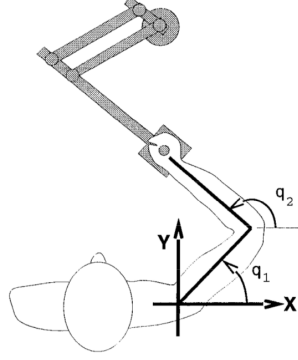


FIGURE 4.4 – Illustration d'un manipulateur robotique utilisé lors d'expériences sur les mouvements d'atteinte. Source : [Shadmehr, 1994].

Considérons une masse ponctuelle m dont la position en fonction du temps t est donnée par les coordonnées cartésiennes $(x(t); y(t))$ et dont la commande motrice est donnée aux muscles respectivement pour chaque axe par $u_x(t)$ et $u_y(t)$. Les équations mathématiques décrivant l'évolution de ce système peuvent alors être trouvées en appliquant simplement le système formé des équations 4.13 et 4.14 à chaque dimension spatiale.

Notre système est alors décrit par les quatre équations suivantes :

$$\begin{cases} m \cdot \ddot{x} + k_v \cdot \dot{x} + k_e \cdot x & = F_x + \lambda \cdot F_y & (4.20) \\ m \cdot \ddot{y} + k_v \cdot \dot{y} + k_e \cdot y & = F_y + \lambda \cdot F_x & (4.21) \\ \tau \cdot \dot{F}_x + F_x & = u_x & (4.22) \\ \tau \cdot \dot{F}_y + F_y & = u_y & (4.23) \end{cases}$$

où un terme de couplage entre F_x et F_y a été ajouté aux équations dynamiques.

Dans ces équations :

- ▷ k_e émule la raideur des muscles.
- ▷ k_v reflète la viscosité des muscles.
- ▷ τ est la constante de temps résultant de la modélisation de l'impédance des muscles comme un filtre passe-bas du premier ordre.
- ▷ λ est le taux de couplage entre x et y .

Notons que lorsque la valeur de λ est nulle, les équations en x et y sont complètement indépendantes l'une de l'autre. Si un tel modèle est implémenté numériquement, il permettra par conséquent aussi d'étudier

le cas unidimensionnel très facilement.

Choisissant maintenant comme vecteur d'état le vecteur

$$\mathbf{x}(t) = \begin{pmatrix} x & y & \dot{x} & \dot{y} & \dot{F}_x & \dot{F}_y \end{pmatrix}^T \quad (4.24)$$

et comme vecteur de contrôle le vecteur

$$\mathbf{u}(t) = \begin{pmatrix} u_x & u_y \end{pmatrix}^T, \quad (4.25)$$

nous réécrivons de manière tout à fait analogue au cas unidimensionnel la représentation d'état temporelle de notre modèle bidimensionnel, soit

$$\underbrace{\begin{pmatrix} \dot{x} \\ \dot{y} \\ \ddot{x} \\ \ddot{y} \\ \dot{F}_x \\ \dot{F}_y \end{pmatrix}}_{\dot{\mathbf{x}}} = \underbrace{\begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ -k_e/m & 0 & -k_v/m & 0 & 1/m & \lambda/m \\ 0 & -k_e/m & 0 & -k_v/m & \lambda/m & 1/m \\ 0 & 0 & 0 & 0 & -1/\tau & 0 \\ 0 & 0 & 0 & 0 & 0 & -1/\tau \end{pmatrix}}_{\mathbf{A}_t} \cdot \underbrace{\begin{pmatrix} x \\ y \\ \dot{x} \\ \dot{y} \\ F_x \\ F_y \end{pmatrix}}_{\mathbf{x}} + \underbrace{\begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1/\tau & 0 \\ 0 & 1/\tau \end{pmatrix}}_{\mathbf{B}_t} \cdot \underbrace{\begin{pmatrix} u_x \\ u_y \end{pmatrix}}_{\mathbf{u}}. \quad (4.26)$$

Il s'agit de ce modèle que nous utiliserons dans nos simulations. Pour rappel, une fois les vecteurs $\mathbf{x}(t)$ et $\mathbf{u}(t)$ et les matrices $\mathbf{A}_t$ et $\mathbf{B}_t$ discrétisés à l'aide de la méthode d'Euler explicite et augmentés en fonction d'un vecteur cible $\mathbf{x}^*$, nous obtenons les vecteurs $\mathbf{x}[k]$ et $\mathbf{u}[k]$ et les matrices $\mathbf{A}$ et $\mathbf{B}$ qui sont alors insérés dans notre représentation LQG pour former le système

$$\begin{cases} \mathbf{x}[k+1] &= \mathbf{A} \cdot \mathbf{x}[k] + \mathbf{B} \cdot \mathbf{u}[k] + \xi[k] \\ \mathbf{y}[k] &= \mathbf{H} \cdot \mathbf{x}[k] + \omega[k]. \end{cases} \quad (4.27)$$

$$(4.28)$$

4.3 Discussion

Nous avons développé dans ce chapitre un modèle simplifié du bras, considéré comme une masse ponctuelle en mouvement dans un plan. Ce modèle s'inspire de modèles déjà utilisés dans la littérature, ainsi que le présente la section C.1 de l'annexe C.

La facilité mathématique se paye bien sûr au prix d'un réalisme biologique moindre : précisons donc d'emblée que cette simplification, par sa nature même, ne se veut pas intimement réaliste. Il est en effet utopique de vouloir réduire la dynamique d'un modèle à deux articulations au cas d'une particule ponctuelle, notamment à cause du couplage existant entre les axes x et y et des termes non-linéaires présents comme le présente l'annexe C. Un mouvement rectiligne de la main, par exemple, va solliciter à la fois l'articulation du coude et celle de l'épaule, puisqu'une articulation seule ne peut produire qu'un mouvement circulaire, comme l'illustre la figure 4.5 à l'échelle relative du corps humain⁴. Or la sollicitation relative de l'articulation de l'épaule par rapport à celle du coude va varier en fonction de la position de la main.

Les propriétés de l'articulation de l'épaule étant différentes de celles du coude⁵ c'est toute la dynamique qui va être impactée par les angles que forment les articulations du coude et de l'épaule. Contrairement à notre modèle, il est donc fort à parier que les paramètres m , k_e , k_v et λ :

4. Longueurs respectives de l'avant-bras et du bras : 33 cm et 34 cm [Shadmehr, 1994]. Demie largeur d'épaule : 24 cm [Motmans, 2005]. Point de départ de la main : 30 cm en face du thorax [Wagner, 2008].

5. Par exemple, l'inertie vue par l'articulation du coude comprend le bras et la main, tandis que celle vue par l'articulation de l'épaule comprend et l'avant-bras et le bras et la main.

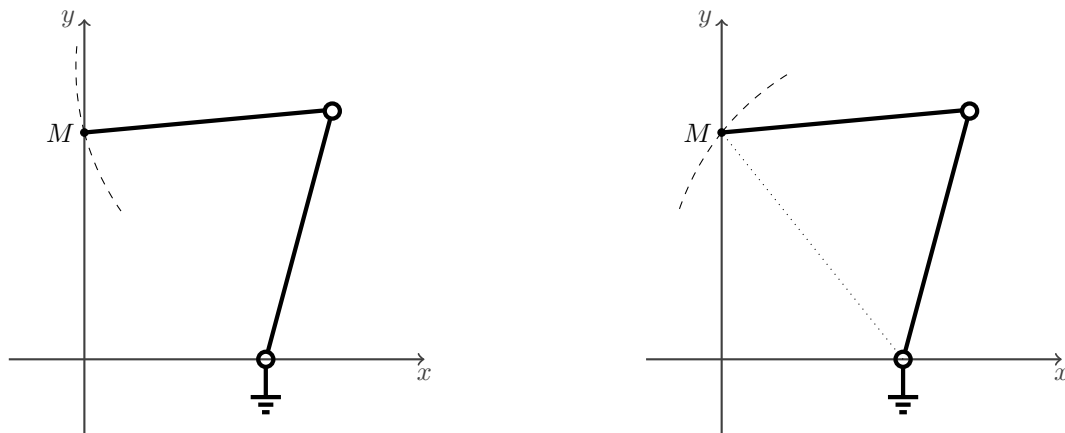


FIGURE 4.5 – Gauche : trajectoire produite lorsque seule l’articulation du coude est actionnée. Droite : trajectoire produite lorsque seule l’articulation de l’épaule est actionnée.

- ▷ Ne soient pas constants mais dépendent de la position de la main,
- ▷ Ne soient pas identiques dans la direction x et dans la direction y .

Cette réduction n’ira d’ailleurs pas sans poser quelques problèmes lors de l’estimation des paramètres biomécaniques du bras comme nous le verrons dans le chapitre suivant, chapitre dans lequel nous poursuivons les discussions entamées ici.

Pourquoi préférer une telle approche dans ce cas ? Notre motivation est que le modèle de la masse ponctuelle, bien qu’élémentaire, nous permettra de simplifier énormément le problème du contrôle robuste en nous offrant d’une part une meilleure analyse des mécanismes en jeu et d’autre part en nous évitant de passer trop de temps à élaborer un modèle complexe plutôt que de nous concentrer sur le cœur de notre problématique : parvenir à un contrôle robuste malgré le délai présent dans le système.

Une fois le problème du contrôle résolu, libre à nous de l’étendre à d’autres modèles !

Valeur des paramètres biomécaniques

Notre modèle du bras est maintenant au point, mais nous n'avons pas encore abordé le choix de valeurs numériques pour les paramètres biomécaniques. Quelles valeurs choisir pour les paramètres m , k_e , k_v , τ et λ ? L'objet de ce chapitre sera donc la recherche de paramètres valides dans notre système cartésien d'axes x et y .

Cependant, la réalité biologique du corps humain fait qu'une articulation, par exemple celle du coude, est décrite par un mouvement de rotation plutôt qu'un mouvement de translation, soit par l'équivalent rotoïde de l'équation 4.8 :

$$\mathcal{I} \cdot \ddot{\theta}(t) + \kappa_v \cdot \dot{\theta}(t) + \kappa_e \cdot \theta(t) = \mathcal{T}(t) \quad (5.1)$$

avec θ l'angle du coude, $\mathcal{I}$ l'inertie du bras calculée par rapport au coude et $\mathcal{T}$ le torque appliqué sur le bras, ainsi que l'illustre la figure 5.1.

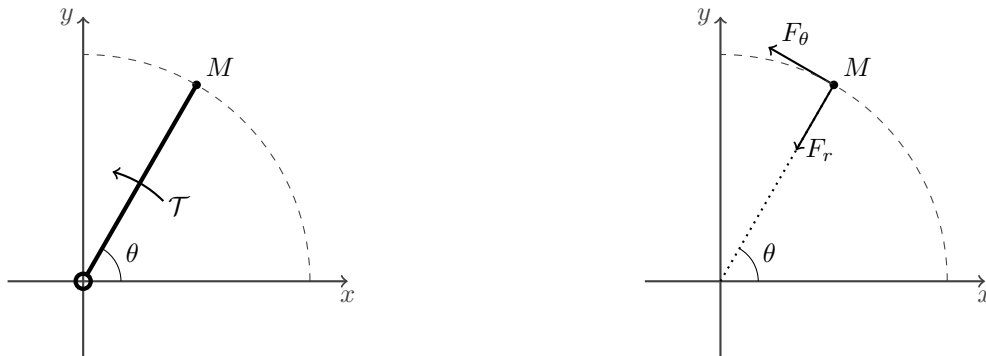


FIGURE 5.1 – Gauche : Représentation simplifiée du bras et de l'articulation du coude. Droite : Représentation équivalente par une masse ponctuelle.

Et ce sont les paramètres $\mathcal{I}$, κ_e et κ_v , pour un mouvement angulaire naturel du bras, qui seront en pratique mesurés. C'est d'ailleurs logique puisque comme nous en avons discuté au chapitre 4, les paramètres k_e , k_v , m et λ pour des mouvements rectilignes sont susceptibles et de changer fortement en fonction de la position de la main.

Bien que la formulation de notre modèle bidimensionnel soit très simple, nous nous heurtons donc ici au revers de la médaille : comment traduire les propriétés dynamiques d'un système effectuant des mouvements rotatifs en un modèle équivalent ponctuel ?

Il n'y a malheureusement pas de bonne réponse à cette question. . . les paramètres k_e , k_v , m et λ dépendant de la position de la main et du mouvement à effectuer, il faudrait les modéliser à l'aide de mesures récoltées ou provenant d'une simulation d'un modèle complet du bras. Et encore, ces modèles mathématiques ne seraient valides que dans le cas de certaines trajectoires étudiées, qui nécessiteraient sans doute une linéarisation.

Conscient de ces limites, nous allons nous contenter de trouver un ordre de grandeur de valeurs pour ces paramètres : nous étudierons de toute façon la sensibilité du modèle à ces paramètres lors des simulations numériques.

5.1 Paramètres viscoélastiques k_e et k_v

Dans un article de 2014, Crevecoeur et Scott [Crevecoeur, 2014] ont estimé, à l'aide de courbes représentant la force développée par les muscles en fonction de leur longueur et de la vitesse de déplacement, la raideur et la viscosité des muscles comme valant respectivement

$$\kappa_e = 1.61 \text{ Nm/rad}, \quad (5.2)$$

$$\kappa_v = 0.14 \text{ Nm} \cdot \text{s/rad}, \quad (5.3)$$

ces valeurs ayant été calculées pour des déplacements angulaires du coude soumis à un torque¹.

Le système ponctuel équivalent au système du bras et de l'articulation du coude, également représenté à la figure 5.1, serait une masse subissant une force tangentielle (dûe au torque) et une force radiale (dûe au mouvement circulaire). Une méthode approximative afin d'avoir une idée des valeurs de k_e et de k_v consiste alors à ne considérer que les composantes tangentielles à la trajectoire de la particule

$$F_\theta = \frac{\mathcal{T}}{L}, \quad (5.4)$$

$$x_\theta = L \cdot \theta, \quad (5.5)$$

$$\dot{x}_\theta = L \cdot \dot{\theta}, \quad (5.6)$$

$$\ddot{x}_\theta = L \cdot \ddot{\theta}, \quad (5.7)$$

avec L la longueur du bras et x_θ la distance parcourue sur l'arc de cercle (à partir, sans perte de généralité, de l'angle $\theta_0 = 0$) où on considère en pratique que la masse ne fait jamais un tour complet du cercle².

Dès lors on transpose facilement l'équation ?? à ces variables :

$$\frac{\mathcal{I}}{L^2} \cdot \ddot{x}_\theta + \frac{\kappa_v}{L^2} \cdot \dot{x}_\theta + \frac{\kappa_e}{L^2} \cdot x_\theta = F_\theta. \quad (5.8)$$

1. L'auteur de ce mémoire est pour l'introduction dans la langue française du mot *torque* selon lui nettement plus explicite que l'appellation *moment d'une force*. D'après le Littré de 1877, « moment » vient du latin *momentum*, étant lui-même la contraction de *movimentum* signifiant « mouvement », et désigne à l'origine « le temps que dure un simple mouvement ». « Torque » viendrait lui par contre du latin *torquere* qui désignerait « l'action de tordre », et dont l'étymologie serait par conséquent plus proche du phénomène décrit.

2. Les articulations du bras ne permettent de toute façon pas un tel mouvement.

Définissant

$$m_\theta = \frac{\mathcal{I}}{L^2}, \quad (5.9)$$

$$k_{v_\theta} = \frac{\kappa_v}{L^2}, \quad (5.10)$$

$$k_{e_\theta} = \frac{\kappa_e}{L^2}, \quad (5.11)$$

nous revenons alors à une forme d'équation d'une particule effectuant un mouvement de « translation » :

$$m_\theta \cdot \ddot{x}_\theta + k_{v_\theta} \cdot \dot{x}_\theta + k_{e_\theta} \cdot x_\theta = F_\theta. \quad (5.12)$$

Attention ! Il ne s'agit cependant pas d'un vrai mouvement de translation : notre masse ponctuelle parcourt en réalité la circonférence du cercle dans un mouvement circulaire. Le mouvement serait une translation si nous pouvions ouvrir le cercle et le déplier jusqu'à ce qu'il devienne un segment, ou pour le formuler autrement, dans le cas limite où le rayon du cercle serait infini et réduirait le cercle à une droite.

Par conséquent, les paramètres m_θ , k_{v_θ} et k_{e_θ} ne pourront s'appliquer à un mouvement de translation que dans la mesure où l'arc de cercle décrit par l'extrémité du bras peut s'apparenter à un segment de droite. En pratique, c'est le cas soit lorsque le rayon du cercle L est assez grand, soit lorsque le déplacement x_θ est assez petit pour pouvoir négliger la courbure du cercle du point de vue du mouvement :

$$x_\theta \ll L. \quad (5.13)$$

En résumé, nous réalisons l'approximation décrite à la figure 5.2, valable seulement dans une configuration de départ proche de celle présentée³, uniquement pour un petit mouvement⁴ dans la direction positive de l'axe y et sous l'hypothèse forte que seul le coude participe à ce mouvement. Soit :

$$m_\theta \cdot \ddot{x}_{y^+} + k_{v_\theta} \cdot \dot{x}_{y^+} + k_{e_\theta} \cdot x_{y^+} \approx F_{y^+}. \quad (5.14)$$

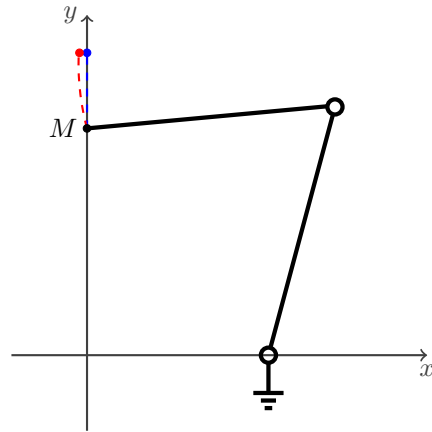


FIGURE 5.2 – Approximation curviligne (trajectoire rouge) d'un mouvement rectiligne de 10 cm dans la direction y (trajectoire bleue).

Aussi grossière que soit cette approximation, il s'agit de la manière la plus simple d'avoir une idée de l'ordre de grandeur des paramètres dont nous avons besoin. En utilisant les valeurs de raideur et d'élasticité

3. Longueurs respectives de l'avant-bras et du bras : 33 cm et 34 cm [Shadmehr, 1994]. Demie largeur d'épaule : 24 cm [Motmans, 2005]. Point de départ de la main : 30 cm en face du thorax [Wagner, 2008].

4. Et encore... la distance à la cible $d = 10$ cm n'est pas tout à fait négligeable devant la longueur du bras $L = 33$ cm.

mesurées par Crevecoeur et Scott [Crevecoeur, 2014] nous trouvons alors en généralisant :

$$k_e = \frac{1.61}{0.33^2} \approx 14.78 \text{ N/m}, \quad (5.15)$$

$$k_v = \frac{0.14}{0.33^2} \approx 1.29 \text{ N} \cdot \text{s/m}. \quad (5.16)$$

La valeur de m étant plus sujet à caution (l'inertie du bras n'est par exemple pas la même que celle de l'avant-bras, et il faut aussi tenir compte de la masse de la main) nous préférons choisir par simplicité une masse de 1 kg et étudier l'influence d'un changement de valeur lors des simulations.

5.2 Constante de temps des muscles τ

Afin de choisir la constante de temps des muscles, τ , nous nous basons sur les valeurs données dans la littérature et répertoriées à la section C.2 de l'annexe C. Celles-ci tournent généralement autour des 40 ms voire des 60 ms .

Nous décidons de garder cette dernière valeur, ce qui nous donne une fréquence de coupure d'environ 2.65 Hz .

5.3 Taux de corrélation λ

Comme le mettent en évidence Wagner et Smith [Wagner, 2008], le bras humain couplé au manipulateur robotique présente une anisotropie inertielle telle qu'une perturbation appliquée dans une direction produira un mouvement non seulement dans cette direction mais également dans la direction perpendiculaire.

Seules deux directions orthogonales, les *directions propres*, donnent lieu à une force qui ne produira effectivement une accélération que dans la même direction que la force appliquée. Les auteurs identifient cette direction propre comme étant un angle variant en fonction des individus entre 43 et 47°, et ce pour un point de départ de la main situé à 30 cm en face du thorax.

La répercussion sur notre modèle est que les équations dynamiques 6.13 et 6.14 sans terme de couplage ($\lambda = 0$) ne sont valables que dans un repère défini de telle manière que les axes correspondent aux directions propres. Si u et v sont les axes correspondant aux directions propres, les équations dynamiques dans ce repère sont alors effectivement découplées :

$$\begin{pmatrix} m & 0 \\ 0 & m \end{pmatrix} \cdot \begin{pmatrix} \ddot{u} \\ \ddot{v} \end{pmatrix} + \begin{pmatrix} k_v & 0 \\ 0 & k_v \end{pmatrix} \cdot \begin{pmatrix} \dot{u} \\ \dot{v} \end{pmatrix} + \begin{pmatrix} k_e & 0 \\ 0 & k_e \end{pmatrix} \cdot \begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} F_u \\ F_v \end{pmatrix}. \quad (5.17)$$

Pour les utiliser dans un repère quelconque, nous pouvons utiliser la matrice de rotation

$$\mathbf{R}(\varphi) = \begin{pmatrix} \cos(\varphi) & -\sin(\varphi) \\ \sin(\varphi) & \cos(\varphi) \end{pmatrix} \in SO(2) \quad (5.18)$$

qui tourne le repère d'un angle φ dans le sens horaire. Or l'application d'une telle matrice de rotation à un tenseur diagonal dont les éléments diagonaux sont identiques ne change pas ce tenseur :

$$\mathbf{R}(\varphi) \cdot \begin{pmatrix} \alpha & 0 \\ 0 & \alpha \end{pmatrix} \cdot \mathbf{R}^{-1}(\varphi) = \begin{pmatrix} \alpha & 0 \\ 0 & \alpha \end{pmatrix}. \quad (5.19)$$

Par conséquent l'application du changement de repère ne fera pas apparaître d'anisotropie ! Le seul moyen de produire une telle anisotropie est donc d'utiliser des paramètres différents m , k_e et k_v pour chaque direction

propre :

$$\begin{pmatrix} m_u & 0 \\ 0 & m_v \end{pmatrix} \cdot \begin{pmatrix} \ddot{u} \\ \ddot{v} \end{pmatrix} + \begin{pmatrix} k_{v_u} & 0 \\ 0 & k_{v_v} \end{pmatrix} \cdot \begin{pmatrix} \dot{u} \\ \dot{v} \end{pmatrix} + \begin{pmatrix} k_{e_u} & 0 \\ 0 & k_{e_v} \end{pmatrix} \cdot \begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} F_u \\ F_v \end{pmatrix}. \quad (5.20)$$

Changement d'axe donnant lieu à l'équation anisotropique générale suivante lors du changement au repère d'axe x et y ⁵ :

$$\begin{pmatrix} m_{xx} & m_{xy} \\ m_{yx} & m_{yy} \end{pmatrix} \cdot \begin{pmatrix} \ddot{x} \\ \ddot{y} \end{pmatrix} + \begin{pmatrix} k_{v_{xx}} & k_{v_{xy}} \\ k_{v_{yx}} & k_{v_{yy}} \end{pmatrix} \cdot \begin{pmatrix} \dot{x} \\ \dot{y} \end{pmatrix} + \begin{pmatrix} k_{e_{xx}} & k_{e_{xy}} \\ k_{e_{yx}} & k_{e_{yy}} \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} F_x \\ F_y \end{pmatrix}. \quad (5.25)$$

Posant maintenant heuristiquement

$$\lambda \cdot F_y = -(m_{xy}\ddot{y} + k_{v_{xy}}\dot{y} + k_{e_{xy}}y), \quad (5.26)$$

$$\lambda \cdot F_x = -(m_{yx}\ddot{x} + k_{v_{yx}}\dot{x} + k_{e_{yx}}x), \quad (5.27)$$

nous trouvons un système couplé via un terme de couplage λ :

$$\begin{cases} m_x \cdot \ddot{x} + k_{v_x} \cdot \dot{x} + k_{e_x} \cdot x & = F_x + \lambda \cdot F_y \\ m_y \cdot \ddot{y} + k_{v_y} \cdot \dot{y} + k_{e_y} \cdot y & = F_y + \lambda \cdot F_x. \end{cases} \quad (5.28)$$

$$\quad (5.29)$$

Le terme λ que nous avons introduit précédemment dans les équations du muscle permet donc de rendre compte de l'anisotropie des muscles. Cependant, en utilisant des paramètres identiques pour les axes x et y il ne nous est paradoxalement pas possible d'évaluer sa valeur. En conséquence, nous posons pour l'instant que $\lambda = 0$ et étudierons l'influence de ce paramètre lors des simulations.

5.4 Discussion

Nous avons tenté dans ce chapitre de trouver des valeurs plausibles de paramètres adaptés à notre modèle. La table 5.1 résume les valeurs des paramètres dérivées dans cette section, et à titre de comparaison, la section C.2 de l'annexe C présente un tableau comparatif des différentes valeurs de paramètres biomécaniques utilisées dans la littérature.

Comme nous l'avons signalé, il s'agit du défaut lié d'un modèle simple : les paramètres mesurés sont difficilement transposables et dépendant alors, comme nous l'avons vu, de la configuration de départ de la main ainsi que de l'orientation dans laquelle s'effectue le mouvement. Laissant maintenant de côté les discussions liées à notre modèle, nous aimerions attirer l'attention sur d'autres aspects liés aux neurosciences.

Une des matières sujette à débat concerne la valeur des paramètres viscoélastiques des muscles. En effet, un point de vue répandu dans le domaine des neurosciences concernant les mouvements est que le cerveau exploite les propriétés élastiques des muscles afin de stabiliser les mouvements durant le contrôle moteur. Shadmehr et Mussa-Ivaldi [Shadmehr, 1994] dans leur article avaient par exemple émis l'idée que les propriétés viscoélastiques des muscles faisaient en sorte que le bras soit stable par rapport à sa trajectoire malgré la

5. Pour un paramètre α , les liens entre les valeurs exprimées dans les deux repères sont les suivants :

$$\alpha_{xx}(\phi) = \alpha_u \cos^2(\varphi) + \alpha_v \sin^2(\varphi), \quad (5.21)$$

$$\alpha_{xy}(\phi) = \alpha_{yx} \quad (5.22)$$

$$= (\alpha_u - \alpha_v) \cdot \sin(\varphi) \cos(\varphi), \quad (5.23)$$

$$\alpha_{yy}(\phi) = \alpha_u \sin^2(\varphi) + \alpha_v \cos^2(\varphi). \quad (5.24)$$

Paramètre	Valeur
m	$= 1 \text{ kg}$
k_e	$= 14.78 \text{ N/m}$
k_v	$= 1.29 \text{ N} \cdot \text{s/m}$
τ	$= 0.06 \text{ s}$
f_c	$\approx 2.65 \text{ Hz}$
λ	$= 0$

TABLE 5.1 – Paramètres biomécaniques retenus.

perturbation présente. Ils avaient par ailleurs assumé une raideur fixe, tout en sachant que la raideur diminue nettement avec la distance à la cible.

Par leurs mesures récentes, Crevecœur et Scott [Crevecœur, 2014] ont notamment pointé la difficulté à représenter la raideur des muscles due au fait que cette raideur varie en fonction de l’extension du muscle. En effet, les expériences réalisées mettent en évidence une raideur relativement importante pour de faibles perturbations (mouvements du coude inférieurs à 5°) contrastant avec une raideur réduite lors de larges perturbations. En tenant compte de la raideur « à long terme », les auteurs ont mis en évidence des paramètres de raideur et de viscosité inférieurs d’un facteur 10 à ceux habituellement utilisés dans la littérature⁶. Ce constat tend par conséquent à infirmer le point de vue selon lequel les propriétés élastiques des muscles empêchent à eux seuls l’instabilité due aux perturbations

Ce chapitre clôturait la mise au point de notre modèle du bras, et également la première partie de ce mémoire. Nous avons maintenant tous les outils en main pour nous lancer dans les simulations numériques, simulations numériques que nous entamerons dès les prochains chapitres.

6. $k_e = 1.61 \text{ Nm/rad}$ et $k_v = 0.14 \text{ Nm} \cdot \text{s/rad}$ contre $k_e = 16 \text{ Nm/rad}$ et $k_v = 2.4 \text{ Nm} \cdot \text{s/rad}$.

Récapitulatif

Avant d'attaquer la suite de ce mémoire, profitons d'un bref chapitre pour prendre une vue d'ensemble synthétique des principaux outils mathématiques et des modèles développés jusqu'à présent.

Représentation d'état : discrète et linéaire avec bruits additifs gaussiens.

$$\begin{cases} \mathbf{x}[k+1] &= \mathbf{A}\mathbf{x}[k] + \mathbf{B}\mathbf{u}[k] + \xi[k] \\ \mathbf{y}[k] &= \mathbf{H}\mathbf{x}[k] + \omega[k] \end{cases} \quad (6.1)$$

Fonction de coût : quadratique.

$$J = E \left\{ \sum_{k=1}^{N-1} (\mathbf{x}^T \mathbf{Q} \mathbf{x}[k] + \mathbf{u}^T \mathbf{Q}_u \mathbf{u}[k]) + \mathbf{x}^T \mathbf{Q}_N \mathbf{x}[N] \right\} \quad (6.3)$$

Estimation optimale : filtre de Kalman.

$$\begin{cases} \hat{\mathbf{x}}[1] &= \mathbf{m}_x[1] \end{cases} \quad (6.4)$$

$$\begin{cases} \hat{\mathbf{x}}[k+1] &= \mathbf{A}\hat{\mathbf{x}}[k] + \mathbf{B}\mathbf{u}[k] + \mathbf{K}(\mathbf{y}[k] - \mathbf{H}\hat{\mathbf{x}}[k]) \end{cases} \quad (6.5)$$

$$\mathbf{K}[k] = \mathbf{A} \cdot \boldsymbol{\Sigma}_x[k] \cdot \mathbf{H}^T \cdot (\mathbf{H} \cdot \boldsymbol{\Sigma}_x[k] \cdot \mathbf{H}^T + \mathbf{R}_\omega)^{-1} \quad (6.6)$$

$$\begin{cases} \boldsymbol{\Sigma}_x[1] &= \mathbf{R}_x[1,1] \end{cases} \quad (6.7)$$

$$\begin{cases} \boldsymbol{\Sigma}_x[k+1] &= \mathbf{A} \cdot \boldsymbol{\Sigma}_x[k] \cdot \mathbf{A}^T - \mathbf{K}[k] \cdot \mathbf{H} \cdot \boldsymbol{\Sigma}_x[k] \cdot \mathbf{A}^T + \mathbf{R}_\xi \end{cases} \quad (6.8)$$

Contrôle optimal : feedback linéaire.

$$\mathbf{u}[k] = -\mathbf{L}\hat{\mathbf{x}}[k] \quad (6.9)$$

$$\mathbf{L}[k] = (\mathbf{B}^T \mathbf{S}[k+1] \mathbf{B} + \mathbf{Q}_u)^{-1} \cdot \mathbf{B}^T \mathbf{S}[k+1] \mathbf{A} \quad (6.10)$$

$$\begin{cases} \mathbf{S}[N] &= \mathbf{Q}_N \end{cases} \quad (6.11)$$

$$\begin{cases} \mathbf{S}[k] &= \mathbf{A}^T \mathbf{S}[k+1] \mathbf{A} - \mathbf{A}^T \mathbf{S}[k+1] \mathbf{B} \cdot \mathbf{L}[k] + \mathbf{Q}[k] \end{cases} \quad (6.12)$$

Modèle du bras : masse ponctuelle bidimensionnelle.

$$\begin{cases} m \cdot \ddot{x} + k_v \cdot \dot{x} + k_e \cdot x & = F_x + \lambda \cdot F_y \end{cases} \quad (6.13)$$

$$\begin{cases} m \cdot \ddot{y} + k_v \cdot \dot{y} + k_e \cdot y & = F_y + \lambda \cdot F_x \end{cases} \quad (6.14)$$

$$\begin{cases} \tau \cdot \dot{F}_x + F_x & = u_x \end{cases} \quad (6.15)$$

$$\begin{cases} \tau \cdot \dot{F}_y + F_y & = u_y \end{cases} \quad (6.16)$$

Vecteur d'état et de contrôle.

$$\mathbf{x}(t) = \begin{pmatrix} x & y & \dot{x} & \dot{y} & \dot{F}_x & \dot{F}_y \end{pmatrix}^T \quad (6.17)$$

$$\mathbf{u}(t) = \begin{pmatrix} u_x & u_y \end{pmatrix}^T \quad (6.18)$$

Matrice dynamique et de contrôle.

$$\mathbf{A}_t = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ -k_e/m & 0 & -k_v/m & 0 & 1/m & \lambda/m \\ 0 & -k_e/m & 0 & -k_v/m & \lambda/m & 1/m \\ 0 & 0 & 0 & 0 & -1/\tau & 0 \\ 0 & 0 & 0 & 0 & 0 & -1/\tau \end{pmatrix} \quad (6.19)$$

$$\mathbf{B}_t = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1/\tau & 0 \\ 0 & 1/\tau \end{pmatrix} \quad (6.20)$$

Calibration de la simulation

Maintenant que nous disposons à la fois d'un contrôleur optimal et d'un modèle du bras, nous allons entrer dans le cœur de ce mémoire en commençant à simuler numériquement des mouvements d'atteintes. Dans ce chapitre, nous calibrerons notre simulation en étudiant les effets des différents paramètres, avant d'introduire des perturbations et des délais dans les chapitres ultérieurs.

Disposant des paramètres biomécaniques du bras développés au chapitre 5, il nous faut encore décider des éléments suivants :

- ▷ Le pas de discrétisation de la simulation et sa durée.
- ▷ L'état de départ de la main ainsi que celui de la cible à atteindre.
- ▷ La forme de la matrice d'observation $\mathbf{H}$.
- ▷ La variance des bruits de traitement et de mesure ainsi que la variance initiale.
- ▷ La fonction de coût à utiliser.

La première partie de ce chapitre sera donc consacrée à ces matières tandis que dans la deuxième nous analyserons l'impact des paramètres sur les résultats donnés par la simulation.

7.1 Paramètres de la simulation

Conformément avec les résultats présentés dans la littérature¹, nous aimerions obtenir une durée de mouvement tournant autour de $T = 500 \text{ ms}$ pour une cible située à une distance de 10 cm . Comme pas de discrétisation, nous démarrerons avec une valeur, également usuelle dans la littérature, de $\Delta t = 10 \text{ ms}$.

La position de départ de notre masse ponctuelle sera le centre du repère cartésien. Elle sera de plus initialement au repos, c'est-à-dire avec une vitesse et une accélération initiales nulles dans les deux directions. Le vecteur d'état de départ $\mathbf{x}[1]$ est donc donné par

$$\mathbf{x}[1] = \begin{pmatrix} x_1 & y_1 & v_{x_1} & v_{y_1}^* & F_{x_1}^* & F_{y_1}^* \end{pmatrix}^T = \begin{pmatrix} 0 \text{ m} & 0 \text{ m} & 0 \text{ m/s} & 0 \text{ m/s} & 0 \text{ N} & 0 \text{ N} \end{pmatrix}^T. \quad (7.1)$$

La cible à atteindre sera quant à elle située à une distance de $\rho = 10 \text{ cm}$ du point de départ de la main. L'orientation de cette cible pourra être changée, mais nous démarrerons arbitrairement avec un angle de

1. Voir la section C.3 de l'annexe C.

$\varphi = 60^\circ$. Ceci nous donne comme coordonnées cartésiennes

$$\begin{cases} x^* &= 0.1 \cdot \cos(60^\circ) = 0.05 \text{ m} \\ y^* &= 0.1 \cdot \sin(60^\circ) \approx 0.087 \text{ m}. \end{cases} \quad (7.2)$$

Nous stipulerons également que la main doit non seulement atteindre la cible mais également s'y arrêter, et par conséquent le vecteur cible sera spécifié par

$$\mathbf{x}^* = \begin{pmatrix} x^* & y^* & v_x^* & v_y^* & F_x^* & F_y^* \end{pmatrix}^T = \begin{pmatrix} 0.05 \text{ m} & 0.087 \text{ m} & 0 \text{ m/s} & 0 \text{ m/s} & 0 \text{ N} & 0 \text{ N} \end{pmatrix}^T. \quad (7.4)$$

Enfin, nous aimerions des résultats présentant moins de 5 mm d'erreur sur la position finale, soit

$$\sqrt{e_x^2[N] + e_y^2[N]} < 0.005 \text{ m} \quad (7.5)$$

où $e_x[k] = x^* - x[k]$ et $e_y[k] = y^* - y[k]$ représentent les erreurs en position.

Il s'agit d'une tolérance assez faible puisqu'en examinant les tracés de la figure 1.2 du chapitre 1, on se rend compte que même en l'absence de champ de force les participants commettent des erreurs de quelques cm sur la position.

En guise de matrice d'observation $\mathbf{H}$, nous choisissons d'utiliser la matrice d'identité $\mathbf{I}$. Comme nous l'avons discuté dans la section 2.5 du chapitre 2 cela signifie en pratique que nous nous ne intéressons pas au problème d'observabilité, soit au problème de la reconstruction du vecteur $\mathbf{x}$ à partir de la mesure $\mathbf{y}$.

7.1.1 Paramètres des bruits

La variance des bruits de traitement R_ξ et de mesure R_ω est initialisée à une valeur de 10^{-4} . Le bruit de traitement ne s'applique que sur les composantes de forces du vecteur d'état, tandis que le bruit de mesure s'applique quant à lui sur toutes les composantes du vecteur de mesure.

Les vecteurs de bruit augmentés sont donc de la forme :

$$\xi[k] = \begin{pmatrix} 0 & 0 & 0 & 0 & \xi_{F_x}[k] & \xi_{F_y}[k] & 0 & \cdots & 0 \end{pmatrix}^T, \quad (7.6)$$

$$\omega[k] = \begin{pmatrix} \omega_x[k] & \omega_x[k] & \omega_{v_x}[k] & \omega_{v_y}[k] & \omega_{F_x}[k] & \omega_{F_y}[k] & 0 & \cdots & 0 \end{pmatrix}^T. \quad (7.7)$$

Les composantes de ces vecteurs de bruit sont aléatoirement sélectionnés à chaque pas de temps à partir d'une distribution gaussienne de moyenne nulle et de variance correspondante.

Enfin nous allons considérer une variance de $R_x = 10^{-6}$ pour l'estimation initiale de l'état, soit deux ordres de grandeur moindre que le bruit de traitement, et ce pour tenir compte du fait que, la main restant à l'arrêt à la position de départ avant le début du mouvement, celle-ci a le temps d'être bien estimée par le système nerveux. Nous ajoutons donc également un vecteur ω_Σ de bruit gaussien à moyenne nulle et de variance R_x sur l'estimation de l'état initial. Nous avons donc :

$$\begin{cases} \hat{\mathbf{x}}[1] &= \mathbf{x}[1] + \omega_\Sigma \end{cases} \quad (7.8)$$

$$\begin{cases} \Sigma_x[1] &= R_x \cdot \mathbf{I}_{12 \times 12}. \end{cases} \quad (7.9)$$

7.1.2 Paramètres de la fonction de coût

Pour rappel, la fonction de coût J est constituée d'un terme de coût courant déterminé par la matrice $\mathbf{Q}$, d'un terme de coût de contrôle déterminé par $\mathbf{Q}_u$ et d'un terme de coût final déterminé par $\mathbf{Q}_N$ et prend la forme suivante :

$$J = E \left\{ \sum_{k=1}^{N-1} (\mathbf{x}^T \mathbf{Q} \mathbf{x}[k] + \mathbf{u}^T \mathbf{Q}_u \mathbf{u}[k]) + \mathbf{x}^T \mathbf{Q}_N \mathbf{x}[N] \right\}. \quad (7.10)$$

La matrice déterminant le coût de contrôle sera simplement conditionnée par un poids w_u identique pour u_x et u_y :

$$\mathbf{Q}_u = w_u \cdot \mathbf{I}_{2 \times 2}. \quad (7.11)$$

Concernant le coût sur l'état, nous démarrons en expérimentant un coût dynamique nul durant le mouvement ($\mathbf{Q} = \mathbf{0}_{12 \times 12}$) et un coût final qui pénalise le carré de la différence de l'état final $\mathbf{x}_\Delta[N]$ par rapport à l'état souhaité $\mathbf{x}^*$:

$$\mathbf{x}^T \mathbf{Q} \mathbf{x}[k] = 0 \quad (7.12)$$

$$\begin{aligned} \mathbf{x}^T \mathbf{Q}_N \mathbf{x}[N] &= w_x \cdot (x^* - x[N])^2 + w_{v_x} \cdot (v_x^* - v_x[N])^2 + w_{F_x} \cdot (F_x^* - F_x[N])^2 \\ &\quad + w_y \cdot (y^* - y[N])^2 + w_{v_y} \cdot (v_y^* - v_y[N])^2 + w_{F_y} \cdot (F_y^* - F_y[N])^2, \end{aligned} \quad (7.13)$$

où $\mathbf{x}$ est le vecteur d'état augmenté² et $\mathbf{w} = (w_x \ w_y \ w_{v_x} \ w_{v_y} \ w_{F_x} \ w_{F_y})^T$ est un vecteur de poids à déterminer.

C'est ici que le fait d'avoir utilisé un vecteur d'état augmenté prend enfin son sens. En effet, un tel coût s'obtient facilement³ en utilisant une matrice

$$\mathbf{Q}_N = \begin{pmatrix} w_x & 0 & 0 & 0 & 0 & 0 & -w_x & 0 & 0 & 0 & 0 & 0 \\ 0 & w_y & 0 & 0 & 0 & 0 & 0 & -w_y & 0 & 0 & 0 & 0 \\ 0 & 0 & w_{v_x} & 0 & 0 & 0 & 0 & 0 & -w_{v_x} & 0 & 0 & 0 \\ 0 & 0 & 0 & w_{v_y} & 0 & 0 & 0 & 0 & 0 & -w_{v_y} & 0 & 0 \\ 0 & 0 & 0 & 0 & w_{F_x} & 0 & 0 & 0 & 0 & 0 & -w_{F_x} & 0 \\ 0 & 0 & 0 & 0 & 0 & w_{F_y} & 0 & 0 & 0 & 0 & 0 & -w_{F_y} \\ -w_x & 0 & 0 & 0 & 0 & 0 & w_x & 0 & 0 & 0 & 0 & 0 \\ 0 & -w_y & 0 & 0 & 0 & 0 & 0 & w_y & 0 & 0 & 0 & 0 \\ 0 & 0 & -w_{v_x} & 0 & 0 & 0 & 0 & 0 & w_{v_x} & 0 & 0 & 0 \\ 0 & 0 & 0 & -w_{v_y} & 0 & 0 & 0 & 0 & 0 & w_{v_y} & 0 & 0 \\ 0 & 0 & 0 & 0 & -w_{F_x} & 0 & 0 & 0 & 0 & 0 & w_{F_x} & 0 \\ 0 & 0 & 0 & 0 & 0 & -w_{F_y} & 0 & 0 & 0 & 0 & 0 & w_{F_y} \end{pmatrix}. \quad (7.14)$$

Il reste à déterminer les valeurs à donner aux poids. Commenant par les poids du vecteur $\mathbf{w}$, nous souhaitons privilégier la précision sur la position finale plutôt que sur la vitesse ou la force finale :

$$\mathbf{w} = (w_x \ w_y \ w_{v_x} \ w_{v_y} \ w_{F_x} \ w_{F_y})^T = (1 \ 1 \ 0.1 \ 0.1 \ 0.01 \ 0.01)^T, \quad (7.15)$$

où ce ne sont pas tant les valeurs absolues données aux poids qui importent mais bien leurs valeurs relatives les unes par rapport aux autres.

Le poids de contrôle est quant à lui initialisé à $w_u = 10^{-6}$. Sa valeur sera par la suite calibrée pour permettre une durée de mouvement déterminée.

2. $\mathbf{x} = (x \ y \ v_x \ v_y \ F_x \ F_y \ x^* \ y^* \ v_x^* \ v_y^* \ F_x^* \ F_y^*)^T$.

3. Pour le remarquer, il suffit de considérer un exemple simplifié où le vecteur d'état augmenté serait $\mathbf{x}_x = (x \ x^*)^T$. Dans ce cas

$$(x \ x^*) \cdot \begin{pmatrix} w_x & -w_x \\ -w_x & w_x \end{pmatrix} \cdot \begin{pmatrix} x \\ x^* \end{pmatrix} = w_x \cdot (x^* - x)^2,$$

et en généralisant cette constatation à notre vecteur d'état $\mathbf{x}$, nous trouvons la matrice $\mathbf{Q}_N$ de l'équation 7.14.

7.2 Structure du programme

Jetons maintenant un bref coup d'œil à la structure du programme utilisé afin de réaliser les simulations. Celui-ci est implémenté à l'aide du langage **GNU Octave**, dont la syntaxe est identique à celle de **Matlab**, et dont le code est disponible dans l'annexe **F**.

Sa structure, schématisée à la figure 7.1, est la suivante :

- A) À l'aide des paramètres reçus en entrée le reste des paramètres utiles à la simulation est initialisé.
- B) Les matrices d'états sont formées et les matrices augmentées calculées.
- C) Si une perturbation est présente (voir chapitre 8), la matrice de perturbation est également formée.
- D) La réalisation des vecteurs de bruit est précalculée.
- E) Les gains du filtre de Kalman sont calculés par itération à l'aide du modèle direct et des matrices de covariance des bruits.
- F) Les matrices de coût sont formées.
- G) Les gains de contrôle sont calculés itérativement à l'aide du modèle direct et des matrices de coût.
- H) Les différents vecteurs (état, mesure, estimation) sont initialisés.
- I) L'évolution du système est simulée, avec un éventuel délai sur les mesures (voir chapitre 10).
- J) Les variables et les graphiques d'intérêt sont exportés.

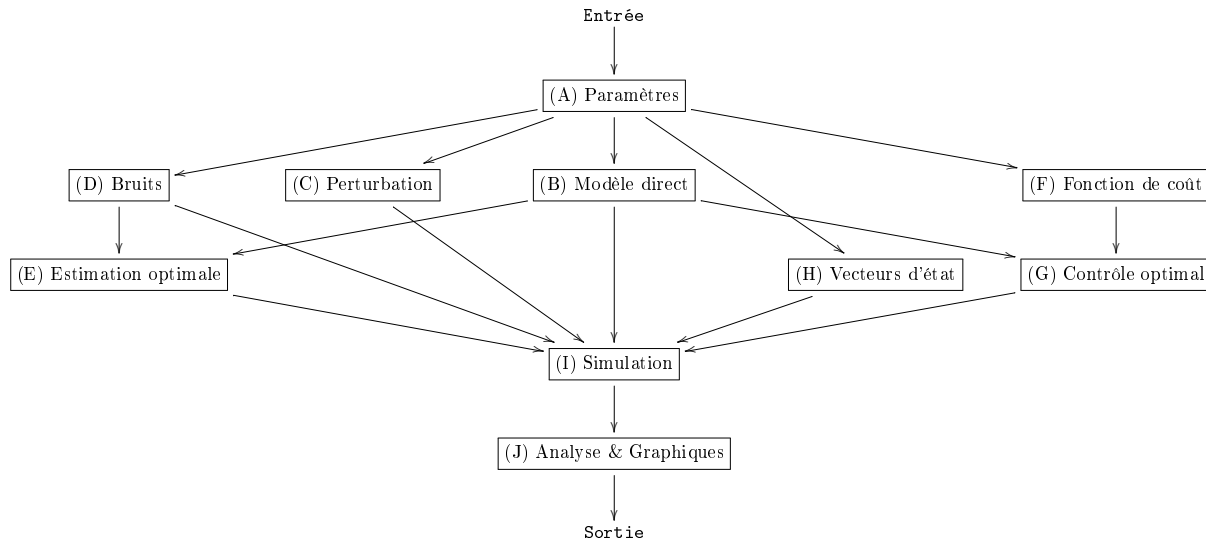


FIGURE 7.1 – Structure du programme.

7.3 Simulation

Dans cette section, nous allons simuler l'évolution de notre modèle en l'absence de perturbation et de délais. Nous commencerons par utiliser les paramètres initiaux fixés au début de ce chapitre et nous étudierons ensuite l'impact de leur modification sur la simulation.

7.3.1 Paramètres initiaux

Les paramètres initiaux utilisés pour la simulation sont résumés dans la table 7.1.

Paramètre	Valeur	Paramètre	Valeur	Paramètre	Valeur
T	$= 350 \text{ ms}$	R_ξ	$= 10^{-4}$	m	$= 1 \text{ kg}$
Δt	$= 0.01 \text{ s}$	R_ω	$= 10^{-4}$	k_e	$= 14.78 \text{ N/m}$
ρ	$= 0.1 \text{ m}$	R_x	$= 10^{-6}$	k_v	$= 1.29 \text{ N} \cdot \text{s/m}$
ϕ	$= 60^\circ$	w_x, w_y	$= 1$	τ	$= 0.06 \text{ s}$
x^*	$= 0.05 \text{ m}$	w_{v_x}, w_{v_y}	$= 0.1$	λ	$= 0$
y^*	$\approx 0.087 \text{ m}$	w_{F_x}, w_{F_y}	$= 0.01$		
		w_u	$= 10^{-6}$		

TABLE 7.1 – Paramètres initiaux utilisés pour la simulation.

Après simulation avec ces paramètres, nous trouvons la trajectoire indiquée dans la figure 7.2. Nous observons que le contrôleur parvient bien à guider la main jusqu'à la cible dans le temps imparti avec une précision de l'ordre du millimètre⁴. La trajectoire est bien une ligne droite et nous pouvons observer que l'estimation donnée par le filtre de Kalman suit bien cette trajectoire. Un point qu'il est important de souligner, car celui-ci aura des répercussions ultérieures, est que, la fonction de coût ne pénalisant les erreurs d'état qu'au tout dernier moment, le contrôleur répartit le mouvement sur toute la période de temps dont il dispose.

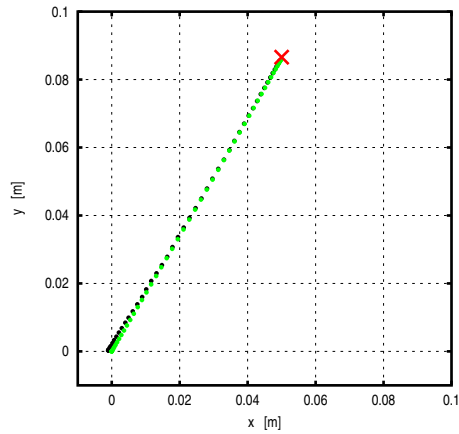


FIGURE 7.2 – Trajectoire (vert) de la main dans le plan xy . Les pointillés noirs indiquent la position estimée, la croix rouge la cible à atteindre.

La figure 7.3 présente quant à elle respectivement la position, la vitesse et la force dans l'axe x ⁵ ainsi

4. Des résultats détaillés sont disponibles dans la table D.1 de l'annexe D.

5. Notons que les résultats pour l'axe y sont complètement similaires à ceux de l'axe x .

que le coût en fonction du temps en fonction du temps. Remarquons le profil caractéristique en forme de cloche de la vitesse et le fait que la position varie sans-à-coups (“smooth trajectory”). La main est de plus bien à l’arrêt à la fin du mouvement, puisque la vitesse et la force tendent vers zéro. Enfin notons le coût de contrôle plus important au début, diminuant ensuite et subissant un regain avant la fin du mouvement. Le profil de ce coût est à mettre en relation avec celui de la force : au début du mouvement le contrôleur doit donner un premier foyer d’impulsions afin d’accélérer la particule vers la cible et vers la fin du mouvement un autre foyer d’impulsions afin de la ralentir.

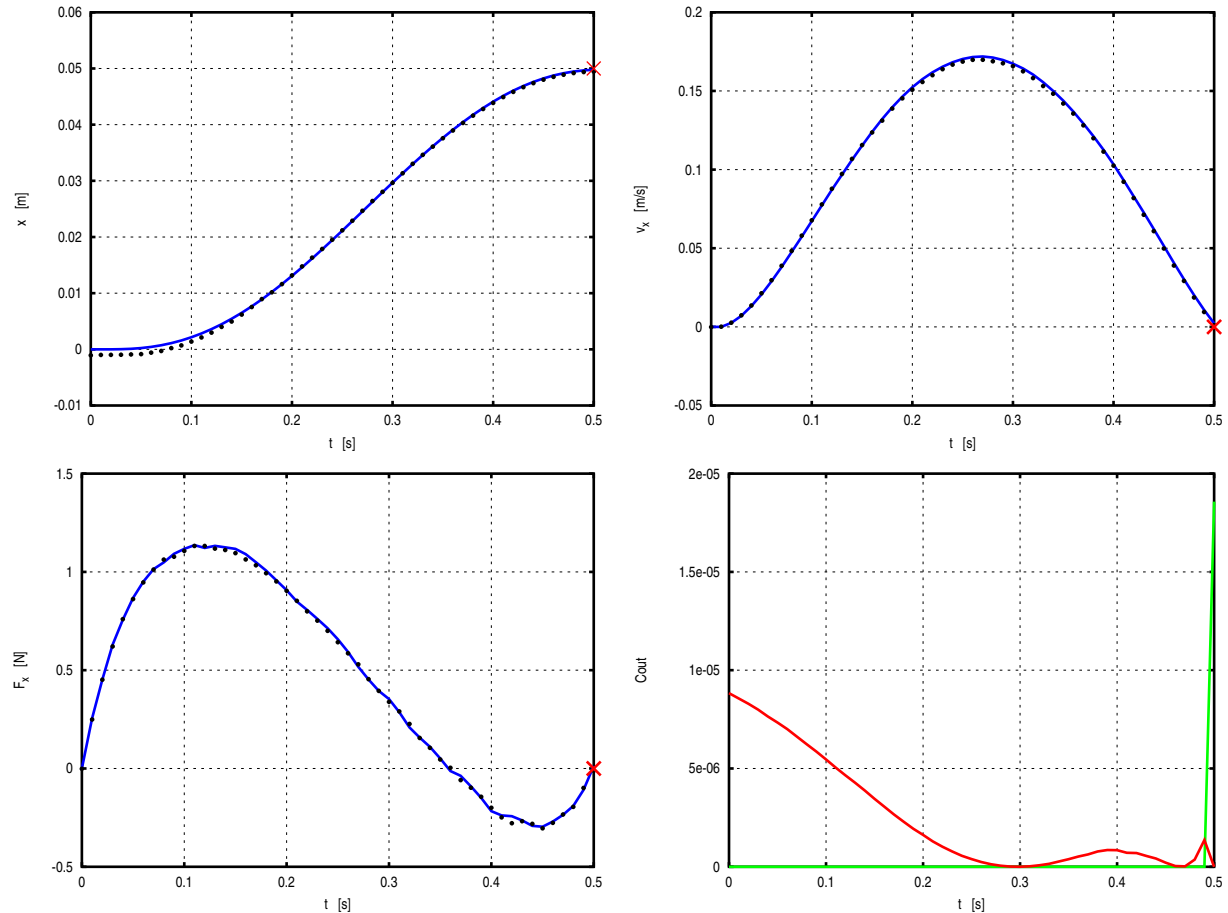


FIGURE 7.3 – Position (haut gauche), vitesse (haut droit) et force (bas gauche) dans l’axe x en fonction du temps. Les pointillés noirs indiquent l’état estimé, la croix rouge l’objectif à atteindre. Bas droit : Coût dynamique (vert) et coût de contrôle (rouge) en fonction du temps.

7.3.2 Variation des paramètres

Étudions maintenant les effets de la variation des différents paramètres sur la simulation. Partant des paramètres de base de la table C.2, nous varierons un paramètre à la fois.

Commençons par faire varier le paramètre de raideur k_e . En le multipliant par dix ($k_e = 147.8 \text{ N/m}$) afin d’obtenir les valeurs usuelles de la littérature⁶, nous obtenons un effet intéressant et facilement interprétable : de manière surprenante il est plus économique pour le contrôleur de prendre de l’élan plutôt

6. Voir la discussion du chapitre 5.

que d'aller directement vers la cible, comme le montre la figure 7.4. Ceci est en réalité une conséquence de l'étalement du contrôle sur tout le temps imparti. Notons d'ailleurs que le même phénomène de prise d'élan apparaît lorsque le temps T de la simulation est suffisamment augmenté.

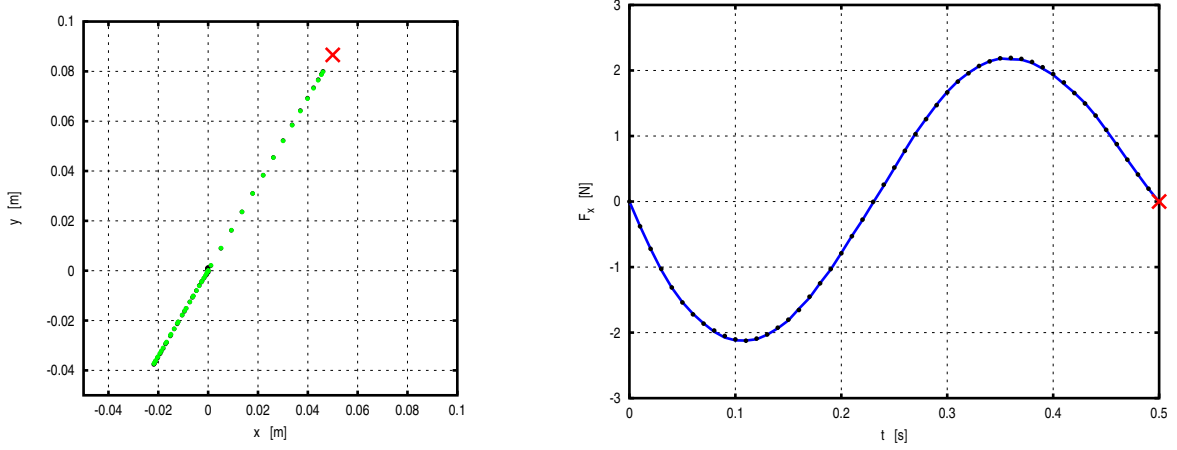


FIGURE 7.4 – Trajectoire (gauche) et force dans l'axe x (droite) lorsque la raideur est multipliée par 10 ($k_e = 147.8 \text{ N/m}$), montrant un phénomène de « prise d'élan ».

Pour arriver à la valeur usuelle donné dans la littérature avec la viscosité, il faut multiplier celle-ci par vingt ($k_v = 25.8 \text{ N} \cdot \text{m/s}$). Le contrôleur n'arrive alors plus à amener le système jusqu'à sa cible et comme le montre la gauche de la figure 7.5, le profil en cloche de la vitesse est alors nettement accentué, avec une vitesse maximale moindre, comme si le profil initial s'était en quelque sorte « diffusé ».

En ce qui concerne m , l'utilisation d'une masse plus faible retarde le début du mouvement comme le montre le profil de vitesse différé de la droite de la figure 7.5. Lorsque la masse descend en-dessous d'un certain seuil ($\approx 0.25 \text{ kg}$) la raideur devient plus prépondérante et on voit l'apparition du même phénomène de « prise d'élan ». Lorsque la masse est encore diminuée ($\approx 0.1 \text{ kg}$) ce phénomène disparaît, et le départ est alors de nouveau nettement différé.

Nous pouvons conjecturer que l'utilisation d'une masse plus importante va quant à elle nécessiter une durée de mouvement plus grande. Et effectivement, lorsque la masse augmente, rapidement le contrôleur n'atteint plus la précision requise de 5 mm . Lorsque le contrôleur semble ne pas disposer d'assez de temps pour pouvoir réaliser son mouvement il existe deux tactiques afin d'y remédier. La première consiste évidemment à augmenter le temps de la simulation T . Mais il est également possible d'y arriver en gardant le même temps de simulation mais en diminuant alors le poids du coût de contrôle w_u .

Intéressons-nous maintenant à la constante de temps τ des muscles. Puisque notre modèle intègre un filtre du premier ordre, cette constante va directement dicter la vitesse à laquelle la force s'adapte à la commande reçue. Augmenter cette constante va donc augmenter le temps de réaction de la force, ce qui fait rapidement chuter la précision du mouvement. Est-ce qu'alors diminuer la valeur de τ va augmenter la précision du mouvement ? La réponse est oui, mais jusqu'à un certain point. En effet, la constante de temps diminuant, la fréquence de coupure diminue également et va affecter le profil de la force en le rendant oscillatoire comme le montre la gauche de la figure 7.6. Par conséquent, comme le fait d'augmenter le temps de la simulation ou de diminuer le poids de contrôle, le fait d'écourter la constante de temps des muscles améliore la précision du contrôleur pour un mouvement où celui-ci n'a pas assez de temps pour atteindre la cible. Toutefois, le phénomène d'oscillation empêchant de descendre en-deça d'un certain seuil, cette correction seule n'est pas

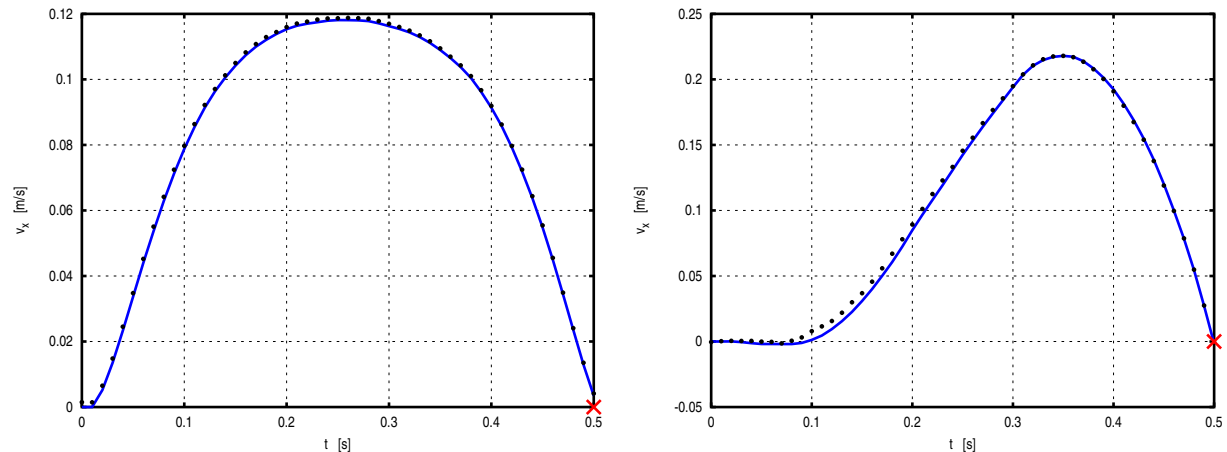


FIGURE 7.5 – Gauche : vitesse dans l’axe x lorsque la viscosité est multipliée par 20 ($k_v = 25.8 \text{ N} \cdot \text{m/s}$), mettant en évidence la « diffusion » du profil de vitesse. Droite : vitesse dans l’axe x pour une masse de $m = 0.25 \text{ kg}$ montrant un départ différé.

assez efficace pour rétablir la précision du mouvement.

Passons au paramètre λ . Lorsque ce terme de couplage est actif, le profil de la force dans l’axe x s’en trouve modifié, comme si celui-ci était bruité ainsi que l’illustre la droite de la figure 7.6. Au-delà de $\lambda = 0.6$ le profil de force dans l’axe x change et prend maintenant des valeurs négatives, mais à mesure que λ approche de 1 retourne lentement à son profil initial. Ce n’est pas le cas du profil de force dans l’axe y qui lui reste bien lisse lorsque λ varie. Puisque l’angle de la cible $\varphi = 60^\circ$ implique que la distance y à parcourir est plus grande que la distance x , il nous vient à l’idée d’observer ce qui se passe lorsque l’angle complémentaire $\varphi = 30^\circ$ est utilisé. Et effectivement, un phénomène symétrique se produit : c’est maintenant le profil de force dans l’axe x qui semble épargné et celui dans l’axe y qui est sujet aux modifications décrites précédemment. Suivant la même intuition, une analyse plus attentive révèle que les valeurs seuils de modifications des profils de vitesse ainsi que le taux de modifications d’un axe par rapport à l’autre varient en fonction de l’angle à la cible.

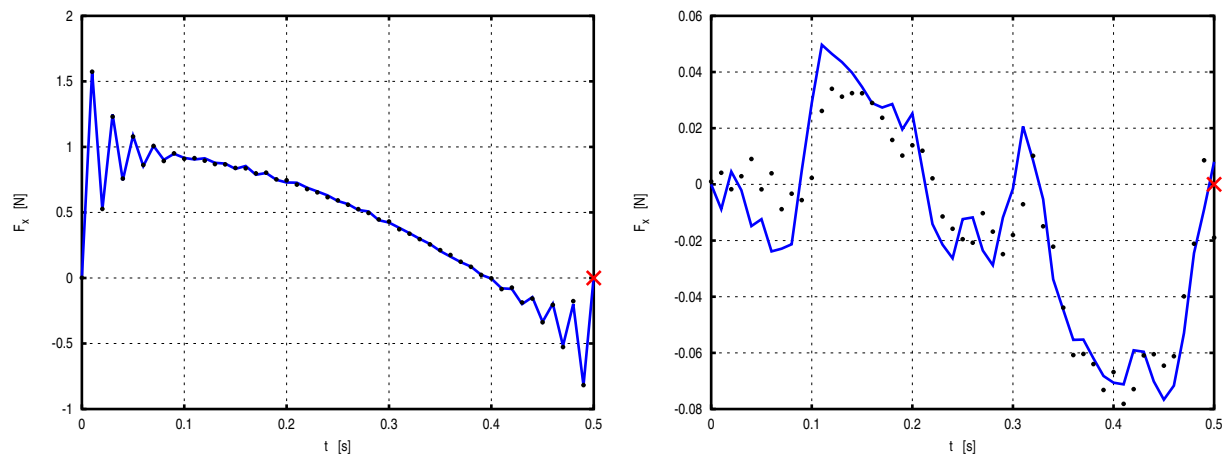


FIGURE 7.6 – Gauche : force dans l’axe x lorsque $\tau = 0.006$ présentant un phénomène oscillatoire. Droite : force dans l’axe x lorsque $\lambda = 0.6$ permettant d’observer un phénomène de « bruitage ».

Nous en arrivons alors à l'influence des paramètres de bruit. Comme nous pouvons le prévoir, augmenter la variance initiale R_x rend les estimations initiales plus imprécises ainsi que le montre la figure 7.7, ce qui nécessite un peu de temps avant que celles-ci ne redeviennent correctes. Une variance de bruit de traitement R_ξ plus importante va quant à elle, en rendant le profil de force plus erratique, donner lieu à des trajectoires plus exotiques ainsi que l'esquisse également la figure 7.7. Enfin, le contrôleur est pour l'instant complètement robuste à l'augmentation de bruit de mesure via le paramètre R_w puisque aucune perturbation n'est présente et que celui-ci peut en conséquence utiliser le modèle direct du système pour réaliser des prédictions très précises.

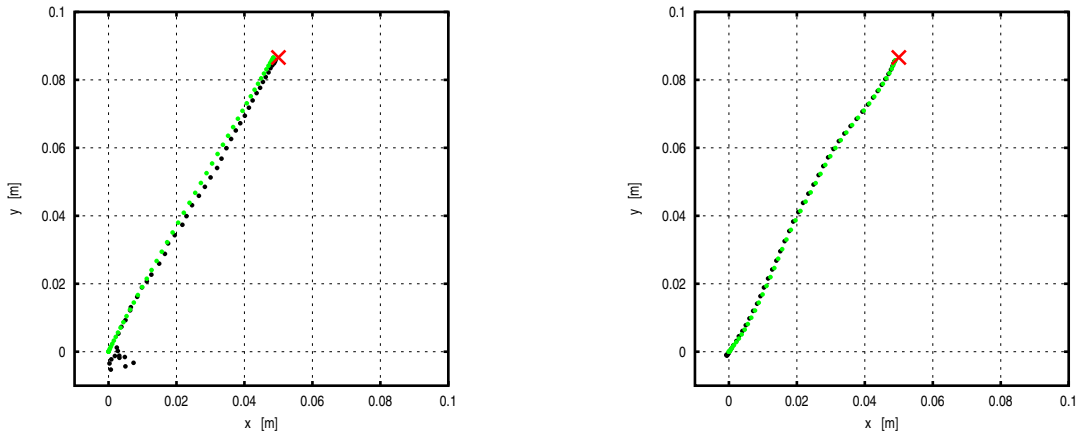


FIGURE 7.7 – Gauche : trajectoire lorsque la variance initiale est augmentée ($R_x = 10^{-2}$) donnant lieu à plus d'imprécisions sur les premières estimations (pointillés noirs). Droite : trajectoire lorsque la variance du bruit de traitement est augmentée ($R_\xi = 10^{-1}$) entraînant un mouvement plus erratique.

Mentionnons également que la simulation s'adapte bien aux changements du vecteur de poids $\mathbf{w}$. Notamment on aura que le coût du mouvement diminue nettement si les poids sur la vitesse et sur la force sont mis à zéro, puisque le contrôleur ne s'occupe alors que de faire aller la main jusqu'à la cible sans s'inquiéter de la vitesse ou de la force finale et n'a donc pas besoin d'appliquer un contrôle nécessitant d'asservir ces valeurs. Cette situation se rencontre en pratique dans les expériences où l'instruction donnée aux participants est de simplement passer par l'emplacement d'une cible sans devoir s'y arrêter.

Pour terminer, la table E.1 de l'annexe E répertorie la tolérance du contrôleur aux variations de paramètres.

7.4 Discussion

Jusqu'à présent, les résultats obtenus coïncident bien avec les caractéristiques des mouvements d'atteinte en l'absence de perturbation : en adéquation avec la figure 7.8 la trajectoire à la cible est bien une droite, et de plus le profil de vitesse exhibe une silhouette à l'allure de cloche. En effet, en comparant les tracés de la vitesse tangentielle relevés par Shadmehr et Mussa-Ivaldi en 1994 et reproduits à la figure 7.9, on remarque bien le même profil bombé. En outre, la durée du mouvement d'atteinte tourne bien autour de 500 ms et la magnitude de la vitesse, avoisinant 0.5 m/s, se situe également dans le même ordre de grandeur que celui de notre simulation.

Enfin, un autre phénomène lié à la discrétisation du modèle est à discuter. En effet, diminuer le pas de temps de la simulation devrait améliorer la précision des résultats. Cependant, due à la nature discrète du

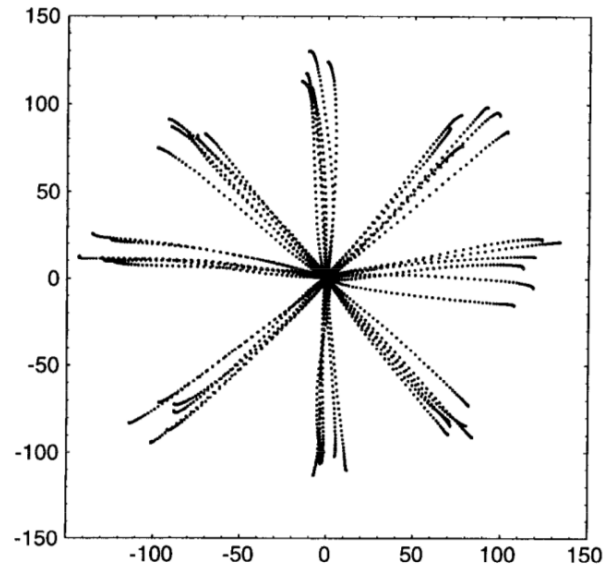


FIGURE 7.8 – Trajectoires typiques lors de mouvements d’atteinte en l’absence de perturbation. Les déplacements sont indiqués en *mm*. Source : [Shadmehr, 1994].

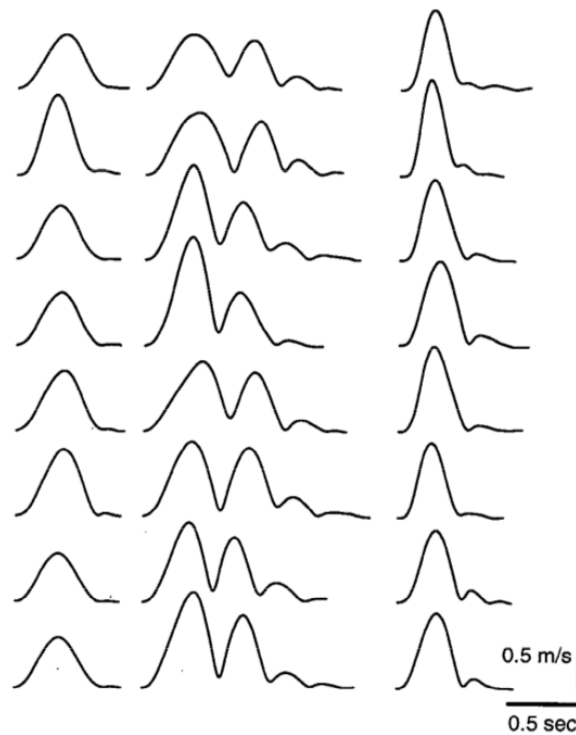


FIGURE 7.9 – Profils typiques de la vitesse tangentielle lors de mouvements d’atteinte. Les profils de haut en bas correspondent à des cibles d’orientation $0, 45, \dots, 315^\circ$. Gauche : profils en l’absence de perturbation. Centre : profils en présence d’un champ de force. Droite : profils en présence d’un champ de force après adaptation. Source : [Shadmehr, 1994].

bruit, plus la simulation comporte de pas discrets et plus il y a de bruit qui l'entache. Par conséquent le fait d'utiliser un trop petit pas de discrétisation Δt nuit rapidement à la précision du contrôleur. Ce point paradoxal assez fâcheux est de nouveau dû à la distribution du mouvement sur l'ensemble du temps disponible, et nous verrons dans le chapitre 9, comment remédier à cette invraisemblance.

Auparavant, le prochain chapitre investiguera l'impact d'un champ de force perturbateur sur notre modèle.

Modélisation de la perturbation

Les résultats obtenus par les simulations du chapitre 7 ont montré que notre modèle était capable de simuler le comportement des participants lors de mouvements d'atteinte en l'absence de perturbation. Mais que se passe-t-il lorsqu'un champ de force perturbateur est soudain introduit dans l'environnement, champ de force dont le système nerveux n'a pas encore eu le d'assimiler dans son modèle interne ?

Cette perturbation induite par le champ de force peut être modélisée en ajoutant un terme $\Delta \mathbf{A}$ constant et par conséquent déterministe à la matrice dynamique $\mathbf{A}$:

$$\mathbf{x}[k+1] = (\mathbf{A} + \Delta \mathbf{A}) \mathbf{x}[k] + \mathbf{B} \mathbf{u}[k] + \xi[k]. \quad (8.1)$$

Bien que l'évolution du système soit maintenant décrite par une matrice dynamique $(\mathbf{A} + \Delta \mathbf{A})$, le contrôleur qui ignore cette nouvelle dynamique continue quant à lui de planifier les gains d'estimation et de contrôle en utilisant un modèle direct erroné $\hat{\mathbf{A}} = \mathbf{A}$.

C'est cette disparité entre le modèle interne au contrôleur et le modèle de l'environnement qui va rendre l'action du filtre de Kalman vraiment significative : puisque le modèle interne incomplet donne lieu à des erreurs de prédictions, l'utilisation adéquate de la mesure $\mathbf{y}$ permet de rectifier celles-ci jusqu'à un certain point.

Dans ce chapitre, nous commencerons par décider du modèle mathématique de la perturbation en nous basant sur les expériences déjà réalisées avant de simuler des mouvements d'atteinte dans lesquels cette perturbation est présente. Nous terminerons par une discussion des résultats.

8.1 Modèle de la perturbation

En nous inspirant des perturbations utilisées lors d'expériences sur les mouvements d'atteinte, nous choisissons spécifiquement une force induite par l'environnement sur le bras proportionnelle à la vitesse de ce dernier :

$$\underbrace{\begin{pmatrix} \Delta F_x \\ \Delta F_y \end{pmatrix}}_{\Delta \mathbf{F}} = \underbrace{\begin{pmatrix} G_{xx} & G_{xy} \\ G_{yx} & G_{yy} \end{pmatrix}}_{\mathbf{G}} \cdot \underbrace{\begin{pmatrix} \dot{x} \\ \dot{y} \end{pmatrix}}_{\mathbf{v}_x}. \quad (8.2)$$

Avec notre modèle bidimensionnel du bras, cette relation sera par conséquent représentée par une matrice

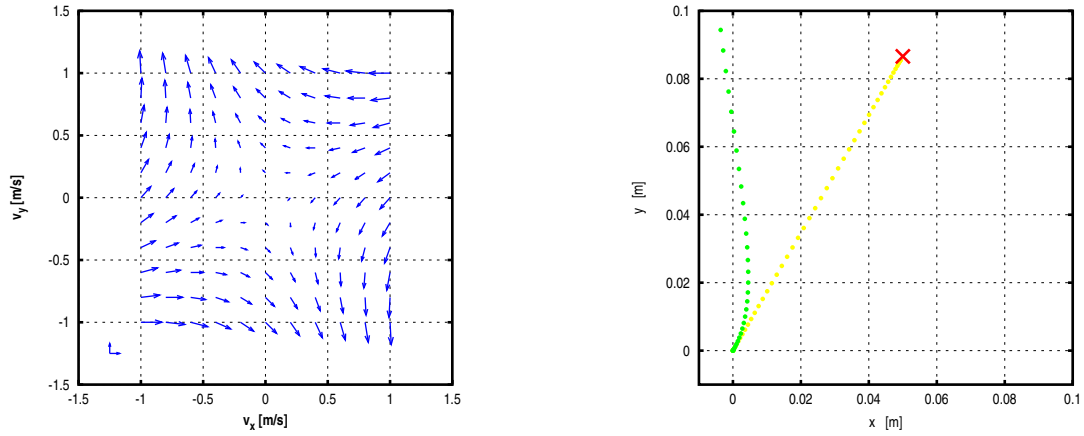


FIGURE 8.1 – Gauche : profil du champ de force utilisé par Shadmehr et Mussa-Ivaldi (1994). Les flèches de référence dans le bas du coin gauche représentent chacune une magnitude de 10 N . Droite : trajectoire (vert) sous l'influence d'un champ de force de magnitude vingt-cinq fois plus élevée. Les pointillés jaune indiquent la trajectoire sans bruits ni perturbation.

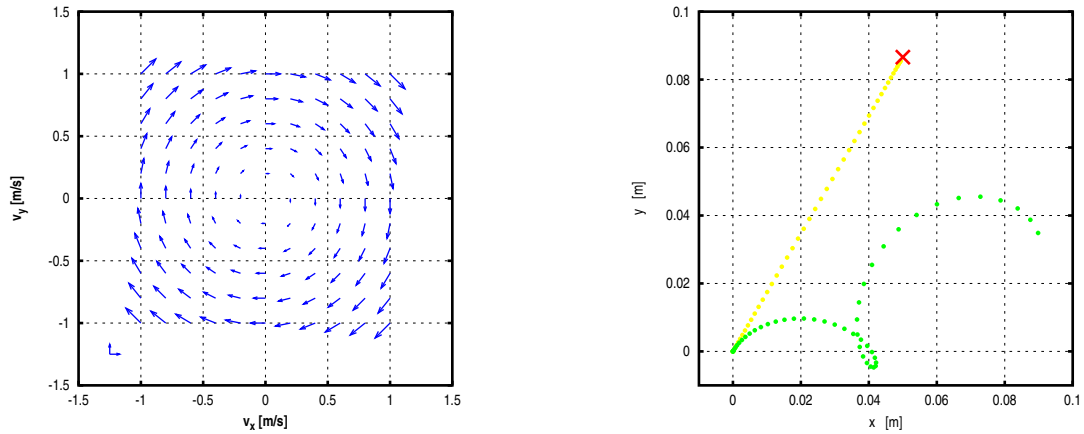


FIGURE 8.2 – Gauche : profil du champ de force utilisé par Wagner et Smith (2008). Les flèches de référence dans le bas du coin gauche représentent chacune une magnitude de 10 N . Droite : trajectoire (vert) sous l'influence d'un champ de force de magnitude quarante-cinq fois plus élevée. Les pointillés jaune indiquent la trajectoire sans bruits ni perturbation.

$\Delta \mathbf{A}$ de la forme

$$\Delta \mathbf{A} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & G_{xx} & G_{xy} & 0 & 0 \\ 0 & 0 & G_{yx} & G_{yy} & 0 & 0 \end{pmatrix}. \quad (8.3)$$

Pour leurs expériences, Shadmehr et Mussa-Ivaldi [Shadmehr, 1994] ont utilisé le champ de force représenté à la gauche de la figure 8.1 et provenant d’une matrice

$$\mathbf{G}_1 = \begin{pmatrix} -10.1 & -11.2 \\ -11.2 & 11.1 \end{pmatrix} N \cdot s/m. \quad (8.4)$$

Avec ce champ de force, les déplacements compris entre 45° et 225° sont amenés à converger vers le coin supérieur droit, tandis que ceux compris entre 225° et 45° sont poussés vers le coin inférieur gauche, comme le montre également la droite de la figure 8.1 où le champ de force a été multiplié par vingt-cinq afin de rendre ses effets prépondérants par rapport à ceux du contrôleur.

D’autres formes de champ de force sont également possibles. Wagner et Smith [Wagner, 2008] par exemple ont quant eux utilisé le champ de force représenté à la gauche de la figure 8.2, soit un champ de force provenant de la matrice

$$\mathbf{G}_2 = \begin{pmatrix} 0 & -15 \\ 15 & 0 \end{pmatrix} N \cdot s/m. \quad (8.5)$$

Le champ de force appliqué est alors un tourbillon : la force est toujours perpendiculaire à la direction de déplacement et va donc seulement changer cette direction de déplacement sans l’accélérer. Ceci est également illustré à la droite de la figure 8.2 où le champ de force a été multipliée par quarante-cinq.

8.2 Simulation

Voyons maintenant comment notre mouvement d’atteinte est modifié lorsqu’un champ de force est actif. Les trajectoires résultantes sont illustrés à gauche de la figure 8.3 pour le champ de force $\mathbf{G}_1$ utilisé par Shadmehr et Mussa-Ivaldi. La figure 8.4 présente quant à elle les graphiques de la position, la vitesse et la force dans l’axe x ainsi que du coût en fonction du temps.

Comme nous pouvons le remarquer, il n’y a étonnamment pas de différence significative avec les tracés obtenus au chapitre précédent en l’absence de perturbations¹. Et des essais avec le champ $\mathbf{G}_2$ conduisent au même constat...

Nous fûmes alors curieux de savoir si en augmentant la magnitude du champ de force, nous arriverions à observer une correction significative de la part du contrôleur. La droite de la figure 8.3 présente la trajectoire générée pour un champ de magnitude dix fois plus élevée. Le résultat montre que le contrôleur n’effectue pas de réelles corrections de trajectoire durant le mouvement mais se laisse tout simplement emporter par la perturbation.

Une idée en entraînant une autre, il nous vient à l’esprit que ce manque de réactivité pourrait être dû au coût trop important du contrôle! Nous avons alors relancé la simulation mais cette fois avec un poids de coût de contrôle nettement réduit ($w_u = 10^{-8}$). Les résultats, présentés dans la figure 8.5, confirment cette

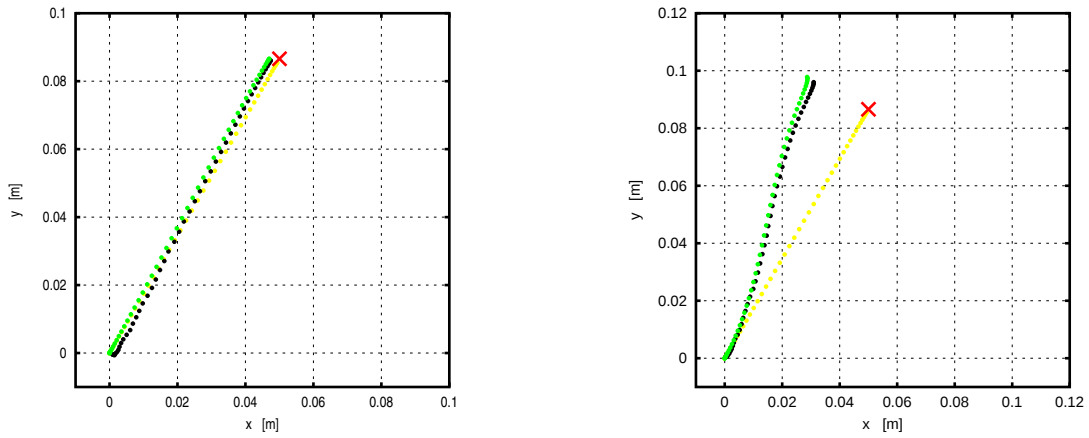


FIGURE 8.3 – Gauche : trajectoire (vert) lorsque la perturbation $\mathbf{G}_1$ est active. Droite : trajectoire (vert) pour une perturbation $10 \cdot \mathbf{G}_1$ démontrant l'absence d'une réelle action corrective de la part du contrôleur. Les pointillés noirs indiquent la position estimée, les pointillés jaunes la trajectoire sans bruits ni perturbation, la croix rouge la cible à atteindre.

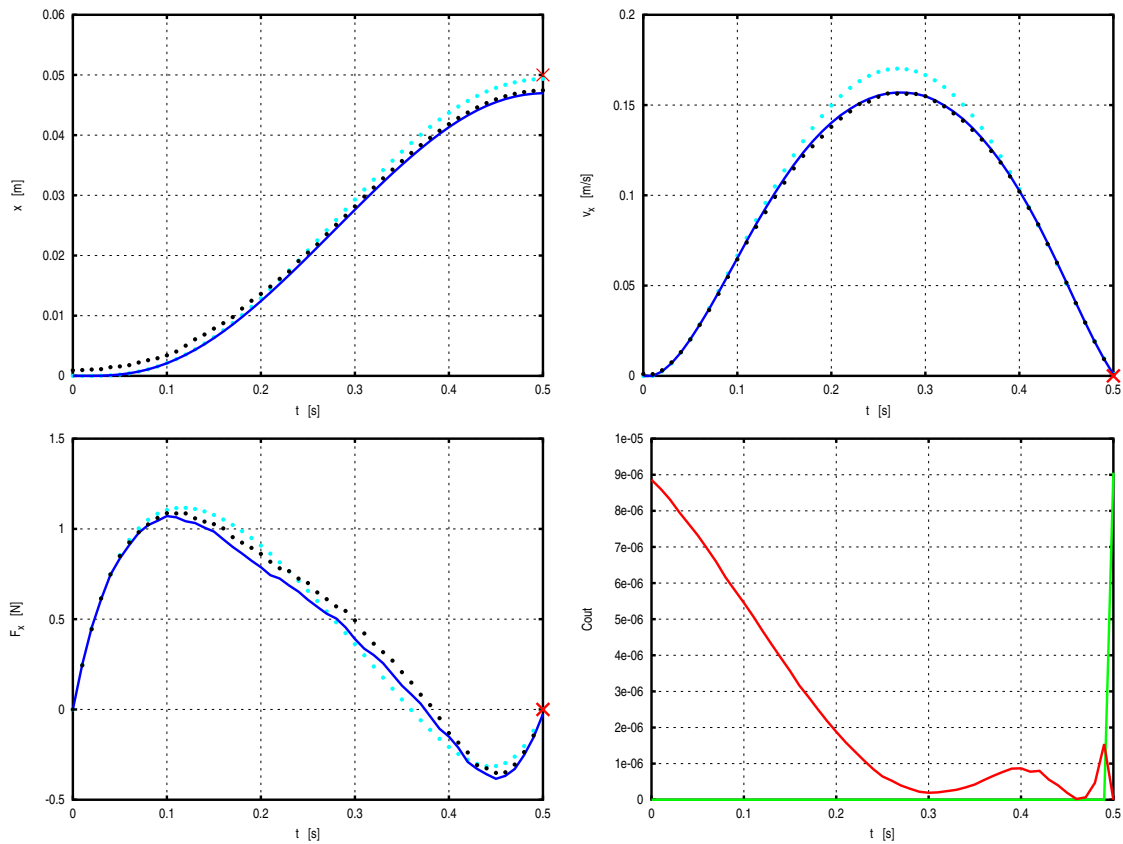


FIGURE 8.4 – Position (haut gauche), vitesse (haut droit) et force (bas gauche) dans l'axe x en fonction du temps lorsque la perturbation $\mathbf{G}_1$ est active. Les pointillés noirs indiquent l'état estimé, les pointillés cyan la trajectoire sans bruits ni perturbation, la croix rouge l'objectif à atteindre. Bas droit : coût dynamique (vert) et coût de contrôle (rouge) en fonction du temps.

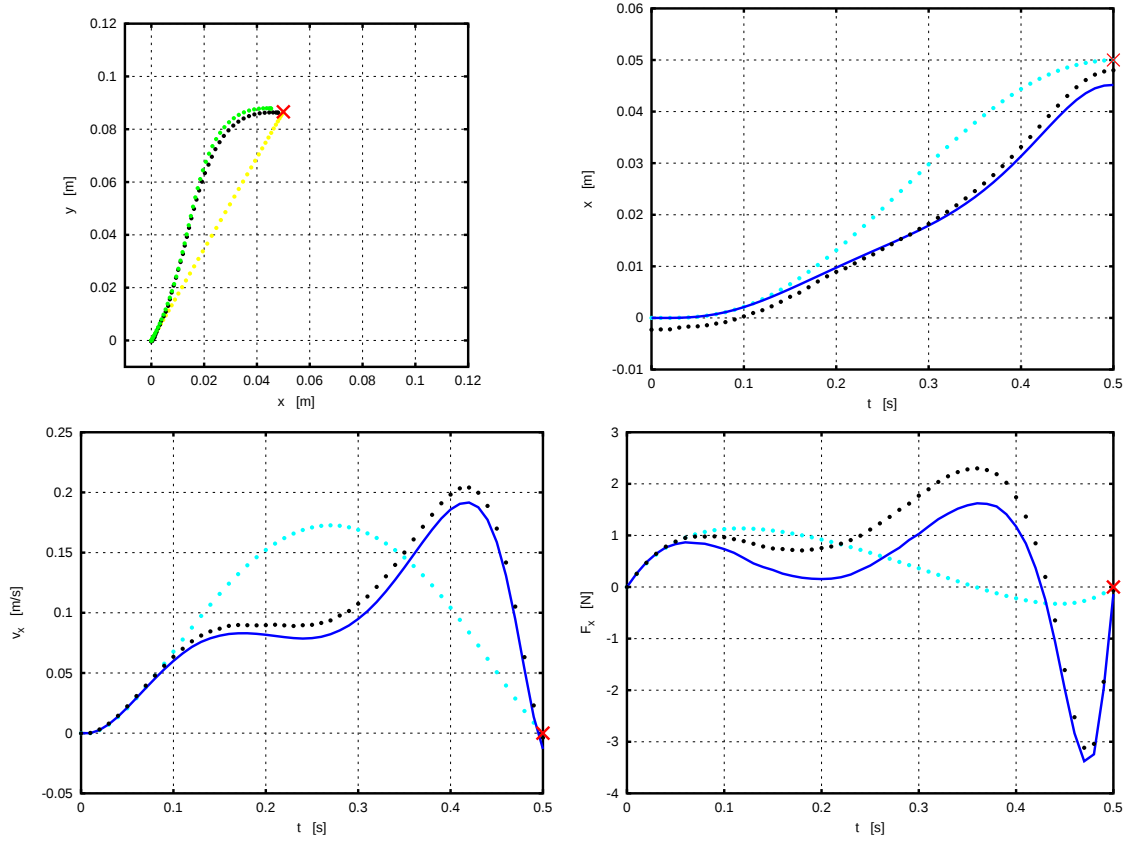


FIGURE 8.5 – Trajectoire (haut gauche) et position (haut droit) vitesse (bas gauche) et force (bas droit) dans l’axe x en fonction du temps lorsqu’une perturbation $10 \cdot \mathbf{G}_1$ est active avec un poids de coût de contrôle réduit ($w_u = 10^{-8}$), allouant au contrôleur la liberté de réaliser une action corrective.

intuition.

Il est donc bien possible d’observer un mouvement correctif du contrôleur mais à condition d’utiliser une perturbation beaucoup plus importante. Nous reviendrons sur ce point lors de notre discussion à la fin de ce chapitre.

8.3 Variation des paramètres

L’impact des paramètres sur la simulation a été abordé en profondeur dans le chapitre précédent, et les conclusions dégagées peuvent facilement être transposées au cas où une perturbation est active. Cependant, la présence d’un champ de force rend maintenant possible l’observation de deux nouveaux phénomènes.

Premièrement, la présence d’un champ de force va nous permettre de mettre évidence un phénomène ébauché à la gauche de la figure 8.6 appelé le *principe d’intervention minimum*. Lorsque, par exemple, les poids w_y , w_{v_y} et w_{F_y} du vecteur $\mathbf{w}$ sont mis à zéro, le contrôleur ne s’occupe plus de corriger les perturbations intervenant dans l’axe y mais seulement d’atteindre l’objectif par rapport à l’axe x .

1. Des résultats détaillés sont disponibles dans la table D.2 de l’annexe D.

Et deuxièmement, nous pouvons maintenant tester la performance du contrôleur face au bruit de mesure. Au-delà de $R_\omega = 10^{-3}$, le contrôleur devient imprécis. En effet, lorsque la variance du bruit de mesure est trop élevée, le filtre de Kalman privilégie nettement la prédiction à l'aide du modèle direct par rapport à celle donnée par les mesures. Puisque celui-ci est incorrect, le contrôleur pense atteindre la cible alors que la trajectoire réelle est complètement erronée. Le contrôle illustré à la droite de la figure 8.6 pour une valeur de $R_\omega = 10^{-1}$ est donc quasiment équivalent à un contrôle « à l'aveugle », c'est-à-dire sans mesure.

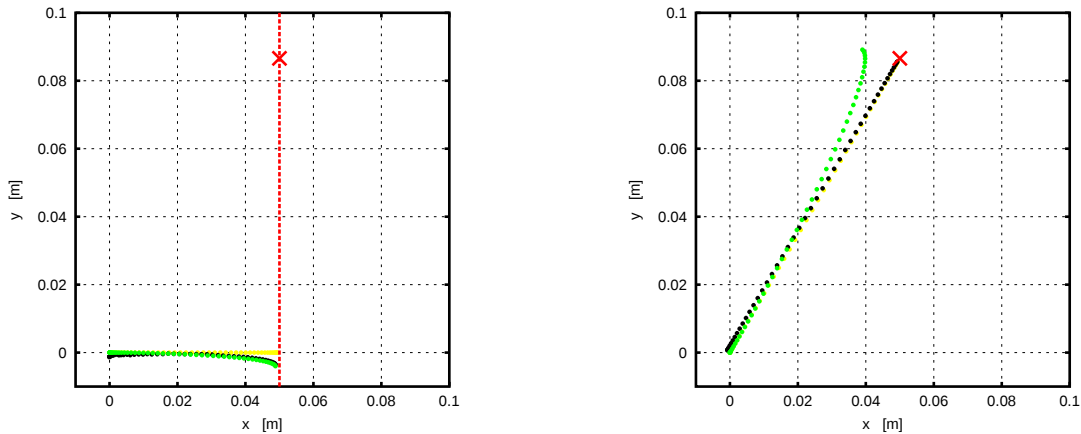


FIGURE 8.6 – Gauche : illustration du principe d'intervention minimum lorsque les poids associés à l'axe y sont nuls. Le contrôleur ne s'occupe plus de corriger la trajectoire dans l'axe y tandis que l'objectif dans l'axe x (rouge) est bien atteint. Droite : trajectoire lorsque le bruit de mesure est augmenté ($R_\omega = 10^{-1}$) donnant lieu à une estimation (noir) se basant presque uniquement sur le modèle direct.

8.4 Discussion

Bien que le contrôleur soit robuste face aux perturbations, à condition toutefois de diminuer le coût attaché au contrôle, et parvienne à atteindre la cible, les résultats obtenus dans ce chapitre sont plutôt déroutants. Contrairement aux résultats de Shadmehr et Mussa-Ivaldi, nous n'observons une vive correction donnant lieu à un profil caractéristique en crochet qu'à condition d'utiliser des champs de force nettement plus importants.

En outre, en retournant à la figure 7.9 du chapitre 7 afin de connaître le profil de vitesse en présence d'un champ de force, nous remarquons que le tracé obtenu à la figure 8.5 est comparativement assez discordant. Bien que la courbe obtenue commence à se rapprocher d'un profil de bimodal, le pic de vitesse est clairement observé à la fin du mouvement, tandis que chez les participants cette crête est mesurée au début du mouvement.

Cette non-concordance des résultats avec les expériences nous conduit directement à nous poser deux questions :

- ▷ Cette grande robustesse du contrôleur est-elle due à notre modèle du bras simplifié ne prenant en compte ni les non-linéarités du bras ni les effets de couplage importants ?
- ▷ Le contrôle optimal donnant un résultat assez robuste, le système nerveux utilise-t-il réellement une stratégie optimale lors de mouvements d'atteinte ? À l'instar des critiques concernant la théorie du choix optimal en économie², on peut en effet se demander si le cerveau possède les capacités cogni-

2. Voir par exemple [Ferrière, 2011].

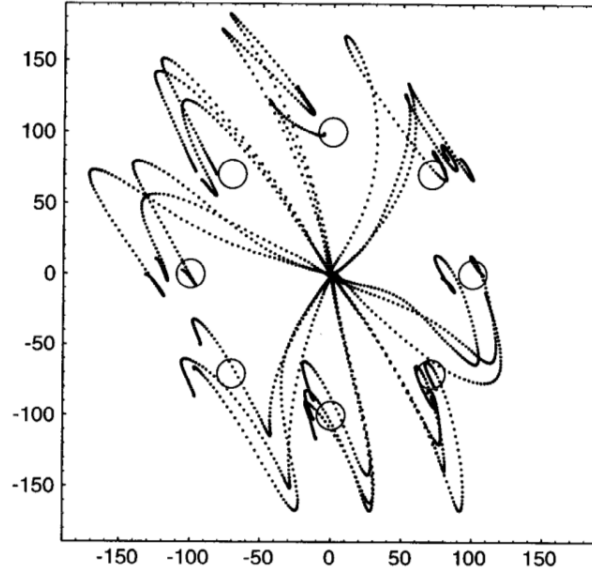


FIGURE 8.7 – Trajectoires typiques lors de mouvements d’atteinte en présence d’un champ de force. Les déplacements sont indiqués en *mm*. Source : [Shadmehr, 1994].

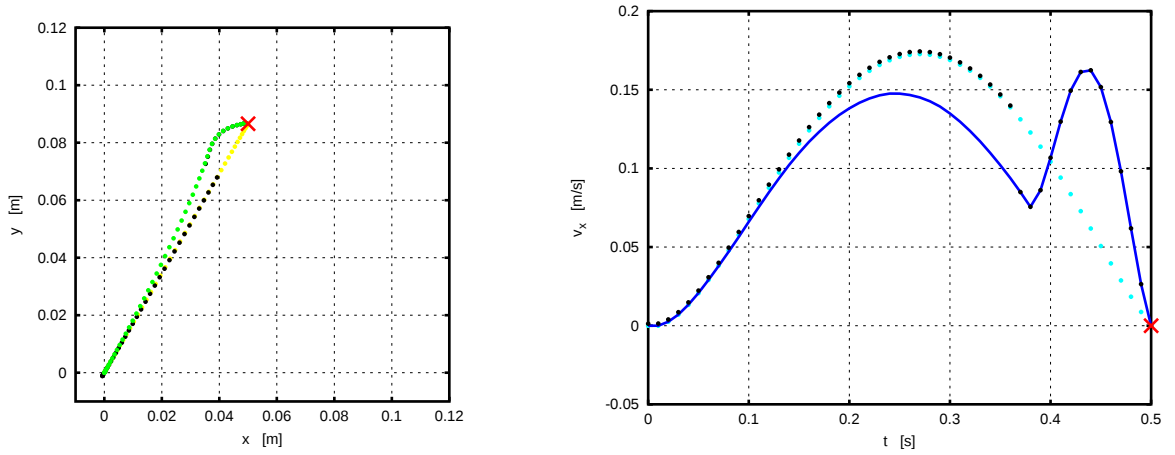


FIGURE 8.8 – Trajectoire (gauche) et vitesse dans l’axe x (droite) pour un contrôle aveugle ($R_\omega = 10^0$) durant les trois premiers quarts du mouvement suivi d’un contrôle à information complète sur l’état ($R_\omega = 10^{-8}$) pour la dernière partie du mouvement avec $w_u = 10^{-8}$. La transition entre les deux types de contrôle est marquée par un saut dans l’estimation (pointillés noirs).

tives, les informations, ou même un temps suffisant pour prendre la décision optimale ou si celui-ci se contente d'effectuer un contrôle « satisfaisant ».

Cette interpellation concernant le caractère optimal ou suffisant du contrôle peut également s'apprécier à la lumière de la droite de la figure 8.6 où le contrôle s'effectue à l'aveugle. En effet, en observant le tracé de la trajectoire engendrée en se basant uniquement sur le modèle direct, on imagine facilement la manière dont on pourrait faire apparaître artificiellement un crochet dans le mouvement : en déclenchant vers la fin du mouvement un contrôle prenant en compte les mesures de position, celui-ci effectuerait vraisemblablement un mouvement correctif en lacet vers la cible.

Et effectivement, en bricolant un peu la simulation il est possible d'obtenir un crochet dans la trajectoire. Nous obligeons le contrôleur à n'utiliser que le modèle direct pendant les trois premiers quarts du mouvement (contrôle aveugle). Par contre pendant le dernier quart, nous le forçons expressément à utiliser les mesures, entachées de très peu de bruit, plutôt que le modèle direct³ (contrôle avec information complète sur l'état). Les graphes résultants, pour un champ de force $\mathbf{G}_1$ et un poids $w_u = 10^{-8}$, sont présentés dans la figure 8.8.

Bien que réalisée par simple curiosité, cette dernière simulation nous donne des pistes de réflexions intéressantes. Nous pouvons par exemple nous demander si le système nerveux utilise un contrôle optimal sur toute la trajectoire ou l'active seulement à partir d'un certain moment. En effet, si celui dispose d'un modèle direct, il lui est possible de comparer les observations effectuées aux prédictions calculées. Tout à fait hypothétiquement, on pourrait imaginer que celui-ci ne déclenche un contrôle plus intense que lorsqu'un certain critère est rempli, par exemple à partir d'un certain seuil d'erreur sur la position.

D'autre part, nous n'avons pas encore introduit de délai dans notre modèle, délai qui sera justement l'objet du chapitre 10. La présence de délai va être particulièrement intéressante puisque celui-ci va induire naturellement un compromis entre contrôle aveugle et contrôle via estimation optimale puisque les mesures ne seront plus immédiatement accessibles au contrôleur.

Mais préalablement à l'introduction de délais, il serait peut être encore plus intéressant de modifier auparavant la fonction de coût utilisée jusqu'à présent pour des raisons que nous expliciterons dans le chapitre suivant.

Avant de refermer ce volet de notre mémoire, il nous faut encore mentionner l'existence d'une littérature dédiée au problème du contrôle robuste de systèmes dont le modèle dynamique présente des incertitudes. Dans ce mémoire, nous considérons qu'aucune information n'est disponible concernant la perturbation. Cependant certaines lois de contrôle optimales sont effectivement dérivables si des informations à propos de la structure de la perturbation sont disponibles. Deux approches typiques existent [Yedavalli, 2014] :

- ▷ Soit on considère que les éléments individuels de la matrice de perturbation $\Delta\mathbf{A}$ peuvent être bornés :

$$\max\{\Delta_{ij}A(t)\} \leq \epsilon_{ij}. \quad (8.6)$$

Dans ce cas, la perturbation est localisable au sein de la matrice $\Delta\mathbf{A}$ et la perturbation est dite *structurée*.

- ▷ Soit on considère que la localisation de la perturbation n'est pas clairement identifiable au sein de la matrice de perturbation et qu'on dispose seulement d'une borne sur la norme de la matrice. La perturbation est alors dite *déstructurée*.

3. En pratique, cet effet est réalisé en spécifiant une grande valeur pour la variance du bruit de mesure ($R_\omega = 10^0$) de $t = 0$ à $t = 0.375$ s, puis une valeur très petite ($R_\omega = 10^{-8}$) de $t = 0.375$ à $t = 0.5$ s.

Exploration de la fonction de coût

Dans ce chapitre, nous aimerions modifier la fonction de coût du contrôleur de manière à ce que celui-ci soit capable de choisir seul le temps de mouvement requis afin d'atteindre la cible.

En effet, comme nous l'avons remarqué dans le chapitre 7, notre contrôleur répartit son contrôle et par conséquent le mouvement, sur tout le temps fixé par la simulation. En plus de donner lieu à certains phénomènes incohérents, comme par exemple une prise d'élan dans la direction opposée à la cible en cas de raideur trop élevée ou une simulation beaucoup moins précise lorsque le pas de temps est diminué, ce comportement ne correspond pas à celui d'un être humain n'ayant aucune bonne raison de brider son mouvement. Il devrait donc exister, dans le cas d'une situation stable, un temps minimum endéans lequel le contrôleur est capable d'exécuter un mouvement correct : le fait de disposer de plus de temps que nécessaire ne devrait pas influencer les résultats donnés par la simulation.

Par ailleurs, nous avons pu constater au chapitre précédent que les dernières simulations avec champ de force sont peu concluantes, notamment au niveau du profil de vitesse. Il s'agit peut-être d'un signal nous indiquant qu'il est temps de changer d'approche.

Nous verrons donc dans un premier temps comment modifier la fonction de coût que nous utilisons pour que ce coût augmente avec le temps avant de simuler et discuter les résultats.

9.1 Fonction de coût croissante

En nous inspirant des modèles d'urgence ("urgency-gating models") utilisé dans la recherche cognitive en prise de décision ("decision making"), il nous vient à l'idée d'introduire explicitement une dépendance au temps dans la fonction de coût utilisée par le contrôleur. À la place d'utiliser un coût dynamique courant nul $\mathbf{Q}[k] = 0$ et uniquement un coût terminal $\mathbf{Q}_N$ non nul, nous allons utiliser un coût dynamique qui croît avec le temps.

L'idée d'utiliser un coût croissant atteignant progressivement la valeur finale $\mathbf{Q}_N$ peut par exemple se traduire par l'utilisation d'une fonction puissance

$$\mathbf{Q}[k] = \left(\frac{k-1}{N-1} \right)^a \cdot \mathbf{Q}_N \quad (9.1)$$

où le paramètre $a > 0$ détermine l'allure de la courbe.

En effet, si $k \in \{1, 2, \dots, N\}$, cette famille de fonction satisfait $\mathbf{Q}[1] = 0$ et $\mathbf{Q}[N] = \mathbf{Q}_N$ ainsi que l'illustre la figure 9.1.

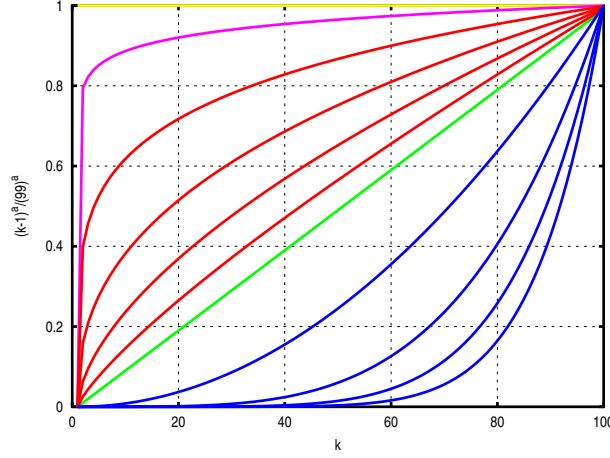


FIGURE 9.1 – Illustration de la fonction $\left(\frac{k-1}{N-1}\right)^a$ avec $N = 100$ pour différentes valeurs du paramètre a . Jaune : $a = 0$. Magenta : $a = 0.05$. Rouge : $a = \{0.2, 0.4, 0.6, 0.8\}$. Vert : $a = 1$. Bleu : $a = \{2, 4, 6, 8\}$.

Nous ne pouvons cependant pas utiliser cette fonction telle quelle. En effet, puisque la matrice $\mathbf{Q}_N$ est construite à partir du vecteur de poids

$$\mathbf{w} = \begin{pmatrix} w_x & w_y & w_{v_x} & w_{v_y} & w_{F_x} & w_{F_y} \end{pmatrix}^T = \begin{pmatrix} 1 & 1 & 0.1 & 0.1 & 0.01 & 0.01 \end{pmatrix}^T \quad (9.2)$$

et que l'objectif final est d'atteindre la cible avec une vitesse nulle $v^* = 0$ et une force nulle $F^* = 0$ dans les deux axes, utiliser la relation 9.1 reviendrait à demander à notre contrôleur d'observer une vitesse et une force nulle durant tout le trajet, rendant le mouvement impossible!

Par conséquent, la matrice $\mathbf{Q}_N$ de l'équation 9.1 est à remplacer par une matrice $\tilde{\mathbf{Q}}_N$ construite à partir du vecteur de poids

$$\mathbf{w} = \begin{pmatrix} w_x & w_y & w_{v_x} & w_{v_y} & w_{F_x} & w_{F_y} \end{pmatrix}^T = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 \end{pmatrix}^T. \quad (9.3)$$

En d'autres mots, le coût dynamique durant le trajet ne sera dû qu'aux erreurs de position, le contrôleur ne prenant en compte les contraintes de vitesse et de force qu'à la fin du mouvement.

9.2 Simulation

Nous allons maintenant tester les effets de ce nouveau coût dynamique sur notre simulation. Pour ce faire, nous gardons les paramètres habituels¹ mais changeons simplement le temps de la simulation. Nous spécifions dès lors $T = 0.75$ s, qui est plus que le temps nécessaire au mouvement afin d'observer le comportement du contrôleur. De plus, nous décidons d'utiliser pour la fonction de coût dynamique le paramètre $a = 1$, soit un coût dynamique courant qui croît linéairement avec le temps.

La trajectoire résultante, sans bruits ni perturbation, est illustrée dans la figure 9.2 tandis que les graphes de la position, la vitesse, la force dans l'axe x ainsi que le coût en fonction du temps sont présentés dans la

1. Voir le tableau 7.1 du chapitre 7.

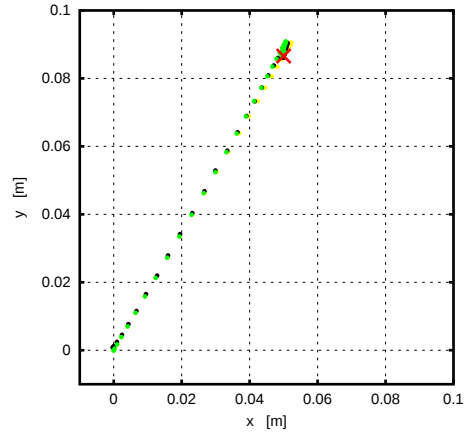


FIGURE 9.2 – Trajectoire (vert) de la main dans le plan xy . Les pointillés noirs indiquent la position estimée, les pointillés jaunes la trajectoire sans bruits ni perturbation, la croix rouge la cible à atteindre.

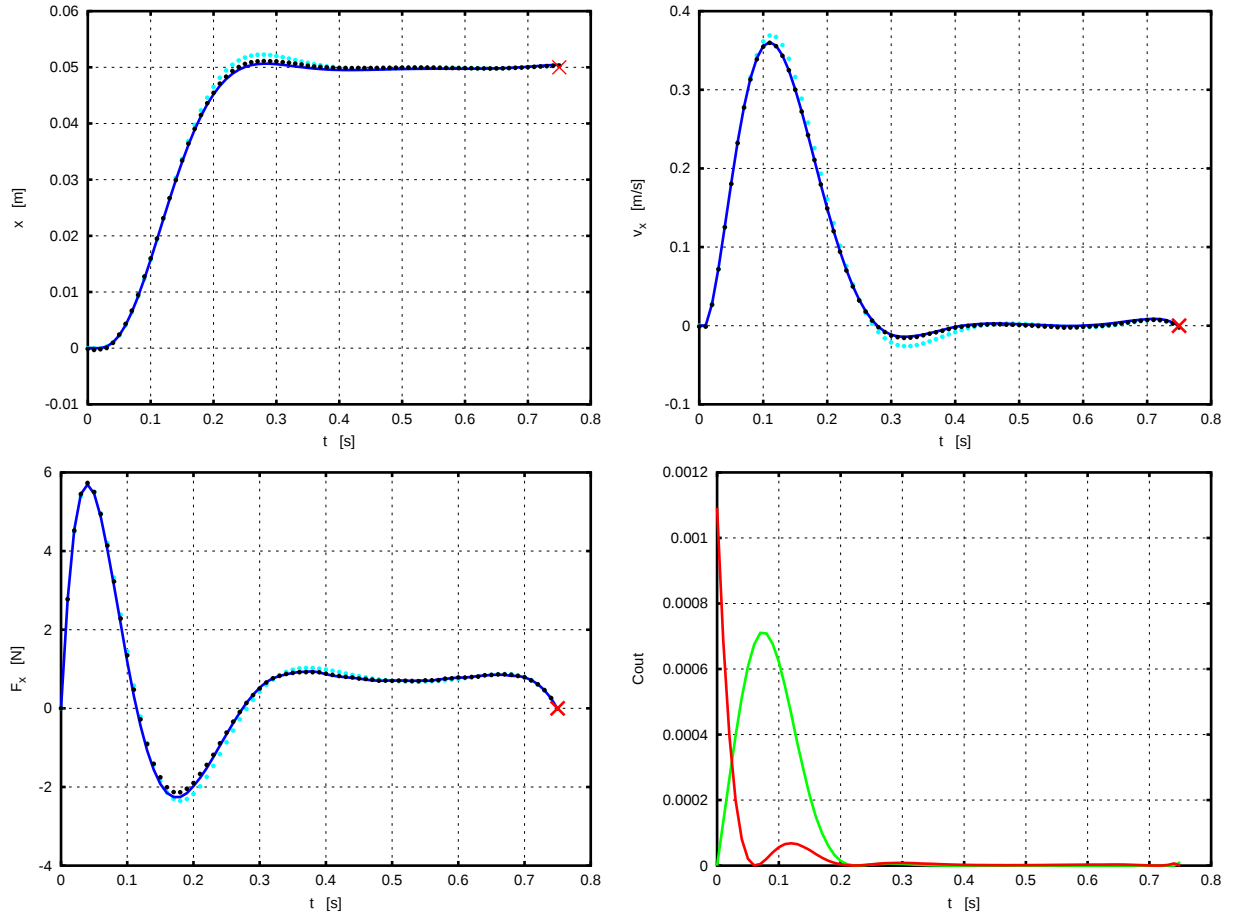


FIGURE 9.3 – Position (haut gauche), vitesse (haut droit) et force (bas gauche) dans l'axe x en fonction du temps. Les pointillés noirs indiquent l'état estimé, les pointillés cyan les profils sans bruits ni perturbation, la croix rouge l'objectif à atteindre. Bas droit : Coût dynamique (vert) et coût de contrôle (rouge) en fonction du temps.

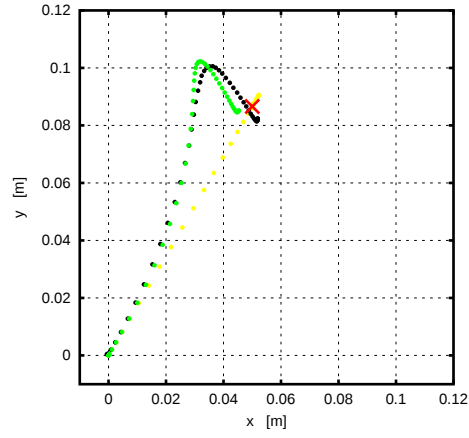


FIGURE 9.4 – Trajectoire (vert) de la main dans le plan xy en présence d'un champ de force $\mathbf{G}_1$ de magnitude quinze fois plus élevée.

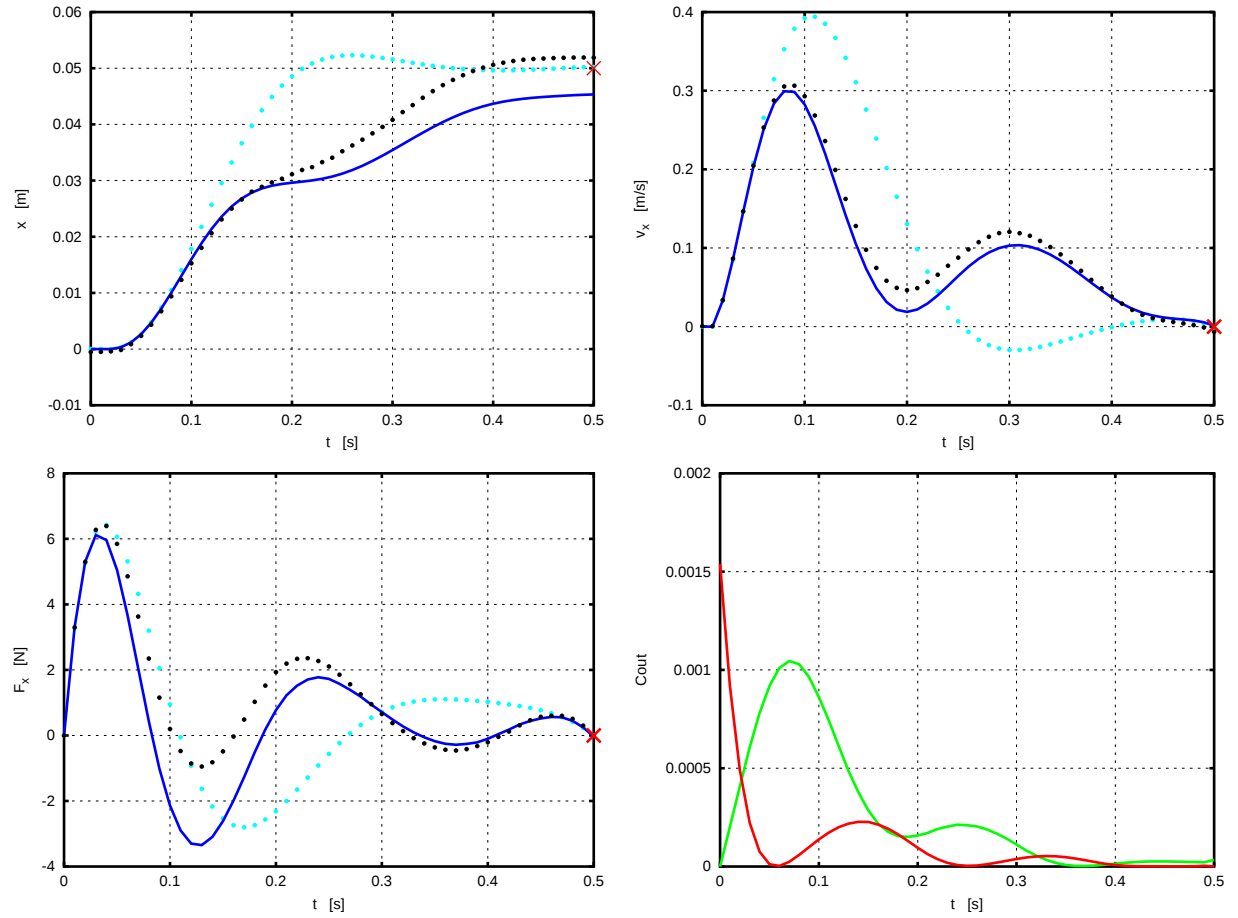


FIGURE 9.5 – Position (haut gauche), vitesse (haut droit) et force (bas gauche) dans l'axe x en fonction du temps en présence d'un champ de force $\mathbf{G}_1$ de magnitude quinze fois plus élevée. Bas droit : Coût dynamique (vert) et coût de contrôle (rouge) en fonction du temps.

figure 9.3.

On remarque d'emblée que le contrôle s'effectue en moins de 400 *ms*, ce qui assez prometteur ! En effet, contrairement aux simulations précédentes où les pointillés étaient plus denses au début et à la fin du mouvement, ici ils ne sont concentrés qu'à la fin du mouvement. Cela paraît plus logique, puisqu'il n'y a pas de raison pour que la main s'attarde à la position de départ : à la place, on voit bien que la main démarre ici directement et attend ensuite à la position d'arrivée plutôt que de répartir le mouvement sur tout le temps disponible.

Le profil de la position dans l'axe x rappelle quant à lui celui d'un système du second ordre, c'est-à-dire d'une réponse exponentielle avec un léger dépassement. Et Le profil de vitesse correspond de nouveau bien à la cloche caractéristique.

En contrepartie, on remarque cependant que le coût dynamique, aussi bien que le coût de contrôle ont augmenté d'un ordre de grandeur 10^3 . Le profil de ces coûts est également bien différent : le coût dynamique prend maintenant une allure bombée, le coût final n'étant plus prépondérant. De plus, les deux coûts semblent surtout importants durant les premières 200 *ms* du mouvement, comme si après ce délai la main n'avait qu'à se laisser guider par sa dynamique.

En présence d'une perturbation, les profils sont de nouveau extrêmement proches de ceux sans champ de force²

Nous avons donc encore une fois décidé de tester la robustesse du contrôleur à un champ de force supérieur à celui utilisé par Shadmehr et Mussa-Ivaldi. Celui-ci, plus performant que le précédent est capable d'effectuer le mouvement d'atteinte pour des champs de magnitude jusqu'à quinze fois plus élevé, et ce sans modification du coût de contrôle ($w_u = 10^{-6}$) ainsi que le montrent les figures 9.4 et 9.5.

Le profil de vitesse obtenu est particulièrement intéressant : celui-ci est bien bimodal et exhibe un pic de vitesse au début de la trajectoire, ce qui se rapproche nettement des profils mesurés lors des expériences.

9.3 Variation des paramètres

Parmi les effets notables de la variation des paramètres sur la simulation en l'absence de perturbation certains méritent d'être mentionnés :

- ▷ Lorsque k_e est suffisamment élevé, le profil de vitesse est nettement impacté, ainsi que l'illustre la gauche de la figure 9.6. Cependant, le phénomène de prise d'élan a maintenant bien disparu.
- ▷ Le fait de diminuer le pas de temps Δt n'a maintenant plus d'influence néfaste sur la précision de la simulation, ce qui était exactement un des problèmes qui nous avaient motivé à changer d'approche au début de ce chapitre.
- ▷ Lorsque le terme de couplage λ est activé, la tactique du contrôleur n'est plus de démarrer en mouvement direct vers la cible mais plutôt de démarrer dans une direction préférentielle et seulement en fin de mouvement de bifurquer vers la cible. Cette stratégie permet peut-être au contrôleur de minimiser les coûts de contrôle dus au couplage, mais ceci ne correspond pas au comportement humain lors d'expériences. La droite de la figure 9.6 illustre une telle trajectoire pour une valeur de $\lambda = 0.75$ où la trajectoire préférentielle semble être de 45° . Notons que cette direction préférentielle varie en fonction de λ et en fonction de l'angle de la cible φ .
- ▷ Enfin, signalons l'utilisation d'autres valeurs de a ne modifie pas sensiblement les résultats.

2. Des résultats détaillés sont disponible dans le tableau D.3 de l'annexe D.

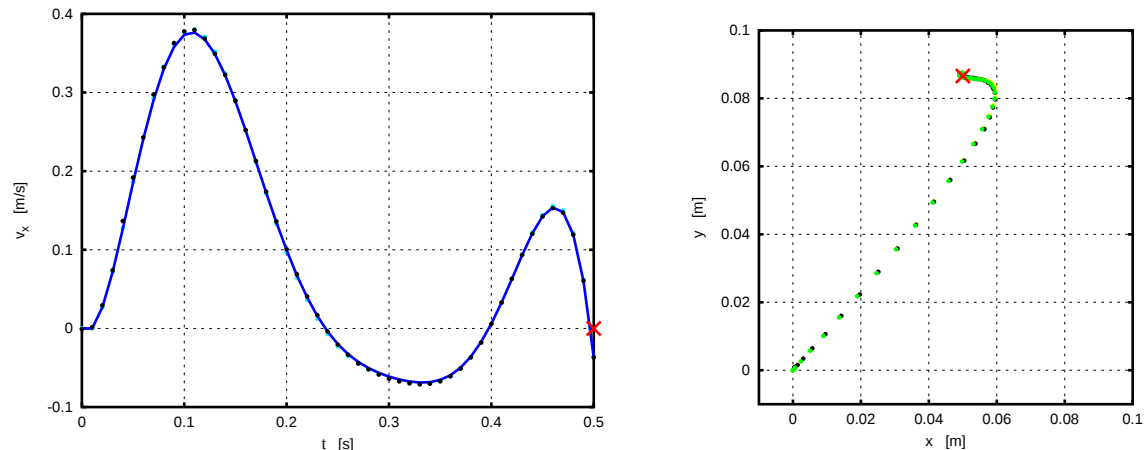


FIGURE 9.6 – Gauche : profil de vitesse lorsque k_e est multiplié par 20 donnant lieu à un profil de vitesse bimodal. Droite : trajectoire lorsque $\lambda = 0.75$ montrant une direction de mouvement différente de l’orientation de la cible.

9.4 Discussion

Cette précision et cette robustesse accrues pour un temps de mouvement moindre semblent indiquer que la modification opérée au niveau de la fonction de coût a rendu le contrôleur bien plus performant !

Pour s’en convaincre, il est possible d’effectuer la même étude de robustesse vis-à-vis des paramètres, sans perturbation et en spécifiant $T = 0.5$ s pour reproduire les mêmes conditions que celles étudiées dans le chapitre 7. Les résultats de la table E.2 de l’annexe E sont à comparer avec ceux de la table E.1 : ceux-ci démontrent la supériorité de ce nouveau contrôleur sur le précédent.

Celui-ci tolère notamment que la raideur soit multipliée par 20 et la viscosité par 70, et fonctionne également lorsque ces termes sont nuls, rendant la simulation assez insensible à ces paramètres. Les approximations effectuées au chapitre 5 pour trouver les valeurs viscoélastiques sembleraient donc n’avoir pas d’impact significatif sur l’allure de la simulation.

En outre il suffit, avec cette fonction de coût modifiée, de 200 ms au contrôleur pour effectuer un mouvement précis.

Enfin, nous avons vu qu’il était possible d’obtenir un profil correspondant aux expériences de Shadmehr et Mussa-Ivaldi mais à la condition d’utiliser une fois encore un champ de force nettement plus élevé. Nous sommes de nouveau renvoyés à nos interrogations du chapitre précédent : est-ce simplement dû au fait que notre modèle ne tient pas compte des non-linéarités et des couplages du bras ? Ou bien s’agit-il plutôt de notre contrôleur qui serait « trop robuste » ?

Dans le prochain chapitre, nous introduirons enfin du délai dans la simulation.

Le système sensorimoteur humain présente un délai d'environ 50 *ms* [Crevecœur, 2014]. Concrètement, cela signifie que ce délai sépare le moment où une observation est effectuée et l'instant où la commande motrice est envoyée aux muscles.

Jusqu'à présent, nous avons considéré que la mesure $\mathbf{y}$ effectuée par notre contrôleur sur le système lui était directement disponible. Il pouvait par conséquent se baser sur celle-ci afin d'adapter son contrôle suivant les formules données par la théorie du contrôle optimal développée dans le chapitre 3.

Ce ne sera plus le cas si du délai est présent dans le système, ce qui va avoir un impact direct sur la qualité du contrôle effectué. En effet, le contrôleur devra attendre que l'information donnée par la mesure lui parvienne avant de pouvoir l'utiliser. En attendant celle-ci, il va être obligé de prédire l'évolution du système. Sauf qu'en présence de perturbation, nous savons que les prédictions effectuées par celui-ci sont fausses, et en l'absence de mesures directement disponibles il est difficile pour le contrôleur de rectifier les erreurs commises.

Cette problématique va en général entraîner l'instabilité du système. Est-il possible de trouver une loi de contrôle permettant, même en présence de délai, de parvenir à un contrôle stable et correct ?

Il s'agit de la question que nous adresserons dans ce chapitre. Partant d'une extension simple de notre contrôleur au cas avec délai, nous verrons que les délais entraînent un comportement erroné du contrôleur. Nous chercherons par conséquent ensuite une loi de contrôle empirique permettant de réussir le contrôle avant de discuter les résultats obtenus.

10.1 Contrôle avec délai de mesure

Si un délai de mesure $t_d = d \cdot \Delta t$ est présent dans le système, l'observation $\mathbf{y}$ effectuée à l'instant discret $[k]$ ne sera disponible au contrôleur qu'à l'instant discret $[k + d]$. Autrement dit, le contrôle optimal à un temps $[k]$ ne peut se baser que sur l'information concernant l'état $[k - d]$, soit

$$\mathbf{u}[k] = -\mathbf{L}[k] \cdot \hat{\mathbf{x}}[k \mid k - d]. \quad (10.1)$$

Le contrôleur utilisera par conséquent l'estimateur optimal $\hat{\mathbf{x}}$ du temps $[k]$ connaissant seulement la mesure $\mathbf{y}[k - d]$. Mais que vaut cet estimateur optimal ?

Connaissant le modèle interne du système ($\mathbf{A}$ et $\mathbf{B}$), une approche heuristique permettant de prédire l'état $[k]$ du système en n'ayant d'informations que sur l'état $[k - d]$ consiste à appliquer le modèle interne d

fois. En effet en l'absence de mesure disponible l'estimateur optimal au temps discret suivant est donné par

$$\hat{\mathbf{x}}[k+1 | k] = \mathbf{A}\hat{\mathbf{x}}[k] + \mathbf{B}\mathbf{u}[k] + \mathbf{E}\{\xi[k]\} \quad (10.2)$$

$$= \mathbf{A}\hat{\mathbf{x}}[k] + \mathbf{B}\mathbf{u}[k] \quad (10.3)$$

puisque par hypothèse le bruit de traitement a une moyenne nulle. Remarquons qu'il s'agit simplement de la formule du filtre de Kalman ou la matrice d'observation $\mathbf{H}$, et par conséquent le gain $\mathbf{K}$ serait nuls.

Exemple 10.1. Imaginons un système avec un pas de discrétisation de $\Delta t = 10 \text{ ms}$ présentant un délai de $t_d = 30 \text{ ms}$, soit $d = 3$. Par conséquent la mesure $\mathbf{y}[1]$ effectuée en $t = 0 \text{ ms}$ ne sera disponible qu'à la troisième itération, soit l'état discret $k = 4$ correspondant à $t = 30 \text{ ms}$.

En attendant cette donnée, nous nous contentons d'effectuer des prévisions pour $\mathbf{x}[2]$ et $\mathbf{x}[3]$ à partir de la position initiale estimée $\hat{\mathbf{x}}[1]$:

$$\hat{\mathbf{x}}_1[2] = \mathbf{A}\hat{\mathbf{x}}[1] + \mathbf{B}\mathbf{u}[1], \quad (10.4)$$

$$\hat{\mathbf{x}}_1[3] = \mathbf{A}\hat{\mathbf{x}}_1[2] + \mathbf{B}\mathbf{u}[2] \quad (10.5)$$

$$= \mathbf{A} \cdot (\mathbf{A}\hat{\mathbf{x}}[1] + \mathbf{B}\mathbf{u}[1]) + \mathbf{B}\mathbf{u}[2] \quad (10.6)$$

où l'indice z de $\hat{\mathbf{x}}_z$ est là pour rappeler que l'estimation est effectuée avec la seule connaissance de $\hat{\mathbf{x}}[z]$ et est par conséquent seulement temporairement optimale.

Au temps $k = 4$, l'information $\mathbf{y}[1]$ est enfin disponible! Nous pouvons désormais utiliser le filtre de kalman afin de calculer une nouvelle estimation, cette fois optimale, de $\mathbf{x}[2]$:

$$\hat{\mathbf{x}}[2] = \mathbf{A}\hat{\mathbf{x}}[1] + \mathbf{B}\mathbf{u}[1] + \mathbf{K}(\mathbf{y}[1] - \mathbf{H}\hat{\mathbf{x}}[1]). \quad (10.7)$$

Il suffit ensuite de corriger notre ancienne estimation de $\mathbf{x}[3]$ en propageant l'information $\hat{\mathbf{x}}[2]$ et de prédire $\mathbf{x}[4]$ à partir de la prédiction corrigée de $\mathbf{x}[3]$, soit

$$\hat{\mathbf{x}}_2[3] = \mathbf{A}\hat{\mathbf{x}}[2] + \mathbf{B}\mathbf{u}[2], \quad (10.8)$$

$$\hat{\mathbf{x}}_2[4] = \mathbf{A}\hat{\mathbf{x}}_2[3] + \mathbf{B}\mathbf{u}[3]. \quad (10.9)$$

À l'instant $k = 5$, l'observation $\mathbf{y}[2]$ sera disponible. Il sera alors possible calculer $\hat{\mathbf{x}}[3]$ via le filtre de Kalman, de mettre à jour $\hat{\mathbf{x}}_z[4]$, de prédire $\hat{\mathbf{x}}_z[5]$ et ainsi de suite.

En généralisant maintenant à un délai d , voici l'algorithme à suivre par le contrôleur :

- ▷ Si $k \leq d$ utiliser le modèle direct afin de prédire l'état k à partir de l'estimation initiale $\hat{\mathbf{x}}[1]$.
- ▷ Si $k > d$, mettre à jour l'estimation $\hat{\mathbf{x}}[k-d+1]$ avec le filtre de Kilman, et utiliser le modèle direct afin de prédire l'état k à partir de la nouvelle estimation $\hat{\mathbf{x}}[k-d+1]$.

Notons par conséquent que le contrôleur n'utilisera jamais une estimation optimale lors du contrôle, seulement une estimation temporairement optimale. De fait, au temps k (si $k > d$), le contrôleur utilisera l'estimation basée sur la mesure $k-d$ c'est-à-dire sur l'estimation optimale de l'état $k-d+1$, soit la loi de contrôle :

$$\mathbf{u}[k] = -\mathbf{L}[k] \cdot \hat{\mathbf{x}}_{k-d+1}[k] \quad (10.10)$$

$$= -\mathbf{L}[k] \cdot \mathbf{p}[k]. \quad (10.11)$$

Notations

- ▷ $\hat{\mathbf{x}}_z[k]$ avec l'indice z est l'estimation temporairement optimale de $\mathbf{x}[k]$ effectuée en se basant sur l'estimation au temps z via le modèle direct car l'information $\mathbf{y}[k]$ n'est pas encore arrivée.
- ▷ $\hat{\mathbf{x}}[k]$ sans indice z est l'estimation définitivement optimale (a posteriori) de $\mathbf{x}[k]$ basée sur la mesure $\mathbf{y}[k]$ via le filtre de Kalman.
- ▷ $\mathbf{p}[k]$ est l'estimation de $\mathbf{x}[k]$ effectivement utilisée par le contrôleur au temps k (a priori), soit $\mathbf{p}[k] = \hat{\mathbf{x}}_{k-d+1}[k]$.

10.2 Simulation

Nous effectuons maintenant une simulation avec le champ de force utilisé par Shadmehr et Mussa-Ivaldi, en utilisant la fonction de coût dynamique croissante développée dans le chapitre précédent et avec un délai de 50 *ms* dans le système. Comme on le constate dans la figure 10.1, les résultats ne sont pas très satisfaisants : la main en bout de course au lieu de se diriger vers la cible, part complètement dans la direction opposée !

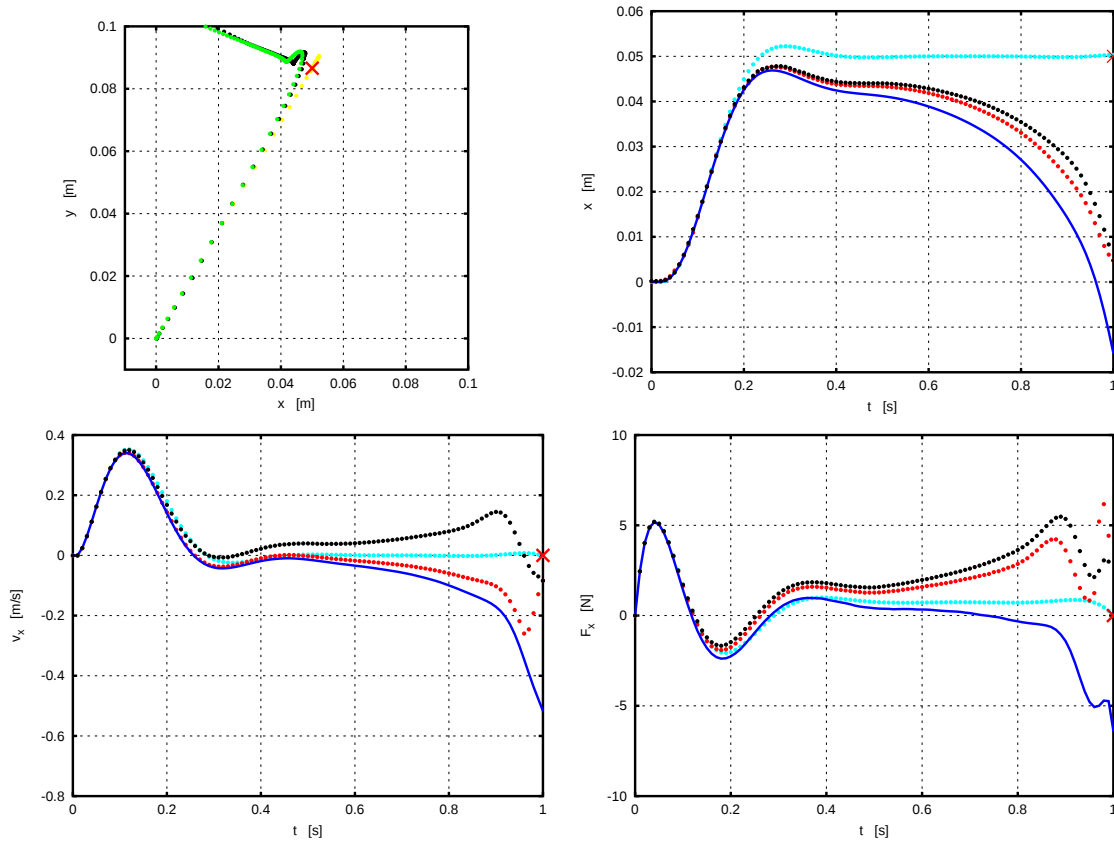


FIGURE 10.1 – Trajectoire (haut gauche) et position (haut droit), vitesse (bas gauche) et force (bas droit) dans l'axe x en présence d'un délai de 50 *ms*. Les pointillés noirs indiquent l'estimation utilisé par le contrôleur au moment du contrôle tandis que les pointillés rouges indiquent l'estimation mise à jour a posteriori à l'aide des informations reçues. Les pointillés jaunes ou cyans indiquent le profil sans bruit, ni perturbation, ni délai et la croix rouge l'objectif à atteindre.

En investiguant les estimations utilisées lors du contrôle (pointillés noirs), on se rend compte que celles-ci sont complètement erronées en ce qui concerne la vitesse et la force. Pire encore, même les estimations mises à jour a posteriori (pointillés rouges) sont fausses pour le profil de force. Utiliser des prédictions provenant d'un modèle direct n'est peut-être finalement pas une si bonne stratégie que ça...

En effet, si l'estimation du profil de force est si mauvaise, c'est qu'il s'agit justement de la variable d'état sur laquelle va agir directement le champ de force. Le modèle direct ne peut prévoir ce champ de force et se révèle par conséquent, en l'absence de mesures correctrices disponibles, assez médiocre.

Avant de pousser la réflexion plus loin, notons tout de même que le contrôleur parvient cependant à effectuer le mouvement lorsque les délais sont inférieurs ou égaux à 30 ms. Dans ce cas, le profil est similaire à celui sans délai. Lorsque la perturbation n'est pas présente, le mouvement est également correct, puisque le modèle utilisé est alors fiable. Mais le contrôleur parvient aussi à effectuer le mouvement pour un délai de 50 ms à condition de durcir légèrement le poids du coût de contrôle ($w_u = 10^{-5}$). Les profils sont de nouveaux très similaires à ceux sans délais. Cependant, une telle astuce ne fait que repousser le vrai problème sans lui donner de solution.

Et que se passerait-il si nous contrôlions le système avec une connaissance complète de l'état (mesure parfaite) arrivant simplement avec du retard¹? Comme nous le constatons sur la figure 10.2, le contrôle est tout simplement instable...

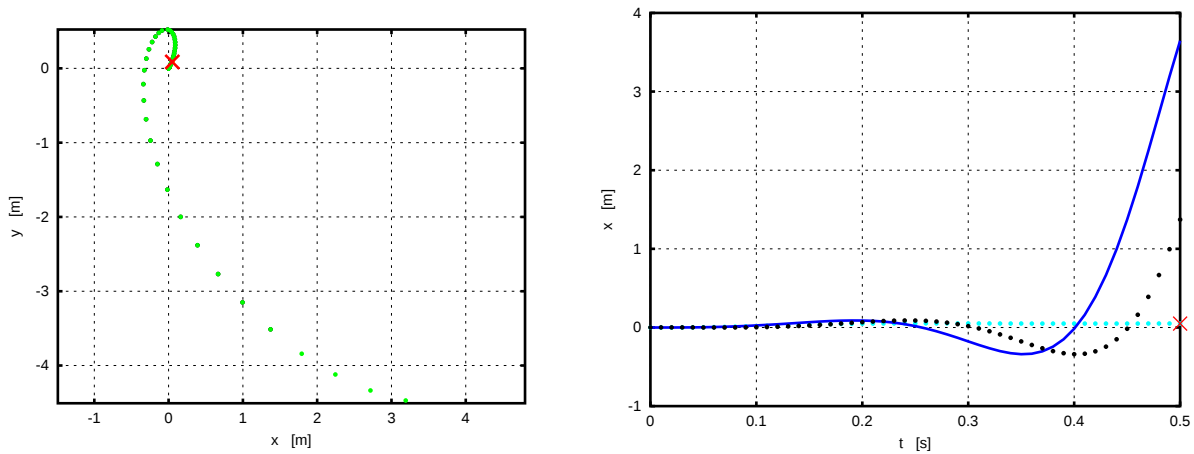


FIGURE 10.2 – Trajectoire (gauche) et position dans l'axe x (droite) pour un contrôle avec information complète arrivant avec un retard de 50 ms.

...il va nous falloir trouver une méthode afin de stabiliser le système!

10.3 Contrôle avec feedback sur les erreurs d'estimation

Après avoir longuement tâtonné en essayant sans succès diverses astuces dans l'espoir de parvenir à un résultat stable, une idée nous est venue à force d'observer la figure 10.1.

Prenons par exemple le profil de vitesse v_x : on voit bien qu'il existe un grand écart entre la prédiction a priori utilisée par le contrôleur au moment du contrôle (pointillés noirs) et celle donnée a posteriori par

1. Jusque $k < d + 1$, le contrôleur utilise comme loi de contrôle $\mathbf{u}[k] = -\mathbf{L}[1] \cdot \mathbf{x}[1]$, ensuite $\mathbf{u}[k] = -\mathbf{L}[k - d] \cdot \mathbf{x}[k - d]$.

la prédiction optimale (pointillés rouges). Ne pourrait-on utiliser cet écart entre la prédiction a priori et la prédiction a posteriori comme une mesure de l'erreur commise et d'utiliser cette mesure pour rectifier notre contrôle ultérieur ?

En effet, on peut voir cet écart comme une erreur statique de prédiction due au modèle direct qui ne connaît pas la dynamique de l'environnement. Par conséquent une première idée intuitive consisterait à simplement retrancher cette erreur de la prédiction future.

Exemple 10.2. Illustrons cette idée en reprenant notre exemple initial où nous l'avons laissé. Pour rappel, nous considérons $d = 3$.

Au temps $k = 5$, la mesure $\mathbf{y}[2]$ est disponible. Nous calculons par conséquent l'estimation optimale $\hat{\mathbf{x}}[3]$ par le filtre de Kalman et la propageons via le modèle direct pour obtenir une estimation $\hat{\mathbf{x}}_3[5]$ de $\mathbf{x}[5]$. Jusque là, rien de nouveau : suivant la stratégie initiale, nous utiliserions simplement $\hat{\mathbf{x}}_3[5]$ pour notre le contrôle.

Cependant, lorsque $\hat{\mathbf{x}}[3]$ est disponible il est également possible de calculer une autre information. En effet, au temps $k = 3$, le contrôleur avait utilisé la prédiction $\mathbf{p}[3] = \hat{\mathbf{x}}_1[3]$ pour son contrôle. Il est alors possible de calculer l'erreur d'estimation commise antérieurement comme étant l'écart

$$\epsilon[3] = \hat{\mathbf{x}}[3] - \mathbf{p}[3], \quad (10.12)$$

soit la différence entre l'estimation a priori et l'estimation a posteriori.

Et nous aimerions maintenant utiliser cet écart afin de corriger la prédiction $\hat{\mathbf{x}}_3[5]$. Un premier essai naïf nous pousse à simplement retrancher cette erreur de la prédiction, soit la loi de contrôle

$$\mathbf{u}[5] = -\mathbf{L}[5] \cdot (\hat{\mathbf{x}}_3[5] - \epsilon[3]). \quad (10.13)$$

En généralisant cet exemple, nous trouvons l'algorithme suivant :

- ▷ Si $k \leq d$ utiliser le modèle direct afin de prédire l'état k à partir de l'estimation initiale $\hat{\mathbf{x}}[1]$.
- ▷ Si $k > d$:
 - a) Mettre à jour l'estimation $\hat{\mathbf{x}}[k - d + 1]$ avec le filtre de Kilman.
 - b) Utiliser le modèle direct afin de prédire l'état k à partir de la nouvelle estimation $\hat{\mathbf{x}}[k - d + 1]$.
 - c) Calculer l'erreur de prédiction qui avait été commise au temps $k - d$.
 - d) Retrancher l'erreur de prédiction de l'estimation de l'état k donnée par le modèle direct avant de l'utiliser dans le contrôleur.

Après simulation, nous nous sommes rendus compte que ce contrôleur avec rétroaction sur les prédictions passées fonctionnait parfaitement pour un délai de 50 ms , et mieux encore, celui-ci est même capable de supporter des délais de 100 ms ! Enfin, comme test final, nous voulions savoir si nous étions capable de reproduire le crochet très net obtenu au chapitre précédent : la réponse est oui, mais il faut alors de nouveau utilisé un champ de force nettement plus important (multiplié par 10) comme le montre la figure 10.3.

Notons que le pas de temps de cette dernière simulation a été divisé par deux ($\Delta t = 5 \text{ ms}$) et le délai adapté en conséquence ($d = 10$) afin de permettre d'observer plus nettement le comportement de notre loi

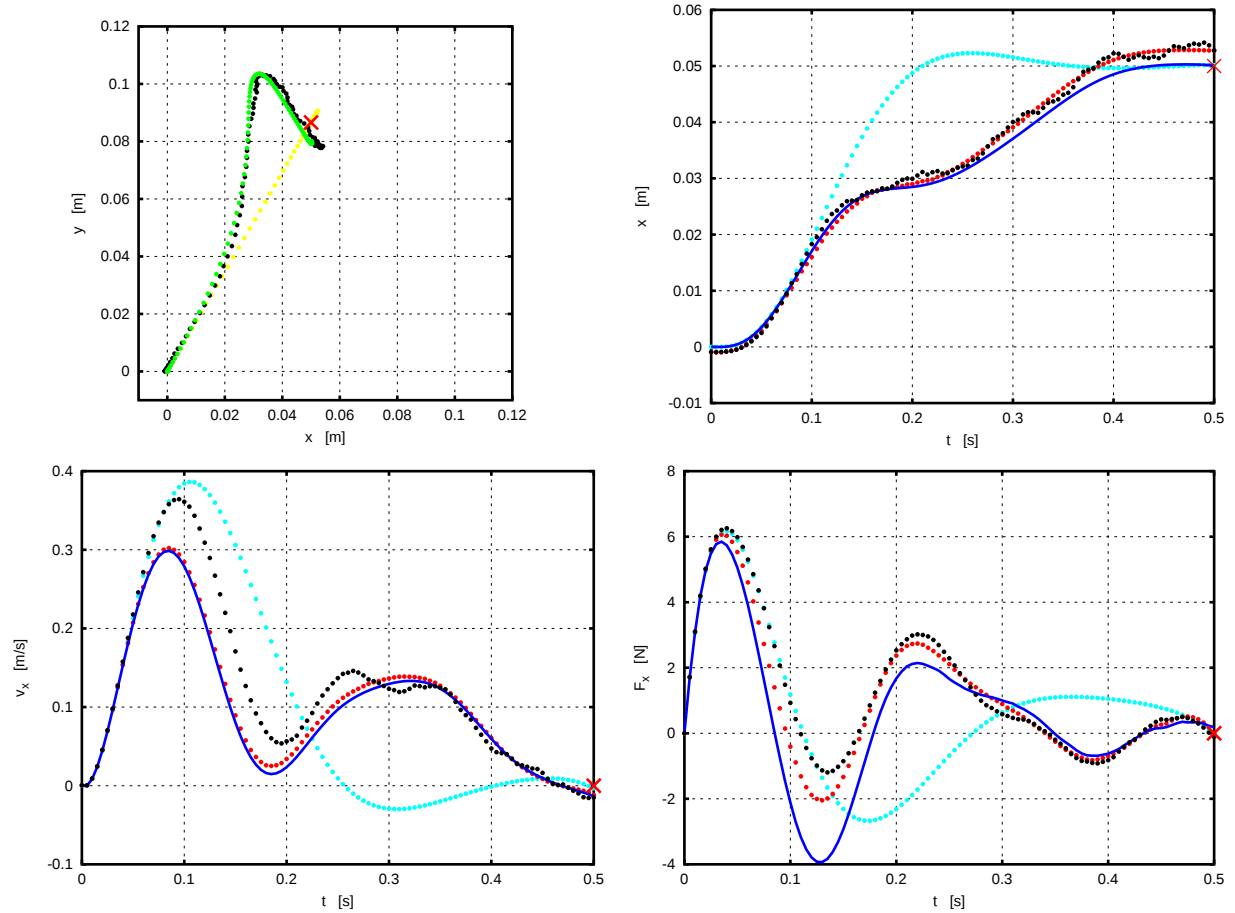


FIGURE 10.3 – Trajectoire (haut gauche) et position (haut droit), vitesse (bas gauche) et force (bas droit) dans l'axe x en présence d'un délai de 50 ms avec un champ de perturbation $10 \cdot \mathbf{G}_1$ et un contrôleur utilisant une action correctrice par feedback sur les erreurs d'estimations.

de contrôle. Comme nous pouvons le voir sur cette figure, les prédictions au moment du contrôle (pointillés noirs) sont maintenant très proches des prédictions optimales (pointillés rouges) et suivent nettement mieux les profils réels (bleu) malgré le retard de 50 ms . Mission accomplie!

10.4 Discussion

Il semble possible de stabiliser un système avec délai et perturbation, et ce de manière relative robuste : la méthode empirique que nous avons trouvée consiste à utiliser un feedback à partir du calcul de l'erreur d'estimation commise au moment du contrôle.

Cependant, nous avons utilisé dans ce chapitre une méthode basée sur l'intuition consistant à simplement retrancher l'écart de prédiction calculé de l'estimation donnée par le modèle direct. Comme nous pouvons le voir dans la figure 10.3, l'estimation utilisée au moment du contrôle oscille autour de l'estimation optimale. En effet, le fait de retrancher l'écart d'estimation à chaque pas de temps fait que la correction est à chaque fois trop importante, entraînant cet effet d'oscillation.

Que faire de ces informations sur l'erreur de prédiction commise au moment du contrôle alors ? Faut-il

les propager via le modèle direct avant de les appliquer ? Faut-il les pondérer ?

Nous ne possédons pas encore la réponse mais sommes convaincus qu'il devrait être possible bâtir une théorie mathématique utilisant cet écart de prédiction, voir tous les écarts de prédiction connus jusqu'à un état donné, afin de trouver une loi de contrôle optimale pour les systèmes présentant du délai.

Maintenant que nous sommes parvenus à mettre au point un contrôleur performant, il pourrait être intéressant d'envisager l'utilisation de modèles plus complexes du bras. Nous n'avons pas résister à l'envie d'amorcer une telle démarche en tentant d'appliquer les méthodes développées jusqu'ici à un modèle du bras à deux articulations découplées. Cette dernière validation sera l'objet du prochain chapitre.

Modélisation découplée du bras

Ce mémoire touche bientôt à sa fin. Avant de mettre celui-ci en perspective dans le chapitre 12, nous aimerions appliquer les méthodes développées jusqu'à présent au cas d'un modèle du bras à deux articulations.

Comme nous en avons discuté dans le chapitre 4 et comme le démontrent les résultats de l'annexe B, les équations dynamiques pour un modèle à deux articulations sont hautement non-linéaires et de plus couplées entre elles.

Mais il est également possible d'utiliser un modèle nettement simplifié qui néglige tout simplement les couplages existant entre les articulations du coude et de l'épaule et considère que celles-ci agissent totalement indépendamment l'une de l'autre. Bien qu'encore assez éloigné de la réalité, ce modèle est un bon point de départ et ne nécessite de plus que de légères adaptations par rapport au modèle de masse ponctuelle utilisé jusqu'ici.

Nous commencerons par dériver le modèle cinématique inverse du bras, modèle inverse qui nous permettra de spécifier quel est la configuration que l'épaule et le coude doivent adopter pour que la main soit positionnée à un emplacement défini. Nous spécifierons ensuite quels sont les équations dynamiques décrivant notre modèle et les valeurs des paramètres utilisés. Enfin nous observerons les simulations résultantes et terminerons par une brève discussion.

11.1 Modèle cinématique inverse

Un *modèle cinématique inverse* permet de relier les coordonnées cartésiennes de l'extrémité d'une chaîne cinématique aux coordonnées internes des articulations de cette chaîne. En d'autres mots, un modèle inverse nous permet de déterminer quels angles de l'épaule θ_1 et du coude θ_2 permettent d'atteindre une position définie (x, y) dans le plan. Cela nous permettra de spécifier à la fois la configuration initiale et la configuration finale souhaitée dans notre simulation.

Pour y arriver, nous nous inspirerons du livre de référence en robotique de Lynch et Park [Lynch, 2017] et des notations des diagrammes représentés à la figure 11.1.

Commençons par dériver l'angle γ . Celui-ci est facilement déterminé à partir des coordonnées (x, y) :

$$\gamma = \arctan\left(\frac{y}{x}\right). \quad (11.1)$$

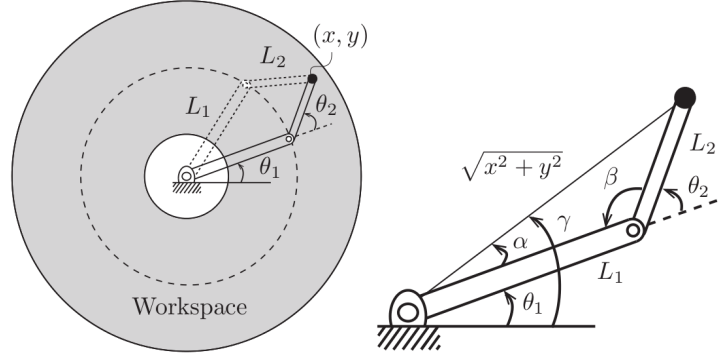


FIGURE 11.1 – Diagrammes utilisés afin de dériver le modèle cinématique inverse du bras. Source : [Lynch, 2017].

Ensuite, la loi des cosinus nous permet de déterminer les angles α et β :

$$\alpha = \arccos\left(\frac{r^2 + L_1^2 - L_2^2}{2L_1 r}\right), \quad (11.2)$$

$$\beta = \arccos\left(\frac{L_1^2 + L_2^2 - r^2}{2L_1 L_2}\right), \quad (11.3)$$

$$(11.4)$$

avec $r^2 = \sqrt{x^2 + y^2}$.

Le modèle cinématique inverse est alors donné par

$$\begin{cases} \theta_1 &= \gamma \pm \alpha, \\ \theta_2 &= \pm(\beta - \pi), \end{cases} \quad (11.5)$$

$$(11.6)$$

où le signe à choisir dépend de la configuration du bras.

Comme le montre la gauche de la figure 11.1, deux solutions existent en effet pour un même point (x, y) : une avec le coude vers le haut (+) et l'autre avec le coude vers le bas (-). Conformément à l'anatomie humaine, nous garderons la configuration dans laquelle le coude est orienté vers le haut.

Avant de passer à la suite, notons qu'il est également possible de déterminer les coordonnées x et y à partir des angles θ_1 et θ_2 , ce qui nous permettra de visualiser nos résultats. Dans ce cas, il faut recourir au *modèle cinématique direct* :

$$\begin{cases} x &= L_1 \cos(\theta_1) + L_2 \cos(\theta_1 + \theta_2), \\ y &= L_1 \sin(\theta_1) + L_2 \sin(\theta_1 + \theta_2). \end{cases} \quad (11.7)$$

$$(11.8)$$

11.2 Équations dynamiques et paramètres utilisés

Nous utilisons comme équations dynamiques celles de deux articulations rotoïdes découplées¹ auxquelles nous greffons les équations de filtres passe-bas du premier ordre représentant la commande musculaire, soit

1. Voir l'équation 5.1 du chapitre 5.

$$\begin{cases} \mathcal{I}_1 \cdot \ddot{\theta}_1 + \kappa_v \cdot \dot{\theta}_1 + \kappa_e \cdot \theta_1 & = \mathcal{T}_1 \\ \mathcal{I}_2 \cdot \ddot{\theta}_2 + \kappa_v \cdot \dot{\theta}_2 + \kappa_e \cdot \theta_2 & = \mathcal{T}_2 \\ \tau \cdot \dot{\mathcal{T}}_1 + \mathcal{T}_1 & = u_1 \\ \tau \cdot \dot{\mathcal{T}}_2 + \mathcal{T}_2 & = u_2 \end{cases} \quad \begin{matrix} (11.9) \\ (11.10) \\ (11.11) \\ (11.12) \end{matrix}$$

où θ_1 et θ_2 représentent respectivement les angles de l'épaule et du coude.

Ces équations sont l'équivalent angulaire de notre modèle ponctuel du chapitre 4 et sont à interpréter de la même manière. Il est cependant nécessaire d'adapter les paramètres utilisés.

Pour la longueur et la masse de l'avant-bras (L_1, m_1) et de l'avant-bras (L_2, m_2), nous considérons les paramètres utilisés par Shadmehr et Mussa-Ivaldi [Shadmehr, 1994] :

$$L_1 = 0.33 \text{ m}, \quad (11.13)$$

$$L_2 = 0.34 \text{ m}, \quad (11.14)$$

$$m_1 = 1.52 \text{ kg}, \quad (11.15)$$

$$m_2 = 1.93 \text{ kg}. \quad (11.16)$$

L'inertie est alors calculée en émettant l'hypothèse que la masse est répartie uniformément sur le bras et l'avant-bras. De plus, puisque nous ignorons les effets de couplage, nous nous autorisons à calculer simplement l'inertie à partir de l'articulation :

$$\begin{aligned} \mathcal{I}_i &= \int_0^{L_i} \frac{m_i}{L_i} \cdot r^2 \, dr, \\ &= \frac{m_i L_i^2}{3}, \quad i. \end{aligned} \quad (11.17)$$

Soit

$$\mathcal{I}_1 \approx 0.055 \text{ kg} \cdot \text{m}^2, \quad (11.18)$$

$$\mathcal{I}_2 \approx 0.074 \text{ kg} \cdot \text{m}^2. \quad (11.19)$$

Les paramètres de viscoélasticité κ_v et κ_e utilisés sont ceux mesurés par Crevecoeur et Scott [Crevecoeur, 2014] et nous gardons par ailleurs la valeur de τ choisie précédemment :

$$\kappa_e = 1.61 \text{ Nm/rad}, \quad (11.20)$$

$$\kappa_v = 0.14 \text{ Nm} \cdot \text{s/rad}, \quad (11.21)$$

$$\tau = 0.06 \text{ s}. \quad (11.22)$$

11.3 Simulation

Pour être simulé, ce nouveau modèle ne nécessite que de légères adaptations dans notre programme. Comme position de départ, nous plaçons la main à 30 cm en face du thorax [Wagner, 2008] en considérant une demie-largeur d'épaule de 24 cm [Motmans, 2005]. La cible est toujours située à 10 cm de cette position de départ avec une orientation à 60°.

Précisons que le contrôle s'effectue sur les variables *internes* au modèle, θ_1 et θ_2 , tandis que la trajectoire est visualisée dans les coordonnées *externes*, x et y , d'où la nécessité d'une conversion opérée par un modèle

cinématique direct.

Pour la première simulation, le champ de force perturbateur n'est pas présent. Par contre le délai est lui bien activé et nous utilisons la fonction de coût modifiée développée dans le chapitre 9. Les résultats sont présentés dans la figure 11.2. On notera que le contrôle s'effectue en un temps très court (environ 150 *ms*).

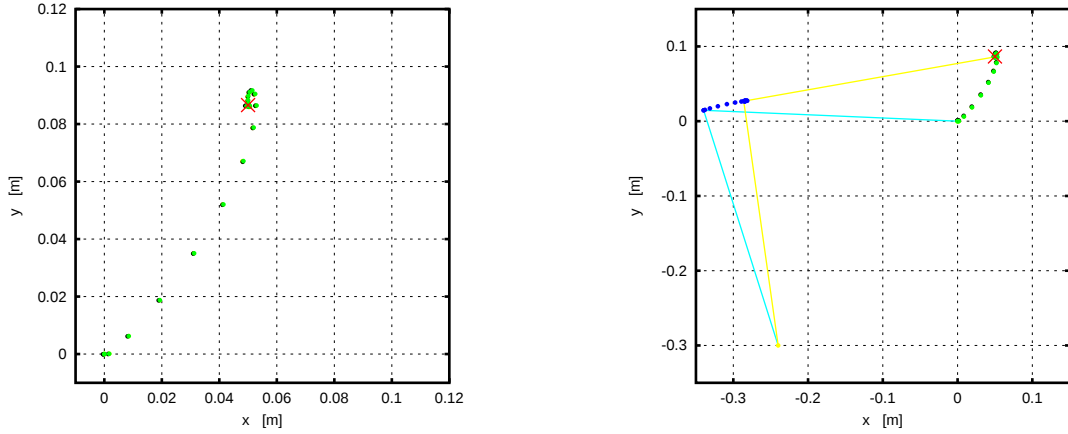


FIGURE 11.2 – Droite : trajectoire (vert) de l'extrémité du bras en l'absence de perturbations mais avec un délai de 50 *ms*. Gauche : mise en évidence des positions de départ (cyan) et d'arrivée (jaune) du bras.

Pour la deuxième simulation, nous activons maintenant la perturbation. À cet effet, nous utilisons l'équivalent en coordonnées angulaires du champ de force de Shadmehr et Mussa-Ivaldi [Shadmehr, 1994] :

$$\mathbf{G}_\theta = \begin{pmatrix} 1.66 & 0.64 \\ 0.64 & -1.54 \end{pmatrix} Nm \cdot s/rad. \quad (11.23)$$

Les résultats sont présentés dans la figure 11.3, où nous avons également effectuée une simulation avec un champ de magnitude dix fois plus élevée donnant lieu à un contrôle instable.

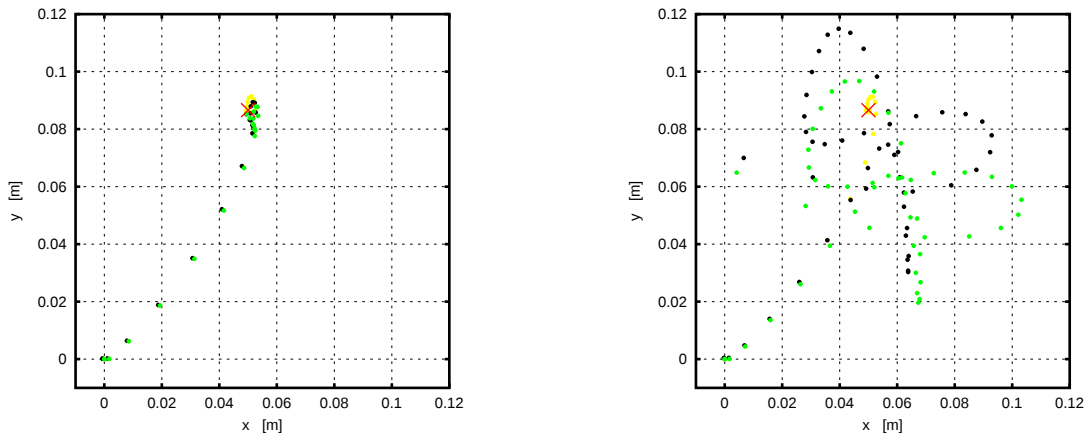


FIGURE 11.3 – Simulations avec perturbation. Droite : trajectoire stable pour une perturbation $\mathbf{G}_\theta$. Gauche : trajectoire instable pour une perturbation $10 \cdot \mathbf{G}_\theta$.

11.4 Discussion

Bien que le modèle utilisé dans ce chapitre soit toujours assez éloigné de la réalité, il a été intéressant de tester notre contrôleur sur celui-ci.

Les résultats en eux-mêmes ne devraient pas nous surprendre : si notre contrôleur est capable d'effectuer son contrôle pour des variables découplées dans le cas d'une masse ponctuelle, il devrait également être capable d'effectuer son contrôle sur ce nouveau modèle, puisque aucun terme de couplage n'est présent.

La vraie nouveauté est que, le contrôle s'effectuant sur les coordonnées internes du bras, la trajectoire est modélisée de manière différente. Cependant, l'introduction de perturbation ne donne toujours pas lieu à la vive correction mesurée lors d'expériences.

Enfin, on remarque qu'en l'absence de perturbation le trajet à la cible n'est pas une ligne droite mais une courbe. Ceci est la conséquence du fait que la fonction de coût pénalise l'erreur sur les variables internes et pas l'erreur sur la position. Serait-il possible, tout en restant dans la cadre d'un formalisme LQG, de modifier la fonction de coût afin que le contrôle continue à s'effectuer sur les variables internes mais que les contraintes du mouvement soit spécifiées par rapport aux coordonnées externes x et y ?

C'est sur cette dernière piste de réflexion que se clôture le travail de recherche de ce mémoire. Dans le prochain chapitre nous aborderons les perspectives et les pistes d'amélioration évoquées par nos résultats avant de passer à la conclusion.

Perspectives et pistes d'amélioration

Tout au long de ce mémoire, les discussions distillées à chaque fin de chapitre nous ont permis de dégager de nombreux points de vue intéressants et de commentaires critiques au sujet du domaine vaste et pluridisciplinaire que constitue l'étude des mouvements d'atteinte. Le but de ce chapitre n'est donc pas de répéter ces commentaires mais bien d'examiner dans une perspective plus large toutes les notions développées dans ce mémoire.

En guise de première prise de recul, nous aimerions donner un bref aperçu des points qui n'ont été volontairement que peu voire pas du tout abordés dans les précédents chapitres mais qui sont liés à notre problématique de manière très intime. Nous envisagerons ensuite les éléments de ce mémoire appelant à un travail de recherche plus approfondi.

12.1 Perspectives

Dans ce mémoire nous n'avons pas traité la question très importante des bases neurologiques du contrôle moteur. En effet, il ne suffit pas d'émettre l'hypothèse que le système nerveux utilise des stratégies de contrôle optimal lors de mouvements, ni même que les simulations donnent des résultats cohérents avec cette hypothèse pour pouvoir conclure que le cerveau utilise effectivement un mécanisme similaire. Un tel postulat doit être étayé par des indices effectifs du déroulement de processus similaires à ceux décrit par la théorie du contrôle dans le système nerveux. Ces techniques de calcul doivent par ailleurs être compatibles avec les capacités computationnelles du cerveau. Une partie de la littérature neuroscientifique se consacre par conséquent à la recherche et à la compréhension des structures nerveuses jouant un rôle dans la programmation et l'exécution des mouvements moteurs.

Dans la même optique, nous n'avons pas non plus évoqué les recherches cognitives en prise de décision (« decision making ») dont une partie s'occupe d'étudier la manière dont le système nerveux planifie ou modifie ses mouvements lors d'une tâche donnée, en fonction par exemple de l'environnement dans lequel s'effectue la tâche, de la présence d'obstacles dans cet environnement ou encore des consignes liées à la tâche à réaliser.

Enfin un autre sujet passionnant mais sortant du cadre de ce mémoire concerne la vaste question de l'apprentissage (« learning ») lors de mouvements d'atteinte. Cette autre pan de la question s'intéresse spécifiquement à comprendre par exemple comment le cerveau parvient à construire un modèle de la perturbation lorsque celui-ci y est confronté plusieurs fois et quels signaux d'erreur déclenchent cette adaptation.

12.2 Pistes d'amélioration

Comme dans tout travail de recherche, il y a certains aspects que nous n'avons pas eu l'occasion de développer plus en profondeur, mais qui constitueraient un prolongement intéressant aux questions adressées dans ce mémoire.

L'étape logique suivante qui permettrait de réellement valider nos résultats serait bien sûr de développer un contrôleur pour un modèle du bras à deux articulations qui tiendrait compte des couplages et des non-linéarités des équations dynamiques. Une première approche potentielle pourrait être de linéariser ces équations pour pouvoir continuer à utiliser les techniques mathématiques liées au formalisme LQG. Mais un autre angle d'attaque pourrait consister à utiliser des techniques de contrôle non-linéaires. Il serait en tout cas intéressant d'observer si la perturbation appliquée à ce modèle non-linéaire entraînerait une trajectoire présentant un crochet sans devoir augmenter la magnitude du champ de force.

Dans un autre ordre d'idée, notre approche pragmatique de la problématique du contrôle de mouvement d'atteinte dans un champ de force a pour conséquence que nous ne nous sommes pas attardés sur une analyse mathématique formelle de la question. Un important travail pourrait être réalisé à ce sujet car la théorie du contrôle permet de caractériser par des critères mathématiques spécifiques la stabilité et la robustesse d'un système asservi. Ce travail théorique pourrait également s'appliquer à l'étude d'un système dont le modèle présente des incertitudes et ouvrirait la voie à de nouvelles pistes d'exploration.

Pour terminer, nous pensons qu'il pourrait être furieusement intéressant de creuser en profondeur les deux résultats complètement inattendus mais très enthousiasmant de ce mémoire, à savoir qu'une fonction de coût dépendant du temps améliore nettement la robustesse du contrôleur, et qu'un système présentant du délai peut être stabilisé par une loi de contrôle tenant compte des erreurs d'estimation commises. Ceux-ci invitent en effet à développer une analyse mathématique permettant de généraliser, optimiser et justifier notre approche intuitive et empirique.

À travers douze chapitres, ce mémoire a étudié diverses problématiques liées au contrôle robuste inspiré par l'être humain de mouvements d'atteinte dans un champ de force. En particulier, nous nous sommes intéressés à la question de savoir si un contrôle stable et robuste était possible malgré la présence dans le système contrôlé de bruits, de perturbations, de délais et de contraintes biologiques liées au corps humain.

Le chapitre 1 a introduit notre thème de recherche en prenant pour point de départ un article de Shadmehr et Mussa-Ivaldi mettant en évidence les trajectoires en crochet observées chez les participants lors de mouvements d'atteinte dans un champ de force. Nous avons vu qu'une hypothèse plausible pour expliquer la forme de ces trajectoires est que le système nerveux utilise une stratégie de contrôle robuste mais que les simulations cherchant à reproduire ces résultats en se basant sur la théorie du contrôle optimal se heurtaient à deux difficultés : l'introduction de délais rend les systèmes asservis instables et les paramètres viscoléastiques des muscles, largement surestimés, ne peuvent jouer leur supposé rôle prépondérant dans la stabilisation du système.

Dans le chapitre 2 nous avons établi le cadre mathématique Linéaire-Quadratique-Gaussien nous permettant de décrire l'évolution d'un système dynamique. Pour ce faire, nous sommes partis de la représentation d'état d'un système d'équations différentielles que nous avons discrétisées à l'aide de la méthode d'Euler explicite. Nous avons ensuite appliqué cette discrétisation au cas des systèmes linéaires et sommes passés à une représentation d'état augmenté. Enfin, nous avons introduit un signal de mesure et du bruit gaussien additif dans notre représentation avant de spécifier un critère d'optimalité pour le contrôle associé au système via une fonction de coût quadratique.

Les résultats mathématiques permettant d'estimer de manière optimale l'état d'un système de manière récursive via un filtre de Kalman qui combine astucieusement prédiction et mesure ont été présentés dans le chapitre 3. Nous avons ensuite résolu la question duale qui consiste à trouver une loi de contrôle optimale permettant de minimiser une fonction de coût donnée et qui est donnée par un feedback linéaire sur l'état courant via des gains de contrôle précalculés.

Nous avons alors dérivé dans le chapitre 4 un modèle simplifié du bras l'assimilant à une masse ponctuelle en mouvement dans le plan. Partant d'un cas unidimensionnel, nous avons développé notre modèle en couplant un modèle biomécanique analogique de Maxwell simulant le comportement viscoélastique du bras et un filtre passe-bas du premier ordre modélisant l'impédance de la réponse musculaire face à une commande motrice. Nous avons ensuite étendu ces équations au cas bidimensionnel et calculé les matrices d'état correspondantes.

Dans le chapitre 5, nous avons défini un ordre de grandeur pour la valeur des paramètres biomécaniques de notre modèle à partir des mesures effectuées par Crevecoeur et Scott et à l'aide de l'équation d'une particule en rotation dans le plan. Nous avons ensuite fixé une valeur pour la constante de temps des muscles et discuté de l'anisotropie inhérente aux mouvements du bras.

Le chapitre 6, très bref, présentait un résumé synthétique des quatre chapitres précédents.

Par après, l'ensemble des paramètres de notre simulation a été fixé dans le chapitre 7, et la structure du programme implémenté parcourue rapidement. Les simulations sans perturbation nous ont permis d'observer des tracés correspondant bien aux mesures relevées lors d'expériences sur les mouvements d'atteinte. Nous avons alors étudié les effets des variations de paramètres sur la simulation et tester la robustesse du contrôleur face à celles-ci.

Dans le chapitre 8, nous avons examiné les répercussions d'un champ de force perturbateur proportionnel à la vitesse de la main sur les trajectoires simulées. La robustesse du contrôleur nous a poussé à devoir augmenter nettement la magnitude de la perturbation et à réduire les coûts associés au contrôle afin de pouvoir observer une action corrective correspondant aux trajectoires mesurées sur les participants, avec toutefois un profil de vitesse ne correspondant pas.

L'objet du chapitre 9 a été de parvenir à modifier la fonction de coût utilisée afin que le contrôleur soit capable de décider lui-même du temps nécessaire afin d'atteindre la cible, et ce en vue d'éliminer certaines incohérences liées à la fonction de coût utilisée précédemment. Pour ce faire, nous avons utilisé un coût dynamique sur l'erreur en position croissant linéairement avec le temps et avons pu constater la nette amélioration de performance de notre contrôleur en résultant. Bien que la magnitude de la perturbation nécessite toujours d'être augmentée afin d'observer une trajectoire en crochet, nous avons pu observer un profil de vitesse nettement plus concluant.

Le chapitre 10 a été l'occasion d'ajouter un délai de 50 ms entre la réception et la prise de mesure, ce qui a entraîné l'incapacité du contrôleur à effectuer un contrôle correct et l'apparition d'un comportement instable même en cas d'information complète sur l'état du système. Nous sommes finalement parvenus à stabiliser le système par une loi de contrôle empirique utilisant l'information disponible a posteriori concernant les erreurs d'estimation commises au moment du contrôle.

En guise de dernière validation, nous avons, dans le chapitre 11, testé notre loi de contrôle sur un modèle du bras à deux articulations mais ignorant totalement les effets de couplage existant entre ces articulations. Pour ce faire, nous avons calculé le modèle cinématique inverse du bras et adapté les équations dynamiques et les paramètres de notre modèle précédent à celui-ci. Les simulations ont permis d'observer les effets d'un contrôle et d'une perturbation en coordonnées internes sur la trajectoire résultante.

Enfin, dans le chapitre 12 nous avons passé en revue quelques-uns des sujets qui n'ont pas été abordés dans ce mémoire, notamment les questions liées aux bases neurologiques du contrôle moteur, à la prise de décision ou encore à l'apprentissage. Nous avons ensuite envisagé les pistes d'amélioration qui pourraient être approfondies et parmi lesquelles figurent par exemple la recherche de méthodes de contrôle non-linéaires sur un modèle plus complexe du bras, l'analyse mathématique formelle de la stabilité et de la robustesse incluant éventuellement les perturbations et les délais ou encore l'exploration des différents points de discussion abordés dans ce mémoire.

En conclusion, nous sommes finalement parvenus à obtenir un contrôleur performant, stable et robuste capable de simuler des mouvements d'atteinte inspirés par l'être humain dans champ de force, et ce en utilisant une fonction de coût dynamique croissant avec le temps et une action corrective tenant compte des erreurs d'estimation commises en présence de délai. Ces résultats encourageant soulignent la vraisemblance

de l'utilisation d'une telle stratégie par le système nerveux et apportent une solution au problème du contrôle optimal dans les systèmes avec délai. Toutefois, un travail important afin de généraliser les résultats de ce travail et de valider ceux-ci sur des modèles du bras plus complexes est encore nécessaire.

La recherche en neurosciences est un domaine passionnant et complexe situé à la croisée des sciences, soulevant de nombreuses questions et jalonné de multiples défis. Si, comme je l'ai mentionné en exergue de ce recueil, la rédaction de ce mémoire a été un travail de longue haleine, je ne me suis en tout cas jamais ennuyé tant j'ai été emporté par le sujet.

J'espère, à travers ces quelques pages, avoir réussi à transmettre au lecteur un peu de la curiosité qui m'a accompagné tout au long de ce travail !

Annexes

Programmation dynamique

La programmation dynamique est une technique mathématique permettant de résoudre des problèmes de décision séquentielle et d'optimisation. Elle a été développée dans les années 50 par Richard Bellman et est basée sur le principe d'optimalité qui permet de calculer des solutions par récurrence. En effet, une politique optimale a la propriété que, quels que soient l'état et la décision initiales, les décisions restantes doivent rester optimales par rapport à l'état résultant de la première décision.

Dans cette annexe nous développons, d'après les notes de cours de Mäkilä [Mäkilä, 2004], le formalisme de l'équation de Bellman permettant de résoudre des problèmes d'optimisation de manière récursive, et ce pour le cas déterministe ainsi que le cas stochastique. En effet, dans le chapitre 3, lors de la recherche d'un contrôle optimal permettant de minimiser une fonction de coût, nous appliquons l'équation de Bellman dans le cas stochastique au cas d'un système LQG.

A.1 Équation de Bellman

A.1.1 Cas déterministe

Théorème A.1. Soit une fonction $f(u_1, u_2)$ telle que son minimum existe en (u_1^o, u_2^o) :

$$\min_{u_1, u_2} f(u_1, u_2) = f(u_1^o, u_2^o). \quad (\text{A.1})$$

Alors, pour trouver ce minimum, on peut d'abord trouver le minimum de la fonction en u_2 , puis en u_1 :

$$\min_{u_1, u_2} f(u_1, u_2) = \min_{u_1} \left\{ \min_{u_2} f(u_1, u_2) \right\}. \quad (\text{A.2})$$

Exemple A.1. Considérons un problème de décision à deux étages où u_1 et u_2 sont les variables de décision.

Il s'agit du problème de minimisation suivant

$$\min_{u_1, u_2} \{f(x_1, u_1, x_2, u_2)\} = \min_{u_1, u_2} \{f_1(x_1, u_1) + f_2(x_2, u_2)\}, \quad (\text{A.3})$$

avec $x_2 = g(x_1, u_1)$.

Alors, nous pouvons trouver ce minimum de la manière suivante :

$$\min_{u_1, u_2} \{f(x_1, u_1, x_2, u_2)\} = \min_{u_1} \left\{ f_1(x_1, u_1) + \min_{u_2} \{f(x_2, u_2)\} \right\}, \quad (\text{A.4})$$

où afin de rendre la notation plus compacte, nous pouvons poser

$$V(x_2) = \min_{u_2} \{f(x_2, u_2)\}. \quad (\text{A.5})$$

Théorème A.2. (Généralisation) Soit un problème d'optimisation séquentielle consistant à trouver

$$\min_{u_1, \dots, u_N} \{f(x_1, u_1, \dots, x_N, u_N)\} = \min_{u_1, \dots, u_N} \{f_1(x_1, u_1) + \dots + f_N(x_N, u_N)\}, \quad (\text{A.6})$$

avec $x_{k+1} = g_k(x_k, u_k)$.

Alors ce problème d'optimisation peut être formulé par :

$$\min_{u_1, \dots, u_N} \{f(x_1, u_1, \dots, x_N, u_N)\} = V_1(x_1), \quad (\text{A.7})$$

où les $V_k(x_k)$ sont donnés récursivement par l'équation de Bellman :

$$\begin{cases} V_N(x_N) &= \min_{u_N} \{f_N(x_N, u_N)\} \\ V_k(x_k) &= \min_{u_k} \{f_k(x_k, u_k) + V_{k+1}(x_{k+1})\}. \end{cases} \quad (\text{A.8})$$

$$\quad \quad \quad (\text{A.9})$$

A.1.2 Cas stochastique

Théorème A.3. (Information complète) Soient X, Y deux variables aléatoires, $u = u(Y)$ une variable de décision, $\ell(X, Y, u)$ une fonction de coût à minimiser et ξ_k, ω_k deux variables aléatoires.

Nous considérons un problème d'optimisation stochastique consistant à trouver

$$\min_{u_1, \dots, u_N} E \{ \ell_1(X_1, u_1) + \dots + \ell_N(X_N, u_N) \}, \quad (\text{A.10})$$

$$(\text{A.11})$$

avec $x_{k+1} = g_k(x_k, u_k, \xi_k)$.

Alors ce problème d'optimisation peut être formulé par

$$\min_{u_1, \dots, u_N} E \{ \ell_1(X_1, u_1) + \dots + \ell_N(X_N, u_N) \} = E \{ V_1(x_1) \}, \quad (\text{A.12})$$

où les $V_k(x_k)$ sont donnés récursivement par l'équation de Bellman :

$$\begin{cases} V_N(x_N) &= \min_{u_N} [\ell_N(x_N, u_N)] \\ V_k(x_k) &= \min_{u_k} [\ell_k(x_k, u_k) + E \{ V_{k+1}(x_{k+1}) \}]. \end{cases} \quad (\text{A.13})$$

$$\quad \quad \quad (\text{A.14})$$

Théorème A.4. (Information incomplète) Soit le même problème d'optimisation stochastique où u_k est maintenant seulement fonction d'une observation y_k :

$$y_k = h_k(x_k, \omega_k). \quad (\text{A.15})$$

Alors ce problème d'optimisation peut être formulé par

$$\min_{u_1, \dots, u_N} E \{ \ell_1(X_1, Y_1, u_1) + \dots + \ell_N(X_N, Y_N, u_N) \} = E \{ V_1(Y_1) \}, \quad (\text{A.16})$$

où les $V_k(y_k)$ sont donnés récursivement par l'équation de Bellman :

$$\begin{cases} V_N(y_N) &= \min_{u_N} E [\ell_N(X_N, y_N, u_N) \mid y_N] \end{cases} \quad (\text{A.17})$$

$$\begin{cases} V_k(y_k) &= \min_{u_k} E [\ell_k(X_k, y_k, u_k) + V_{k+1}(Y_{k+1}) \mid y_k, u_k]. \end{cases} \quad (\text{A.18})$$

Modélisation non-linéaire du bras

Cette annexe expose brièvement la méthodologie utilisée afin de dériver les équations dynamiques d'un système plan à deux articulations rotoïdes tel que représenté dans la figure B.1 en s'inspirant des livres de référence sur la robotique de Spong, Hutchinson et Viyasagar [Spong, 2005] et de Lynch et Park [Lynch, 2017]. Les calculs intermédiaires, assez fastidieux seront cependant omis mais le lecteur intéressé pourra les trouver dans les ouvrages susmentionnés.

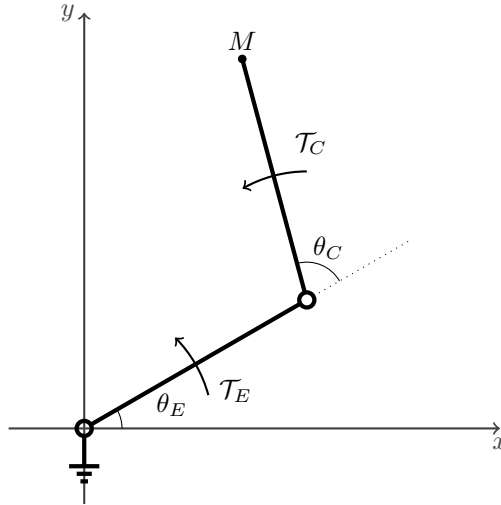


FIGURE B.1 – Représentation de l'avant-bras et du bras comme un système plan.

Un choix de coordonnées généralisées est donné par

$$\mathbf{q} = \begin{pmatrix} q_1 \\ q_2 \end{pmatrix} = \begin{pmatrix} \theta_E \\ \theta_C \end{pmatrix}, \quad (\text{B.1})$$

avec les forces généralisées correspondantes :

$$\mathbf{\Gamma} = \begin{pmatrix} \Gamma_1 \\ \Gamma_2 \end{pmatrix} = \begin{pmatrix} \tau_E \\ \tau_C \end{pmatrix}. \quad (\text{B.2})$$

Par la méthode d'Euler-Lagrange, nous pouvons dériver les équations dynamiques du système sous la forme

$$\mathbf{M}(\mathbf{q}) \cdot \ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \cdot \dot{\mathbf{q}} = \mathbf{\Gamma} \quad (\text{B.3})$$

avec

$$\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{2 \times 2} \quad : \quad \text{la matrice de masse,} \quad (\text{B.4})$$

$$\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^{2 \times 2} \quad : \quad \text{la matrice regroupant les effets centripètes et de Coriolis.} \quad (\text{B.5})$$

Pour dériver cette équation en fonction de nos coordonnées généralisées, il nous faut tout d'abord dériver les expressions des énergies potentielle $\mathcal{P}(\mathbf{q})$ et cinétique $\mathcal{K}(\mathbf{q}, \dot{\mathbf{q}})$ du système. Nous négligeons dans un premier temps les effets viscoléastiques des muscles.

Concernant l'énergie potentielle, puisque notre système est plan et que la gravité est compensée par le manipulateur robotique, nous savons que celle-ci est toujours constante et nous pouvons la fixer sans perte de généralité à 0 :

$$\mathcal{P}(\mathbf{q}) = 0. \quad (\text{B.6})$$

Concernant l'énergie cinétique, nous pouvons la calculer pour chaque partie de notre système et ensuite sommer les différentes énergies cinétiques trouvées :

$$\mathcal{K}(\mathbf{q}, \dot{\mathbf{q}}) = \sum_i \mathcal{K}_i(\mathbf{q}, \dot{\mathbf{q}}). \quad (\text{B.7})$$

L'énergie cinétique de chaque partie est calculée comme étant la somme de l'énergie cinétique provenant de la vitesse linéaire du centre de masse et de l'énergie cinétique provenant de la vitesse angulaire, où la matrice d'inertie est calculée à partir du centre de masse. Notons que pour trouver la vitesse linéaire du centre de masse, il est nécessaire de dériver le modèle cinématique direct du système.

Passant outre le détail de ces calculs, une fois l'énergie cinétique connue, le lagragien $\mathcal{L}$ du système peut ensuite être formé par l'équation

$$\mathcal{L}(\mathbf{q}, \dot{\mathbf{q}}) = \mathcal{K}(\mathbf{q}, \dot{\mathbf{q}}) - \mathcal{P}(\mathbf{q}) \quad (\text{B.8})$$

et les équations dynamiques du système sont finalement obtenues comme étant

$$\frac{d}{dt} \left\{ \frac{\partial \mathcal{L}}{\partial \dot{\mathbf{q}}} \right\} - \frac{\partial \mathcal{L}}{\partial \mathbf{q}} = \mathbf{\Gamma}, \quad (\text{B.9})$$

cette dernière équation pouvant être réécrite sous la forme donnée par l'équation [B.3](#).

Considérant que la masse et la longueur de l'avant-bras et du bras sont donnés respectivement par m_1 , L_1 et m_2 , L_2 , que leur masse est par ailleurs uniformément répartie donnant lieu aux positions respectives pour les centres de masse $l_1 = \frac{L_1}{2}$ et $l_2 = \frac{L_2}{2}$ et qu'enfin leur inertie respective calculée à partir de ce centre de masse vaut respectivement $\mathcal{I}_1$ et $\mathcal{I}_2$, nous trouvons après calcul le modèle dynamique suivant :

$$\mathbf{M}(\mathbf{q}) = \begin{pmatrix} m_1 l_1^2 + m_2 (L_1^2 + l_2^2 + 2L_1 l_2 \cos(q_2)) + \mathcal{I}_1 + \mathcal{I}_2 & m_2 (l_2^2 + L_1 L_2 \cos(q_2)) + \mathcal{I}_2 \\ m_2 (l_2^2 + L_1 L_2 \cos(q_2)) + \mathcal{I}_2 & m_2 l_2^2 + \mathcal{I}_2 \end{pmatrix} \quad (\text{B.10})$$

$$\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) = \begin{pmatrix} -m_2 \cdot L_1 l_2 \sin(q_2) \cdot \dot{q}_2 & -m_2 \cdot L_1 l_2 \sin(q_2) \cdot (\dot{q}_1 + \dot{q}_2) \\ m_2 \cdot L_1 l_2 \sin(q_2) \cdot \dot{q}_1 & 0 \end{pmatrix} \quad (\text{B.11})$$

Par conséquent, non seulement ces équations sont fortement couplées, mais elles sont en plus trigonométriques en $\ddot{\mathbf{q}}$ et quadratiques en $\dot{\mathbf{q}}$ ce qui nous empêche de les utiliser dans le cadre d'une représentation d'état linéaire. Par ailleurs, elles ne tiennent pas du tout compte des effets viscoléastiques liés aux muscles...

Paramètres utilisés dans la littérature

Dans cette annexe, nous présentons les modèles utilisés et des tableaux comparatifs des différentes valeurs de paramètres rencontrées dans la littérature.

C.1 Modèles utilisés

- ▷ [Shadmehr, 1994] Modèle non-linéaire du bras à deux articulations.
- ▷ [Todorov, 2005] Modèle unidimensionnel pour un mouvement de translation.
- ▷ [Diedrichsen, 2007] Modèle bidimensionnel dérivé d'une approximation linéaire du modèle non-linéaire.
- ▷ [Crevecœur, 2014] Modèle à une articulation.

C.2 Paramètres biomécaniques

Source	m	k_e	k_v	τ
[Shadmehr, 1994]	1.93 kg 1.52 kg ¹	$\begin{pmatrix} -15 & -6 \\ 6 & -15 \end{pmatrix} Nm/rad$	$\begin{pmatrix} -2.3 & -0.9 \\ -0.9 & -2.4 \end{pmatrix} Nm \cdot s/rad$	
[Todorov, 2005]	1 kg			$\tau_1 = 40 ms$ $\tau_2 = 40 ms$
[Diedrichsen, 2007]	0.5 kg		129.9 N · s/m	$\tau_1 = 40 ms$ $\tau_2 = 40 ms$
[Izawa, 2008]	0.3 kg			
[Crevecœur, 2014]		1.61 Nm/rad	0.14 Nm · s/rad	66 ms

TABLE C.1 – Paramètres biomécaniques utilisés dans la littérature.

1. Valeurs données respectivement pour le bras et l'avant-bras.

C.3 Paramètres de simulation

Source	ρ	ϕ	Δt	T
[Shadmehr, 1994]	10 cm	$0, 45, \dots, 315^\circ$		
[Todorov, 2005]	10 cm		10 <i>ms</i>	250 ... 350 <i>ms</i>
[Diedrichsen, 2007]	10 cm			< 700 <i>ms</i>
[Izawa, 2008]	20 cm	$-15 \dots 15^\circ$	10 <i>ms</i>	
[Wagner, 2008]	10 cm	$43 \dots 47^\circ$		450 ... 550 <i>ms</i>
[Crevecœur, 2014]		90°	5 <i>ms</i>	300 ... 600 <i>ms</i>

TABLE C.2 – Paramètres de simulation utilisés dans la littérature.



Résultats des simulations

Dans cette annexe, nous présentons des résultats détaillés des valeurs finales calculées lors des simulations.

Notations

Pour une variable d'état α :

- ▷ L'erreur est donnée par $e_\alpha = \alpha^* - \alpha$.
- ▷ L'erreur d'estimation est donnée par $\epsilon_\alpha = \alpha - \hat{\alpha}$.

Par ailleurs l'erreur sur la cible est donnée par $e_R = \sqrt{e_x^2 + e_y^2}$.

Tables

Objectif	Valeur	Erreur	Estimation	Erreur d'estimation	Unités
$x^* = 0.05$	$x = 0.04977$	$e_x = 0.00023$	$\hat{x} = 0.04945$	$\epsilon_x = 0.00032$	[m]
$y^* = 0.0866$	$y = 0.08574$	$e_y = 0.00086$	$\hat{y} = 0.08558$	$\epsilon_y = 0.00016$	[m]
		$e_R = 0.00089$			[m]
$v_x^* = 0$	$v_x = 0.00220$	$e_{v_x} = -0.00220$	$\hat{v}_x = 0.00107$	$\epsilon_{v_x} = 0.00113$	[m/s]
$v_y^* = 0$	$v_y = 0.00345$	$e_{v_y} = -0.00345$	$\hat{v}_y = 0.00317$	$\epsilon_{v_y} = 0.00028$	[m/s]
$F_x^* = 0$	$F_x = 0.01327$	$e_{F_x} = -0.01327$	$\hat{F}_x = -0.01128$	$\epsilon_{F_x} = 0.02455$	[N]
$F_y^* = 0$	$F_y = 0.00939$	$e_{F_y} = -0.00939$	$\hat{F}_y = 0.00582$	$\epsilon_{F_y} = 0.00357$	[N]

TABLE D.1 – Résultats lors d'un essai sans délai ni perturbation. Coût dynamique total : $18.57 \cdot 10^{-6}$. Coût de contrôle total : $126.14 \cdot 10^{-6}$.

Objectif	Valeur	Erreur	Estimation	Erreur d'estimation	Unités
$x^* = 0.05$	$x = 0.04699$	$e_x = 0.00301$	$\hat{x} = 0.04745$	$\epsilon_x = 0.00054$	[m]
$y^* = 0.0866$	$y = 0.08663$	$e_y = -0.00003$ $e_R = 0.00301$	$\hat{y} = 0.08610$	$\epsilon_y = 0.00054$	[m] [m]
$v_x^* = 0$	$v_x = 0.00078$	$e_{v_x} = -0.00078$	$\hat{v}_x = 0.00045$	$\epsilon_{v_x} = 0.00032$	[m/s]
$v_y^* = 0$	$v_y = -0.00079$	$e_{v_y} = 0.00079$	$\hat{v}_y = 0.00206$	$\epsilon_{v_y} = -0.00285$	[m/s]
$F_x^* = 0$	$F_x = -0.02528$	$e_{F_x} = 0.02528$	$\hat{F}_x = -0.00305$	$\epsilon_{F_x} = -0.02224$	[N]
$F_y^* = 0$	$F_y = 0.01390$	$e_{F_y} = -0.0139$	$\hat{F}_y = 0.00304$	$\epsilon_{F_y} = -0.02224$	[N]

TABLE D.2 – Résultats lors d'un essai avec le champ de force utilisé par Shadmehr et Mussa-Ivaldi, sans délai. Coût dynamique total : $9.056 \cdot 10^{-6}$. Coût de contrôle total : $130.99 \cdot 10^{-6}$.

Objectif	Valeur	Erreur	Estimation	Erreur d'estimation	Unités
$x^* = 0.05$	$x = 0.05046$	$e_x = -0.00046$	$\hat{x} = 0.05029$	$\epsilon_x = 0.00017$	[m]
$y^* = 0.0866$	$y = 0.08757$	$e_y = -0.00096$ $e_R = 0.00107$	$\hat{y} = 0.08720$	$\epsilon_y = 0.00036$	[m] [m]
$v_x^* = 0$	$v_x = -0.00078$	$e_{v_x} = 0.00078$	$\hat{v}_x = -0.00218$	$\epsilon_{v_x} = 0.0014$	[m/s]
$v_y^* = 0$	$v_y = -0.00468$	$e_{v_y} = 0.00468$	$\hat{v}_y = -0.00333$	$\epsilon_{v_y} = -0.00135$	[m/s]
$F_x^* = 0$	$F_x = -0.01857$	$e_{F_x} = -0.01857$	$\hat{F}_x = 0.00833$	$\epsilon_{F_x} = -0.01024$	[N]
$F_y^* = 0$	$F_y = 0.00858$	$e_{F_y} = -0.00858$	$\hat{F}_y = 0.00365$	$\epsilon_{F_y} = -0.01024$	[N]

TABLE D.3 – Résultats lors d'un essai avec le champ de force utilisé par Shadmehr et Mussa-Ivaldi, sans délai, avec une fonction de coût modifiée. Coût dynamique total : $7.619 \cdot 10^{-3}$. Coût de contrôle total : $3.157 \cdot 10^{-3}$.

Robustesse du contrôleur aux variations de paramètres

Cette annexe présente des tables répertoriant les bornes inférieures et supérieures approximatives des paramètres permettant d'atteindre la précision de 5 mm souhaitée et donnent par conséquent une idée de la tolérance du contrôleur aux variations de ces paramètres.

Paramètre	Valeur initiale	Borne inférieure	Borne supérieure	Unités
T	0.5	0.4		s
Δt	0.01	0.025	0.1	s
w_u	10^{-6}		$3.5 \cdot 10^{-6}$	
R_x	10^{-6}		10^2	
R_ξ	10^{-4}		10^{-1}	
m	1	0.005	1.9	kg
k_e	14.78	0	$5.5 \cdot k_e$	N/m
k_v	1.29	0	$6.5 \cdot k_v$	$N \cdot s/m$
τ	0.06	$\tau/100$	$3.5 \cdot \tau$	s
λ	0	0	0.75	

TABLE E.1 – Robustesse approximative du contrôleur aux variations de paramètres lorsque ni perturbation ni délai ne sont présents.

Paramètre	Valeur initiale	Borne inférieure	Borne supérieure	Unités
T	0.5	0.2		s
Δt	0.01		0.1	s
w_u	10^{-6}		10^{-5}	
R_x	10^{-6}		10^2	
R_ξ	10^{-4}		$5 \cdot 10^{-1}$	
m	1	0.005	5	kg
k_e	14.78	0	$20 \cdot k_e$	N/m
k_v	1.29	0	$70 \cdot k_v$	$N \cdot s/m$
τ	0.06	$\tau/100$	$5 \cdot \tau$	s
λ	0	0	0.95	

TABLE E.2 – Robustesse approximative du contrôleur aux variations de paramètres pour une fonction de coût modifiée lorsque ni perturbation ni délai ne sont présents.

Code GNU Octave

Cette annexe présente le code GNU Octave, dont la syntaxe est identique à celle de Matlab, utilisé afin d'implémenter le contrôleur et d'effectuer les simulations. Afin d'alléger la taille de cette annexe, nous n'avons toutefois pas inclus la partie du code calculant et affichant les résultats souhaités en ligne de commande ni celle s'occupant de dessiner, formater et exporter les figures d'intérêt.

```

1  %.title      ctrl.m
2  %.author     Francesco Sicurella
3  %.date       04/08/2017
4  %.thesis     Robust control of human-inspired reaching movements in a force field
5  %            ELME2990 – UCL 2016–2017
6  %.synopsis
7  % ctrl(T, dt, phi, w, wu, flag_Q, a, m, ke, kv, tau, lambda, flag_ksi, R_ksi, R_x, ...
   %      R_omega, flag_control, flag_E, G, flag_d, d, flag_C)
8  %.description
9  % Optimal feedback control for a 2D point mass.
10 %.parameters
11 % T : Duration [s].
12 % dt : Timestep [s].
13 % Boundary conditions.
14 % phi : Angle to target [deg].
15 % Cost function.
16 % w : Weights for the dynamic cost : [w_x w_y w_vx w_vy w_Fx w_Fy].
17 % wu : Weight for the control cost.
18 % flag_Q : If <true>, uses a running cost with [w_x w_y].
19 % a : Power of the running cost power function.
20 % Biomechanics.
21 % m : Mass [kg].
22 % ke : Stiffness [N/m].
23 % kv : Viscosity [N s/m].
24 % tau : Muscle first-order filter time constant [s].
25 % lambda : Correlation between x and y.
26 % Process noise.
27 % flag_ksi : If <true>, process noise (on forces) is activated.
28 % R_ksi : Ksi noise variance.
29 % Estimation.
30 % R_x : Initial state variance.
31 % R_omega : Omega noise variance.
32 % Control.
33 % flag_control : If 'complete', control with complete state information : u = -Lx
34 %               If 'incomplete', control with incomplete state information : u = -Lp
35 %               If 'blind', control based only on direct model : u = -LX
36 %               If 'hasteful', control based only on observation : u = -Ly
37 %               If 'none', no control : u = 0

```

```

38 % Perturbation
39 %   flag_E : If <true>, perturbation is activated.
40 %   G       : Perturbation submatrice :  $f = G \cdot v$ .
41 % Delay.
42 %   flag_d : If <true>, delay is activated.
43 %   d       : Steps of delay
44 %   flag_C : If <true> corrective posteriori action on the delayed system is applied.
45 function [] = ctrl(T, dt, phi, w, wu, flag_Q, a, m, ke, kv, tau, lambda, flag_ksi, R_ksi, ...
    R_x, R_omega, flag_control, flag_E, G, flag_d, d, flag_C);
46
47 %%%%%%%%%%%%%%
48 % PARAMETERS %
49 %%%%%%%%%%%%%%
50 % Simulation.
51 n = T/dt; % Number of iterations.
52 N = n+1;
53
54 % Boundary conditions.
55 xi = [0 0 0 0 0 0]'; % Initial condition [x y x' y' Fx Fy].
56
57 rho = 0.1; % Distance to target [m].
58
59 x_star = rho*cosd(phi);
60 y_star = rho*sind(phi);
61
62 xf = [x_star y_star 0 0 0 0]'; % Final condition [x* y* x'* y'* Fx* Fy*] : [m] [m] [m/s] ...
    [m/s] [N] [N].
63
64 %%%%%%%%%%%%%%
65 % STATE MATRICES %
66 %%%%%%%%%%%%%%
67 At = [ 0 0 1 0 0 0 ;
68        0 0 0 1 0 0 ;
69        -ke/m 0 -kv/m 0 1/m lambda/m ;
70        0 -ke/m 0 -kv/m lambda/m 1/m ;
71        0 0 0 0 -1/tau 0 ;
72        0 0 0 0 0 -1/tau ]; % Dynamic matrix : 6x6
73
74 Bt = [ 0 0 ;
75        0 0 ;
76        0 0 ;
77        0 0 ;
78        1/tau 0 ;
79        0 1/tau ]; % Control matrix : 6x2
80
81 %%%%%%%%%%%%%%
82 % AUGMENTED STATE %
83 %%%%%%%%%%%%%%
84 A = [ dt*At + eye(6,6) zeros(6,6) ;
85        zeros(6,6) eye(6,6) ]; % 12x12
86
87
88 B = [ dt*Bt ;
89        zeros(6,2) ]; % 12x2
90
91 H = eye(12, 12); % Observation matrix : 12x12
92
93 %%%%%%%%%%%%%%
94 % PERTURBATION %
95 %%%%%%%%%%%%%%
96 Et = [ 0 0 0 0 0 0 ;
97        0 0 0 0 0 0 ;
98        0 0 0 0 0 0 ;
99        0 0 0 0 0 0 ;
100       0 0 G(1,1) G(1,2) 0 0 ];

```

```

101         0    0    G(2,1)  G(2,2)  0    0    ]; % Perturbation matrice (Delta A) : 6x6
102
103 E = zeros(12, 12);
104
105 if (flag_E)
106     E = [    dt*Et    zeros(6, 6) ;
107           zeros(6, 12)    ]; % Perturbation matrice augmented : 12x12
108 end
109
110 %%%%%%%%%%
111 % NOISE %
112 %%%%%%%%%%
113 % Process noise (on Fx and Fy).
114 ksi = zeros(12, n);
115
116 Ksi = [ zeros(4, 12)    ;
117         0 0 0 0 R_ksi    0        zeros(1, 6) ;
118         0 0 0 0 0        R_ksi    zeros(1, 6) ;
119         zeros(6, 12)    ]; % Covariance matrix for process noise : 12x12
120
121 if (flag_ksi)
122     for k=1:n
123         ksi(5:6, k) = random('norm', 0, sqrt(R_ksi), 2, 1); % random('norm', <mean>, ...
124                               <standard deviation>, <rows>, <columns>).
125     end
126 end
127
128 % Initial noise.
129 sigma = zeros(6, 1);
130
131 if (R_x > 0)
132     sigma = random('norm', 0, sqrt(R_x), 6, 1);
133 end
134
135 % Measurement noise.
136 omega = zeros(12, n);
137
138 Omega = R_omega*eye(12, 12); % Covariance matrix for measurement noise : 12x12
139
140 for k=1:n
141     omega(1:6, k) = random('norm', 0, sqrt(R_omega), 6, 1);
142 end
143
144 %%%%%%%%%%
145 % OPTIMAL ESTIMATION : Kalman Filter %
146 %%%%%%%%%%
147 K = zeros(12, 12, N); % Filter gains.
148
149 Sigma = zeros(12, 12, N); % Covariance matrix of the estimation error.
150 Sigma(:, :, 1) = R_x*eye(12, 12);
151
152 for k=1:n
153     K(:, :, k) = A*Sigma(:, :, k)*H'*(H*Sigma(:, :, k)*H' + Omega)^-1;
154     Sigma(:, :, k+1) = (A - K(:, :, k)*H)*Sigma(:, :, k)*A' + Ksi;
155 end
156
157 %%%%%%%%%%
158 % COST FUNCTION : J = x'Qx + u'Q_u u %
159 %%%%%%%%%%
160 % Dynamic cost.
161 Q = zeros(12, 12, N); % Running cost.
162
163 Qn = zeros(12, 12); % Final cost condition : J[N] = x'Qx[N]
164
165 for i=1:2

```

```

165     Qn(i, i)      = w(i);
166     Qn(i, i+6)    = -w(i);
167     Qn(i+6, i)    = -w(i);
168     Qn(i+6, i+6)  = w(i);
169 end
170
171 if (flag_Q)
172     for i=1:n
173         Q(:, :, i) = ((i-1)/(n-1)).^a*Qn;
174     end
175 end
176
177 for i=3:6
178     Qn(i, i)      = w(i);
179     Qn(i, i+6)    = -w(i);
180     Qn(i+6, i)    = -w(i);
181     Qn(i+6, i+6)  = w(i);
182 end
183
184 Q(:, :, N) = Qn;
185
186     % Control cost.
187 Qu = wu*eye(2, 2);
188
189 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
190 % OPTIMAL CONTROL : u = -Lx %
191 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
192 L = zeros(2, 12, n);    % Control gains.
193
194 S = zeros(12, 12, N);
195 S(:, :, N) = Qn;
196
197 for k=n:-1:1
198     %L(:, :, k) = (B'*S(:, :, k+1)*B + Qu)^(-1)*B'*S(:, :, k+1)*A; % Matrix inversion : x ...
199     %               = A^-1*b <=> x = A \ b.
200     L(:, :, k) = (B'*S(:, :, k+1)*B + Qu) \ B'*S(:, :, k+1)*A;    % Same computation as ...
201     %               the previous row but faster.
202
203     S(:, :, k) = A'*S(:, :, k+1)*(A - B*L(:, :, k)) + Q(:, :, k);
204 end
205
206 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
207 % STATE INITIALIZATION %
208 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
209 % State.
210 x = zeros(12, N);
211 x(1:6, 1) = xi;
212 x(7:12, 1) = xf;
213
214 % State without noise nor perturbation for comparison.
215 X = x;
216
217 % Observed state.
218 y = zeros(12, n);
219
220 % Predicted state : x^
221 p = x;
222 p(1:6, 1) = xi + sigma; % Initial state uncertainty.
223
224 % Updated predicted state (updated each turn).
225 z = x;
226 z(1:6, 1) = xi + sigma; % Initial state uncertainty.
227
228 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
229 % ITERATION %

```

```

228 %%%%%%%%%%%
229 for k=1:n
230     % Observation.
231     y(:, k) = H*x(:, k) + omega(:, k);
232
233     % System evolution.
234     X(:, k+1) = (A - B*L(:, :, k))*X(:, k);
235
236     if strcmp(flag_control, 'complete')
237         x(:, k+1) = (A + E - B*L(:, :, k))*x(:, k) + ksi(:, k);
238     end
239     if strcmp(flag_control, 'incomplete')
240         x(:, k+1) = (A + E)*x(:, k) - B*L(:, :, k)*p(:, k) + ksi(:, k);
241     end
242     if strcmp(flag_control, 'blind')
243         x(:, k+1) = (A + E)*x(:, k) - B*L(:, :, k)*X(:, k) + ksi(:, k);
244     end
245     if strcmp(flag_control, 'hasteful')
246         x(:, k+1) = (A + E)*x(:, k) - B*L(:, :, k)*y(:, k) + ksi(:, k);
247     end
248     if strcmp(flag_control, 'none')
249         x(:, k+1) = (A + E)*x(:, k) + ksi(:, k);
250     end
251
252     % Estimation.
253     if (flag_d)
254         % /\ Prediction of state k+1 when it's still state k /\
255         if (k < d) % Until no information is available...
256             z(:, k+1) = (A - B*L(:, :, k))*z(:, k); % ...prediction with forward model.
257         else
258             z(:, k-d+2) = (A - B*L(:, :, k-d+1))*z(:, k-d+1) + K(:, :, k-d+1)*(y(:, k-d+1) ...
259                 - H*z(:, k-d+1)); % Kalman filter for the delayed measure...
260             for i=(k-d+2):k % ...and propagation of the new information.
261                 z(:, i+1) = (A - B*L(:, :, i))*z(:, i);
262             end
263             if (flag_C)
264                 if (k < d)
265                     p(:, k+1) = z(:, k+1);
266                 else
267                     p(:, k+1) = z(:, k+1) - (p(:, k-d+2) - z(:, k-d+2));
268                 end
269             else
270                 p(:, k+1) = z(:, k+1);
271             end
272         else
273             p(:, k+1) = (A - B*L(:, :, k))*p(:, k) + K(:, :, k)*(y(:, k) - H*p(:, k));
274         end
275     end
276 %%%%%%%%%%
277 % EXIT %
278 %%%%%%%%%%
279 end

```

Bibliographie

- [Aström, 1970] Aström, K. J. (1970). *Introduction to stochastic control theory*. Division of automatic control, Lund Institute of Technology, Lund, Sweden.
- [Aström, 2009] Aström, K. J. & Murray, R. M. (2009). *Feedback Systems, An Introduction for Scientists and Engineers*. Princeton University Press.
- [Crevecœur, 2016] Crevecœur, F. (2016). *Control Theory and Applications to Neuroscience*. Université Catholique de Louvain. Notes du cours LGBIO2060.
- [Crevecœur, 2014] Crevecœur, F. & Scott, S. H. (2014). Beyond muscles stiffness : Importance of state-estimation to account for very fast motor corrections. *PLOS Computational Biology*, 10(10).
- [Diedrichsen, 2007] Diedrichsen, J. (2007). Optimal task-dependent changes of bimanual feedback control and adaptation. *Current Biology*, 17 :1675–1679.
- [Evans, 1983] Evans, L. C. (1983). *An Introduction to Mathematical Optimal Control Theory*. Department of Mathematics, University of California, Berkeley. Notes de cours.
- [Ferrière, 2011] Ferrière, M. (2011). Les principales critiques de la théorie du choix rationnel. *Idées économiques et sociales*, 3(165) :37–45. Disponible en ligne à l'adresse : <http://www.cairn.info/revue-idees-economiques-et-sociales-2011-3-page-37.htm>.
- [Harris, 1998] Harris, C. M. & Wolpert, D. M. (1998). Signal-dependent noise determines motor planning. *Nature*, 394 :780–784.
- [Henrotte, 2016] Henrotte, F. & Marchandise, E. (2016). *Biomechanics*. École Polytechnique de Louvain, Université Catholique de Louvain. Notes du cours LGBIO2040.
- [Izawa, 2008] Izawa, J. & Shadmehr, R. (2008). On-line processing of uncertain information in visuomotor control. *The Journal of Neuroscience*, 28(44) :11360–11368.
- [James, 2004] James, M. (2004). *An Overview of Risk-Sensitive Stochastic Optimal Control*. Workshop : Stochastic control, communications and numerics perspective, UniSA, Adelaide. Présentation consultable à l'adresse : <http://users.cecs.anu.edu.au/~Matthew.James/docs/adelaide04james.pdf>.
- [Jones, 2002] Jones, K., Hamilton, A. & Wolpert, D. (2002). Sources of signal-dependent noise during isometric force production. *Journal of Neurophysiology*, 88 :1533–1544.
- [Legat, 2012] Legat, V. (2012). *Méthodes Numériques*. École Polytechnique de Louvain, Université Catholique de Louvain. Notes du cours FSAB1104.
- [Lynch, 2017] Lynch, K. M. & Park, F. C. (2017). *Modern Robotics : Mechanics, Planning, and Control*. Cambridge University Press. Chapitre 8.

- [Motmans, 2005] Motmans, R. & Ceriez, E. (2005). *DINBelg 2005 : Mesures corporelles de la population belge*. Ergonomie RC, Leuven. Site consultable à l'adresse : <http://www.dinbelg.be/anthropometrie.htm>.
- [Mäkilä, 2004] Mäkilä, P. (2004). *Kalman Filtering and Linear Quadratic Gaussian Control*. Institute of Automation and Control, Tampere University of Technology. Notes du cours 7604120.
- [Purves, 2015] Purves, D., Augustine, G. J., Fitzpatrick, D., Hall, W., Lamantia, A.-S. & White, L. (2015). *Neurosciences*. de boeck, 5^e edition.
- [Shadmehr, 1994] Shadmehr, R. & Mussa-Ivaldi, F. A. (1994). Adaptative representation of dynamics during learning of a motor task. *The Journal of Neuroscience*, 14(5) :3208–3224.
- [Spong, 2005] Spong, M. W., Hutchinson, S. & Viyasagar, M. (2005). *Robot Modeling and Control*. Wiley. Chapitre 7.
- [Todorov, 2005] Todorov, E. (2005). Stochastic optimal control and estimation methods adapted to the noise characteristics of the sensorimotor system. *Neural Computation*, 17 :1084–1108.
- [Wagner, 2008] Wagner, M. J. & Smith, M. A. (2008). Shared internal models for feedforward and feedback control. *The Journal of Neuroscience*, 28(42) :10663–10673.
- [Yedavalli, 2014] Yedavalli, R. K. (2014). *Robust Control of Uncertain Dynamic Systems, A Linear State Space Approach*. Springer.

