Online identification and control of robots using algebraic differentiators

Qi Guo

N° d'ordre : *285*

ECOLE CENTRALE DE LILLE

THESE

Présentée en vue
d'obtenir le grade de

DOCTEUR

En

Spécialité : Automatique, génie informatique, traitement du signal et des images

Par

GUO QI

DOCTORAT DELIVRE PAR L'ECOLE CENTRALE DE LILLE

Titre de la thèse :
Identification et commande en ligne des robots avec utilisation
de différentiateurs algébriques

Soutenue le 17 décembre 2015 devant le jury d'examen :

Président	Christine Chevallereau	Directeur CNRS
Rapporteur	Hugues Garnier	Professeur à l'université de Lorraine
Rapporteur	Philippe Poignet	Professeur à l'université Montpellier 2
Membre	Sette Diop	Chargé de recherche CNRS au Supélec
Directeur de thèse	Wilfrid Perruquetti	Professeur à l'Ecole Centrale de Lille
Co-Directeur de thèse	Maxime Gautier	Professeur emérite à l'Université de Nantes

Thèse préparée dans le Laboratoire CRIStAL
Ecole Doctorale SPI 072 (EC Lille)
PRES Université Lille Nord-de-France

Order Number : *285*

ECOLE CENTRALE DE LILLE

THESIS

Presented in order to
obtain the diploma of

DOCTOR

On

Specialty : Automation, computer engineering, signal processing and images

By

GUO QI

DOCTOR DELIVERED BY L'ECOLE CENTRALE DE LILLE

Title of the thesis :
Online identification and control of robots using algebraic differentiators

Defended on 17 December, 2015 in front of the jury :

President	Christine Chevallereau	Director CNRS
Referees	Hugues Garnier	Professor at université de Lorraine
Referees	Philippe Poignet	Professorà at université Montpellier 2
Examiners	Sette Diop	Researcher CNRS at Supélec
Supervisor	Wilfrid Perruquetti	Professor at Ecole Centrale de Lille
Co-supervisor	Maxime Gautier	Professor emeritus at Université de Nantes

Thesis prepared in the Laboratory CRIStAL
Ecole Doctorale SPI 072 (EC Lille)
PRES Université Lille Nord-de-France

This thesis has been prepared at

Centre de Recherche en Informatique, Signal, et Automatique de Lille (CRIStAL UMR 9189)

Bâtiment M3, Université Lille 1
59655, Villeneuve d'Ascq
France

☎ (33)(0)3 28 77 85 41
Web Site http://cristal.univ-lille.fr/

The Ph.D work presented in this thesis was carried out in Project-team SyNeR (Systèmes Hybrides Non Linéaires et à Retards) in Ecole Centrale de Lille, Project-team Non-A (Non-Asymptotic estimation for online systems) in INRIA Lille-Nord Europe and Project-team Robotics of Laborotory IRCCyN (Institut de Recherche en Communications et Cybernétique de Nantes UMR 6597 CNRS) in Ecole Centrale de Nantes. The Ph.D funding was supported by China Scholarship Council.

Je dédie ce travail
à tous ceux qui le méritent

À mon cher directeur !

À mon cher co-directeur !

Et à mes chers amis !

Contents

List of Tables

List of Figures

Résumé français

Introduction

La plupart des contrôleurs avancés ont besoin d'une bonne connaissance du modèle dynamique du robot pour leur mise en œuvre. Le modèle dynamique du robot peut être développé selon la méthode de Newton-Euler ou la méthode de Lagrange. Il décrit la dynamique du système en termes de position, vitesse, l'accélération et force ou couple, ainsi que des paramètres dynamiques. Les paramètres dynamiques sont des constantes comme la masse, l'inertie, les moments d'inertie, les paramètres de frottement de chaque articulation du robot, et le moment d'inertie global des éléments composant la chaîne cinématique du réducteur. Puisque les paramètres dynamiques sont inconnus, ils doivent d'être identifiés avant l'opération. De nombreux paramètres ne sont pas mesurables directement et doivent donc être identifiées à partir de mesures sur le robot en fonctionnement. Donc, la procédure d'identification des paramètres dynamiques est nécessaire. Dans la littérature, plusieurs méthodes ont été proposées pour résoudre ce problème, basée sur l'utilisation les 3 modèles:

- **modèle dynamique** (Canudas de Wit et al., 1991; Gautier, 1986; Gautier et al., 2008, 2013; Gautier and Khalil, 1990; Gautier, Khalil, and Restrepo, 1995; Gautier, Vandanjon, et al., 2011; Hollerbach et al., 2008; Khalil and Dombre, 2004; Khosla et al., 1985; Lu et al., 1993);

- **modèle d'énergie** (Gautier, 1996; Gautier and Khalil, 1988);

- **modèle de puissance** (Gautier, 1997).

L'objectif de cette thèse est de développer de nouvelles techniques en identification des paramètres dynamiques des robots et de les comparer avec les techniques existantes. Nous proposons de nouvelles techniques sur deux aspects: reformuler le modèle du robot en utilisant les fonctions modulatrices; appliquer les différentiateurs de Jacobi développés récemment pour résoudre le

problème de dérivation et analyser leur propriétés de filtrage dans le domaine fréquentiel.

Fonctions modulatrices en identification

Soit $l \in \mathbb{N}^*$, $T \in \mathbb{R}_+^*$, et g une fonction satisfaisant des propriétés suivantes:

$$\begin{aligned} (P_1) &: \ g \in \mathcal{C}^l([t_a, t_b]); \\ (P_2) &: \ g^{(i)}(t_a) = g^{(i)}(t_b) = 0, \ pour \ i = 0, 1, ..., l-1; \end{aligned} \tag{1}$$

avec $\mathcal{C}^l([t_a, t_b])$ l'ensemble des fonctions qui sont l–fois continûment dérivable sur la fenêtre de temps $[t_a, t_b]$ avec $l \in \mathbb{N}^*$. La fonction g définit une **fonction modulatrice** d'ordre l sur $[t_a, t_b]$.

Les fonctions modulatrices peuvent être utilisées pour l'identification. Supposons que $x^{(d)}$ est d'ordre d, avec x la variable d'observation et s est un nombre entier. On peut faire diminuer l'ordre de la dérivée $x^{(d)}$ grâce aux fonctions modulatrices. Par exemple, soit g une fonction modulatrice d'ordre l définie sur l'intervalle $[0, T]$, avec $l \geq d$. On multiplie g par $x^{(d)}$ et on intègre par partie le produit g sur la fenêtre de temps $[0, T]$. Ce qui permette les dérivées de x vers les dérivées de g, qui sont analytiquement connues.

$$\int_0^T g x^{(d)} = -\int_0^T \dot{g} x^{(d-1)} = \cdots = (-1)^d \int_0^T g^{(d)} x. \tag{2}$$

Analyse dans le domaine fréquentiel

Les intégrales (2) sont des intégrales de convolution correspondant à un filtrage qui peut être analysé dans le domaine fréquentiel.

Dans (Chen et al., 2011; Collado et al., 2009), les auteurs analysent également la propriété de différenciation dans le domaine fréquentiel. Le calcul pratique de $\int gx$ se fait sous la forme d'une convolution numérique avec une période d'échantillonnage T_s, et conduit à une version discrète sous la forme $\sum_{i=1}^{N} g[i]x[i]$. De cette façon, la convolution avec les fonctions modulatrices peut être analysée comme un filtre à réponse impulsionnelle finie (FIR) dont les coefficients sont les valeurs $g(t)$.

Les fonctions proposées $g_\ell(t)$ sont des fonctions modulatrices d'ordre K sur l'intervalle $[0, T]$, avec K l'ordre maximum de dérivation de x à calcule. Les fonctions $g(t)$ satisfont les deux conditions aux limites: $g_\ell^{(i)}(0) = 0$, et $g_\ell^{(i)}(T) = 0$, quand $i = 0, 1, \ldots, K-1$.

On étude 4 types de fonctions modulatrices:

(1) Les fonctions modulatrices sinusoïdales (SMF): la valeur de la fonction sinusoïdale atteint 0 à chaque demi-période, et on propose $g_\ell(t) = \sin^\ell(\frac{\pi}{T}t)$, avec $\ell \in \mathbb{R}$.

(2) Les fonctions modulatrices de Jacobi (JMF): ce groupe de fonctions est une combinaison de polynômes de Jacobi qui vaut 0 au début et à la fin de l'intervalle $[0, T]$. Il faut faire attention à ce que l'ordre des fonctions modulatrices soit supérieur à $K-1$ et on propose $g_\ell(t) = t^{\ell_1}(t-T)^{\ell_2}$, avec $\ell_1,\ \ell_2 \in \mathbb{R}$ et $\ell_1,\ \ell_2 > K-1$.

(3) Les fonctions modulatrices de Fourier (FMF): la fonction exponentielle $e^{ix} = \cos x + i \sin x$ est une fonction périodique qui atteint 1 à chaque période. Grâce à cette propriété, on peut écrire les fonctions modulatrices de Fourier sous la forme $g_\ell(t) = e^{-i\alpha\ell}(e^{-i\frac{2\pi}{T}t} - 1)^K$, où α est paramètre de réglage et $\ell \in \mathbb{R}$.

(4) Les fonctions modulatrices de Harley (HMF): elles sont basées la méthode de Shinbrot et Pearson Fourier fonctions modulatrices, ce groupe de fonctions est donné par :
$g_\ell(t) = \sum_{j=0}^{n}(-1)^j \begin{pmatrix} n \\ j \end{pmatrix} cas((n+\ell-j)\omega_0 t)$, où $\ell = 0, \pm1, \pm2, \ldots$ est entier, $\omega_0 = \frac{2\pi}{T}$ est la résolution fréquentielle et $\mathrm{cas(x)} = \cos \mathrm{x} + \sin \mathrm{x}$.

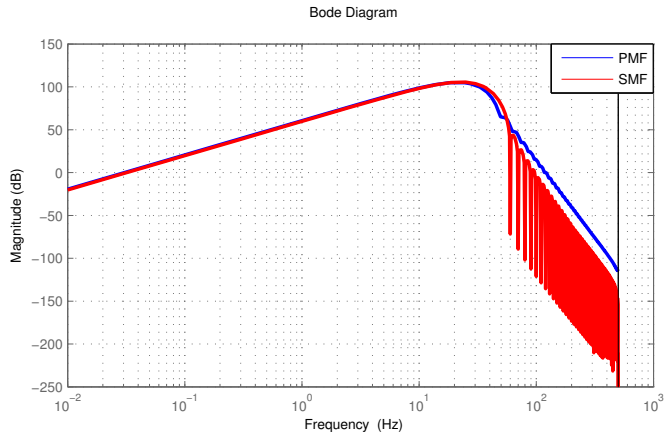
Les 4 types de fonctions modulatrices sont analysées d'un point de vue filtrage en calculant la forme discrète de la convolution dans laquelle les valeurs de $g_\ell(t)$Íéchantillonnée à la période de T_s, sont les coefficients d'un FIR. Par exemple, les diagramme de Bode en amplitude Fig. (1) représentent la réponse fréquentielle de FIR définie pour le calcul de la dérivée seconde avec les 4 types de fonctions modulatrices.

Identification en utilisant les fonctions modulatrices et le modèle de puissance

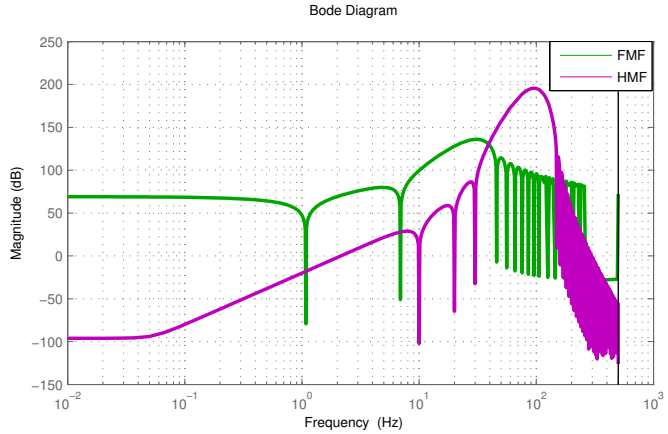
Les fonctions modulatrices peuvent être appliquées à l'identification des paramètres dynamiques des robots, sous la forme d'une convolution avec le modèle de puissance du robot. Rappelons le modèle de puissance (3) d'un robot avec n articulations:

$$\dot{\mathbf{q}}^{\mathbf{T}}\boldsymbol{\Gamma}_{\mathbf{m}} = \frac{d}{dt}(H) + \dot{\mathbf{q}}^{\mathbf{T}}[\mathbf{diag}(\dot{\mathbf{q}})\mathbf{F_v} + \mathbf{diag}(\mathbf{sign}(\dot{\mathbf{q}})\mathbf{F_s} + \boldsymbol{\Gamma}_{\mathbf{off}}]. \tag{3}$$

où $\mathbf{q}$, $\dot{\mathbf{q}}$ sont des vecteurs de la position et de la vitesse, de dimension $(n \times 1)$, Γ_m est le couple de moteur, H est l'énergie totale du robot, $H(\mathbf{q}, \dot{\mathbf{q}}) = E(\mathbf{q}, \dot{\mathbf{q}} + \mathbf{U}(\mathbf{q})$ est la somme de l'énergie cinétique de l'articulation E et de l'énergie potentielle U. H et linéaire par rapport aux parametres inertials K de robot. $H(\mathbf{q}, \dot{\mathbf{q}}) =$



(a) JMF and SMF



(b) FMF and HMF

Figure 1 – Diagramme de Bode du filtre FIR correspondant à la dérivation d'ordre 2 par fonctions modulatrices avec $\ell = 10$

$\mathbf{h}(\mathbf{q}, \dot{\mathbf{q}})\mathbf{K}$, les coefficients de $\mathbf{h}$ définissent les fonctions d'énergie. F_{vj}, F_{sj} sont les coefficients de frottement visqueux et de Coulomb de l'articulation j, Γ_{offj} est un paramètre d'offset (Gautier et al., 2013).

On intègre par partie, l'équation (3) pondérée par des fonctions modulatrices de premier ordre g sur l'intervalle $[t_a, t_b]$ et :

$$\int_{t_a}^{t_b} g\dot{\mathbf{q}}^{\mathbf{T}}\boldsymbol{\Gamma}_{\mathbf{m}}dt = -\int_{t_a}^{t_b} \dot{g}\mathbf{h}\mathbf{K}dt + \int_{t_a}^{t_b} g\dot{\mathbf{q}}^{\mathbf{T}}\mathbf{diag}(\dot{\mathbf{q}})\mathbf{F}_{\mathbf{v}}dt + \int_{t_a}^{t_b} tg\dot{\mathbf{q}}^{\mathbf{T}}\mathbf{diag}(\mathbf{sign}(\dot{\mathbf{q}}))\mathbf{F}_{\mathbf{s}}dt + \int_{t_a}^{t_b} g\dot{\mathbf{q}}^{\mathbf{T}}\boldsymbol{\Gamma}_{\mathbf{off}}dt. \quad (4)$$

Cette équation scalaire correspond aux modèle d'énergie pondéré par g, ce qui évite le calcul numérique de la dérivée d'ordre 2.

Les fonctions modulatrices sont bien adaptées à ce modèle, puisque nous

connaissons l'expression analytique de l'énergie $\mathbf{h}$ et les dérivées des fonctions modulatrices. En calculant (4) par différentes fonctions modulatrices g_ℓ.

On obtient un système linéaire surdéterminé de $n \times n_t \times n_m$ équations:

$$\mathbf{Y_E} = \mathbf{W_E}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\mathbf{X} + \rho_\mathbf{E}, \tag{5}$$

où n_t est le nombre d'intervalles de temps, n_m est le nombre de fonctions modulatrices, ρ_E est le bruit, $\mathbf{Y_E}$ et $\mathbf{W_E}$ sont respectivement le vecteur et la matrice d'observation, qui sont définis comme:

$$\mathbf{Y_E} = \begin{bmatrix} \int_{t_{a1}}^{t_{b1}} g_1 \dot{\mathbf{q}}^T \mathbf{\Gamma_m} \\ \vdots \\ \int_{t_{an_t}}^{t_{bn_t}} g_{n_e} \dot{\mathbf{q}}^T \mathbf{\Gamma_m} \end{bmatrix}, \mathbf{W_E} = \begin{bmatrix} -\int_{t_{a1}}^{t_{b1}} \dot{g}_1 \mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) \\ \vdots \\ -\int_{t_{an_t}}^{t_{bn_t}} \dot{g}_{n_e} \mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) \end{bmatrix}. \tag{6}$$

Le système (5) est résolu en utilisant des techniques de moindres carrés.

Differentiateur de Jacobi

Les différentiateurs numériques introduits dans cette partie sont basés sur des méthodes algébriques. Ils sont d'abord proposés par Fliess et Sira-Ramírez dans un article récent (Fliess, Mboup, et al., 2003; Fliess and Sira-Ramirez, 2004). Ces différentiateurs algébriques sont divisés en deux classes: les différentiateurs basés sur des modèles et les différentiateurs sans modèle. Les premiers (Fliess and Sira-Ramrez, 2004; Tian et al., 2008). ont été principalement utilisés pour les systèmes linéaires. Ils ont été étendus aux différentiateurs sans modèle, qui peuvent être utilisés pour les systèmes non linéaires et divers problèmes en traitement du signal. Le premier facteur de différentiation sans modèle a été introduit dans (Fliess, Join, et al., 2004) en appliquant la méthode algébrique de développement tronqué en série de Taylor du signal à différentier. Puis, deux différentiateurs sans modèle ont été étudiés dans (Mboup et al., 2007, 2009a). En outre, il a été montré que le différentiateur causal peut également être obtenu en projetant le signal sur la base orthogonale de Jacobi. Ensuite, il a été significativement amélioré en admettant un retard choisi par le concepteur (Mboup et al., 2007, 2009a). Dans (Liu et al., 2011c), un différenciateur central de Jacobi a été proposé, pour une utilisation hors ligne.

Les différentiateurs de Jacobi présentent les avantages suivants: leur calcul

par une intégrale définie sur une fenêtre glissante dans le cas continu, leur comportement de filtre passe-bande et correspond à une convolution dans le cas discrèt. En outre, le caractère passe-bande dans les hautes fréquences qui est robuste par rapport aux bruits (Fliess, 2006). Une étude théorique démontre que les erreurs sont des fonctions fortement non linéaires des paramètres de conception et que les erreurs sont limitées. D'autre part, quelques travaux expérimentaux montrent la relation entre les erreurs et les paramètres de conception (Liu et al., 2009a, 2011a, 2012a). Cependant, il n'existe pas encore de méthode efficace de conception en différentiateur de Jacobi, car les paramètres sont fortement couplés. Pour résoudre ce problème, j'ai proposé une écriture du différentiateur Jacobi sous la forme d'un FIR, ce qui permet d'analyser son comportement fréquentiel et fournit un outil de conception.

On présente ici les différentiateurs causaux de Jacobi, utilisés aussi pour les différentiateurs centrales de Jacobi. Soit une mesure bruitée $x^\omega : I \to \mathbb{R}$, $x^\omega(t) = x(t) + \omega(t)$, où I est un interval ouvert de temps fini $\mathbb{R}^+$, $x \in \mathcal{C}^n(I)$ avec $n \in \mathbb{N}$, et ω est un bruit. L'objectif est d'estimer la dérivée d'ordre n de x en utilisant x^ω. On applique les polynômes de Jacobi pour calculer le développement série orthogonale pour estimer la $n^{i\psi me}$ dérivée, (Mboup et al., 2007, 2009a).

D'abord, pour tout $t_0 \in I$, on introduit $D_{t_0} := \{t \in \mathbb{R}_+^* ; t_0 - t \in I\}$. La définition du polynôme d'ordre i décalé sur l'intervalle $[0,1]$ est (voir dans(Abramowitz et al., 1965) pp. 774-775), où $\mu, \kappa \in]-1, +\infty[$:

$$P_i^{(\mu,\kappa)}(\tau) = \sum_{j=0}^{i} \binom{i+\mu}{j} \binom{i+\kappa}{i-j} (\tau - 1)^{i-j} \tau^j. \tag{7}$$

Définissons sur $\mathcal{L}^2([0,1])$ un produit scalaire $\langle \cdot, \cdot \rangle_{\mu,\kappa}^{(0,1)}$ avec la fonction de poids $\hat{w}_{\mu,\kappa}(\tau) = (1-\tau)^\mu \tau^\kappa$, notons $\forall\, g_1, g_2 \in \mathcal{C}[0,1]$,

$$\langle g_1, g_2 \rangle_{\mu,\kappa}^{(0,1)} = \int_0^1 \hat{w}_{\mu,\kappa}(\tau) g_1(\tau) g_2(\tau) d\tau. \tag{8}$$

Et la norme associée au polynôme orthogonaux décalé de Jacobi d'ordre i est donnée par: $\|P_i^{(\mu,\kappa)}\|_{\mu,\kappa}^2 = \frac{1}{2i+\mu+\kappa+1} \frac{\Gamma(\mu+i+1)\Gamma(\kappa+i+1)}{\Gamma(\mu+\kappa+i+1)\Gamma(i+1)}$, où $\Gamma(n)$ est la fonction Gamma (voir (Abramowitz et al., 1965) p. 255), avec $\Gamma(n) = (n-1)!$.

Le calcu sous la forme d'un différentiateur causal de Jacobi est présenté en détail dans (Liu et al., 2012a). Ici, nous donnons l'expression continue analytique du différentiateur causal de Jacobi, qui calcule la $n^{i\psi me}$ dérivée à l'instant

t_0, $\forall \xi \in [0,1], \forall t_0 \in I$,

$$D^{(n)}_{\kappa,\mu,T,q} x(t_0 - T\xi) = \frac{1}{(-T)^n} \int_0^1 Q_{\kappa,\mu,n,q,\xi}(\tau)\, x(t_0 - T\tau) d\tau, \tag{9}$$

avec $\mu, \kappa \in]-1, +\infty[$,

$$C_{\kappa,\mu,n,i} = \frac{(\mu+\kappa+2n+2i+1)\Gamma(\kappa+\mu+2n+i+1)\Gamma(n+i+1)}{\Gamma(\kappa+n+i+1)\Gamma(\mu+n+i+1)}, \tag{10}$$

$$Q_{\kappa,\mu,n,q,\xi}(\tau) = \hat{w}_{\mu,\kappa}(\tau) \sum_{i=0}^{q} C_{\kappa,\mu,n,i} P_i^{(\mu+n,\kappa+n)}(\xi) P_{n+i}^{(\mu,\kappa)}(\tau). \tag{11}$$

Enfin, nous remplaçons x dans (9) par x^ω pour obtenir $D^{(n)}_{\kappa,\mu,T,q} x^\omega(t_0 - T\xi)$ dans le cas bruité.

L'idée de ce différentiateur est d'utiliser une fenêtre d'intégration glissante pour estimer la valeur de $x^{(n)}$ pour chaque $t_0 \in I$ par $D^{(n)}_{\kappa,\mu,T,q} x(t_0 - T\xi)$ avec la valeur fixée $\xi \in [0,1]$, l'optimisation de ξ est donnée dans ((Mboup et al., 2007, 2009a)). Si $\xi \neq 0$, alors on a un retard de $T\xi$.

Il dépend d'un ensemble de paramètres:

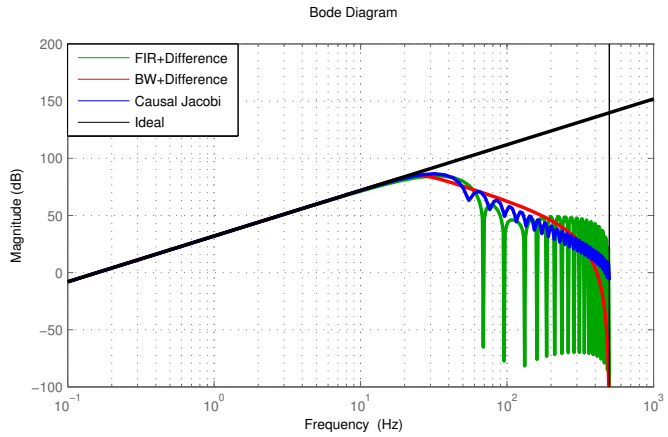
- $\kappa, \mu \in]-1, +\infty[$: les paramètres du polynômes de Jacobi,
- $q \in \mathbb{N}$: l'ordre du développement de la série Jacobi tronqué,
- $T \in D_{t_0}$: la longueur de la fenêtre glissante d'intégration,
- $\xi \in [0,1]$: le paramètre de retard $T\xi$.

L'analyse fréquentielle permet de caractériser le comportement du différentiateurs de Jacobi dans Fig. (2). On compare à un filtre FIR et à un filtre passe-bande composé d'un filtre passe-bas de Butterworth, suivi d'une dérivée par différence. Les résultats montrent que le différentiateur causal de Jacobi présente une phase linéaire au voisinage de la fréquence de coupure avec une propriété de filtrage passe bas en hautes fréquences intéressantes.

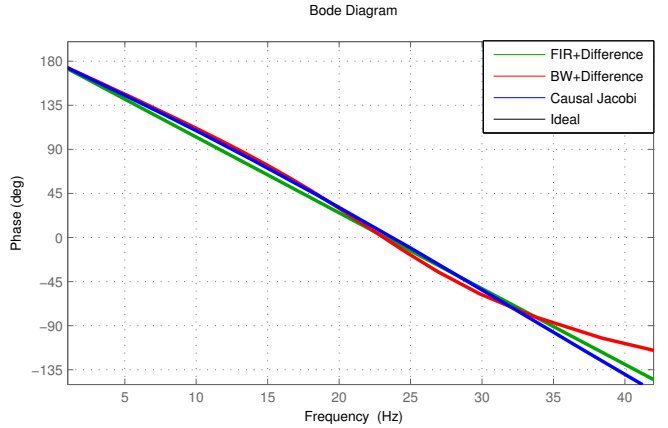
Comparaisons des méthodes d'identificaiton des paramètres dynamiques des robot

Dans le tableau (1), on rappelle les différentes techniques d'identification en robotique.

Une étude comparative des méthodes d'identification est réalisée sur un prototype de robot SCARA sans gravité à deux articulation à entraînement direct développé au laboratoire (IRCCyN) (Presse, 1994) comme indiqué sur Fig.



(a) Amplitude



(b) phase response

Figure 2 – Comparaison des réponses fréquentielles du différentiateur causal de Jacobi et des filtres classiques

(3). Le modèle dynamique dépend de 4 paramètres inertiels minimaux, et de 4 paramètres de frottement:

$$\mathbf{X} = [ZZ_{1R},\ ZZ_2,\ LMX_2,\ LMY_2,\ F_{v1},\ F_{s1},\ F_{v2},\ F_{s2}], \tag{12}$$

avec $ZZ_{1R} = ZZ_1 + M_2L^2$.

Résultats d'identification

Pour étudier les erreurs systématiques dues à la distorsion des différentes filtres, on simule le modèle du robot sans bruit avec la valeurs de $\mathbf{X}$ en unités SI : $\mathbf{X}$ = [3.5 0.06 0.15 0.1 0.3 0.4 0.2 0.12]. Les résultats sont présentés dans la Fig. (4). L'erreur systématique sur chaque paramètre, est calculée avec $e_{\hat{X}ir}$ % = 100 ×

Modèle d'identification	Différentiateurs	Moindre carré
Modèle dynamique (IDIM)	différence & Butterworth	OLS [1]
Modèle de puissance	différentiateurs centrale de Jacobi	WLS [2]
Modèle d'énergie		
Modèle de puissance avec fonctions modulatrices		

Table 1 – méthodes pour l'identification des paramètres dynamiques des robots

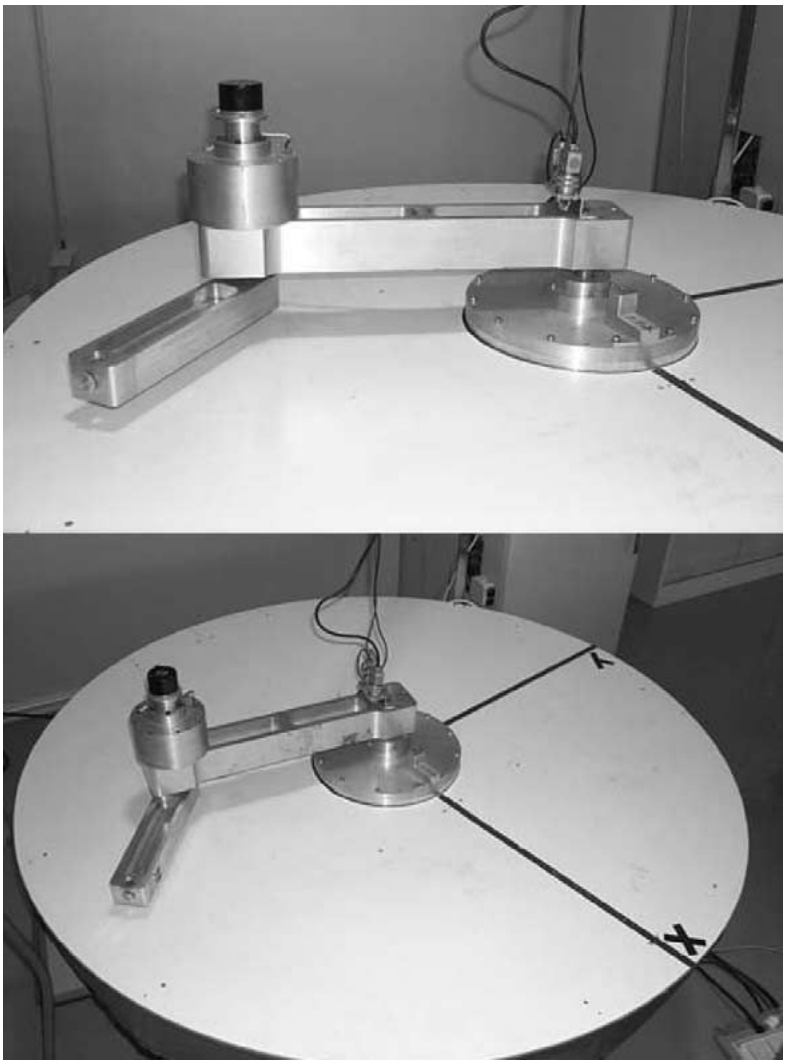


Figure 3 – 2R scara planar prototype robot (Lab IRCCyN)

$|\frac{Xi-\hat{X}i}{Xi}|$. Les résultats montrent que la méthode avec le différentiateur centré de Jacobi a moins d'erreurs systématiques plus faibles que les autres. D'autre part, le modèle de puissance avec les fonctions modulatrices a le résultat le plus proche des valeurs réelles. Ensuite, on réalise des essais d'identification

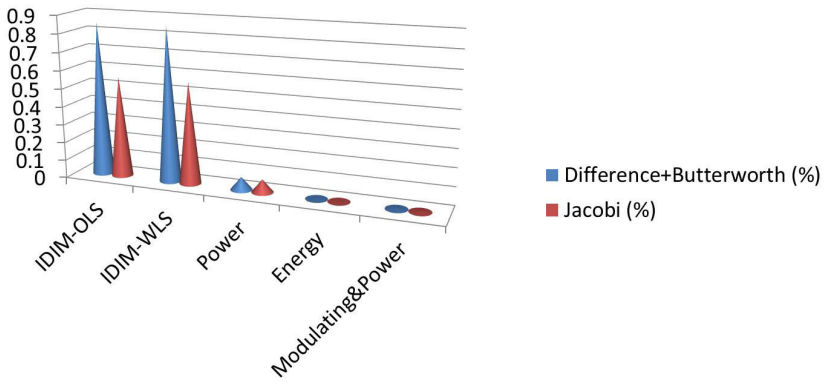


Figure 4 – Erreurs systématiques

sur le prototype du robot scara. Les résultats sont indiqués dans la Fig. (5). L'identification expérimentale montre :
modèle de puissance avec fonctions modulatrices = IDIM-WLS ≥ modèle de puissance ≥ modèle d'énergie ≥ IDIM-OLS .

Conclusion

Ce travail concerne l'identification des paramètres dynamiques des robots. Les contributions suivantes ont été réalisées:

- Proposition d'un **modèle de puissance avec fonctions modulatrices** qui évite la dérivation numérique des fonctions d'énergie.

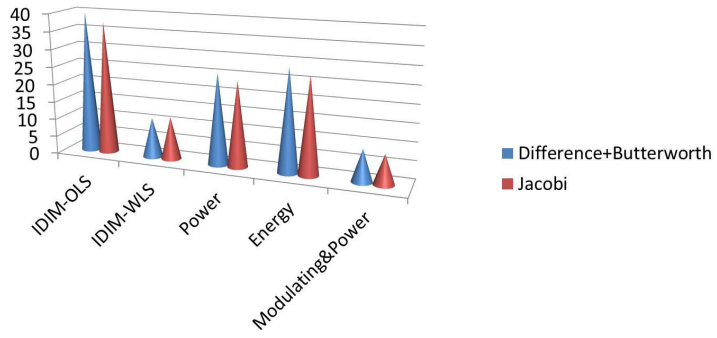
- Analyser dans le domaine fréquentiel des fonctions modulatrices, qui a permis de sélectionner les fonctions modulatrices de filtre passe-bande adapté à l'identification.

- introduction les **différentiateurs de Jacobi** et analyser pour la première fois, dans le domaine fréquentiel pour qualifier et quantifier leurs propriétés de filtrage. La majeur de la thèse a permis de comparer les differentiateurs de Jacobi en terme de distorsion et d'établissement des bruits hautes fréquences.

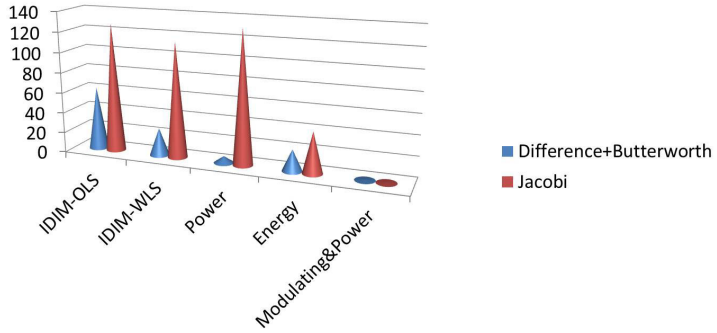
- **comparaison** entre les modèles différents d'identification, dynamic, de puissance d'énergie, des différentiateurs, des techniques de moindre carré pour l'identification des robots.

Un problème important dans l'identification des paramètres dynamiques des robots est de diminuer au maximum l'influence du bruit dans la dérivation des signaux. Dans ce travail, nous proposons des méthodes basées sur des techniques algébriques: utiliser le modèle de puissance avec les fonctions modulatrices ce qui supprime le calcul de l'accélération et le calcul de la dérivée des fonctions d'énergie; appliquer les différentiateurs de Jacobi pour obtenir une bonne estimation des dérivées d'ordre 2.

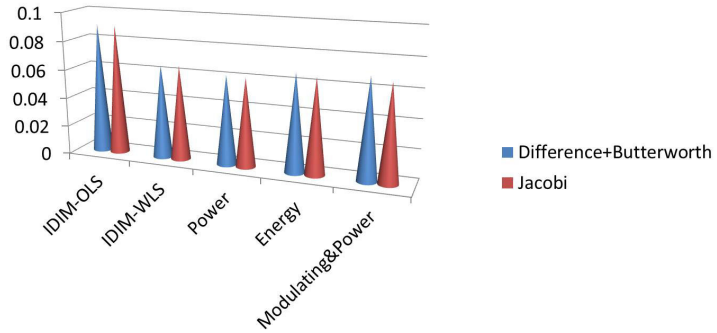
Les travaux futurs concernent une validation sur des robots plus complexes, qui ont un comportement non-linéaire accentué avec un grand nombre de paramètres, ce qui entraîne des difficultés pour l'identification.



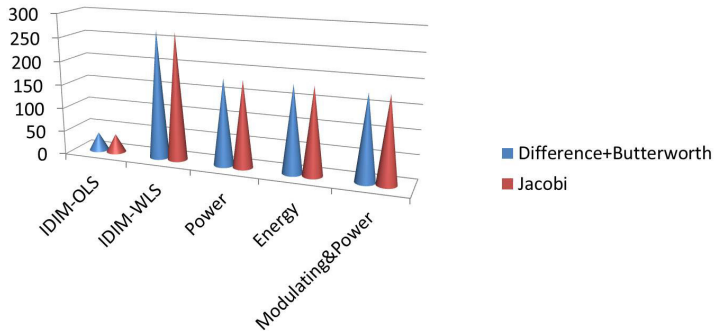
(a) Moyennes de l'écart type relatif (%)



(b) Résidus norme relative maximale $\max\left(\left|\frac{(Y-W\hat{\mathbf{X}})_j}{Y_j}\right|\right)$



(c) Résidus norme relative



(d) Conditionnement

Figure 5 – Comparaison

General Introduction

A robot is a mechanical or virtual artificial agent, usually an electro mechanical machine that is guided by a computer program or electronic circuitry. As a word, robot is drawn from an old Church Slavonic word, robota, for "servitude," "forced labor" or "drudgery." The word, which also has cognates in German, Russian, Polish and Czech, was a product of the central European system of serfdom by which a tenants rent was paid for in forced labor or service.

The first robot can be tracked back to the 4th century BC, when a wooden, mechanical steam-operated bird called The Pigeon was created by the Greek mathematician Archytas. While the research into the functionality and potential uses of robots did not grow substantially until the 20th century. Especially, fully autonomous robots only appeared in the second half of the 20th century and the robotics subject keeps prosperity and development. More recently, "robots" and the derived term "robotics" have come to represent the most modern engineering technologies for a myriad of functions ranging from artificial intelligence experiments and building automobiles to performing delicate surgical procedures.

Robotics are crossing disciplines of mechanical engineering, electrical engineering and computer science that deals with the design, construction, operation and application of robots, as well as the computer systems for their control, sensory feedback and information processing.

With the fast development of robot theory, people gradually became aware of the geometry and dynamic descriptions of the robot, with better and better precision in building mathematical model. Consequently, most of the advanced robot control schemes are proposed with application of the robot model, especially the robot dynamic models, in order to adjust the controller and the increase the control precision. Such control schemes include: computed torque control; predictive control; passivity control; adaptive control.

It requires a good knowledge of the robot dynamic model before implementation. Robot dynamic model can be developed from Newton-Euler method or Lagrange method. It describes the system dynamics in terms of current states,

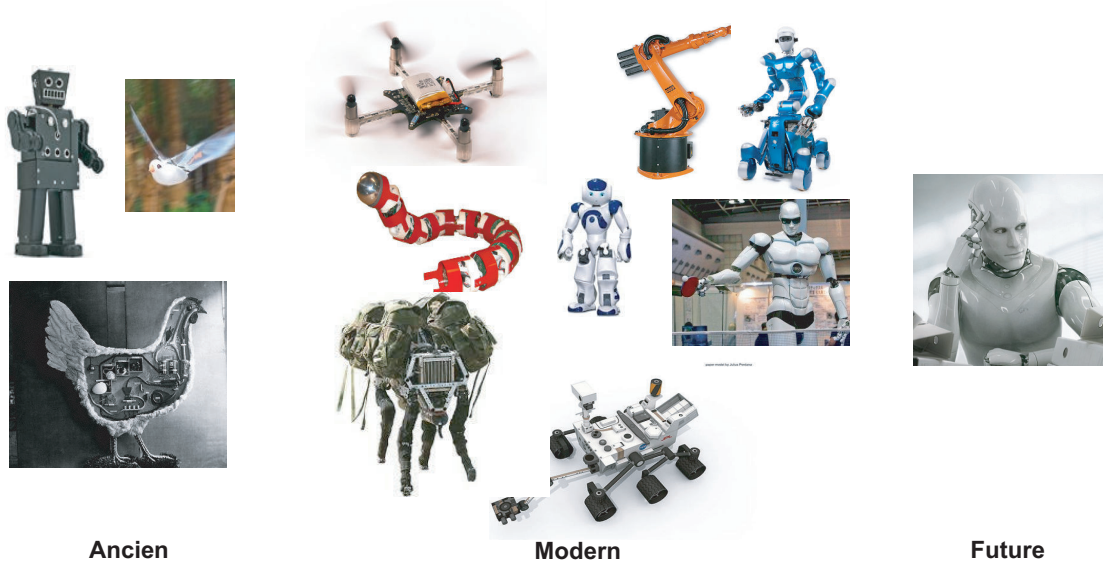


Figure 6 – Development of robots

such as position, velocity, acceleration and force/torque, as well the dynamic parameters. The dynamic parameters are constant at each instant and include the mass, inertia, first moments, joint friction for each robot link, and moment of inertia of the rotor and transmission system of actuators. As the key elements of the robot model, they need to be recognized before operation. However, for some of them, it is not likely to implement a measurement by the existing technology. Thus, the identification procedure is necessary and as a long-standing subject in the robotics.

In this work, we investigate the robot identification issue. The aim is to identify robot's dynamic parameters via a measurement of robot trajectory and torque/force information. Various identification methods have been developed based on different robot models, differentiation approaches, numerical tools, sensor feedback information or cross validation using closed loop simulation. Illustrating results are obtained in the past decades, while there still exist some challenges in the differentiation problem. In robot applications, the obtained measurements are usually noisy, which makes the derivative estimation process be ill-posed in the sense that a small error in measurement can induce a large error in the computed derivatives, specially for high order derivatives. The poor performance of reconstructing derivatives from the noisy measurement will directly induce some bias in the identification matrix, which results in huge errors in the identification results. Moreover, the highly nonlinear property of the robot systems and its large number of dynamic parameters also bring in difficulty in identification. Therefore, more researches and studies should be

carried out in this area.

The Project-team Non-A has developed a group of **algebraic differentiators**, called Jacobi differentiators. They provide the solution as an explicit integral formulae associated with finite signal information within a sliding integration time window. Because of their integration structure, they behave as nature filtering process and are robust to corrupt noises. Compared to other differentiators, Jacobi differentiators can be implemented immediately without prefiltering of the noisy signal due to their integral structure.

Meanwhile based on the algebraic conception, it is possible to transform the high order differential equation into lower order one. Associated with the **modulating functions**, the differential equation is rewritten in a simple expression which could eliminate certain high order terms by using some annihilating integral operator. The robot identification model can also apply this technique in order to avoid the terms with joint acceleration, which is usually badly estimated.

Objective of the thesis

The main objective of this thesis is to develop new techniques in robot dynamic parameters identification methods and compare them with the existing techniques. We propose new approaches in two aspects : reformulate the robot model using both the algebraic method and modulating functions; apply the newly developed Jacobi differentiators in robot differentiation problem and analyse the filtering property in frequency domain. In the end, we discuss if the detailed identification model is needed for control and based on the algebraic method, a simple and fast response adaptive controller is designed.

Outline of the thesis

Chapter 1 gives an overview of the existing approaches for serial robot identification process. The principle of the **identification** procedure is based on the analysis of the 'input/output' behavior of the robot following some planned motion and on estimating the parameters value by minimizing the difference between a function of the real robot variables and its mathematical model. According to the models, the most widely applied approach is based on **robot explicit dynamic model**, requiring the joint force/torque, position, velocity and acceleration information. The other models such as **robot energy model** and

robot power model require only the joint force/torque, position and velocity information. However they need an additional derivative operation on the implicit expression of velocity. As the dynamic parameters are linear with respect to the models above, most of the identification processes form an overdetermined system by a sequence of measurements and obtain the optimal solution using the least-square techniques. Besides, a parallel scheme to identify robot dynamic parameters by minimizing the output error from a closed loop simulation is presented.

Chapter 2 investigates the new identification method. It starts from an algebraic point of view, by which the order of differential equation can decrease using some annihilating operator. Based on this technique, the robot power model is transformed to an energy equation, that does not consider the acceleration variables and the implicit derivative operation. In this sense, the identification model avoids using the inaccurate acceleration information and introduce less errors in the results. Meanwhile, the modulating functions are implemented during the process. We will analyse their magnitude-frequency response and show that they have a low-pass **filtering property** for certain groups of the **modulating functions**.

Chapter 3 discusses the differentiation problem in robot identification issues. The principle and analysis of the causal and central **Jacobi differentiators** are introduced. As well, their frequency domain properties are analysed via a finite impulse response (FIR) filter point of view, indicating clearly their differentiation performance. Comparisons with other differentiators are made both in time domain and frequency domain.

Chapter 4 presents mainly the identification results in simulation and application of different approaches, in order to give a clear understanding of the advantages and draw-backs for each methodology. To be more specific, four different identification models (dynamic, power, energy identification models, modulating functions with power model approach), two differentiators (Jacobi differentiators and central difference with Butterworth filter) and three least square techniques (Ordinary, Weighted, Iterative least squares techniques) are compared.

Chapter 5 studies from the control aspect, some new ultra local robot model, which represents well the robot dynamics. Here "ultra local" means on a small time window. An adaptive controller is proposed so that by estimating the states of the simplified model, the corrupt changes of the robot are detected and updated within a short time window, which offers better dynamic performance of the control scheme. Moreover, the estimation model jumps over the

traditional differentiation problem in robotics, and only needs the joint force/-torque and position information. In the 2-DOF (degrees of freedom) planar robot simulation test, the estimation window is reduced to 0.1 second in presence of noise.

Finally, the thesis is completed with some conclusion and perspectives.

Chapter 1

Overview of the robot identification problem

Accurate dynamic models of robots are required in most advanced control schemes formulated in recent literature (Khatib, 1987; Piltan et al., 2012; Slotine et al., 1987). The precision, performance, stability and robustness of these schemes depend on, to a large extent, the accuracy of the **dynamic parameters**. Adaptive and robust control scheme can tolerate some error in the dynamic parameters, while other schemes designed to achieve perfect feedback linearization, such as computed torque control, assuming precise knowledge of the dynamic parameters. In this sense, the precise determination of the dynamic parameters is useful to most schemes and is crucial to some others. Furthermore, the dynamic parameters are necessary to simulate the robot dynamics.

However, accurate values of the dynamic parameters are typically unknown, even to the robot manufactures, and the measurement for some of them are practically not accessible. Thus, the **indirect identification approaches** are considered through the analysis of the 'input/output' behavior of the robot following some planned motion and on estimating the parameters value by minimizing the difference between a function of the real robot variables and its mathematical model. The identification problem turns out to be an optimization question which searches for the correct robot model with proper dynamic parameters.

The following parts present the principle of identification procedures and the related various techniques. According to the inputs that the identification model needs, there exists three **robot identification models**:

- **robot dynamic model** (Canudas de Wit et al., 1991; Gautier, 1986; Gautier et al., 2008, 2013; Gautier and Khalil, 1990; Gautier, Khalil, and Re-

strepo, 1995; Gautier, Vandanjon, et al., 2011; Hollerbach et al., 2008; Khalil and Dombre, 2004; Khosla et al., 1985; Lu et al., 1993);

- **robot energy model** (Gautier, 1996; Gautier and Khalil, 1988);
- **robot power model** (Gautier, 1997).

The robot dynamic model is the most widely implemented identification model. It establishes the dynamic equations at individual point along the trajectory. The advantages of this model include that it is easy to create the dynamic equations and it has a good excitation in the identification regression matrix, which means the regression always has a solution. By contrast, the dynamic model contains the acceleration variables, which are usually inaccurate computation using the position measurement. While the robot energy model and power model avoid using the acceleration data. Instead, the energy model applies an integral operation on the robot power equation and the power model make use of the differential equation of the energy part. In the following, we will present these identification process. Note that for the rest part of the thesis, we denote the bold mathematical symbols for the vectors or matrix.

1.1 Inverse dynamic identification model with LS

Inverse dynamic identification model with LS method, also named IDIM-LS, is the most applied identification model for robot identification. In order to construct it, first we need to deduce the inverse dynamic model. The dynamics of a rigid robot composed of n moving links calculates the motor torque vector $\mathbf{\Gamma_m}$ as a function of the state variables and their derivatives. It can be deduced from the following **Lagrangian formulation**:

$$\mathbf{\Gamma_m} = \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\mathbf{q}}}\right) - \frac{\partial L}{\partial \mathbf{q}} + \mathbf{\Gamma_f}, \tag{1.1}$$

where $\mathbf{q}$, $\dot{\mathbf{q}}$ are the $(n \times 1)$ vectors of generalized joint positions and velocities, L is the Lagrangian of the system defined as the difference between the **kinetic energy** $E(\mathbf{q}, \dot{\mathbf{q}})$ and the **potential energy** $U(\mathbf{q})$. $E = \frac{1}{2}\dot{\mathbf{q}}^{\mathbf{T}}\mathbf{M}(\mathbf{q})\dot{\mathbf{q}}$, where $\mathbf{M}(\mathbf{q})$ is the $(n \times n)$ robot inertia matrix. $\mathbf{\Gamma_f}$ is the friction torque which is usually modelled at non zero velocity as $\Gamma_{fj} = F_{sj}sign(\dot{q}_j) + F_{vj}\dot{q}_j + \Gamma_{offj}$, where $\dot{q}_j$ is the velocity of joint j, $sign(x)$ denotes the sign function. F_{vj}, F_{sj} are the viscous and Coulomb friction coefficients of joint j, Γ_{offj} is an offset parameter which is the dis-symmetry of the Coulomb friction with respect to the sign of the velocity

and is due to the current amplifier offset which supplies the motor (Gautier et al., 2013).

Develop Eq. (1.1) by replacing L with $E - U$, and it becomes the inverse dynamic model:

$$\mathbf{\Gamma_m} = \mathbf{M(q)\ddot{q}} + \mathbf{C(q,\dot{q})\dot{q}} + \mathbf{Q(q)} + \mathbf{\Gamma_f}, \tag{1.2}$$

where $\mathbf{\ddot{q}}$ is the $n \times 1$ vector of joint acceleration, $\mathbf{M(q)}$ is the $n \times n$ symmetric and positive definite inertia matrix, $\mathbf{C(q,\dot{q})\dot{q}}$ is the $n \times 1$ vector of Coriolis and centrifugal torques, $\mathbf{Q(q)}$ is the $n \times 1$ vector of gravity torques.

The inverse dynamic model is linear with respect to a set of standard dynamic parameters $\mathbf{X}_s$, because E, U and $\mathbf{\Gamma_f}$ are linear with respect to the dynamic parameters (Gautier, 1990). Thus, the model (1.2) can be rewritten as:

$$\mathbf{\Gamma_m} = \mathbf{D_s(q,\dot{q},\ddot{q})X_s} = \sum_{i=1}^{N_s} \mathbf{D_{s}}_i \mathbf{X_{s}}_i, \tag{1.3}$$

where N_s is the total number of the dynamic standard parameters, $\mathbf{D_s}$ is a $n \times N_s$ matrix, and $\mathbf{D_{si}}$ is the i_{th} column of $\mathbf{D_s}$, $\mathbf{X_{si}}$ is the i^{th} element of $\mathbf{X}_s$. $\mathbf{X}_s$ is the vector of standard dynamic parameters:

$$\mathbf{X_s} = \left[\mathbf{X_s^1}^T \ \mathbf{X_s^2}^T \ \cdots \ \mathbf{X_s^n}^T \right],$$

with $\mathbf{X_s^j}^T$ is the dynamic parameters of joint and link j:

$$\mathbf{X_s^j} = [XX_j \ XY_j \ XZ_j \ YY_j \ YZ_j \ ZZ_j \ MX_j \ MY_j \ MZ_j \ M_j \ IA_j F_{vj} \ F_{sj} \ \Gamma_{offj}]^T,$$

where $XX_j \ XY_j \ XZ_j \ YY_j \ YZ_j \ ZZ_j$ are the six components of the inertia matrix of link j; $MX_j \ MY_j \ MZ_j$ are the three components of the first moments; M_j is the mass of link j, IA_j is the total inertia moment for rotor actuator and gears of actuator j; F_{vj}, F_{sj}, Γ_{offj} are the viscous, Coulomb and offset friction parameters of joint j.

According to (Gautier and Khalil, 1990; Mayeda et al., 1990), the set of standard dynamic parameters can be simplified into a set of base inertial parameters, which are the minimum parameters that can be used to describe the robot dynamics. These base parameters are obtained from the standard inertial dynamic parameters by eliminating those that have no effect on the dynamic model and by regrouping those in linear relations. In (Gautier, 1991), symbolic and numerical solutions are presented for any open or closed chain robot ma-

nipulator to get a minimal dynamic model:

$$\mathbf{\Gamma_m} = \mathbf{D}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\mathbf{X}, \tag{1.4}$$

where $\mathbf{X}$ is the $N_b \times 1$ vector of base parameters.

1.1.1 Case study: the 2R scara planar prototype robot of IRCCyN

To illustrate the construction of robot dynamic model, here we present a two joint scara planar robot, called 2R robot for short. As shown in Fig. 1.1, the robot geometry is described in table (1.1) using the modified Denavit and Hartenberg notation (DHM) method, with

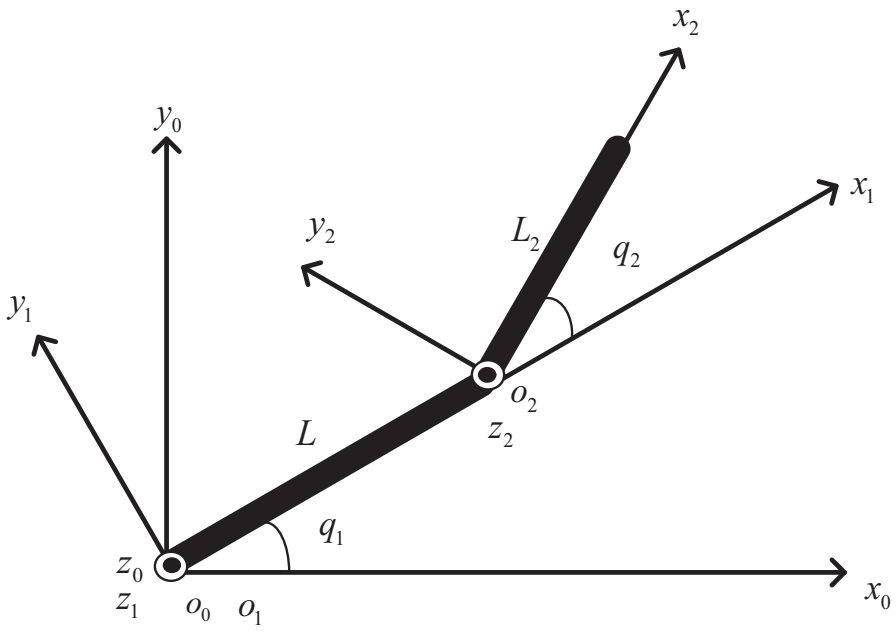
- j denotes the j^{th} joint,
- σ_j denotes the type of joint, 0 for revolute joint, 1 for prismatic joint,
- α_j is the angle between z_{j-1} and z_j about x_{j-1},
- d_j is the distance between z_{j-1} and z_j along x_{j-1},
- θ_j is the angle between x_{j-1} and x_j about z_j,
- r_j is the distance between x_{j-1} and x_j along z_j,
- q_1 and q_2 are joint position for joint 1 and joint 2 respectively,
- L is the length of the first robot link, L_2 is the length of the second robot link.

j	σ	α	d	θ	r
0	0	0	0	q_1	0
1	0	0	L	q_2	0

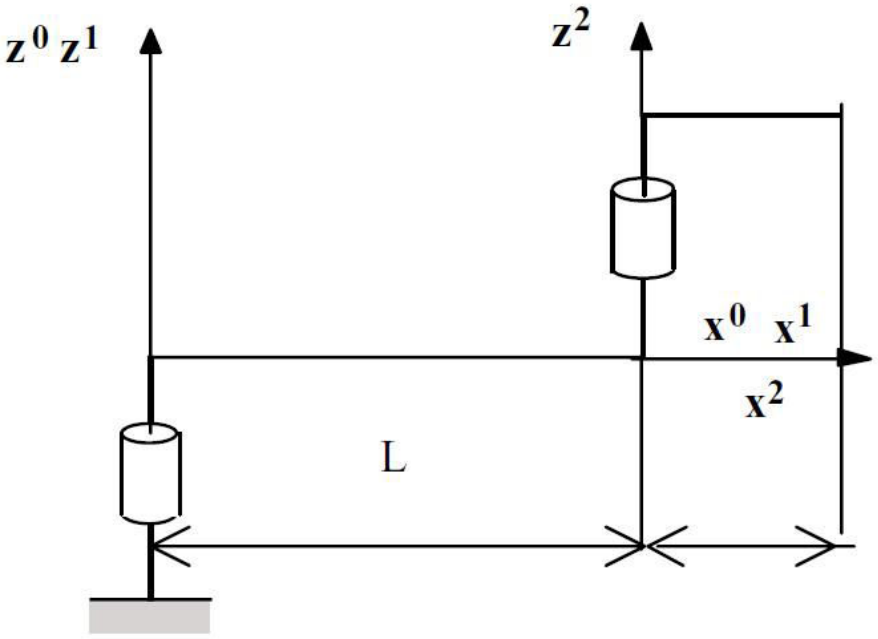
Table 1.1 – Modified Denavit and Hartenberg notation presentation of 2R scara planar robot

Before regrouped into base parameters, there exist 11 standard inertia parameters for each joint, such as in table (1.2):

Notice the special geometric configuration and apply the regrouping rule (Gautier, 1990) on the 2R robot, and it can be concluded as:



(a) 2R scara planar robot



(b) Frame and joint variables

Figure 1.1 – DHM frame of 2R scara planar robot

XX_j	XY_j	XZ_j	YY_j	YZ_j	ZZ_j	MX_j	MY_j	MZ_j	M_j	IA_j

Table 1.2 – Standard inertia parameters of 2R scara planar robot

- joint 1 and joint 2 are direct drive so that the link 1 and 2 are attached to the rotors of motor 1 and 2. Then, the inertia moment ZZ_1 and ZZ_2 are the inertia moment of links plus the inertia moment of rotors IA_1 and IA_2 respectively, so that in the following $IA_j = 0$;
- joint 2 is revolute: YY_2, MZ_2, M_2 can be regrouped with other dynamic parameters of link 2 and 1;
- the axe of joint 2 is parallel to that of joint 1 : XX_2, XY_2, XZ_2, YZ_2 can be eliminated;
- joint 1 is revolute, and the axe is along the gravity direction: only the inertia parameter ZZ_1 is considered;
- for joint 1, ZZ_1 is grouped with M_2 using the following relation $ZZ_{1R} = ZZ_1 + M_2L^2$.

It is shown in table 1.3 that there exist five base inertia parameters for 2R scara planar robot.

j	XX_j	XY_j	XZ_j	YY_j	YZ_j	ZZ_j	MX_j	MY_j	MZ_j	M_j	IA_j
1	0	0	0	0	0	ZZ_{1R}	0	0	0	0	0
2	0	0	0	0	0	ZZ_2	MX_2	MY_2	0	0	0

Table 1.3 – Base parameters of 2R direct drive scara planar robot

In order to compute the Lagrangian formulation, we need to calculate the kinetic energy $E(\mathbf{q}, \dot{\mathbf{q}})$ and the potential energy $U(\mathbf{q})$. Because the robot is moving in a horizontal plan, the potential energy keeps constant, which means the $n \times 1$ vector of gravity torques $\mathbf{Q}(\mathbf{q})$ is null. The kinetic energy calculates from the following relation

$$\begin{aligned} E = &\frac{1}{2}ZZ_2(\dot{q}_1 + \dot{q}_2)^2 + \frac{1}{2}ZZ_{1R}\dot{q}_1^2 \\ &+LMX_2\dot{q}_1\cos(q_2)(\dot{q}_1 + \dot{q}_2) - LMY_2\dot{q}_1\sin(q_2)(\dot{q}_1 + \dot{q}_2). \end{aligned} \tag{1.5}$$

The symmetric (2×2) inertia matrix $\mathbf{M}(\mathbf{q})$ can be deduced from the kinetic energy

$$E = \frac{1}{2}\dot{\mathbf{q}}^T\mathbf{M}(\mathbf{q})\dot{\mathbf{q}}. \tag{1.6}$$

From equation 1.5 and 1.6 we can calculate

$$\begin{aligned}
M(1,1) &= ZZ_{1R} + ZZ_2 + 2LMX_2\cos(q_2) - 2LMY_2\sin(q_2),\\
M(1,2) &= ZZ_2 + LMX_2\cos(q_2) - LMY_2\sin(q_2),\\
M(2,1) &= M(1,2),\\
M(2,2) &= ZZ_2.
\end{aligned} \tag{1.7}$$

The $n \times n$ matrix of Coriolis and centrifugal torques $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ comes from $\mathbf{M}(\mathbf{q})$ and E, which is given as

$$\begin{aligned}
C(i,j) &= \sum_{k=1}^{n} c_{i,jk}\dot{q}_k,\\
c_{i,jk} &= \frac{1}{2}\left[\frac{\partial M(i,j)}{\partial q_k} + \frac{\partial M(i,k)}{\partial q_j} - \frac{\partial M(j,k)}{\partial q_i}\right].
\end{aligned} \tag{1.8}$$

After calculation, we have the explicit form

$$\begin{aligned}
C(1,1) &= -\dot{q}_2(LMY_2\cos(q_2) + LMX_2\sin(q_2)),\\
C(1,2) &= -(\dot{q}_1 + \dot{q}_2)(LMY_2\cos(q_2) + LMX_2\sin(q_2)),\\
C(2,1) &= \dot{q}_1(LMY_2\cos(q_2) + LMX_2\sin(q_2))\\
C(2,2) &= 0.
\end{aligned} \tag{1.9}$$

For friction, because it has two links, the friction torque can be given as

$$\begin{aligned}
\Gamma_f(1) &= F_{v1}\dot{q}_1 + F_{s1}sign(\dot{q}_1) + \Gamma_{off1},\\
\Gamma_f(2) &= F_{v2}\dot{q}_2 + F_{s2}sign(\dot{q}_2) + \Gamma_{off2}.
\end{aligned} \tag{1.10}$$

In all, we have the explicit expression of $\mathbf{M}(\mathbf{q})$, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$, $\mathbf{Q}(\mathbf{q})$ and $\mathbf{\Gamma_f}$, which are all elements of the robot dynamic model. Moreover, because the link length L is unknown and it appears together with MX_2 and MY_2, thus we can group it into LMX_2 and LMY_2, where $LMX_2 = L \times MX_2$, $LMY_2 = L \times MY_2$. Finally, the base dynamic parameters of the 2R scara planar robot can be presented as

$$\mathbf{X_{2R}} = [ZZ_{1R},\ ZZ_2,\ LMX_2,\ LMY_2,\ F_{v1},\ F_{s1},\ \Gamma_{off1},\ F_{v2},\ F_{s2},\ \Gamma_{off2}], \tag{1.11}$$

Corresponding to the mathematical model, the experimental works are carried out a two joints planar direct drive prototype robot manufactured in the laboratory (IRCCyN) (Presse, 1994) as shown in Fig. (1.2), without gravity effect. The description of geometry is the same as in Fig. (1.1) and table (1.1).

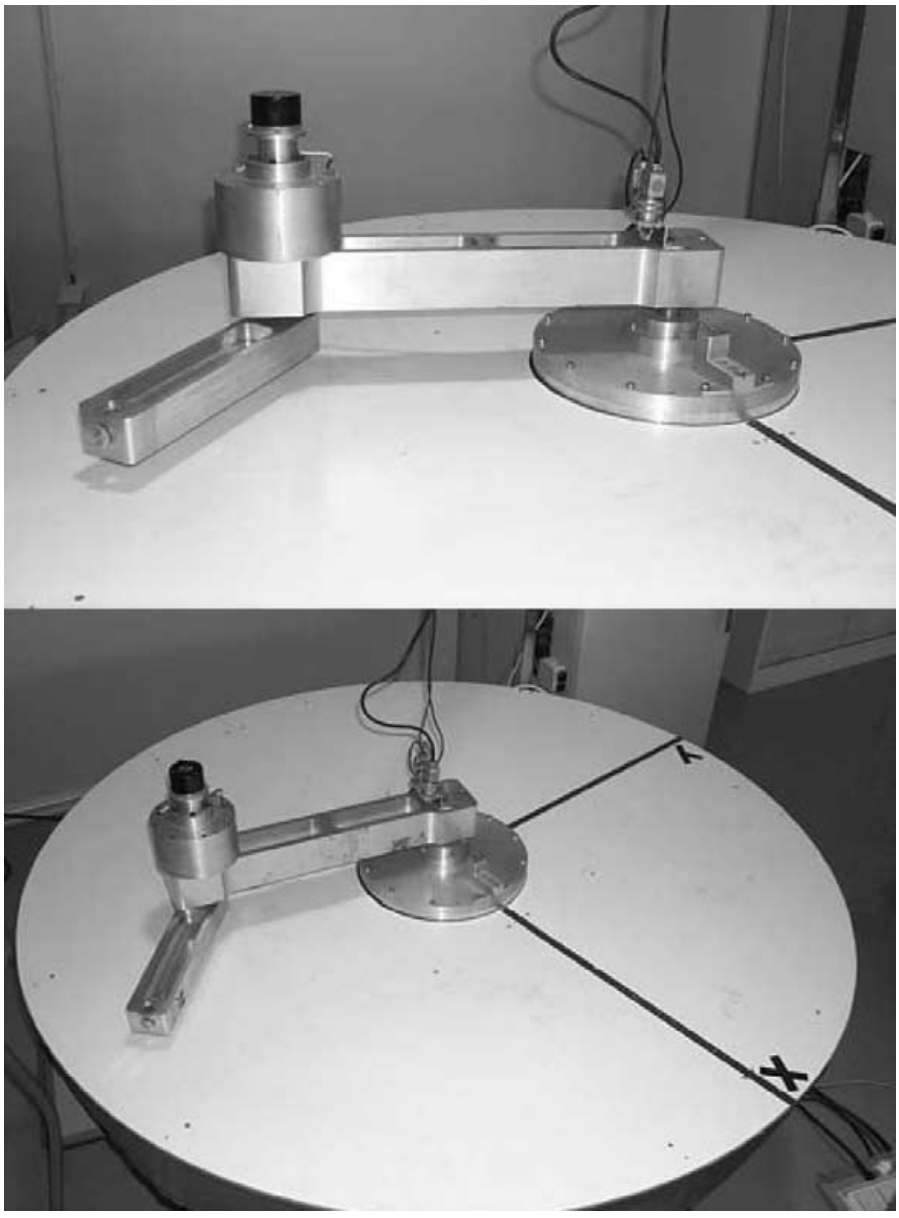


Figure 1.2 – 2R scara planar prototype robot (Lab IRCCyN)

The robot is directly driven by two DC permanent magnet motors supplied by PWM choppers. Recall the base dynamic parameters (1.11) of the 2R scara planar robot model, while in the prototype case we assume (the effect of friction offset parameter Γ_{off} are negligible) the dynamic model depends on eight minimal dynamic parameters, including four friction parameters:

$$\mathbf{X} = [ZZ_{1R},\ ZZ_2,\ LMX_2,\ LMY_2,\ F_{v1},\ F_{s1},\ F_{v2},\ F_{s2}], \tag{1.12}$$

with $ZZ_{1R} = ZZ_1 + M_2L^2$.

The robot motion is driven by a PD controller with a reference of a successive point to point trajectories using the 5^{th} order polynomial trajectory generator. The joint position $\mathbf{q}$ and torque $\mathbf{\Gamma}_m$ are collected at a 100 Hz sampling rate, where each torque Γ_m is calculated as

$$\Gamma_{mj} = G_{Tj} V_{Tj},$$

with G_{Tj} the drive chain gain which is considered as a constant in the frequency range of the robot dynamics. This trajectory has been calculated in order to obtain a good condition number (see in 1.1.4) of the observation matrix ($Cond(W) = 290$).

1.1.2 IDIM-LS

The **inverse dynamic identification model** is based on the measured or estimated data of $\mathbf{\Gamma_m}, \mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}$, which are collected during robot tracking of reference trajectories.

The principle is to establish the identification model (1.4) at a sufficient number of samples t_i, with $i = 1, \ldots, n_s$, satisfying $n \times n_s \gg N_b$, in order to get an over-determined linear system of $n \times n_s$ equations:

$$\mathbf{Y} = \mathbf{W}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\mathbf{X} + \rho, \tag{1.13}$$

where ρ is a noise, $\mathbf{Y}$ and $\mathbf{W}$ are the vector of torques and the observation matrix, respectively, which are defined as follows:

$$\mathbf{Y} = \begin{bmatrix} \mathbf{\Gamma_m}(1) \\ \vdots \\ \mathbf{\Gamma_m}(n_s) \end{bmatrix}, \mathbf{W} = \begin{bmatrix} \mathbf{D}(1) \\ \vdots \\ \mathbf{D}(n_s) \end{bmatrix}. \tag{1.14}$$

Take the 2R scara planar prototype robot referred in (1.1.1) for example, at

each sampling instant, we build the dynamic equivalence equation:

$$\mathbf{\Gamma_m} = \mathbf{D}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\mathbf{X}, \tag{1.15}$$

where $\mathbf{\Gamma_m} = \begin{bmatrix} \Gamma_{m1} \\ \Gamma_{m2} \end{bmatrix}$, and Γ_{m1}, Γ_{m2} are joint torques respectively, $\mathbf{D}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$ is a $n \times N_b$ observation matrix, which has the following expression

$$\begin{aligned}
&D(1,1) = \ddot{q}_1, \\
&D(1,2) = \ddot{q}_1 + \ddot{q}_2, \\
&D(1,3) = (2\ddot{q}_1 + \ddot{q}_2)\cos(q_2) - \dot{q}_2(\dot{q}_2 + 2\dot{q}_1)\sin(q_2), \\
&D(1,4) = -(2\ddot{q}_1 + \ddot{q}_2)\sin(q_2) - \dot{q}_2(\dot{q}_2 + 2\dot{q}_1)\cos(q_2), \\
&D(1,5) = \dot{q}_1, \\
&D(1,6) = sign(\dot{q}_1), \\
&D(1,7) = 0, \\
&D(1,8) = 0, \\
&D(2,1) = 0, \\
&D(2,2) = \ddot{q}_1 + \ddot{q}_2, \\
&D(2,3) = \ddot{q}_1 \cos(q_2) + \dot{q}_1 \dot{q}_1 \sin(q_2), \\
&D(2,4) = -\ddot{q}_1 \sin(q_2) + \dot{q}_1 \dot{q}_1 \cos(q_2), \\
&D(2,5) = 0, \\
&D(2,6) = 0, \\
&D(2,7) = \dot{q}_2, \\
&D(2,8) = sign(\dot{q}_2). \qquad (1.16) \\
& \qquad (1.17)
\end{aligned}$$

1.1.3 Identifiability of the dynamic parameters

The **dynamic parameters** are divided into three groups: fully identifiable, identifiable in linear combinations and completely unidentifiable. Consequently, the observation matrix **W** corresponding to this set of dynamic parameters could be rank deficient, with the fact that some columns of **W** are linearly dependent with respect to whatever $\mathbf{q}$, $\dot{\mathbf{q}}$, $\ddot{\mathbf{q}}$. In order to obtain a unique solution, a set of independent identifiable parameters, called **base dynamic parameters** or **minimum dynamic parameters**, need to be determined. The selection of base

or minimum dynamic parameters will regroup those are linearly dependent and will eliminate those who have no effect on the dynamic model.

In (Khalil and Dombre, 2004), the authors present both the symbolic methods and numerical methods to determine the base dynamic parameters. Furthermore, a robotic software named SYMORO+ developed by laboratory IRCCyN is proposed to resolve base dynamic parameters, and it has been open-source since 2014 (Khalil, Vijayalingam, et al., 2014) [1] . Actually, the determination of the base dynamic parameters is a prerequisite for identification procedure. It should be noted that the grouped values can be directly computed from the identification model, and reconstruct the robot dynamics with these grouped parameters.

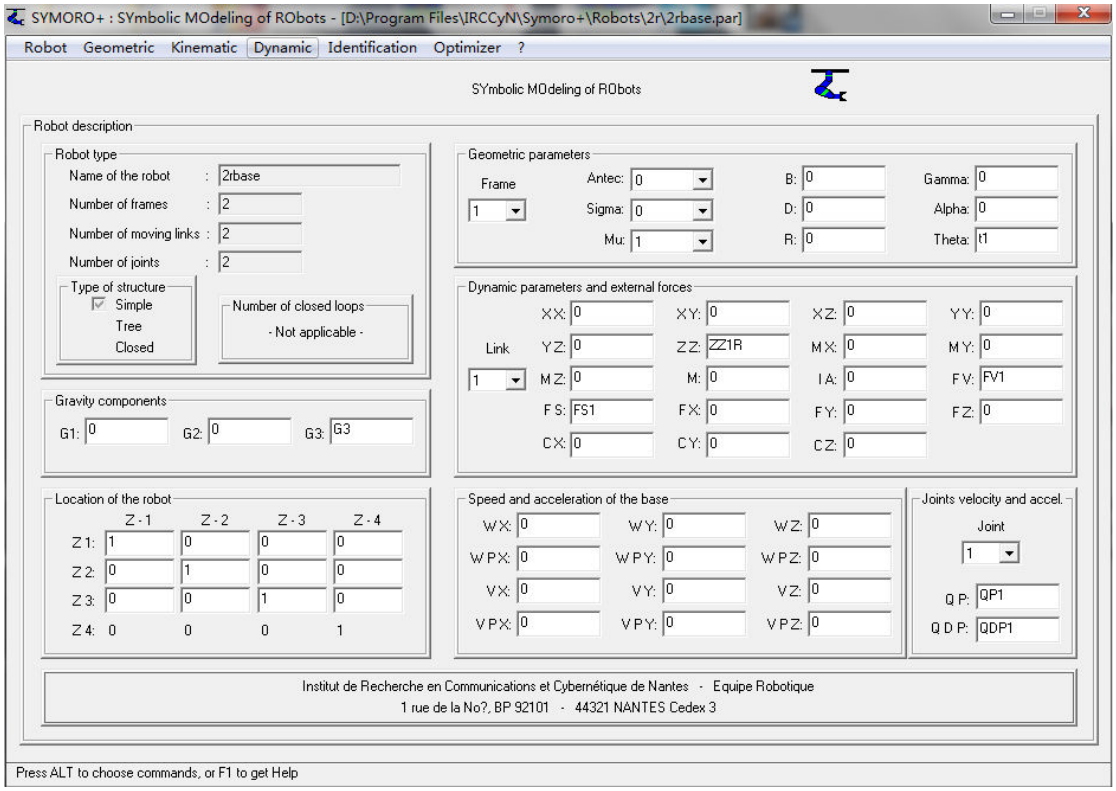


Figure 1.3 – SYMORO+ software

1.1.4 Excitation of the trajectory

The trajectory used in the identification should be carefully selected, in order to improve the least-squares estimation performance, such as convergence rate and the noise immunity. Some techniques are applied to choose the optimal trajectory, namely persistently exciting trajectory.

[1] https://github.com/symoro/symoro.

Persistently excitation condition (PE condition) A function $\omega : \mathbb{R}^+ \rightarrow \mathbb{R}^n$ is persistently excitating if there exist T, δ_1,$\delta_2 > 0$ such that

$$\delta_1 \mathbf{I}_n \leq \int_t^{t+T} \omega(\tau)^T \omega(\tau) d\tau \leq \delta_2 \mathbf{I}_n, \tag{1.18}$$

holds for all $t \geq 0$, where $\mathbf{I}_n$ is the identity matrix of order n.

This criterion can be applied to verify if the trajectory is well excited. As stated in (Antonelli et al., 1999; Gautier and Khalil, 1992), two schemes are usually used:

- calculation of a trajectory satisfying some optimization criteria (Gautier and Poignet, 2001; Gautier and Khalil, 1992; Presse and Gautier, 1993; Swevers, Ganseman, et al., 1997; Swevers, Verdonck, et al., 2007);
- utilization of the sequential sets of special motions, where each motion will excite certain dynamic parameters. As certain parameters are already identified with respect to the global problem, the exciting trajectory is easier to find.

Physically, finding this constraint is equivalent to finding an optimal trajectory that can excite most the identified parameters. Several criteria have been proposed in literature (Presse and Gautier, 1993; Siciliano et al., 2008). For example, here we minimize the condition number and maximizing the smallest singular value of the observation matrix $\mathbf{W}$ as in (Antonelli et al., 1999). Since the optimum trajectory will be executed on the manipulator, parameterizing the optimal trajectory is also an important step. Two most common types are the quintic polynomial trajectory (Antonelli et al., 1999) and periodic trajectory (Swevers, Verdonck, et al., 2007). The former is suitable for most of industrial manipulators which only accepts simple velocity command while the later targets the open-architecture controller which allows user to program an arbitrary trajectory. Here we consider the periodic trajectory as formulated in equation 1.19, which can be parameterized as a sum of finite Fourier series:

$$q_i(t) = q_{i0} + \sum_{j=1}^{N} [a_{ij} \sin(j\omega_f t) - b_{ij} \cos(j\omega_f t)], \tag{1.19}$$

$$\dot{q}_i(t) = \sum_{j=1}^{N} [a_{ij} j\omega_f \cos(j\omega_f t) + b_{ij} j\omega_f \sin(j\omega_f t)], \tag{1.20}$$

$$\ddot{q}_i(t) = \sum_{j=1}^{N} [-a_{ij} (j\omega_f)^2 \sin(j\omega_f t) + b_{ij} (j\omega_f)^2 \cos(j\omega_f t)], \tag{1.21}$$

where i denotes the i^{th} robot joint, ω_f is the fundamental frequency of the excitation trajectory and should not excite the un-modelled dynamics of the manipulator. Then, the problem of finding the optimal trajectory becomes determining the coefficients q_{i0}, a_{ij}, b_{ij} which minimize the following cost function (Presse and Gautier, 1993):

$$f(q_i(t)) = \lambda_1 cond(\mathbf{W}) + \lambda_2 \frac{1}{\sigma_{min}(\mathbf{W})}, \tag{1.22}$$

where the scalar λ_1 and λ_2 represent the relative weights between the condition number of the observation matrix $cond(\mathbf{W})$ and inverse of the minimum singular value $\frac{1}{\sigma_{min}(\mathbf{W})}$. Note that the condition number of the observation matrix [2] is defined as

$$cond(\mathbf{W}) = \frac{\sigma_{max}(\mathbf{W})}{\sigma_{min}(\mathbf{W})}. \tag{1.23}$$

The condition number measures how change in input is propagated to change in output. It has the following relation

$$\frac{\|\Delta \mathbf{X}\|}{\|\mathbf{X}\|} \leq con(\mathbf{W}) \frac{\|\Delta \mathbf{W}\|}{\|\mathbf{W}\|}. \tag{1.24}$$

1.1.5 Data processing for the inverse dynamic identification model

In real application, the measurements or estimations of $\mathbf{\Gamma_m}$, $\mathbf{q}$, $\dot{\mathbf{q}}$, $\ddot{\mathbf{q}}$ are corrupted with noise. Thus, the matrices $\mathbf{Y}$ and $\mathbf{W}$ are perturbed and the LS solution leads to a bias estimation. Because the matrix $\mathbf{W}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$ are highly non linear, it is not possible to get the analytical expression of the bias and the variance. To tackle this problem, two filtering processes should be applied:

- data filtering to decrease noise effect,
- closed loop identification for the tracking the persistently excited trajectories.

$(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$ must be pre-filtered by a low-pass filter $F_q(s)$, with s the derivative operator in order to eliminate high frequency noise. Usually, $\dot{q}, \ddot{q}$ are processed using the product sF_q and s^2F_q on the position respectively. In practice, this

[2] The optimization problem consists of determining the trajectory, which provides a condition number of $\mathbf{W}$ that is close to 1;

process can be carried out using a central difference algorithm to obtain the time derivative. The optimal filter $F_q(s)$ should have a flat amplitude response without phase shift in the range $[0\ \omega_c]$, where the cutoff frequency $\omega_c > (10 \times \omega_{dyn})$, with ω_{dyn} is the bandwidth of the joint position closed loop (Gautier, 1997). Meanwhile, the torque $\mathbf{\Gamma_m}$ is perturbed by high frequency torque ripple from joint drive chain in the closed loop control. Hence, it has to be filtered. Then, $\mathbf{\Gamma_m}$ and $\mathbf{D}(\mathbf{q_{f_q}}, \dot{\mathbf{q}}_{\mathbf{f_q}}, \ddot{\mathbf{q}}_{\mathbf{f_q}})$ are both filtered and down-sampled through a decimate filter composed of a low-pass filter $F_p(s)$, where its cutoff frequency ω_{f_p} is approximated by $5 \times \omega_{dyn}$. The decimate rate n_d can be calculated with $n_d = \frac{\omega_c}{2\omega_{fp}}$ for a FIR filter and $n_d = 0.8 \times \frac{\omega_c}{2\omega_{fp}}$ for an IIR filter, where ω_c is the control rate.

Then, the new filtered linear system is obtained:

$$\mathbf{Y_{fp}} = \mathbf{W_{fp}}(\mathbf{q_{fq}}, \dot{\mathbf{q}}_{\mathbf{fq}}, \ddot{\mathbf{q}}_{\mathbf{fq}})\mathbf{X} + \rho_{\mathbf{fp}}. \tag{1.25}$$

Finally, we solve the LS problem via:

$$\hat{\mathbf{X}} = \mathbf{W}^{+}_{\mathbf{fp}}(\mathbf{q_{fq}}, \dot{\mathbf{q}}_{\mathbf{fq}}, \ddot{\mathbf{q}}_{\mathbf{fq}})\mathbf{Y_{fp}}. \tag{1.26}$$

1.1.6 Resolution of the inverse dynamic identification model

The identification handbooks provide a large variety of deterministic and stochastic methods to estimate $\mathbf{X}$ from the previous system of equations. Most of the schemes solves $\mathbf{X}$ by the **maximum likelihood approach** (Olsen et al., 2002; Swevers, Ganseman, et al., 1997) or the **least squares** (LS) methods. To our knowledge, good experimental results have been obtained by ordinary LS methods, such as those based on the SVD (singular value decomposition) or QR decomposition.

The LS methods minimize the Euclidean length of the residual vector $\hat{\mathbf{X}} = min\|\mathbf{WX} - \mathbf{Y}\|$, which obtains the optimal solution $\hat{\mathbf{X}} = \mathbf{W}^{+}\mathbf{Y}$, where $\mathbf{W}^{+} = (\mathbf{W}^{T}\mathbf{W}^{+})^{-1}\mathbf{W}^{T}$ is the pseudo-inverse matrix of $\mathbf{W}$. If $\mathbf{W}$ is of full rank, the LS solution $\hat{\mathbf{X}}$ is unique. The rank deficiency of $\mathbf{W}$ can come from two aspects:

- structural rank deficiency which is solved by considering the minimal set of parameters;
- data rank deficiency due to a bad choice of noisy samples ($\mathbf{q}$, $\dot{\mathbf{q}}$), which needs to satisfy the persistently excitation condition by a good planning of trajectories (Presse and Gautier, 1993).

It should be noted that the LS estimation is biased because the observation matrix $\mathbf{W}$ is random, and because $\mathbf{W}$ and ρ are realization of random and correlated variables. Furthermore, the elements of the matrix $\mathbf{W}$ are nonlinear functions in $\mathbf{q}$, $\dot{\mathbf{q}}$ and $\ddot{\mathbf{q}}$, which leads one to assume some statistical properties of the noise in order to evaluate the quality of the estimation process (bias and standard deviation). In the following, we give some variables to verify the accuracy of the values obtained using appropriate validation procedures.

Standard deviations $\sigma_{\hat{\mathbf{X}}_i}$ are estimated using classical and simple results from statistics, assuming the matrix $\mathbf{W}$ to be a deterministic one, and ρ is assumed to be a vector of unobserved zero mean independent identically distributed (iid) biases, with a standard deviation σ_ρ such that $C_{\rho\rho} = E(\rho^T\rho) = \sigma_\rho^2\mathbf{I}_r$, where E is the expectation operator. The variance-covariance matrix of the estimation error and standard deviations can be calculated by (de Larminat P et al., 1977):

$$C_{\hat{\mathbf{X}}\hat{\mathbf{X}}} = E[(\mathbf{X}-\hat{\mathbf{X}})(\mathbf{X}-\hat{\mathbf{X}})^T] = \sigma_\rho^2(\mathbf{W}^T\mathbf{W})^{-1},$$

where $\sigma_{\hat{\mathbf{X}}_i} = \sqrt{C_{\hat{\mathbf{X}}\hat{\mathbf{X}}ii}}$ is the diagonal coefficient of $C_{\hat{\mathbf{X}}\hat{\mathbf{X}}}$.

This interpretation has been proposed by Raucent (Raucent, 1990), but we should be careful with the results obtained because the corresponding assumptions are not verified.

An unbiased estimation of σ_ρ is used to get the relative standard deviation $\sigma_{\hat{\mathbf{X}}ri}$ by the expression:

$$\hat{\sigma}_\rho^2 = \frac{\|Y - W\hat{\mathbf{X}}\|^2}{r-c},\ \%\sigma_{\hat{\mathbf{X}}ri} = 100 \times \frac{\sigma_{\hat{\mathbf{X}}i}}{\hat{\mathbf{X}}i},$$

where r is the total number of equations and c is the number of unknown parameters.

The relative standard deviation can be used as a criterion to measure the quality of the identification value for each parameters. For example, if the relative standard deviation of a parameters is greater than ten times the minimum relative standard deviation value, this parameter can be regarded as poorly identified.

1.1.7 Numerical tools and evaluation

The solution to the LS problem $\mathbf{Y} = \mathbf{W}\mathbf{X} + \rho$ mainly depends on the quality of the observation matrix $\mathbf{W}$, the rank and condition number.

The condition number of the observation matrix is an effective tool to give a good prediction of the observability of the parameters Driels et al., 1990. The

optimal identification configuration provide a a condition number of the observation matrix close to 1. As a results, the condition number can be regarded as a criterion to optimize the reference trajectory in order to well excite the observability.

For decomposition of the observation matrix, there exist two methods associated with the LS techniques:

- Singular Value Decomposition (SVD) (Maciejewski et al., 1989),
- QR decomposition (Golub et al., 2012).

This two methods work well and can deal with rank deficient case.

Meanwhile, with different LS techniques, we can obtain different optimal solutions. In this work, we mainly focus on the three approaches:

- Ordinary LS (OLS),
- Weighted LS (WLS),
- Iterative LS (ILS) (Fong et al., 2011; Paige et al., 1982).

The OLS technique is the most widely applied one, which is the optimal linear unbiased estimator when the errors are of homogeneous variance and uncorrelated. The WLS technique is an improvement of the OLS, which by adding weight coefficients to different region, it has focusing accuracy at reliable region and discount the imprecision at the unreliable region. The ILS is a technique more reliable when the observation matrix is ill-conditioned. The numerical techniques discussed above will be presented in appendix.

1.2 Energy model identification

The dynamic model identification is obligatory to access the joint acceleration data, whose estimation value usually has relatively larger error compared to the estimation of joint position and velocity. In order to avoid this, the energy model is proposed first in (Gautier, 1990), which states that the total mechanical energy applied to the robot is equal to the sum of penitential and kinetic energy contained in the system. This model has several advantages:

- the dynamic parameters are linear with respect to the model, and the corresponding base dynamic parameters are the same as those of the dynamic model;

- the computation of the observation matrix is easier than dynamic model;

- the optimization for the exciting trajectory is easier than for the dynamic model.

Denote the total energy of the system also termed **Hamiltonian** by $H = E(\mathbf{q}, \dot{\mathbf{q}}) + U(\mathbf{q})$. According to the energy theory, we have the following relation:

$$dH = (\mathbf{\Gamma_m} - \mathbf{\Gamma_f})^T \dot{\mathbf{q}} dt. \tag{1.27}$$

The energy version comes from the integration of 1.27

$$\int_{t_a}^{t_b} (\mathbf{\Gamma_m} - \mathbf{\Gamma_f})^T \dot{\mathbf{q}} dt = H(t_b) - H(t_a) = \Delta H, \tag{1.28}$$

where t_a, t_b are time instants. Furthermore, the energy item H can be reformulated as:

$$H = [\mathbf{h_1} \ \dots \ \mathbf{h_n}] \begin{bmatrix} \mathbf{K_1} \\ \vdots \\ \mathbf{K_n} \end{bmatrix} = \mathbf{hK}, \tag{1.29}$$

where $\mathbf{K_j}$ is the vector of the base parameters of link j, and $\mathbf{h_j}$ is the vector of energy functions.

Meanwhile, the friction torque $\mathbf{\Gamma_f}$ is linear in friction parameters. Hence, the energy equation is linear in the dynamic parameters and it yields the energy identification model:

$$\int_{t_a}^{t_b} \mathbf{\Gamma_m}^T \dot{\mathbf{q}} dt = [\mathbf{\Delta h} \ \mathbf{\Delta f_v} \ \mathbf{\Delta f_s} \ \mathbf{\Delta \gamma_{off}}] \begin{bmatrix} \mathbf{K} \\ \mathbf{F_v} \\ \mathbf{F_s} \\ \mathbf{\Gamma_{off}} \end{bmatrix}, \tag{1.30}$$

with $\Delta\mathbf{h} = \mathbf{h}(t_b) - \mathbf{h}(t_a)$, $\Delta f_{vj} = \int_{t_a}^{t_b} \dot{q_j}^2 dt$, $\Delta f_{sj} = \int_{t_a}^{t_b} \dot{q_j} sign(\dot{q_j}) dt$, $\Delta \gamma_{offj} = \int_{t_a}^{t_b} \dot{q_j} dt$, $\Delta f_v = [\Delta f_{v1} \ \dots \ \Delta f_{vn}]$, $\Delta f_s = [\Delta f_{s1} \ \dots \ \Delta f_{sn}]$, $\Delta \gamma_{off} = [\Delta \gamma_{off1} \ \dots \ \Delta \gamma_{offn}]$.

Collecting the corresponding $(\mathbf{\Gamma_m}, \mathbf{q}, \dot{\mathbf{q}})$ at a sufficient number of intervals $[t_{a(i)}, t_{b(i)}]$, we obtain a linear overdetermined system of r equations with respect to the base dynamic parameters. Similar to the dynamic model, the energy identification model writes as:

$$\mathbf{Y_e}(\mathbf{\Gamma_m}, \dot{\mathbf{q}}) = \mathbf{W_e}(\mathbf{q}, \dot{\mathbf{q}})\mathbf{X} + \rho_\mathbf{e}, \tag{1.31}$$

where

$$\mathbf{Y_e} = \begin{bmatrix} y_e(1) \\ \vdots \\ y_e(r) \end{bmatrix}, \ \mathbf{W_e} = \begin{bmatrix} \Delta\mathbf{h(1)} & \Delta\mathbf{f_v(1)} & \Delta\mathbf{f_s(1)} & \Delta\gamma_{\mathbf{off}}\mathbf{(1)} \\ \vdots & \vdots & \vdots & \vdots \\ \Delta\mathbf{h(r)} & \Delta\mathbf{f_v(r)} & \Delta\mathbf{f_s(r)} & \Delta\gamma_{\mathbf{off}}\mathbf{(r)} \end{bmatrix},$$

$$\Delta\mathbf{h(i)} = \mathbf{h}[(\mathbf{q},\dot{\mathbf{q}})_{\mathbf{b_{(i)}}}] - \mathbf{h}[(\mathbf{q},\dot{\mathbf{q}})_{\mathbf{a_{(i)}}}], \ y(i) = \int_{t_{a(i)}}^{t_{b(i)}} \mathbf{\Gamma_m}^T \dot{\mathbf{q}} dt,$$
$$(\mathbf{q},\dot{\mathbf{q}})_{\mathbf{a_{(i)}}} = [\mathbf{q}(\mathbf{t_{a(i)}}), \dot{\mathbf{q}}(\mathbf{t_{a(i)}})], \ (\mathbf{q},\dot{\mathbf{q}})_{\mathbf{b_{(i)}}} = [\mathbf{q}(\mathbf{t_{b(i)}}), \dot{\mathbf{q}}(\mathbf{t_{b(i)}})].$$

From the 1.31, the identification for $\mathbf{X}$ is a least squares problem.

It should be noticed that this formulation needs a lower resampling of the energy function h at times $t_a(i)$, $t_b(i)$. More precisely the function h calculated at the acquisition rate ω_c (or control rate) must be low-pass filtered with $F_p(s)$ in order to avoid aliasing. So it is easier and natural to define the sampling times $t_a(i)$ and $t_b(i)$ from the decimate procedure with the ratio n_d, which results in choosing $t_a(i)$ and $t_b(i)$ with a constant value $t_b(i) - t_a(i) = n_d * 2\pi/\omega_c$. And the parallel decimate procedure $F_p(s)$ is applied on the energy model (Gautier, 1996, 1997).

To illustrate the procedure, we construct the energy model on the 2R scara planar robot. Recall the kinetic energy $E(\mathbf{q},\dot{\mathbf{q}})$ in equation 1.5, since the potential energy $U(\mathbf{q})$ is constant in this case, the Hamiltonian energy equals to kinetic energy

$$H = \frac{1}{2}ZZ_2(\dot{q}_1+\dot{q}_2)^2 + \frac{1}{2}ZZ_{1R}\dot{q}_1^2 + LMX_2\dot{q}_1\cos(q_2)(\dot{q}_1+\dot{q}_2) - LMY_2\dot{q}_1\sin(q_2)(\dot{q}_1+\dot{q}_2). \tag{1.32}$$

With the same base parameters $\mathbf{X}_{2R}$, it is easy to specify the energy identification model 1.31 with

$$W_e(i,1) = \frac{1}{2}\dot{q}_1^2 \,|_{t_{a_{(i)}}}^{t_{b_{(i)}}},$$
$$W_e(i,2) = \frac{1}{2}(\dot{q}_1+\dot{q}_2)^2 \,|_{ta_{(i)}}^{t_{b_{(i)}}},$$
$$W_e(i,3) = \cos(q_2)\dot{q}_1(\dot{q}_1+\dot{q}_2) \,|_{t_{a_{(i)}}}^{t_{b_{(i)}}},$$
$$W_e(i,4) = -\sin(q_2)\dot{q}_1(\dot{q}_1+\dot{q}_2) \,|_{t_{a_{(i)}}}^{t_{b_{(i)}}},$$

$$W_e(i,5) = \int_{t_{a_{(i)}}}^{t_{b_{(i)}}} \dot{q_1}^2 dt,$$
$$W_e(i,6) = \int_{t_{a_{(i)}}}^{t_{b_{(i)}}} \dot{q_1} sign(\dot{q_1}) dt,$$
$$W_e(i,7) = \int_{t_{a_{(i)}}}^{t_{b_{(i)}}} \dot{q_2}^2 dt,$$
$$W_e(i,8) = \int_{t_{a_{(i)}}}^{t_{b_{(i)}}} \dot{q_2} sign(\dot{q_2}) dt,$$
$$Y_e(i) = \int_{t_{a_{(i)}}}^{t_{b_{(i)}}} (\Gamma_{m1}\dot{q}_1 + \Gamma_{m2}\dot{q}_2) dt.$$

1.3 Power model identification

The **robot power model** also does not consider the joint acceleration data. Compared to robot energy model, it is more robust with respect to low-frequency noise. As shown in energy model, the integration appeared in equation 1.30 is actually infinite-gain filter at zero frequency. This causes an offset due to the small low-frequency errors, which can produce a large error. Instead, the power model use the differential equation form in order to overcome this problem. The power of the system at instant t writes as:

$$\dot{\mathbf{q}}^{\mathbf{T}}\boldsymbol{\Gamma}_{\mathbf{m}} = \frac{d}{dt}(H) + \dot{\mathbf{q}}^{\mathbf{T}}[\mathbf{diag}(\dot{\mathbf{q}})\mathbf{F_v} + \mathbf{diag}(\mathbf{sign}(\dot{\mathbf{q}})\mathbf{F_s} + \boldsymbol{\Gamma}_{\mathbf{off}}]. \tag{1.33}$$

Since $H = \mathbf{hK}$, the power model estimates the dynamic parameters $\mathbf{X} = [\mathbf{K}, \mathbf{F_v}, \mathbf{F_s}, \boldsymbol{\Gamma}_{\mathbf{off}}]$ in a linear equation:

$$\mathbf{Y_p}(\boldsymbol{\Gamma}_{\mathbf{m}}, \dot{\mathbf{q}}) = \mathbf{W_p}(\mathbf{q}, \dot{\mathbf{q}})\mathbf{X} + \rho_{\mathbf{p}}, \tag{1.34}$$

where

$$\mathbf{Y_p} = \begin{bmatrix} y_p(1) \\ \vdots \\ y_p(r) \end{bmatrix}, \ \mathbf{W_p} = \begin{bmatrix} \frac{d}{dt}(\mathbf{h}(1)) & W_{2p}(1) & W_{3p}(1) & W_{4p}(1) \\ \vdots & \vdots & \vdots & \vdots \\ \frac{d}{dt}(\mathbf{h}(r)) & W_{2p}(r) & W_{3p}(r) & W_{4p}(r) \end{bmatrix},$$

$$y_p(i)\dot{\mathbf{q}}^{\mathbf{T}}(\mathbf{i})\boldsymbol{\Gamma}_{\mathbf{m}}(\mathbf{i}), \ W_{2p}(i) = \dot{\mathbf{q}}^{\mathbf{T}}(\mathbf{i})\mathbf{diag}(\dot{\mathbf{q}}(\mathbf{i})),$$
$$W_{3p}(i) = \dot{\mathbf{q}}^{\mathbf{T}}(\mathbf{i})\mathbf{diag}(\mathbf{sign}(\dot{\mathbf{q}}(\mathbf{i}))), \ W_{4p}(i) = \dot{\mathbf{q}}^{\mathbf{T}}(\mathbf{i}).$$

As we did for the filtered dynamic model, we process the columns $\mathbf{Y}$, $\mathbf{X}_{2p}$, $\mathbf{X}_{3p}$ and $\mathbf{X}_{4p}$ using a low-pass filter $F_p(s)$, while the columns of $\mathbf{h}$ are filtered by $sF_p(s)$. In practice, this process can be carried out using a central difference algorithm to obtain the time derivative of $\mathbf{h}$, then, by using a decimation process $F_p(s)$ to filter all the model. Sufficient numbers of sampling data are needed in order to obtain a linear overdetermined system of r equations with respect to the base dynamic parameters. More precisely, with a 2R scara planar robot, we have

$$
\begin{aligned}
W_p(i,1) &= \frac{d}{dt}(\frac{1}{2}\dot{q}_1^2(i)),\\
W_p(i,2) &= \frac{d}{dt}(\frac{1}{2}(\dot{q}_1(i)+\dot{q}_2(i))^2),\\
W_p(i,3) &= \frac{d}{dt}(\cos(q_2)\dot{q}_1(i)(\dot{q}_1(i)+\dot{q}_2(i))),\\
W_p(i,4) &= \frac{d}{dt}(-\sin(q_2(i))\dot{q}_1(i)(\dot{q}_1(i)+\dot{q}_2(i))),\\
W_p(i,5) &= \dot{q_1}(i)^2,\\
W_p(i,6) &= \dot{q_1}(i)sign(\dot{q_1}(i)),\\
W_p(i,7) &= \dot{q_2}^2(i),\\
W_p(i,8) &= \dot{q_2}(i)sign(\dot{q_2}(i)),\\
Y_p(i) &= \Gamma_{m1}(i)\dot{q}_1(i)+\Gamma_{m2}(i)\dot{q}_2(i).
\end{aligned}
$$

1.4 Closed-loop output error identification (CLOE)

In (Gautier et al., 2013), the authors propose a new approach called **DIDIM (Direct and Inverse Dynamic Identification Models technique)**, which requires only the joint force/torque measurement. It is a closed-loop output error method where the usual joint position output in CLOE method is replaced by the joint force/torque. It is based on a closed-loop simulation of the robot using the direct dynamic model, the same structure of the control law, and the same reference trajectory for both the actual and the simulated robot. And the optimization is to minimize the 2-norm of the error between the actual force/-torque and the simulated force/torque.

The identification scheme is illustrated in Fig. (1.4). Tracking the reference trajectory $(\mathbf{q}, \dot{\mathbf{q}}_r, \ddot{\mathbf{q}}_r)$, the actual closed loop of robot produces the force/torque τ. The simulated closed loop robot implements the same control law to get the output force/torque $\tau_{\mathbf{ddm}}$, with joint position feedback $\mathbf{q}_{ddm}$ computed from

the direct dynamic model. Finally, the output error are minimized by optimizing the model dynamic parameters. This is a nonlinear least-squares problem which is dramatically simplified using the inverse dynamic model to obtain an analytical expression of the simulated force/torque.

Compared to the inverse dynamic identification model (IDIM) procedure, first of all, DIDIM has the advantage that it is still robust with respect to low sampling rate measurement. And DIDIM does not need well tuned band-pass filtering to calculate the velocity and acceleration. While IDIM will behave poor because of the amplitude distortion in the estimation of $\dot{\mathbf{q}}, \ddot{\mathbf{q}}$, with a central difference of $\mathbf{q}$ which is sampled at too low frequency.

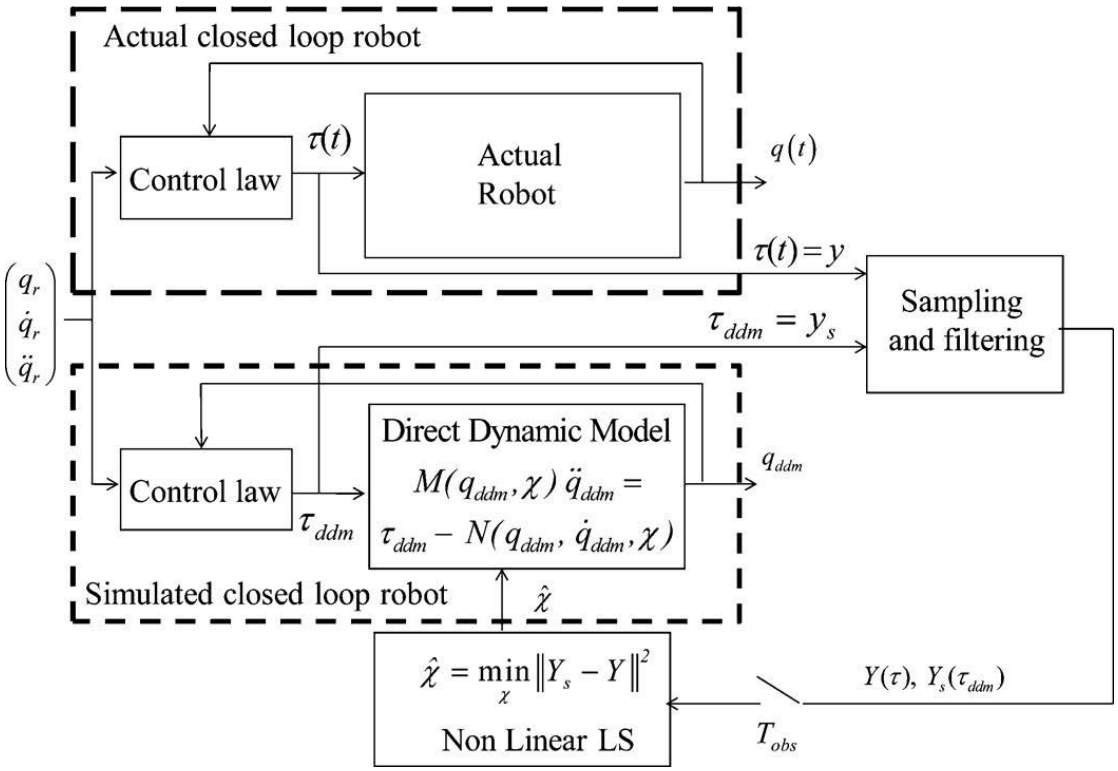


Figure 1.4 – DIDIM identification scheme

1.5 Payload Identification

When a payload is fixed on the terminal link of the robot, the robot dynamics is actual a combination of the payload dynamics and robot dynamics without charge. In order to identify the payload dynamics, there exist four approaches based on robot dynamic model, which are specified in (Khalil, Gautier, et al., 2007):

- identify the payload with dynamic parameters estimated without payload;
- identify globally both the robot parameters and the payload;
- compare the difference of dynamic parameters of robot identified without and with payload;
- use the difference between the joint torques before and after loading the robot on the same trajectory.

To have a conception on payload identification, we explain the first method mentioned above. Assume that $\mathbf{X}$ is already identified when the robot is without payload. By developing the dynamic model of the robot with payload, the identification model becomes as:

$$\mathbf{Y_T} = \mathbf{W_E}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\mathbf{X} + \mathbf{W_L}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\mathbf{X_L} + \rho, \tag{1.35}$$

where
$\mathbf{Y_T}$: vector of force/torque when robot is with payload,
$\mathbf{W_E}$: the observation matrix of robot dynamic parameters without payload,
$\mathbf{X}$: the base dynamic parameters of the robot without payload,
$\mathbf{X_L}$: the 10×1 vector of the inertial parameters of the payload,
$\mathbf{W_L}$: the observation matrix corresponding to the load inertial parameters.

Thus, the load inertial parameters are estimated from (1.35) via:

$$\hat{\mathbf{X}}_\mathbf{L} = (\mathbf{W_L})^+(\mathbf{Y_T} - \mathbf{W}\mathbf{X}). \tag{1.36}$$

1.6 Problems in robot identification

Previously, we present different identification models, where the regressors give the unique and accurate solution in presence of no disturbances. While in real applications, there always exist some errors in the sampling and filtering procedures. As known, a small noise in the measurement can induce large error in the derivatives, especially for high order derivatives. Thus, the joint velocity and acceleration can be inaccurately estimated if the noise component is not well filtered. Besides, the robot identification model are highly nonlinear with respect to the joint position, velocity and acceleration, so that the small error part can be amplified and cause large bias in the final results.

In order to obstacle this problem, intuitively we need to avoid using the derivatives containing large noise or provide the derivation with more precision. Starting from this point, in chapter 2 we will present the identification method based on power model and modulating functions, which allows the elimination of the joint acceleration term, the guaranty of the rank efficiency of the observation matrix and the low-pass filtering effect on the observation matrix; in chapter 3, we will emphasize the differentiation techniques and discuss the newly developed algebraic Jacobi differentiators, by analysing their frequency magnitude response.

Chapter 2

Robot identification using power model and modulating functions

In order to avoid using noisy computation of acceleration data, the energy model and power model are considered as approaches to describe the robot dynamics for identification purpose. However the energy model computes one scalar equation from a period of time, thus in some sense it will lose some diversity in information of the observation matrix and in consequence decrease the identifiability. The power model applies a derivative operation on the energy terms and in fact it still contains the second order derivation.

Inspired by algebraic method, modulating functions are combined with robot power model to replace the derivative operator in the power model by an integral. In (Co and Ydstie, 1990; Daniel-Berhe et al., 1998; Preisig et al., 1993), authors also applied modulating functions in system identification. Integration with modulating functions can decrease the order of a differential system and is in nature a filtering process. These make this transformation interesting in several applications, in particular for parameter identification propose.

There are several advantages of modulating functions: first it avoids the computation of acceleration; then it replaces the derivative operation by an integration operation with modulating functions, which have a nature low-pass filtering; lastly by selecting different parametrization of modulating functions and choosing different integral intervals, the identifiability property can be recovered.

For years many authors have focused on the choice of different modulating functions types because they have different performances with respect to noise. They are looking for certain kind of modulating functions which is adapted in particular application. Several groups are listed here, such as sinusoid modu-

lating functions, polynomial modulating functions, Hermite functions (Jordan et al., 1986), Fourier modulating functions (Pearson et al., 1985), Hartley modulating functions (Unbehauen et al., 1997), (Fedele and Coluccio, 2010) and spline-type functions (Fedele, Picardi, et al., 2009).

In the following part, we will introduce the modulating functions and their applications on decreasing the order of differential equations. Furthermore, analysis on frequency magnitude response is done with respect to different groups of modulating functions, so that certain groups are chosen for the identification procedure. At the end, two robot identification applications are carried out with this type of identification model.

2.1 Modulating functions

Let $l \in \mathbb{N}^*$, $T \in \mathbb{R}_+^*$, and g be a function satisfying the following properties:

$$\begin{aligned} (P_1) &: \ g \in \mathcal{C}^l([t_a, t_b]); \\ (P_2) &: \ g^{(i)}(t_a) = 0, \ for \ i = 0, 1, ..., l-1; \\ (P_3) &: \ g^{(i)}(t_b) = 0, \ for \ i = 0, 1, ..., l-1, \end{aligned} \tag{2.1}$$

where $\mathcal{C}^l([t_a, t_b])$ refers to the set of functions being $l-$times continuously differentiable on $[t_a, t_b]$ with $l \in \mathbb{N}^*$. Then g is called l^{th} **order modulating function** on $[t_a, t_b]$.

Modulating functions transform a differential expression into a sequence of algebraic equations using noisy data signals. Their filtering property makes this method interesting in several real processes.

For example, assume that a relation has variable $x^{(s)}$ of differential order s, where x is the observe variable. We would like to use only the observed variable and have to change the relation involving $x^{(s)}$ "descending" its order of derivation. For this, we choose g a l^{th} order modulating function on time interval $[0, T]$ where $l \geq s$. Multiply g with $x^{(s)}$ and multi-integrate them on $[0, T]$. By partial integration, the variable $x^{(s)}$ decreases its order to x and modulating function g increases its order to $g^{(s)}$ which is analytically known.

$$\int_0^T g x^{(s)} = -\int_0^T \dot{g} x^{(s-1)} = \cdots = (-1)^s \int_0^T g^{(s)} x. \tag{2.2}$$

2.2 Studies of modulating functions

From the above example, the modulating functions allows to transform a differential expression into a sequence of algebraic equations. Moreover, the modulating functions method annihilates the effects of initial conditions and allows the direct use of noisy data signals (Co and Ungarala, 1997). From the filtering aspect modulating functions have low pass filtering property. These features make the modulating functions method desirable for use in several real processes.

2.2.1 Frequency Analysis

Frequency domain property of a modulating function can be analysed considering the integration operator as a filtering process. In (Chen et al., 2011; Collado et al., 2009), the authors also analyze the differentiator frequency domain property. In real computation, the numerical convolution $\int gx$ is actually a discrete operation with at sampling time interval Ts, which calculates the sum of discrete points of a signal x associated with the modulating function g. In discrete version it writes as $\sum_{i=1}^{N} g[i]x[i]$ with interval Ts. In this way the modulating functions can be discretized as a list of weighting coefficients. Moreover these weighting coefficients can be regarded as coefficients of a finite impulse response (FIR) filter with respect to a discrete system with sampling time Ts. By studying the frequency domain behavior of the FIR filter, we can extend the results to integration effect with modulating functions. In the following part we look at several modulating functions and discuss their filtering property in order to give some clue in the choice for applications.

The proposed functions $g_\ell(t)$ are K order modulating functions on interval $[0, T]$, with K the order desired to decrease. They satisfy the two-point boundary conditions, $g_\ell^{(i)}(0) = 0$, and $g_\ell^{(i)}(T) = 0$, when $i = 0, 1, \ldots, K-1$.

(1) Sinusoid based modulating functions (SMF): the sinusoid function value reaches 0 per half period, according to this property, propose $g_\ell(t) = \sin^\ell(\frac{\pi}{T}t)$, with $\ell \in \mathbb{R}$.

(2) Jacobi modulating functions (JMF): this group of functions are a combination of Jacobi polynomials which equal to 0 at each end of interval. Remember the order of each polynomial is larger than $K-1$ and propose $g_\ell(t) = t^{\ell_1}(t-T)^{\ell_2}$, with $\ell_1,\ \ell_2 \in \mathbb{R}$ and $\ell_1,\ \ell_2 > K-1$.

(3) Fourier modulating functions (FMF): as known complex exponential function $e^{ix} = \cos x + i \sin x$ is a periodic function which reaches 1 per period.

Based on this, the fourier modulating functions can be written as $g_\ell(t) = e^{-i\alpha\ell}(e^{-i\frac{2\pi}{T}t} - 1)^K$, where α is a tuning parameter and $\ell \in \mathbb{R}$.

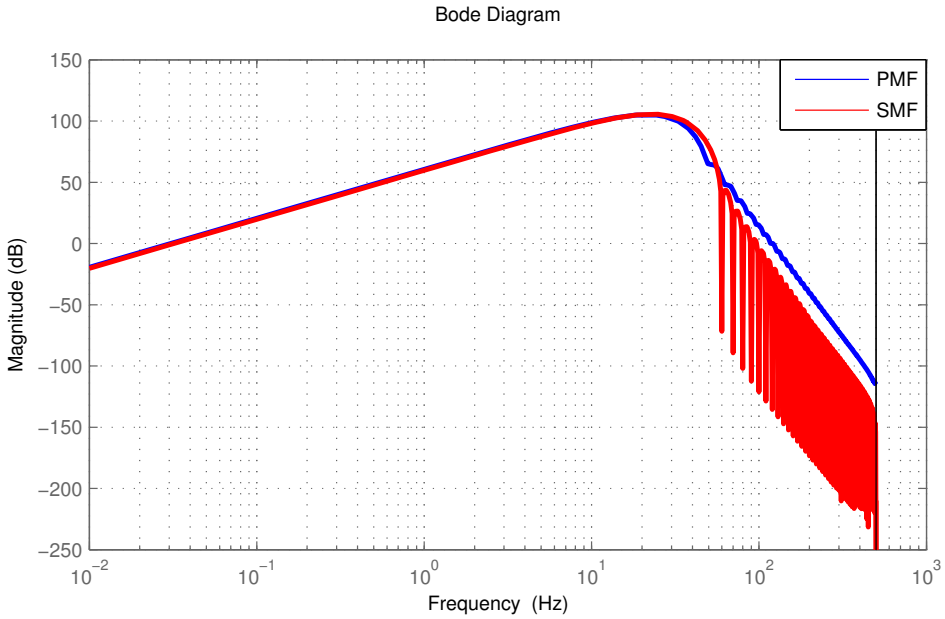
(4) Harley modulating functions (HMF): based on Shinbrot's method of moment functionals and Pearson Fourier modulating functions, this group of modulating functions are given: $g_\ell(t) = \sum_{j=0}^{n}(-1)^j \begin{pmatrix} n \\ j \end{pmatrix} cas((n+\ell-j)\omega_0 t)$, where $\ell = 0, \pm1, \pm2, \ldots$ is integer, $\omega_0 = \frac{2\pi}{T}$ is the resolving frequency, $cas(x) = \cos x + \sin x$.

The integration effect with modulating functions can be analysed via FIR filtering point of view. Suppose the system sampling time is Ts and extract modulating function value $g_\ell(i)$ every Ts second as the coefficient of FIR filter. Then use bode plot to get the frequency contribution to magnitude of the modulating function. For example, take into account the second order derivatives of modulating functions, with system sampling time 1 millisecond, time window 0.1 second, and draw their bode plots.

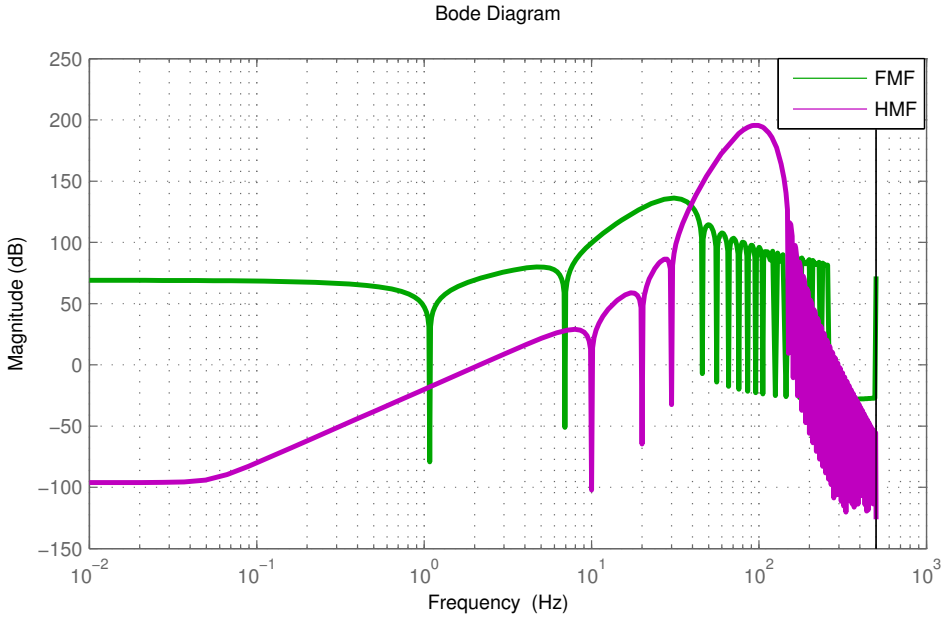
From figure (2.1), the frequency-magnitude response shows that for the groups of JMF and SMF, the filtering property of modulating functions are similar as band-pass filter, because the high frequency component of the signal contributes in a attenuation way to output and frequency higher than 150 Hz is considered to be cut off, as well at low frequency there is attenuation. When noise occurs at high frequency part of the signal, computation of integration using these modulating functions is robust to noise. From another aspect, the discrete version of a definite integral is the product of integrators with time step on the time interval $[t_a, t_b]$. Because the time step is of low frequency, the integral is also low-pass.

While for the groups of FMF and HMF, the band-pass area occurs at high frequency. It turns out that they enlarge the high frequency contribution to integration. Especially HMF can be regarded as a high pass FIR filter because it attenuates greatly the low frequency contribution to magnitude. This property makes these two groups of modulating functions not suitable in normal applications. In the next section, the estimation process considers only the groups of JMF and SMF modulating functions.

In conclusion, integration with modulating functions is an effective approach to decrease the order of input model. As well it has certain filtering property. Compared to filter techniques, it is causal and it has no phase shift because it calculates a scalar. The integration coefficients can be computed off-line so that it can be implemented easily and instantly for on-line applications. These advantages make modulating functions method interesting, but still it has drawback such as it has less excitation in the identifiability compared to the method



(a) JMF and SMF



(b) FMF and HMF

Figure 2.1 – Bode plot of second order derivatives of modulating functions when $\ell = 10$

treating each points of the interval as an independent equation, because the modulating function approach combines all into one scalar.

2.2.2 Configuration Choice

Integration with modulating functions is actually a band-pass filtering procedure. If the sampling time is fixed, the effect depends on two choices: the type of modulating functions and the integration time window.

We have discussed in the previous section, how are the inherent properties of different modulating functions. Indeed, they are all pass-band filter and their pass band locates in different area under same situation.

In other aspect, it should be noticed that if we enlarge the time window of integration, which means more sampling points, the band-pass area will be shifted to lower frequency, meanwhile the magnitude response will be increased. This can be easily understood as time window enlarges, the integration value increases.

From the above, we can make strategy how to tune the configuration for the purpose of filtering. Assume sampling time is already known, the first step is to fix the type of modulating functions to be used. If the cut-off frequency is low, we choose JMF and SMF. The second step is fix the integration time window, which need to be instructed by drawing and analysing the magnitude bode plot. The principle is that the pass band shift to lower frequency as integration time window increases.

2.2.3 An introducing example with one joint robot with gravity effect

Let us consider a simple one joint robot model with gravity torque and with current position q driven by a torque τ. Its classical dynamic model is given by:

$$ZZ\ddot{q} + F_v\dot{q} + MX\sin(\dot{q}) = \tau. \tag{2.3}$$

Divide equation (2.3) by the constant ZZ, then we have the equation of a pendulum:

$$\ddot{q} + \alpha\dot{q} + \beta\sin(q) = \gamma\tau, \tag{2.4}$$

with $\alpha = \frac{F_v}{ZZ}$, $\beta = \frac{MX}{ZZ}$, $\gamma = \frac{1}{ZZ}$. The purpose is to recover on-line the three dynamic parameters α, β, γ described in this model only by using the measured position and the known applied torque. The estimation is carried out with

modulating-like functions. At time instant t, the estimation make use of the data from time interval $[t-T,t]$, where T is the time window length. Consider a modulating-like function $g_\ell(v) = \frac{(v-t+T)^\ell}{\ell!}$, which has the property that $\dot{g}_\ell(v) = g_{\ell-1}(v)$ and g_ℓ is null at time $t-T$. Multiply (2.4) by $g_\ell(v)$ and perform an integration by part on interval $[t-T,t]$, it leads to the following relation:

$$\begin{aligned} &g_\ell(t)\dot{q}(t) - g_{\ell-1}(t)q(t) + \alpha\left[g_\ell(t)q(t) - \int_{t-T}^{t} g_{\ell-1}(t)q(t)dt\right] \\ &+\beta\int_{t-T}^{t} g_\ell(t)\sin(q(t))dt - \gamma\int_{t-T}^{t} g_\ell(t)\gamma(t)dt = -\int_{t-T}^{t} g_{\ell-2}(t)q(t)dt. \quad (2.5) \end{aligned}$$

Take $\mathbf{X}_p = [\dot{q}(t)+\alpha q(t), q(t), \alpha, \beta, \gamma]^T$ as unknown, thus it requires at least 5 equations to find them. And it can be noticed that this relation (2.5) is a scalar equation. Thus it need additional data to form multi-equations, which can be realized by replacing ℓ by a sequence of N_p elements of ℓ_i where $\ell_i \in \mathbb{R}$ and $\ell_i \geq 2$. After setting a sequence of ℓ_i the list of equations are expressed as:

$$\mathbf{A}(t,q,\tau)\begin{bmatrix} \dot{q}(t)+\alpha q(t) \\ q(t) \\ \alpha \\ \beta \\ \gamma \end{bmatrix} = \mathbf{B}(t,q), \quad (2.6)$$

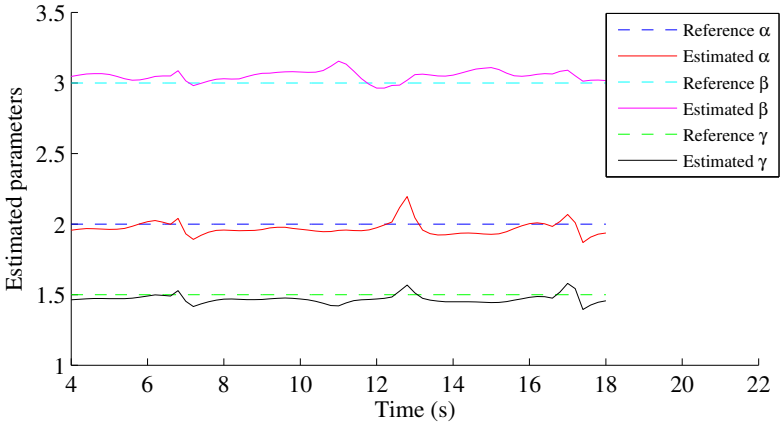
where $\mathbf{A}(t,q,\tau)$ is a $N_p \times 5$ observation matrix and the i-th line of $\mathbf{A}(t,q,\tau)$ is given by

$$\begin{aligned} &\left[g_{\ell_i}(t), -g_{\ell_i-1}(t), -\int_{t-T}^{t} g_{\ell_i-1}(t)q(t)dt, \int_{t-T}^{t} g_{\ell_i}(t)\sin(q(t))dt, -\int_{t-T}^{t} g_{\ell_i}(t)\gamma(t)dt\right] \\ &\qquad = \mathbf{A}_i(t,q,\tau) \quad (2.7) \end{aligned}$$

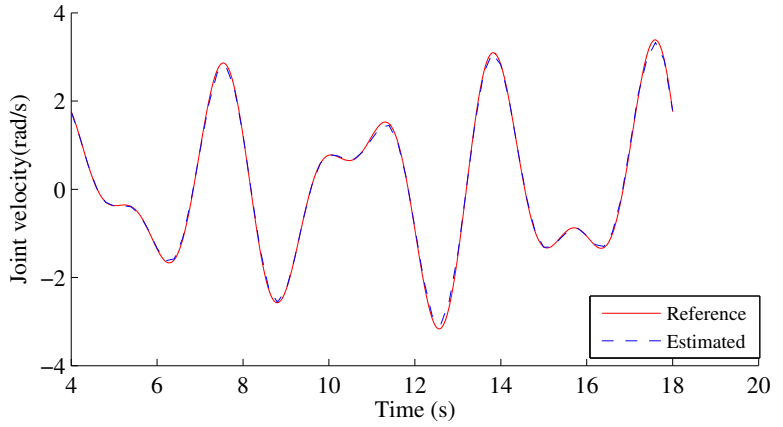
and the i-th element of vector $\mathbf{B}(t,q)$ is given by $\mathbf{B}_i(t,q) = -\int_{t-T}^{t} g_{\ell_i-2}(t)q(t)dt$.

This forms the general over-determined linear system $\mathbf{AX}_p = \mathbf{B}$, which can be resolved by least square approaches. Tests are done in simulation, the measured signal $\tilde{q}(t_i)$ is simulated as a composition of an additive white Gaussian noise $\omega(t_i)$ and the current trajectory $q(t_i)$. The noise level is described as uniform distributed noise of signal to noise ratio 30 dB (SNR, *i.e.* SNR $= 10\log_{10}(\frac{\sum|\tilde{q}(t_i)|}{\sum|\omega(t_i)|})$). Take $\ell = 2 + \frac{i}{100}$, $i = 1, 2, ..., 1800$ and sliding time window length $T = 4$ second.

The estimation result is shown in Fig (2.2,2.3). The average estimation of dynamic parameters α, β, γ are 1.966, 3.0572 and 1.4666, with errors 0.034,



(a) Estimated parameters in 18 seconds

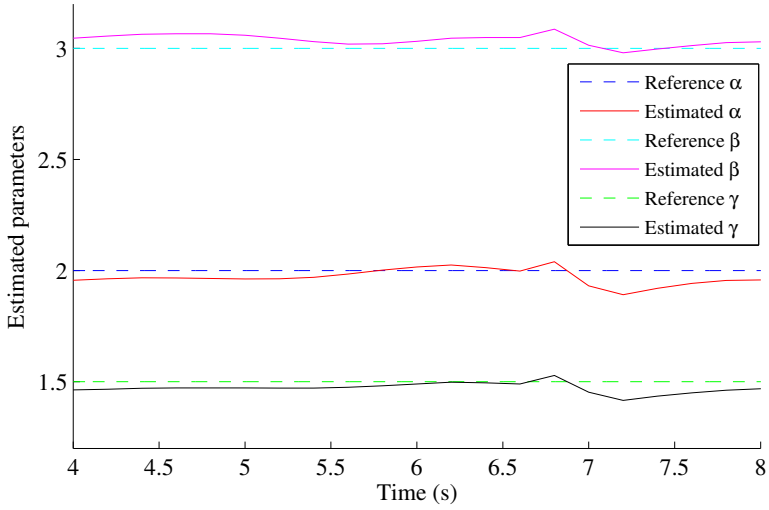


(b) Estimated $\dot{q}$ in 18 seconds

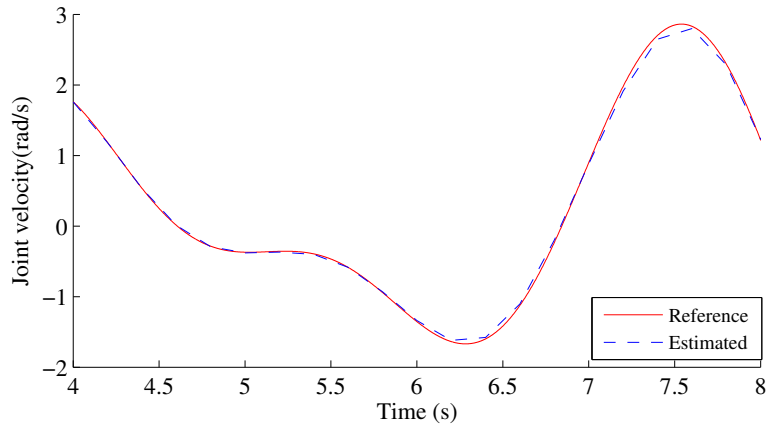
Figure 2.2 – Estimation when q is of SNR= 30dB, with $\ell = 2+\frac{i}{100}$, $i = 1, 2, ..., 1800$ and T=4s

0.0572 and 0.05333. Meanwhile the observed pendulum velocity fits quit well the reference one. The advantages of this method include: all estimations come from only the position and torque data, as well the state velocity is observed; the estimation is carried out without pre-filtering of the noisy position nor torque data. This is because that integration associated this special kind of modulating-like functions has a nature effect of filtering property, which makes the implementation simpler.

In conclusion, for pendulum case we utilize the measured angle position from a certain time window to estimate not only the dynamic parameters, but also the angle velocity and filtered joint position at time t. The estimation gives good result with a time window of 4 seconds. With a large sequence of ℓ, the estimator forms an over-determined observation matrix and can be solved by least square techniques. Meanwhile results show that this estimator is robust



(a) Estimated parameters from 4 to 8 second



(b) Estimated $\dot{q}$ from 4 to 8 second

Figure 2.3 – Zoomed estimation when q is of SNR= 30dB, with $\ell = 2 + \frac{i}{100}$, $i = 1, 2, ..., 1800$ and T=4s

with respect to noise. But the drawback of this method lies on the fact that the estimation is out of precision before it acquires enough sampling data.

2.2.4 An introducing example with one joint robot without gravity effect

Consider a simple one revolute joint robot described by:

$$ZZ\ddot{q} + F_v\dot{q} + F_c sign(\dot{q}) = \tau, \tag{2.8}$$

where ZZ $(kg \cdot m^2)$ is the inertial parameter, F_v (N/(m/s)) and F_c (N) are the viscous and Coulomb friction parameters respectively.

The purpose is to recover in real time the three dynamic parameters ZZ, F_v, F_c described in this model only by using the measured angular position q and the known applied torque. At time instant t, the estimation make use of the data from time interval $[t-T, t]$, where T is the time window length. Consider a second order combination of Jacobi modulating functions $g_\ell(v)$

$$\begin{aligned} g_{\ell 1}(v) &= \left(\frac{v-t+T}{T}\right)^\ell \left(\frac{v-t}{T}\right)^\ell, \\ g_{\ell 2}(v) &= \left(\frac{v-t+T}{T}\right)^\ell \left(\frac{v-t}{T}\right)^{\ell+1}, \\ g_{\ell 3}(v) &= \left(\frac{v-t+T}{T}\right)^\ell \left(\frac{v-t}{T}\right)^{\ell+2}, \end{aligned} \tag{2.9}$$

where $v \in [t-T, t]$ and $\ell = \{\ell \in \mathbb{N} | \ell = 10, 11, \ldots, 20\}$. Multiplying (2.8) by the combination of modulating functions $g_\ell(v)$ and perform an integration by part on interval $[t-T, t]$ and its half interval, it gives

$$\begin{aligned} &ZZ \int_{t-T}^{t} g_{\ell-2}(t)q(t)dt - F_v \int_{t-T}^{t} g_{\ell-1}(t)q(t)dt + F_c \int_{t-T}^{t} g_\ell(t) sign(\dot{q})dt \\ &= \int_{t-T}^{t} g_\ell(t)\tau dt. \end{aligned} \tag{2.10}$$

Take $\mathbf{X}_p = [ZZ, F_v, F_c]^T$ as unknown, thus it requires at least 3 equations to find them. Notice that this relation is a scalar equation, and by replacing ℓ by a sequence of integers ℓ_i. After developing a sequence of N_{1R} elements ℓ_i, a list of equations are expressed as:

$$\mathbf{A}(t, q, \dot{q})[ZZ, F_v, F_c] = \mathbf{B}(t, \tau), \tag{2.11}$$

where $\mathbf{A}(t,q,\dot{q})$ is of dimension $N_{1R}\times 3$ and the i-th line of $\mathbf{A}(t,q,\dot{q})$ is given by

$$A_i(t,q,\dot{q}) = [\int_{t-T}^{t} g_{\ell_i-2}(t)q(t)dt, -\int_{t-T}^{t} g_{\ell_i-1}(t)q(t)dt, \int_{t-T}^{t} g_{\ell_i}(t)sign(\dot{q})dt],$$

and the i-th element of vector $\mathbf{B}(t,\tau)$ is given by $\mathbf{B}(t,\tau) = \int_{t-T}^{t} g_{\ell_i}(t)\tau dt$.

A simulation test is carried out with measurement in joint position and torque associated with a normally disturbed random noise whose signal to noise ratio (SNR) is 30 dB. In the off-line identification case, the data are collected with sampling time $T_s = 1$ millisecond. Because the modulating function with power model identification method turns vector equations into scalar equations, in that sense the observation loses some information. In order to ensure the full rank and good condition number of the observation matrix, the estimation time window length should be set relatively long, here we set $T_{est} = 20$ seconds. The inverse dynamic identification model with LS technique method (IDIM-LS) is applied to make a comparison. In the both identification model, the joint position are pre-filtered using a fourth order low-pass Butterworth filter with cut-off frequency at 2 Hz. The joint velocity and acceleration are computed from the filtered joint position by means of Euler central difference algorithm of the low-pass filtered position. The Euler central difference is the finite difference method. Assume a discrete signal sequence $[x[1],\cdots,x[i],\cdots,x[N]]$, with sampling time T_{samp}, the first order derivative is obtained with Euler central difference

$$\begin{aligned}
\dot{x}[1] &= \frac{x[2]-x[1]}{T_{samp}},\\
\dot{x}[i] &= \frac{x[i+1]-x[i-1]}{2T_{samp}},\ 1<i<N\\
\dot{x}[N] &= \frac{x[N]-x[N-1]}{T_{samp}}.
\end{aligned} \tag{2.12}$$

In dynamic identification model, the data are down-sampled through a decimate filter at 10 Hz. The estimation results are obtained with the same reference trajectory and several identifications are processed off-line with time window $T_{est} = 20$ seconds. The average value and variance with respect to real value are shown in table (2.1).

In conclusion, for one joint robot case we utilize only the measured joint position and the computed joint velocity from a certain time window to estimate the dynamic parameters. The estimation gives good off-line identification result with respect to noise. With a large sequence of ℓ, the estimator forms

Parameters	Real	Modulating	IDIM-LS	Variance Modulating	Variance IDIM-LS
ZZ	3	3.0013	3.0018	0.0204	0.0263
F_v	0.4	0.3940	0.4185	0.0354	0.0483
F_c	0.5	0.5053	0.4736	0.0366	0.0667

Table 2.1 – Identification results of 1R robot using IDIM-LS and modulating functions identification approaches

an over-determined observation matrix and can be solved by least square techniques. Compared to IDIM-LS, the modulating approach has good or even better performance in off-line identification. The drawback of this method lies on the fact that the condition number of the observation matrix is relatively larger than that of IDIM-LS method, and the sampling rate should not be too low in order to reduce the integration error.

2.2.5 General case

The idea of solving this kind of differential system is to increase the order of input value by partial integration with modulating functions. For a general system

$$\sum_{i=0}^{N_t} \alpha_i f_i(\theta, \theta^{(1)}, ..., \theta^{(n)}) = \gamma, \tag{2.13}$$

where N_t is the number of terms, n is the highest order of derivative of θ, α_i are constant parameters, $\theta^{(i)}$ is i-th order of derivative of θ and f_i is a general function. Now suppose a family of modulating functions $g_\ell(v)$ satisfying

$$g_\ell^{(i)}(0) = g_\ell^{(i)}(t) = 0,\ i \leq n. \tag{2.14}$$

Then, multiply $g_\ell(v)$ with the general system formulation and do integration on the interval $[0, t]$. When the function $f_i(\theta, \theta^{(1)}, ..., \theta^{(n)})$ is analytically integrable, one can perform integration by part according to partial integration theory using (2.14)

$$\int_0^t \alpha_i g_\ell f_i(\theta, \theta^{(1)}, ..., \theta^{(n)}) dv = -\int_0^t \alpha_i g_\ell^{(1)} f_i^{[1]}(\theta, \theta^{(1)}, ..., \theta^{(n-1)}) dv, \tag{2.15}$$

where $f_i^{[1]}(\theta, \theta^{(1)}, ..., \theta^{(n-1)})$ is the analytical form of first time integral function of
$f_i(\theta, \theta^{(1)}, ..., \theta^{(n)})$.

In a similar way, if $f_i(\theta, \theta^{(1)}, ..., \theta^{(n)})$ is K times integrable, the highest order derivative of θ can be degraded to $(n-K)$-th order, which avoids using noisy high order derivatives. With this method, there is no need to access high order values and it can be numerically more precise, thus gives better estimation result. Finally in the K times integrable case, (2.15) can be written as a counterpart of

$$\int_0^t \alpha_i g_\ell f_i(\theta, \theta^{(1)}, ..., \theta^{(n)}) dv = (-1)^k \int_0^t \alpha_i g_\ell^{(k)} f_i^{[k]}(\theta, \theta^{(1)}, ..., \theta^{(n-k)}) dv. \quad (2.16)$$

2.3 Identification with modulating functions and power model

Recall the robot power model (1.33):

$$\dot{\mathbf{q}}^{\mathbf{T}} \mathbf{\Gamma_m} = \frac{d}{dt}(H) + \dot{\mathbf{q}}^{\mathbf{T}} [\mathbf{diag}(\dot{\mathbf{q}}) \mathbf{F_v} + \mathbf{diag}(\mathbf{sign}(\dot{\mathbf{q}}) \mathbf{F_s} + \mathbf{\Gamma_{off}}]. \quad (2.17)$$

Integrate both sides of equation (1.33) with a first order modulating function g on the time interval $[t_a, t_b]$ and apply integration by part which leads to:

$$\begin{aligned} &\int_{t-T}^{t} g \dot{\mathbf{q}}^{\mathbf{T}} \mathbf{\Gamma_m} dt = -\int_{t-T}^{t} \dot{g} \mathbf{h} \mathbf{K} dt + \\ &\int_{t-T}^{t} g \dot{\mathbf{q}}^{\mathbf{T}} \mathbf{diag}(\dot{\mathbf{q}}) \mathbf{F_v} dt + \int_{t-T}^{t} g \dot{\mathbf{q}}^{\mathbf{T}} \mathbf{diag}(\mathbf{sign}(\dot{\mathbf{q}})) \mathbf{F_s} dt + \int_{t-T}^{t} g \dot{\mathbf{q}}^{\mathbf{T}} \mathbf{\Gamma_{off}} dt \end{aligned} \quad (2.18)$$

This equation presents an energy balance and is a scalar equation. The modulating functions are well adapted in this model, because we know the analytical expression of the energy part $\mathbf{h}$ and the derivatives of the modulating functions. In order to identify multi-unknowns, we need to construct more equations whose number is equal to or larger than that of the base dynamic parameters. The construction process is considered as choices of different modulating functions g and with different parametrization.

With the above procedures, we obtain an over-determined linear system of $n \times n_t \times n_m$ equations similar as (1.13):

$$\mathbf{Y_E} = \mathbf{W_E}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) \mathbf{X} + \rho_{\mathbf{E}}, \quad (2.19)$$

where n_t is the number of intervals, n_m is the number of the modulating func-

tions, ρ_E is the noise, $\mathbf{Y_E}$ and $\mathbf{W_E}$ are the vector of integrals and the observation matrix, respectively, which are defined as follows:

$$\mathbf{Y_E} = \begin{bmatrix} \int_{t_{a1}}^{t_{b1}} g_1 \dot{\mathbf{q}}^T \mathbf{\Gamma_m} \\ \vdots \\ \int_{t_{an_t}}^{t_{bn_t}} g_{n_e} \dot{\mathbf{q}}^T \mathbf{\Gamma_m} \end{bmatrix}, \mathbf{W_E} = \begin{bmatrix} -\int_{t_{a1}}^{t_{b1}} \dot{g}_1 \mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) \\ \vdots \\ -\int_{t_{an_t}}^{t_{bn_t}} \dot{g}_{n_e} \mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) \end{bmatrix}. \tag{2.20}$$

The system (2.19) can be solved using the different LS techniques which are discussed in appendix.

2.3.1 Simulation on 2R robot

This simulation part utilizes the robot model described in chapter (1.1.1). There are eight minimal dynamic parameters $\mathbf{X}$ and recall (1.12)

$$\mathbf{X} = [ZZ_{1R},\ ZZ_2,\ LMX_2,\ LMY_2,\ F_{v1},\ F_{s1},\ F_{v2},\ F_{s2}],$$

The simulation tests are running with value $\mathbf{X}$ which is all in SI Units: $\mathbf{X}$ = [3.5 0.06 0.12 0.005 0.05 0.5 0.01 0.1]. And the sampling rate $\omega_c = 2\pi{\times}100\ rad/s$.

In off-line identification, the estimation time window T_{est} is set to be 20 seconds. A reference trajectory is selected using a fifth order polynomial trajectory.

We can see that the estimated friction parameters are more disturbed, because $F_v\dot{q}$ is small in value in the robot model, and F_s is not well excited with polynomial trajectory, instead it can be excited with trapezoid velocity in the trajectory. Meanwhile, it is because the friction parameters are in term with velocity data whose estimation has more bias with respect to real value. For example, the viscous friction parameters correspond to term $\dot{\mathbf{q}}^2$, where the errors are in the order of noise square and have great influence on these parameters.

To carry out the estimation, apply the same combination of Jacobi modulating functions $g_\ell(v)$ described in (2.9), with $v \in [t-T, t]$ and $\ell = \{\ell \in \mathbb{N} | \ell = 10, 11, \ldots, 20\}$. Here we use QR factorization method to solve the least square problem. Notice that similar as the decimate procedure composed of a low-pass filter $F_p(s)$ in the IDIM-LS technique, the modulating function with power model approach also needs a lower resampling procedure. As discussed in (Gautier, 1997), the decimate rate $n_d = 0.8 \times \frac{\omega_c}{2\omega_{fp}}$ for an IIR filter. Here, we take $n_d = 10$.

As discussed in chapter (1.2), the selection of definite integration time inter-

val $[t_a(i), t_b(i)]$ is actually a decimate procedure with the ratio n_d, which results in choosing $t_a(i)$ and $t_b(i)$ with a constant value $t_b(i) - t_a(i) = n_d * 2\pi/\omega_c$. In the same way, the decimate procedure $F_p(s)$ is applied on the modulating function with power model approach. With $n_e = 5000$ samples and sampling rate $\omega_c = 2\pi \times 100\ rad/s$, $t_b(i) - t_a(i) = 0.1s$.

The joint position and torque data are pre-filtered using a forward-backward Butterworth with a cutoff frequency $\omega_{fq} = 0.8\frac{\omega_c/2}{5}$. Then, the joint velocity is computed using Euler central difference algorithm.

In order show the systematic error caused by the modulating functions with power model approach and the filtering procedure, we carry out the simulation without noise in the measurement. The results are shown in table (2.2), where real parameters values $\mathbf{X}$, the identified values $\hat{\mathbf{X}}$ and the relative standard deviation $\sigma_{\hat{\mathbf{X}}ri}$ % are presented. The results prove that modulating functions with power model has small systematic error.

Parameters	Real	$\hat{\mathbf{X}}$	$\sigma_{\hat{\mathbf{X}}ri}$ %
ZZ_{1R}	3.5	3.5005	0.0782
F_{v1}	0.05	0.0499	0.0989
F_{s1}	0.5	0.5001	0.0300
ZZ_2	0.06	0.0600	0.0260
LMX_2	0.12	0.1200	0.0211
LMY_2	0.005	0.0050	0.0315
F_{v2}	0.01	0. 0.0100	0.0226
F_{s2}	0.1	0.0999	0.0315

Table 2.2 – Systematic error in simulation using modulating functions with power model approach

2.3.2 Identification on 2R prototype robot

The experimental work is done on the two revolute joints planar prototype robot described in chapter (1.1.1). The joint position $\mathbf{q}$ and the current reference $\mathbf{V}_T$ (the control input) are collected at a 100 Hz sample rate. The measurement joint position is shown in Fig. (2.4) and the motor torques are presented in Fig. (2.5).

Similar as the simulation, the modulating functions chooses a combination of Jacobi modulating functions $g_\ell(v)$ described in (2.9), with $v \in [t-T, t]$ and $\ell = \{\ell \in \mathbb{N} | \ell = 10, 11, \ldots, 20\}$. The decimate procedure composed of a low-pass filter $F_p(s)$ with $n_d = 10$ is implemented in IDIM-OLS. For modulating functions with

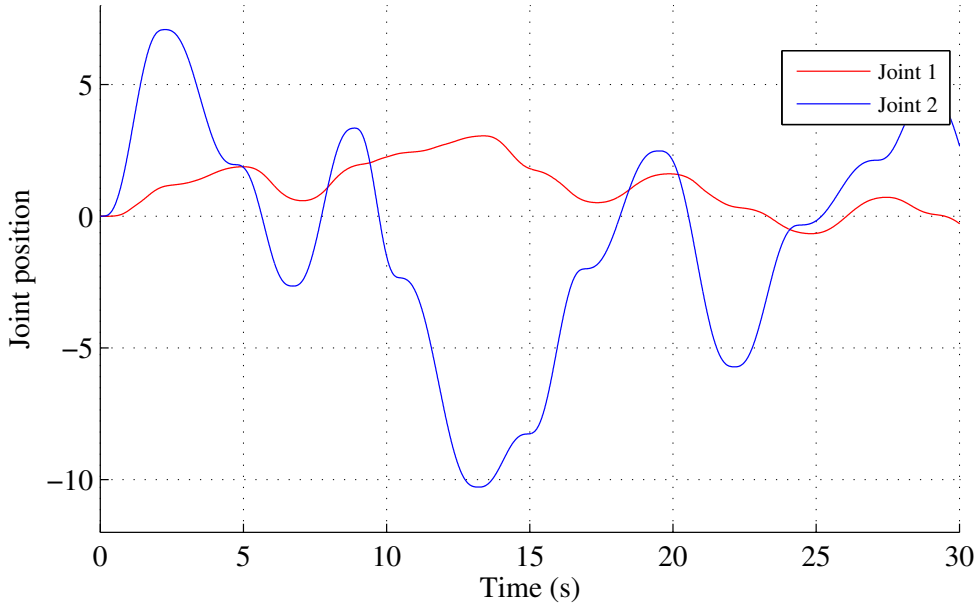


Figure 2.4 – Measured joint position of 2R prototype scara planar robot

power model approach, the selection of definite integration time interval $[t_a(i), t_b(i)]$ can be chosen as $t_b(i) - t_a(i) = 0.5s$, taking the filtering purpose and the numbers of points in integration with modulating functions into a compromise.

The joint position and torque data are pre-filtered using a 12 order forward-backward Butterworth with a cutoff frequency $\omega_{fq} = 0.8\frac{\omega_c/2}{5}$. Then, the joint velocity is computed using Euler central difference algorithm.

Once the identification model is built, the estimation of minimal dynamic parameters $\hat{\mathbf{X}}$ is solved by OLS method. Standard deviations $\sigma_{\hat{\mathbf{X}}_i}$ are estimated using classical and simple results from statistics, considering the matrix $\mathbf{W}$ to be a deterministic one, and ρ to be a zero mean additive independent noise, with standard deviation σ_ρ that $C_{\rho\rho} = E(\rho^T\rho) = \sigma_\rho^2\mathbf{I}_r$, where E is the expectation operator. The variance-covariance matrix of the estimation error and standard deviations can be calculated by:

$$C_{\hat{\mathbf{X}}\hat{\mathbf{X}}} = E[(\hat{\mathbf{X}} - \hat{\mathbf{X}})(\hat{\mathbf{X}} - \hat{\mathbf{X}})^T] = \sigma_\rho^2(\mathbf{W}^T\mathbf{W})^{-1},$$

where $\sigma^2_{\hat{\mathbf{X}}_i} = C_{\hat{\mathbf{X}}\hat{\mathbf{X}}ii}$, the diagonal coefficient of $C_{\hat{\mathbf{X}}\hat{\mathbf{X}}}$.

An unbiased estimation of σ_ρ is used to get the relative standard deviation $\sigma_{\hat{\mathbf{X}}ri}$ by the expression:

$$\hat{\sigma}_\rho^2 = \frac{\|Y - W\hat{\mathbf{X}}\|^2}{r - c}, \ \%\sigma_{\hat{\mathbf{X}}ri} = 100 \times \frac{\sigma_{\hat{\mathbf{X}}i}}{\hat{\mathbf{X}}i},$$

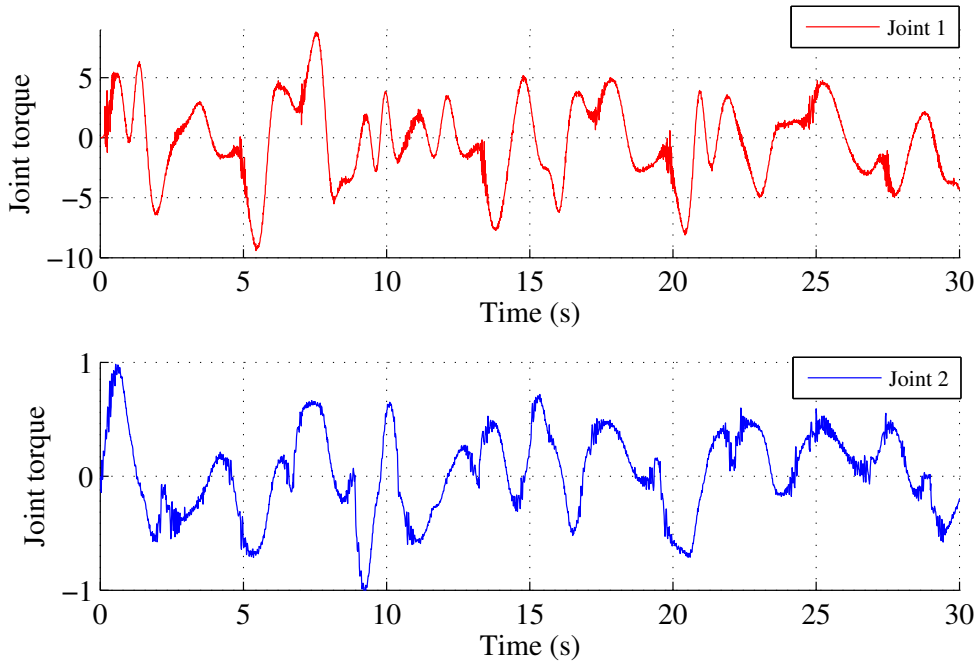


Figure 2.5 – Joint torque of 2R prototype scara planar robot

where r is the total number of equations and c is the number of unknown parameters.

To make the results more clear, a comparison is done with IDIM-OLS technique on the same trajectory, and the identification results are shown in table(2.3). Because the numbers of equations are different, the comparison in the residue norm $\|Y - W\hat{\mathbf{X}}\|$ is meaningless. Instead, we compare in the relative residue norm $\frac{\|Y-W\hat{\mathbf{X}}\|}{\|Y\|}$, where the modulating functions with power model approach has better relative residue norm than IDIM-OLS method. Meanwhile, according to the relative standard deviation $\sigma_{\hat{\mathbf{X}}ri}$ % results, the results obtained by the modulating functions with power model are better than those of IDIM-OLS method. Thus, we can conclude that the modulating functions with power model approach behaves better than IDIM-OLS method in this 2R prototype scara robot case.

2.3.3 Simulation of payload dynamic parameters on 4R robot

The 4R robot model are constructed in the modified Denavit and Hartenberg notation (DHM) frame described in table (2.4), and the direct dynamic model and inverse dynamic model are computed using the software SYMORO introduced in chapter (1.1.3).

We consider the payload identification from the power point of view, with

	IDIM-OLS			Modulating & power model		
Parameters	$\hat{\mathbf{X}}$	$2\sigma_{\hat{\mathbf{X}}}$	$\sigma_{\hat{\mathbf{X}}r}$ %	$\hat{\mathbf{X}}$	$2\sigma_{\hat{\mathbf{X}}}$	$\sigma_{\hat{\mathbf{X}}r}$ %
ZZ_{1R}	3.4836	0.0332	0.4765	3.4452	0.0187	0.2713
F_{v1}	0.2460	0.0895	18.1825	0.1484	0.0663	22.3353
F_{s1}	0.4330	0.0491	5.6677	0.5696	0.0749	6.5788
ZZ_2	0.0596	0.0041	3.4481	0.0648	0.0007	0.5726
MX_2	0.1253	0.0027	1.0865	0.1199	0.0020	0.8143
MY_2	0.0006	0.0026	218.2929	0.0033	0.0011	16.5420
F_{v2}	0.0139	0.0153	55.0943	0.0187	0.0026	7.0349
F_{s2}	0.1274	0.0438	17.2103	0.1016	0.0159	7.8086
$\|Y - W\hat{\mathbf{X}}\|$	4.9148			8.8995		
$\frac{\|Y-W\hat{\mathbf{X}}\|}{\|Y\|}$	0.0905			0.0699		
$\frac{\|Y-W\hat{\mathbf{X}}\|}{\|\sqrt{r-c}\|}$	0.2168			0.1864		
$\max(\left\|\frac{(Y-W\hat{\mathbf{X}})_j}{Y_j}\right\|)$	62.1104			0.7561		
Nb equation	522			2288		
Cond(W)	38.5368			178.7265		

Table 2.3 – Comparison between IDIM-OLS and modulating function using power model method, 2R prototype robot

j	σ	α	d	θ	r
0	0	0	0	q_1	0
1	0	$-\frac{\pi}{2}$	0	q_2	0
2	0	0	$L2$	q_3	0
3	0	0	$L3$	q_4	0

Table 2.4 – DHM configuration of 4R scara planar robot

application of the modulating functions. Assume the robot dynamic parameters are already identified before loading. Then the robot power model has an additional term of the payload energy :

$$\dot{\mathbf{q}}^{\mathbf{T}}\boldsymbol{\Gamma}_{\mathbf{m}} = \frac{d}{dt}(H) + \frac{d}{dt}(H_p) + \dot{\mathbf{q}}^{\mathbf{T}}[\mathbf{diag}(\dot{\mathbf{q}})\mathbf{F_v} + \mathbf{diag}(\mathbf{sign}(\dot{\mathbf{q}})\mathbf{F_s} + \boldsymbol{\Gamma}_{\mathbf{off}}], \tag{2.21}$$

where $\frac{d}{dt}(H_p)$ is the *Hamiltonian* energy of the payload. In a similar way, integrate both sides of equation (1.33) with a first order modulating function g on the time interval $[t_a, t_b]$ and apply integration by part which leads to:

$$\int_{t-T}^{t} \dot{g}\mathbf{h_p}\mathbf{X_p}dt = -\int_{t-T}^{t} g\dot{\mathbf{q}}^{\mathbf{T}}\boldsymbol{\Gamma}_{\mathbf{m}}dt - \int_{t-T}^{t} \dot{g}\mathbf{h}\mathbf{K}dt + \int_{t-T}^{t} g\dot{\mathbf{q}}^{\mathbf{T}}\mathbf{diag}(\dot{\mathbf{q}})\mathbf{F_v}dt$$
$$+ \int_{t-T}^{t} g\dot{\mathbf{q}}^{\mathbf{T}}\mathbf{diag}(\mathbf{sign}(\dot{\mathbf{q}}))\mathbf{F_s}dt + \int_{t-T}^{t} g\dot{\mathbf{q}}^{\mathbf{T}}\boldsymbol{\Gamma}_{\mathbf{off}}dt, \tag{2.22}$$

moreover, $\mathbf{h_p}$ is the vector of energy functions corresponding to payload, $\mathbf{X_p}$ is the payload dynamic parameters.

This gives linear formulation of the identification model, because $\mathbf{X_p}$ is constant. And it only depends on the joint force/torque, position and velocity information, which is more practical to implement.

The simulations are running with a four revolute joints robot model , assuming that the dynamic parameters of the robot X have already been identified without carrying a payload. When a payload is fixed on the terminal link of the robot, we assume that the payload has influence on the last joint link and causes the variation of the dynamic parameters of last joint by:

$$\Delta X = [\Delta XX\ \Delta XY\ \Delta XZ\ \Delta YY\ \Delta YZ\ \Delta ZZ\ \Delta MX\ \Delta MY\ \Delta MZ\ \Delta M].$$

δX = needs to be identified in order to update the controller. In simulation, we assume that $\Delta X = [0.2,\ 0.3,\ 0.2,\ 0.2,\ 0.1,\ 0.3,\ 0.1,\ 0.5,\ 0.4,\ 2]$, which is all in SI Units.

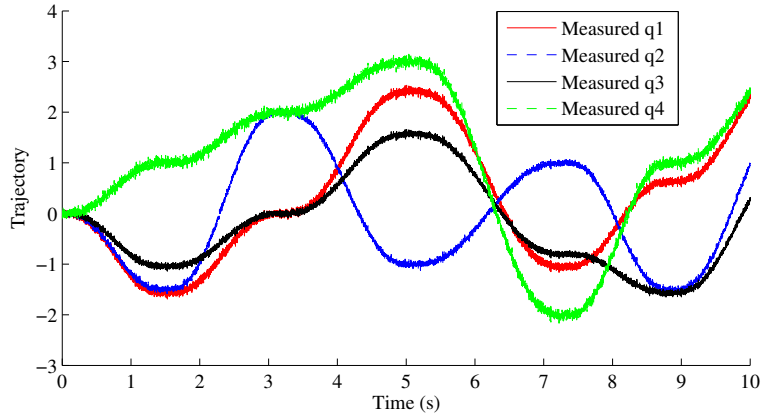
With known robot dynamic parameters, the robot is installed with the unknown payload at instant $t = 0$. The robot is driven by a computed torque PID controller with the known robot dynamic parameters. The robot motion follows a successive point to point reference trajectory using a classical 5^{th} order polynomial trajectory generator.

The measurements of joint position is the superposition of real joint position and a normally disturbed random noise whose signal to noise ratio (SNR) is 30 dB, with sampling time $T_s = 1$ millisecond. The off-line estimation time window length T_{est} for identification of the payload dynamics is set as 10 seconds, in order to ensure good rank and condition number of the observation matrix.

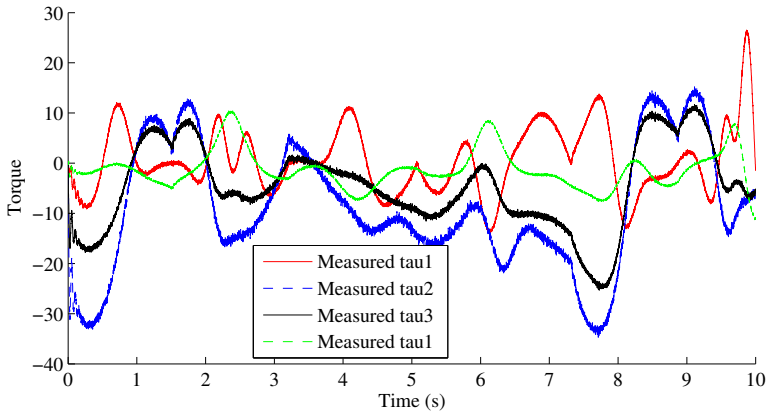
As discussed in subsection (2.2.1), we implement the sinusoid based and Jacobi polynomial based modulating functions because they have better attenuation on the high frequency noise. The Jacobi modulating functions $g_\ell(v)$ described in (2.9), with $v \in [t-T, t]$ and $\ell = \{\ell \in \mathbb{N} | \ell = 10, 11, \ldots, 20\}$. The sinusoid based modulating functions are $g_\ell(v) = \sin^\ell\left(\frac{\pi}{T}(v - t + T)\right)$, with $\ell = \{\ell \in \mathbb{N} | \ell = 10, 11, \ldots, 20\}$.

The integrals definite are selected in simulation with $T = n_d * 2\pi/\omega_c$. We set the sampling frequency $\omega_c = 1000$ Hz and $n_d = 10$, thus the definite integral length $T = 0.01s$.

The joint velocity is computed using Euler central difference from the filtered joint position $\mathbf{q}$. The filter is a eighth order forward-backward Butterworth filter with cutoff frquency 2.5 Hz. The joint torque is also filtered with



(a) Measurement position



(b) Measurement torque

Figure 2.6 – Measurement with normally disturbed random noise of SNR=30dB

cutoff frequency 10 Hz. The simulated measurement of joint position and torque are shown in figure (2.6) and the identification results are listed in table (2.5).

$[t_a,\ t_b]$ with configuration Z_1				
Parameters	Real	Modulating	IDIM-OLS	IDIM+M
ΔXX	0.2000	0.2137	0.1961	0.2138
ΔXY	0.3000	0.2992	0.3118	0.2991
ΔXZ	0.2000	0.2534	0.2078	0.2474
ΔYY	0.2000	0.1656	0.2199	0.1757
ΔYZ	0.1000	0.0898	0.1085	0.0923
ΔZZ	0.3000	0.2434	0.2905	0.2494
ΔMX	0.1000	0.0566	0.0953	0.0635
ΔMY	0.5000	0.4713	0.4854	0.4739
ΔMZ	0.4000	0.3796	0.4003	0.3824
ΔM	2.0000	2.0085	1.9862	2.0069

Table 2.5 – Comparison between IDIM-OLS and modulating functions identification approaches

2.4 Conclusion

The modulating function approach is an extension of the robot energy identification method. Compared to the energy model, the integration with different groups of modulating function will ensure the condition number and rank efficiency of the observation matrix. Compared to robot dynamic model, the terms containing joint acceleration disappear because of the integration, so that only joint position, velocity and joint force/torque are required. Once the joint velocities are well estimated, this modulating function with power model method can give a good estimation of dynamic parameters. While the friction parameters are difficult to estimate because they are less important in the model and sensitive to noise. In the end, several simulation tests show that the identification model with power model and modulating functions is an efficient identification approach. And the experimental identification on the 2R prototype robot shows that the modulating functions with power model method has better precision than IDIM-OLS method.

In the fourth chapter, comparisons are done among all robot identification models. It will prove that the modulating functions with robot power model method is a robust identification approach.

Chapter 3

Differentiation in parameters identification of robots

3.1 Introduction

The robot identification issues have been widely studied in the past decades, but there still exist several open questions. One of them concerns with estimating the derivatives of an unknown signal from its discrete, potentially using noisy measurement. The numerical differentiation is ill-posed in the sense that a small error in the measurement can produce a large error in the estimated derivatives, specially in the case of high order derivatives. Therefore, various numerical methods have been developed to obtain stable algorithms which are robust against additive noises. They mainly fall into the following categories:

- the finite difference methods (Khan et al., 2000; Qu, 1996; Rahul et al., 2006; Ramm et al., 2001),
- the Savitzky Golay methods (Barak, 1995; Diop, Grizzle, Moraal, et al., 1994; Gorry, 1990a; Savitzky et al., 1964a)
- the wavelet differentiation methods (Diop, Grizzle, and Chaplais, 2000; Leung et al., 1998; Nie et al., 2002; Shao and Ma, 2003; Shao, Pang, et al., 2000)
- the Fourier transform methods (Dou et al., 2010; Fu et al., 2010; Kauppinen et al., 1981; Qian, Fu, and Feng, 2006; Qian, Fu, Xiong, et al., 2006; Yang, 2008)
- the mollification methods (Hao et al., 1995; Murio et al., 1998; Murio, 1993),

- the Tikhonov regularization methods (Cullum, 1971; Hanke et al., 2001; Nakamura et al., 2008; Wang, Jia, et al., 2002; Wei et al., 2005),
- the algebraic methods (Liu et al., 2009b, 2011b,d, 2012a; Mboup, 2007; Mboup et al., 2009b),
- the differentiation by integration methods (Lanczos, 1956; Rangarajana et al., 2005; Wang and Wen, 2010), i.e. using the Lanczos generalized derivatives.
- observer design in the control literature (Chitour, 2002; Levant, 1998, 2003; Polyakov et al., 2014).

The numerical differentiators discussed in this chapter are based on algebraic methods. They are rooted in a recent algebraic parametric method introduced by Fliess and Sira-Ramírez (Fliess, Mboup, et al., 2003; Fliess and Sira-Ramirez, 2004). These algebraic differentiators are divided into two classes: model-based differentiators and model-free differentiators. The formers were obtained by applying the algebraic method to a differential equation which defines a class of linear systems (Fliess and Sira-Ramrez, 2004; Tian et al., 2008). Hence, they were mainly used for linear systems. However they have been extended to the model-free differentiators, which can be used for nonlinear systems and various problem in signal processing. The first model-free differentiator was introduced in (Fliess, Join, et al., 2004) by applying the algebraic method to the truncated Taylor series expansion of the signal to differentiate. Then, two model-free differentiators were studied in (Mboup et al., 2007, 2009a), where the so-called Jacobi differentiator is the most used. Moreover, it was shown that the causal Jacobi differentiator can also be obtained by taking the truncated Jacobi orthogonal series expansion of the signal to be differentiated. Then, it was significantly improved by admitting a known time delay chosen by the designer (Mboup et al., 2007, 2009a). In (Liu et al., 2011c), a central Jacobi differentiator was proposed, which is devoted to off-line applications.

The Jacobi differentiator is a non-asymptotic differentiator and has the following advantages. First it is given by an integral formula in the continuous case, which can be considered as a low-pass filter and corresponds to a convolution in the discrete case. Thus, estimations at different instants can be obtained using a sliding integration window of finite length. Moreover it shows robust properties with respect to corrupting noises (Fliess, 2006). Regarding to error analysis of Jacobi differentiator, theoretical works have been done to

demonstrate that the errors are highly nonlinear functions of the designing parameters and the errors are bounded. As well some experimental works show the relation between the error and the designing parameters (Liu et al., 2009a, 2011a, 2012a). However there does not exist yet an effective approach to design the Jacobi differentiator because the parameters are highly coupled. To investigate, this chapter will consider from a FIR filter point of view, to show the cut-off property of the Jacobi differentiator regarding to magnitude frequency response.

In this section, we will construct high order precise numerical derivative differentiators of a smooth functions from an algebraic framework. For this, we consider two cases. In the first case, we use the sampling data given before the point at which the derivative value we want to estimate. The such obtained estimator is called causal differentiator. In the second case, the point at which the derivative value we want to estimate is the middle point of the time window used for data. Hence, we get central differentiator, used for off-line purpose.

3.1.1 A motivating example

The classical numerical differentiation methods, generated from an interpolating polynomial (see (Anderssen et al., 1998; Brown et al., 1992)) or a least-squares polynomial (see (Gorry, 1990b; Savitzky et al., 1964b)), is used to approximate a function, the derivatives of which we want to estimate. Then, the derivatives of this polynomial is closely linked to the coefficients of this polynomial. From these, in the recent papers (Mboup et al., 2007, 2009a), a new algebraic parametric differentiation method is presented where an elimination technique such as the one introduced in (Herceg et al., 1986) was used to calculate the effective coefficients.

We begin with a simple example to demonstrate the principle of algebraic parametric technique. Let $p(t) = a_0 + a_1 t$ be a first order polynomial defined on $\mathbb{R}^+$, where a_0 and a_1 are unknown. The aim is to calculate the first order derivative of $p(t)$, through an elimination technique on the operational domain. Applying the Laplace transform to $t^\alpha, \alpha \in \mathbb{R}$, the operational expression writes as $\hat{p} = \frac{a_0}{s} + \frac{a_1}{s^2}$, where $\hat{p}$ is the Laplace transform of $p(t)$. Then, by multiplying both sides by s, we get $s\hat{p} = a_0 + \frac{a_1}{s}$. Thus, we can annihilate the polynomial coefficient a_0 by deriving with respect to s:

$$s\hat{p}^{(1)} + \hat{p} = -\frac{1}{s^2} a_1. \tag{3.1}$$

As in (3.1), it only remains $\hat{p}$ and a_1 in the operational domain. It is necessary to return to time domain in order to calculate a_1 by using knowledge of $p(t)$. Considering that the inverse Laplace transform of $s\hat{p}^{(1)}$ contains the derivative of $\hat{p}$ which is unknown, we need multiply both sides of (3.1) by s^{-2}, in order to construct the correct inverse Laplace form. Finally, we obtain the analytical solution of a_1 which only depends on $p(t)$:

$$a_1 = \frac{3!}{t^3}\int_0^t (2\tau - t)p(\tau)d\tau,\ t > 0. \tag{3.2}$$

From the previous computations, we define the following differential operator:

$$\Pi_{0,0} = \frac{1}{s^2}\cdot\frac{d}{ds}\cdot s, \tag{3.3}$$

which allows to annihilate the lower order coefficient a_0 and calculate the derivative by an integral. From these property, we call such differentiator **integral annihilator**, and this method **differentiation by integration**. In practice, this kind of differentiator is actually a band-pass filtering procedure, where its differentiation property annihilates the low frequency contribution and the integration by definite integrals is in nature low-pass filtering, thus only band frequency is passed.

We can extend the previous polynomial function to higher order straightforwardly. Inspired by this, we can estimate derivatives of a given smooth function, by taking a suitable truncated Taylor series expansion around a given instant and by applying some integration technique to annihilate the undesired coefficients. In the following part, we present in detail the recently developed Jacobi differentiators.

3.2 Causal Jacobi differentiator

Consider a noisy measurement $x^\omega : I \to \mathbb{R}$, $x^\omega(t) = x(t) + \omega(t)$, where I is a finite time open interval of $\mathbb{R}^+$, $x \in \mathcal{C}^n(I)$ with $n \in \mathbb{N}$, and ω is an additive corrupting noise. The objective is to estimate the n^{th} order derivative of x using x^ω. Contrary to (Rangarajana et al., 2005) where the n^{th} order Legendre polynomials were used, we apply the Jacobi polynomials to get the truncated Jacobi orthogonal series to estimate the n^{th} order derivative, which were introduced in (Mboup et al., 2007, 2009a).

First, for any $t_0 \in I$, we introduce the set $D_{t_0} := \{t \in \mathbb{R}_+^*; t_0 - t \in I\}$. Define the

i^{th} order shifted Jacobi orthogonal polynomial on interval $[0,1]$ as follows (see (Abramowitz et al., 1965) pp. 774-775), with parameters $\mu, \kappa \in]-1, +\infty[$:

$$P_i^{(\mu,\kappa)}(\tau) = \sum_{j=0}^{i} \binom{i+\mu}{j} \binom{i+\kappa}{i-j} (\tau-1)^{i-j} \tau^j. \tag{3.4}$$

Define the $\mathcal{L}^2([0,1])$ scalar product $\langle \cdot, \cdot \rangle_{\mu,\kappa}^{(0,1)}$ with the associated weight function $\hat{w}_{\mu,\kappa}(\tau) = (1-\tau)^\mu \tau^\kappa$, let us denote $\forall\, g_1, g_2 \in \mathcal{C}[0,1]$,

$$\langle g_1, g_2 \rangle_{\mu,\kappa}^{(0,1)} = \int_0^1 \hat{w}_{\mu,\kappa}(\tau) g_1(\tau) g_2(\tau) d\tau. \tag{3.5}$$

Hence the associated norm for the i^{th} order shifted Jacobi orthogonal polynomial is given as: $\|P_i^{(\mu,\kappa)}\|_{\mu,\kappa}^2 = \frac{1}{2i+\mu+\kappa+1} \frac{\Gamma(\mu+i+1)\Gamma(\kappa+i+1)}{\Gamma(\mu+\kappa+i+1)\Gamma(i+1)}$, where $\Gamma(n)$ is the classical Gamma function (see (Abramowitz et al., 1965) p. 255), with $\Gamma(n) = (n-1)!$.

The derivation of causal Jacobi differentiator is discussed in detail in (Liu et al., 2012a), and a general demonstration are done in appendix A.2. Here, we give the analytical continuous expression of the causal Jacobi differentiator, which calculates the n^{th} derivative at instant t_0, $\forall \xi \in [0,1], \forall t_0 \in I$,

$$D_{\kappa,\mu,T,q}^{(n)} x(t_0 - T\xi) = \frac{1}{(-T)^n} \int_0^1 Q_{\kappa,\mu,n,q,\xi}(\tau)\, x(t_0 - T\tau) d\tau, \tag{3.6}$$

with $\mu, \kappa \in]-1, +\infty[$,

$$C_{\kappa,\mu,n,i} = \frac{(\mu+\kappa+2n+2i+1)\Gamma(\kappa+\mu+2n+i+1)\Gamma(n+i+1)}{\Gamma(\kappa+n+i+1)\Gamma(\mu+n+i+1)}, \tag{3.7}$$

$$Q_{\kappa,\mu,n,q,\xi}(\tau) = \hat{w}_{\mu,\kappa}(\tau) \sum_{i=0}^{q} C_{\kappa,\mu,n,i} P_i^{(\mu+n,\kappa+n)}(\xi) P_{n+i}^{(\mu,\kappa)}(\tau). \tag{3.8}$$

Finally, we substitute x in (3.6) by x^ω so as to obtain the causal Jacobi differentiator $D_{\kappa,\mu,T,q}^{(n)} x^\omega(t_0 - T\xi)$ in noisy case.

Different from the existing polynomial approaches, the idea of the causal Jacobi differentiator is to use a sliding integration window to estimate the value of $x^{(n)}$ at each $t_0 \in I$ by $D_{\kappa,\mu,T,q}^{(n)} x(t_0 - T\xi)$ with a fixed value of $\xi \in [0,1]$ [1] (see (Mboup et al., 2007, 2009a)). If $\xi \neq 0$, then it produces a delay of value $T\xi$.

It is clear that for each $t_0 \in I$, the causal Jacobi differentiator $D_{\kappa,\mu,T,q}^{(n)} x^\omega(t_0 - T\xi)$ depends on a set of design parameters, except for the order of the desired

[1] According to (Mboup et al., 2009a), an optimal value of ξ is given, we discuss in chapter (3.2.1)

derivative n:

- $\kappa, \mu \in]-1, +\infty[$: the parameters of Jacobi polynomials,
- $q \in \mathbb{N}$: the order of truncated Jacobi series expansion,
- $T \in D_{t_0}$: the length of the sliding integration window,
- $\xi \in [0,1]$: the parameter of time-delay $T\xi$.

3.2.1 Error Analysis in Time Domain

The estimation error of the causal Jacobi differentiator can be decomposed in the continuous case as follows:

$$\begin{aligned}
&D^{(n)}_{\kappa,\mu,T,q} x^{\omega}(t_0 - T\xi) - x^{(n)}(t_0) \\
&= \left(D^{(n)}_{\kappa,\mu,T,q} x(t_0) - x^{(n)}(t_0 - T\xi) \right) + \left(D^{(n)}_{\kappa,\mu,T,q} x^{\omega}(t_0 - T\xi) - D^{(n)}_{\kappa,\mu,T,q} x(t_0 - T\xi) \right) \\
&+ \left(D^{(n)}_{\kappa,\mu,T,q} x(t_0 - T\xi) - D^{(n)}_{\kappa,\mu,T,q} x(t_0) \right) \\
&= e_{R_n}(t_0; \kappa, \mu, T, q) + e_{\omega}(t_0; n, \kappa, \mu, T, q) + e_h(t_0; n, \kappa, \mu, T, q). \qquad (3.9)
\end{aligned}$$

1) **Noise error contributions** $e_{\omega}(t_0; n, \kappa, \mu, T, q)$: in continuous case, consider the noise errors due to the two following categories of noises.

- Integrable noises: In this case the noise is assumed to be a bounded and integrable function on I, which can be divided into two parts (Fliess, Mboup, et al., 2003): the first part is a $(n-1)^{th}$ order polynomial, considered as a *structured perturbation*, and the seconded part is a high frequency perturbation, considered as an *unstructured noise*.

An error bound based on the integral formula given in (3.6) was proposed in (Liu, 2011) (p. 90) for this kind of noise errors. Moreover, it was shown that the Jacobi differentiator $D^{(n)}_{\kappa,\mu,T,q} x^{\omega}(t_0 - T\xi)$ can eliminate a $(n-1)^{th}$ order structured perturbation.

- Non-independent stochastic process noises: consider in this case a class of continuous stochastic processes with finite second moments, whose mean value function and covariance kernel are continuous functions (Parzen, 1999), such as Brownian motion and Poisson process.

Since a stochastic process usually is not bounded, the Bienaymé-Chebyshev inequality was used to give error bounds by calculating the mean value and the variance of the associated noise error (Liu et al., 2011a).

2) **Truncated term error** $e_{R_n}(t_0; \kappa, \mu, T, q)$: Using the Taylor series expansion of x at t_0, an error bound for the amplitude error was provided in (Liu et al., 2012b), where the $(n+q+1)^{th}$ order derivative of x was assumed to be bounded on I.

3) **Delay error** $e_h(t_0; n, \kappa, \mu, T, q)$: Instead of giving an error bound for the delay error, existing studies were based on how to chose ξ and on how to reduce the delay $T\xi$ if $\xi \neq 0$.

• On the one hand, according to (Mboup et al., 2009a), we can take $\xi = \xi^{(n)}_{\kappa,\mu,q}$ in $D^{(n)}_{\kappa,\mu,T,q} x^{\omega}(t_0 - T\xi)$, where $\xi^{(n)}_{\kappa,\mu,q}$ is the smallest root of $P^{(\mu+n,\kappa+n)}_{q+1}$. This choice of ξ significantly improved the Jacobi differentiator by admitting the time-delay $T\xi$. Indeed, it was shown in (Mboup et al., 2009a) that $e_h(t_0; n, \kappa, \mu, T, q, \xi^{(n)}_{\kappa,\mu,q}) < e_h(t_0; n, \kappa, \mu, T, q+1, 0) < e_h(t_0; n, \kappa, \mu, T, q, 0)$.

• On the other hand, $\xi^{(n)}_{\kappa,\mu,q}$ depends on three design parameters. The influence of q is given by the classical orthogonal polynomial theory. The influence of κ and μ was studied in (Liu et al., 2011a).

• Another choice of ξ is to take $\xi = 0.5$ (this case corresponds to the central Jacobi differentiator, see (Liu, 2011; Liu et al., 2011c)). It is the optimal value of ξ which minimizes the noise error contribution. However, the time-delay is equal to $0.5T$. Hence, this case is only considered for off-line applications.

Finally, by numerically[2] calculating the noise error bound, the amplitude error bound and the time-delay, we can know their behaviors with respect to different design parameters. Then, we can deduce the influence of these design parameters on each source of errors. We summarize the obtained results in Table (3.1) (see (Liu, 2011; Liu et al., 2011a) for more details), where the notations $a \uparrow$, $b \nearrow$ and $c \searrow$ mean that if we increase the value for the parameter a then the error b increases and the error c decreases. Consequently, it is interesting to take negative values of κ to reduce both the truncated term error and the noise error contribution (see (Liu et al., 2011a) for more details).

Meanwhile, in discrete case, the integral formula of the causal Jacobi differentiator should be approximated by taking a numerical integration method, which implies a numerical error. Compared to Simpson's integration rule, it indicates the trapezoidal integration rule is the optimal numerical integration method to reduce the noise error contribution. In the following part, we consider in discrete case and study the frequency domain property.

3.2.2 Frequency Domain Analysis

Inspired by (Chen et al., 2011), we investigate the frequency response. Seen from equation (3.6), the causal Jacobi differentiator is a combination of integrals with measurement in $[t_0 - T, t_0]$. The discrete version Jacobi differentiator

[2]It is very difficult to analytically study the behavior of each error bound due to their complex expressions.

	Noise error	Truncated error	Time-delay
$\kappa \uparrow$	↗	↗	↗
$\mu \uparrow$	↗	↘	↘
$q \uparrow$	↗	↘	↘
$T \uparrow$	↘	↗	↗

Table 3.1 – Influence of design parameters on $D^{(n)}_{\kappa,\mu,T,q}x(t_0-T\xi^{(n)}_{\kappa,\mu,q})$ in continuous case.

writes as the sum of a list of weighting coefficients associated with measurement. In this sense, the Jacobi differentiator can be seen as a FIR filter applied to a discrete system with sampling time T_s. By studying the frequency domain behavior of the FIR filter, we can investigate the filtering and differential properties of the Jacobi differentiator. After extracting the weighting coefficients, we can draw the bode plot of Jacobi differentiator, as a digital filter with sampling time T_s.

Given a signal that is the sum of three sinusoidal waves with amplitude 1 and frequency 4 Hz, 9 Hz, 15 Hz respectively. To get the second order derivative, the causal Jacobi differentiator is applied, with comparison of two on-line approaches: Euler causal difference with a traditional FIR filter and a forward Butterworth filter.

FIR filters can be designed as a linear-phase filter (but they do not have to be), where their coefficients are symmetrical around the centre coefficient (McClellan et al., 1973). Linear-phase means all frequency components are shifted in time by the same amount, where no distortion happens with frequency during the filtering process. When the order of linear-phase FIR filter is N_f, the delay $T_{\Delta FIR} = \frac{N_f-1}{2F_s}$, where F_s is the sampling frequency. Butterworth filter is an IIR filter and referred to as a maximally flat magnitude filter. It is widely used in off-line case with a zero phase forward-backward filtering process. So comparison with Butterworth filter technique is interesting.

We simulate the measurement of the signal by superimposing an uniformly distributed random noise of amplitude 0.3 and a sinusoidal noise of amplitude 0.2 with frequency 200 Hz on the signal. The sampling frequency is 1 millisecond. To derive such a signal, the causal Jacobi differentiator is set with $\kappa = 2$, $\mu = 2$, $q = 2$ and the sliding integration window $T = 0.054$ second. In this configuration, the causal Jacobi differentiator causes a delay of $T_{\Delta causal} = T\xi = 0.0189s$. Allowing the same amount of delay as causal Jacobi differentiator, $N_f = 38.8$ but the FIR filter order N_f should be integer thus set $N_f = 39$, with cutoff frequency at 25 Hz. A well tuned forward Butterworth fil-

ter configuration is of order 4 with cutoff frequency at 25 Hz. Then the filtered data is derived by Euler causal difference with transfer function $H(z) = \frac{1-z^{-2}}{2T_s}$.

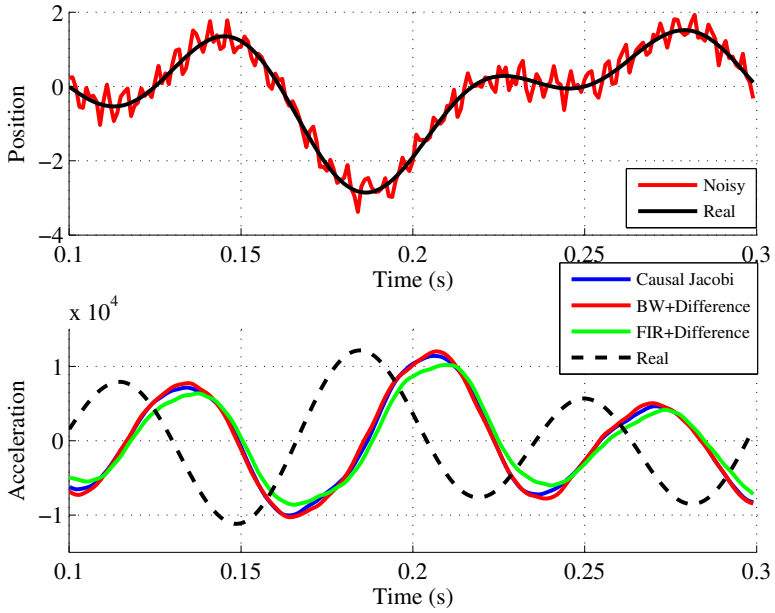
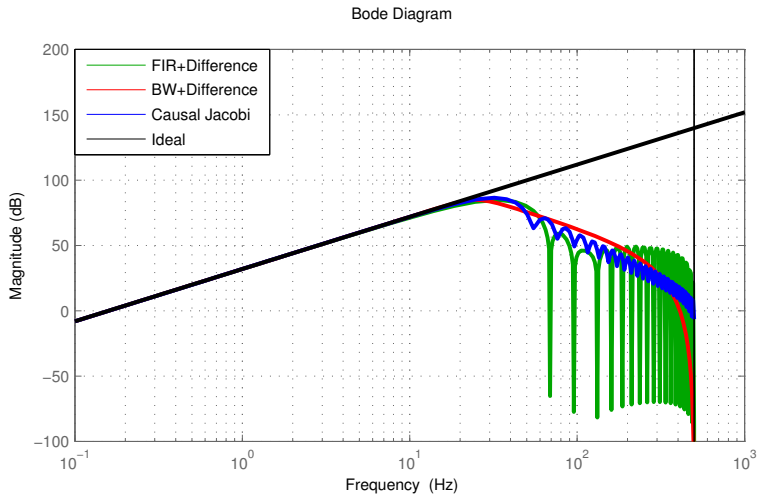


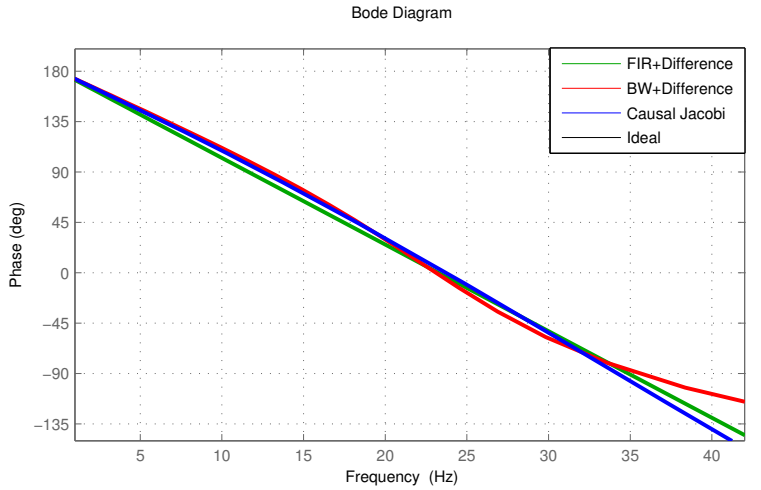
Figure 3.1 – Estimated acceleration with causal Jacobi differentiators

The derivative estimation for acceleration are shown in Fig. (3.1). In the middle figure, the causal Jacobi differentiator and causal difference with linear phase FIR filter are applied allowing the same amount delay analytically known. While the causal difference with Butterworth filter does not guarantee the linear phase property. The original acceleration with black line is computed without noise using the analytical form. As well, in the third figure, the estimations are shifted to right time line in order to compare. The results show that causal Jacobi differentiator is better than FIR filter, and presents a good precision similar to the Butterworth forward filter with Euler causal difference approaches. They are robust to different noises.

The frequency domain analysis is shown in Fig. (3.2). In Fig (3.2(a)) the magnitude frequency response plot, the FIR filter with Euler causal difference method has the fastest descent in the beginning of the unwanted frequency, but in high frequency part the magnitude response is not attenuated. The Butterworth filter with Euler causal difference method forward has the smoothest descent and cuts off completely the high frequency component. Comparatively the causal Jacobi differentor has an intermediate behavior, because it has better descending rate than FIR filter method and less attenuation in high frequency part than Butterworth forward filter method. From Fig (3.2(b)) the phase fre-



(a) magnitude response



(b) phase response

Figure 3.2 – Bode plot of second order causal Jacobi differentiators

quency response, the causal Jacobi differentiator presents a good linear-phase property as FIR filter method. The FIR filter method and causal Jacobi differentiator allows quite the same amount of delay, which is indicated in figure that they have the similar slope of curve. Compared to FIR filter, causal Jacobi differentiator can induce delay in smaller scale such as 0.1 millisecond. While the forward Butterworth filter method has a phase distortion, which makes it less interesting in on-line case.

To investigate influence of parametrization in causal Jacobi differentiator, we analyse in the bode plot as shown in Fig. (3.3).

1) κ mainly influences the low frequency contribution. A good selection

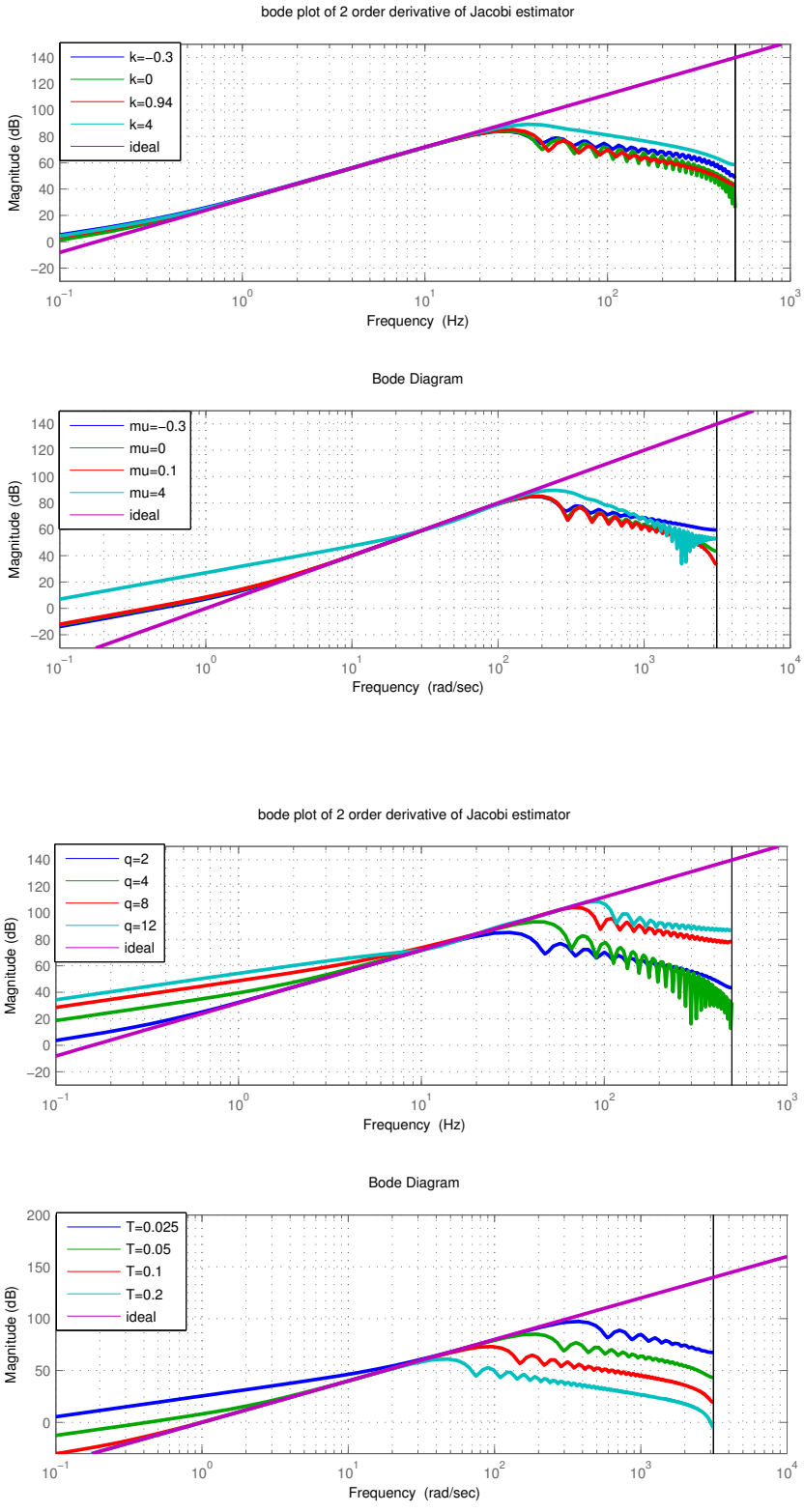


Figure 3.3 – Causal Jacobi differentiator parameters influence on bode plot when $T_s = 0.001s$

of κ can ensure the low frequency response close to 0. As κ increases, the descending behaviors of the causal Jacobi differentiators vary. It can be found that around $\kappa = 0.94$, it has a fast descending of magnitude frequency response. Besides, the cutoff point is slightly pushed to higher frequency when κ increase.

2) μ is related to descending rate in the high frequency part. When μ increase, high frequency response descends faster but low frequency response will increase largely. As μ goes up, similarly the cutoff point slightly moves to higher frequency.

3) q affects the truncated term error. When q increase, the cutoff frequency area largely moves to higher frequency but it causes bias in low frequency response. In most cases, $q = 2$ is the best trade-off condition.

4) T is the sliding integration window and when the sampling time T_s is fixed. When sampling time is fixed, it represents the points required for causal Jacobi differentiator. As T increase, the cutoff frequency largely moves to lower frequency and it presents a robust differentiator property at low frequency.

Conclusion: compared to the time domain error analysis given in table (3.1), the frequency domain analysis gives the corresponding explications. As μ, q increase, the cutoff point of causal Jacobi differentiator moves to higher frequency, which means it estimates with more frequency component, for one part, it should enlarge the high frequency noise error; for other part, more frequency components in the truncated expansion of original signal are utilized so that the truncated error should be decreased. By contrast, as T increases, the situation is opposite. For μ, q, T, the frequency domain analysis fits well with time domain analysis.

While for κ, the effect is not obvious because as κ goes up, the behaviors in frequency domain vary without monotony and their behaviors are similar (as shown in Fig. (3.3(a))). From the figure, there should be an optimal κ, and we can conclude that κ has small influence on differentiator performance.

From the previous analysis, the causal Jacobi differentiator is regarded as a low-pass differentiator. The low-pass property is inherent because it considers the signal as a certain order polynomial in a small time window and uses the truncations to estimate the derivatives. Compared to other online differentiators, its frequency magnitude response performance behaves less robust but it has good linearity in phase and the exact time delay $T\xi$ can be analytically computed.

3.3 Central Jacobi Differentiator

The central Jacobi differentiator can be regarded as a special case of the causal Jacobi differentiator by admitting $\xi = 0.5$. Recall the causal Jacobi differentiator (3.6):

$$D^{(n)}_{\kappa,\mu,T,q}x(t_0 - T\xi) = \frac{1}{(-T)^n}\int_0^1 Q_{\kappa,\mu,n,q,\xi}(\xi)\,x(t_0 - T\tau)d\tau.$$

When $\xi = 0.5$, $D^{(n)}_{\kappa,\mu,T,q}x(t_0 - 0.5T)$ estimates the n^{th} order derivatives at instant $t_0 - 0.5T$, using the data collected from the interval $[t_0 - T, t_0]$. Replace t_0 by $t_0 + 0.5T$, and it becomes the central Jacobi differentiator:

$$D^{(n)}_{\kappa,\mu,T,q}x(t_0) = \frac{1}{(-T)^n}\int_0^1 Q_{\kappa,\mu,n,q,0.5}(\tau)\,x(t_0 + 0.5T - T\tau)d\tau, \tag{3.10}$$

which estimates the n^{th} order derivatives at instant t_0, with a symmetric sampling of causal and anti-causal data from the interval $[t_0 - \frac{T}{2}, t_0 + \frac{T}{2}]$.

3.3.1 Another access to central Jacobi differentiator

In the above, we discuss how to get central Jacobi differentiator with obtained causal Jacobi differentiator. Here, we consider from a base of central Jacobi orthogonal polynomial, to prove the validity of the central Jacobi differentiator.

First introduce the central Jacobi orthogonal polynomial defined on $[-1, 1]$ as follows: (see (Abramowitz et al., 1965)):

$$\overline{P}_i^{(\mu,\kappa)}(\tau) = \sum_{j=0}^{i}\binom{i+\mu}{j}\binom{i+\kappa}{i-j}\left(\frac{\tau-1}{2}\right)^{i-j}\left(\frac{\tau+1}{2}\right)^{j}, \tag{3.11}$$

with $\mu, \kappa \in]-1, +\infty[$. Let us denote $\langle\cdot,\cdot\rangle^{(-1,1)}_{\mu,\kappa}$ as a $\mathcal{L}^2([-1,1])$ scalar product with the associated weight function $\overline{w}_{\mu,\kappa}(\tau) = (1-\tau)^{\mu}(1+\tau)^{\kappa}$. Then, we have $\forall\, \overline{g}_1, \overline{g}_2 \in \mathcal{C}[-1, 1]$,

$$\langle \overline{g}_1, \overline{g}_2\rangle^{(-1,1)}_{\mu,\kappa} = \int_{-1}^{1}\overline{w}_{\mu,\kappa}(\tau)\overline{g}_1(\tau)\overline{g}_2(\tau)d\tau. \tag{3.12}$$

And the associated norm is given as:

$$\|\overline{P}_i^{(\mu,\kappa)}\|^2_{\mu,\kappa} = \frac{2^{\mu+\kappa+1}}{2i+\mu+\kappa+1}\frac{\Gamma(\mu+i+1)\Gamma(\kappa+i+1)}{\Gamma(\mu+\kappa+i+1)\Gamma(i+1)}. \tag{3.13}$$

For any $t_0 \in I$, introduce the set $D'_{t_0} = \{t \in \mathbb{R}^*_+ | [t_0 - t, t_0 + t] \in I\}$, and suppose $h \in D'_{t_0}$. Let x still be a smooth function in $C^n(I)$. According to (Liu et al., 2011c),

the central Jacobi differentiator $\overline{D}^{(n)}_{\kappa,\mu,h,q}x(t_0)$ can be formulated using the above notations (for more details a general demonstration are listed in appendix A.3).

$$\overline{D}^{(n)}_{\kappa,\mu,h,q}x(t_0) = \frac{1}{h^n}\int_{-1}^{1} O_{\kappa,\mu,n,q}(\tau)\, x(t_0+h\tau)d\tau, \tag{3.14}$$

where $2h$ is entire sliding time window,

$$\rho_{n,\kappa,\mu}(\tau) = \frac{2^{-(n+\kappa+\mu+1)}n!\Gamma(2n+\kappa+\mu+2)}{\Gamma(n+\kappa+1)\Gamma(n+\mu+1)}, \tag{3.15}$$

$$O_{\kappa,\mu,n,q}(\tau) = \sum_{i=0}^{q}\overline{P}_i^{\mu+n,\kappa+n}(0)\sum_{j=0}^{i}(-1)^{i+j}\binom{i}{j}\frac{2i+\kappa+\mu+2n+1}{i+\kappa+\mu+2n+1}\rho_{n,\kappa+i-j,\mu+j}(\tau). \tag{3.16}$$

Finally, x is substituted in (3.14) by x^{ω} in order to obtain the central Jacobi differentiator $\overline{D}^{(n)}_{\kappa,\mu,h,q}x^{\omega}(t_0)$ in noisy case. Because the the central Jacobi differentiator depends on the causal and anti-causal data, so the the length of the sliding integration window of central Jacobi differentiator $T = 2h$. Similarly the central Jacobi differentiator depends on the same design parameters as causal Jacobi differentiator.

- $\kappa, \mu \in]-1, +\infty[$: the parameters of Jacobi polynomials,
- $q \in \mathbb{N}$: the order of truncated Jacobi series expansion,
- $h \in D_{t_0}$: half length of the sliding integration window.

3.3.2 Error Analysis in Time Domain

The estimation error of the central Jacobi differentiator can be decomposed in continuous case as follows:

$$\begin{aligned}\overline{D}^{(n)}_{\kappa,\mu,h,q}x^{\omega}(t_0) - x^{(n)}(t_0) &= \left(\overline{D}^{(n)}_{\kappa,\mu,h,q}x - x^{(n)}(t_0)\right) \\ &\quad + \left(\overline{D}^{(n)}_{\kappa,\mu,h,q}x^{\omega}(t_0) - \overline{D}^{(n)}_{\kappa,\mu,h,q}x(t_0)\right) \\ &= e_{\omega}(t_0;n,\kappa,\mu,h,q) + e_{R_n}(t_0;\kappa,\mu,h,q).\end{aligned} \tag{3.17}$$

where $e_{\omega}(t_0;n,\kappa,\mu,h,q)$ and $e_{R_n}(t_0;\kappa,\mu,h,q)$ refer to the noise error contribution and the truncated term error respectively. Corresponding error bounds have been provided in (Liu et al., 2011c). Finally, by numerically calculating these error bounds, their behaviors with respect to different design parameters can be known. Then, the influence of these design parameters on each source of errors can be deduced. The obtained results are summarized in Table (3.2) (see (Liu,

2011; Liu et al., 2011c) for more details). According to Table (3.2), the design parameters' influence on different errors is not the same. Thus, a compromise among these parameters should be taken.

	Noise error contribution	Truncated error
$\kappa \uparrow$	↗	↘
$\mu \uparrow$	↗	↘
$q \uparrow$	↗	↘
$h \uparrow$	↘	↗

Table 3.2 – Influence of design parameters on $\overline{D}^{(n)}_{\kappa,\mu,h,q} x(t_0)$ in continuous case.

3.3.3 Error Analysis in Frequency Domain

The central Jacobi differentiator can also be considered as a FIR filter. After extracting the weighting coefficients, we can draw its magnitude frequency response plot, as a digital filter with sampling time T_s. Central Jacobi differentiator utilizes the non-causal data which means it is off-line differentiator.

Given the same signal mentioned in causal Jacobi case, the central Jacobi differentiator is applied to get the second order derivative. To compare, the Euler central difference methods with a forward-backward Butterworth filter are discussed. The forward-backward Butterworth filter is a zero phase IIR filter and referred to as a maximally flat magnitude filter. It is widely used in various applications so comparison with Butterworth filter technique is interesting.

In the noise-free case, both differentiators get the precise estimation. With noise case, we simulate the measurement of the signal by superimposing together on the signal a normally distributed random noise of amplitude 0.2, a 200 Hz high frequency sinusoidal wave of amplitude 0.2 and a Poisson distributed random noise with mean parameter $\lambda = 0.1$ of amplitude 0.2. The sampling frequency is 1 millisecond. In order to estimate the derivatives of the original signal , the central Jacobi differentiator is applied by taking $\kappa = \mu = 12$, $q = 6$ and the sliding integration window $T = 0.21$ second. A well tuned forward Butterworth filter configuration is of order 6 with cutoff frequency at 25 Hz. The forward-backward process is done by adding poles in the denominator of transfer function with negative values.

The estimation errors in velocity and acceleration are shown in Fig. 3.4. The result shows that central Jacobi differentiator can be accurate and robust as Euler central differentiation with a well tuned Butterworth filter. It can be

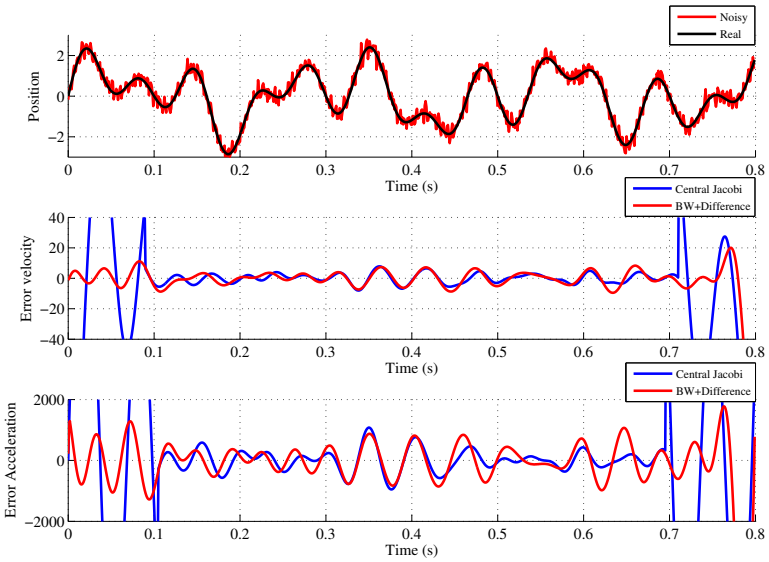


Figure 3.4 – Derivative errors in velocity and acceleration with central Jacobi differentiator

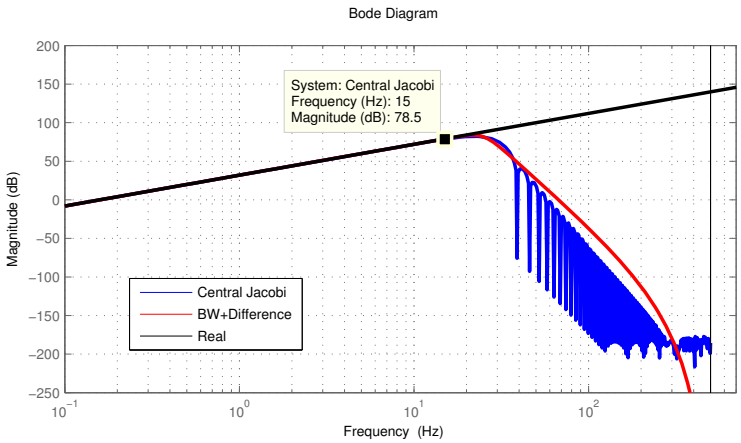


Figure 3.5 – Bode magnitude plot of second order central Jacobi differentiators

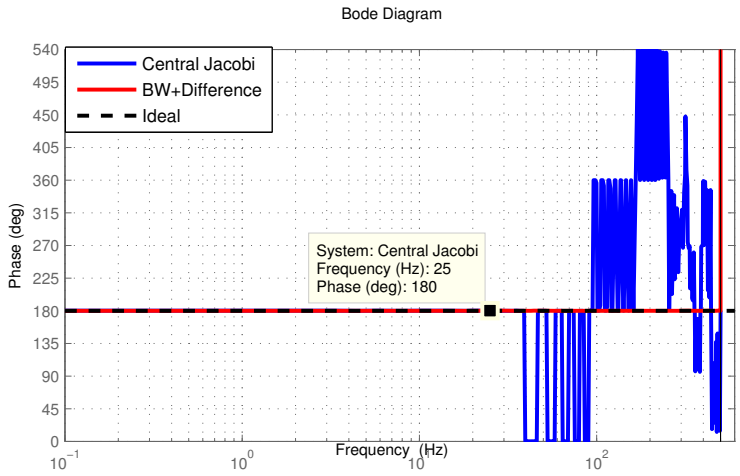


Figure 3.6 – Phase response of second order central Jacobi differentiators

seen that the estimation errors for the central Jacobi differentiator is larger at the beginning and the end, because there is not enough data for the estimation.

In frequency domain, the magnitude bode plots of second order derivative are shown in Fig. 3.5. The black line presents the ideal case with transfer function $H(s) = s^2$ in continuous time. The Jacobi differentiator and Euler differentiation with Butterworth filter are in discrete case. Notice the cutoff frequency of Butterworth filter is at 25 Hz. Under 15 Hz, the magnitude frequency response follows quite well the ideal curve for both Jacobi differentiator and Butterworth method. Above 15 Hz, the magnitude frequency response begins to descend rapidly, especially for the Jacobi differentiator. This means that the Jacobi differentiator has a better cutoff property and the unexpected frequency is attenuated quickly to 0 magnitude response. From the phase bode plot shown in Fig. (3.6), both differentiators have the good phase linear-phase property at low frequency. Above the cutoff frequency 25 Hz, central Jacobi differentiator has several jumps of 180 degrees in phase, and has disturbance in high frequency above 100 Hz, where the magnitude response is much attenuated so that the phase distortion does not make much influence. The distortion may come from numerical error when calculating the coefficients of central Jacobi differentiator.

From the magnitude bode plot, we can analyze in Fig. (3.7) the influence of parametrization in central Jacobi differentiator.

1) $\kappa = \mu$, these parameters are chosen to be identical because this configuration reduce the truncated term error (Liu et al., 2011c). They have two aspects of influence. When the value increases, the descending period will be short-

ened and the unwanted frequency-magnitude response drops fast to 0; on the other hand it will affect the numerical integration error, which mainly causes a bias at 0 Hz frequency-magnitude response.

2) q affects the truncated term error. When q increase, at low frequency part the magnitude response remains the same but at high frequency part, the cutoff frequency increase and the drop rate of descending period remains the same. This means the functioning area is enlarged.

3) T is the sliding integration window and when the sampling time T_s is fixed, it represents the points taken for Jacobi differentiator. When T increase, the cutoff frequency will move to lower frequency and the numerical integration error is reduced, which presents a robust differentiator property at low frequency.

Conclusion: the frequency domain analysis corresponds well with time domain error analysis described in table (3.2). For parameters $\kappa = \mu$,q, as their values increase, the cutoff point moves to high frequency, which indicates noise error will enlarge and truncated error decreases. In the opposite way, it works with parameter T.

From the previous analysis, the Jacobi differentiator is regarded as a low-pass differentiator. The low-pass property is inherent because it considers the signal as a certain order polynomial in a small time window and uses the truncations to estimate the derivatives. From an empirical point of view, the Jacobi differentiation functioning frequency is less than 10% of the sampling frequency. Compared to Euler central differentiation with Butterworth filter, it requires more data thus it is more robust with respect to noise, and its descending period is shorter.

In practice, there are several parameters to tune, which make the regulation difficult to apply. Here we give a general rule to select these parameters. As causal and central Jacobi are similar, we discuss the central case. From the discussion above, κ, μ, q, T all have influence on the filtering property. First, q is easier to choose according to the order of derivative we want to compute. For example, $q = 2$ if we want acceleration. Second, it is a good choice to select $\kappa = \mu$ and fix the value. Normally, the value can be chosen within $]-1, 10[$. Then, last we select the time window T, which is more important, because it has direct influence on the pass frequency of the Jacobi differentiator. The analyse can be done by drawing bode plot.

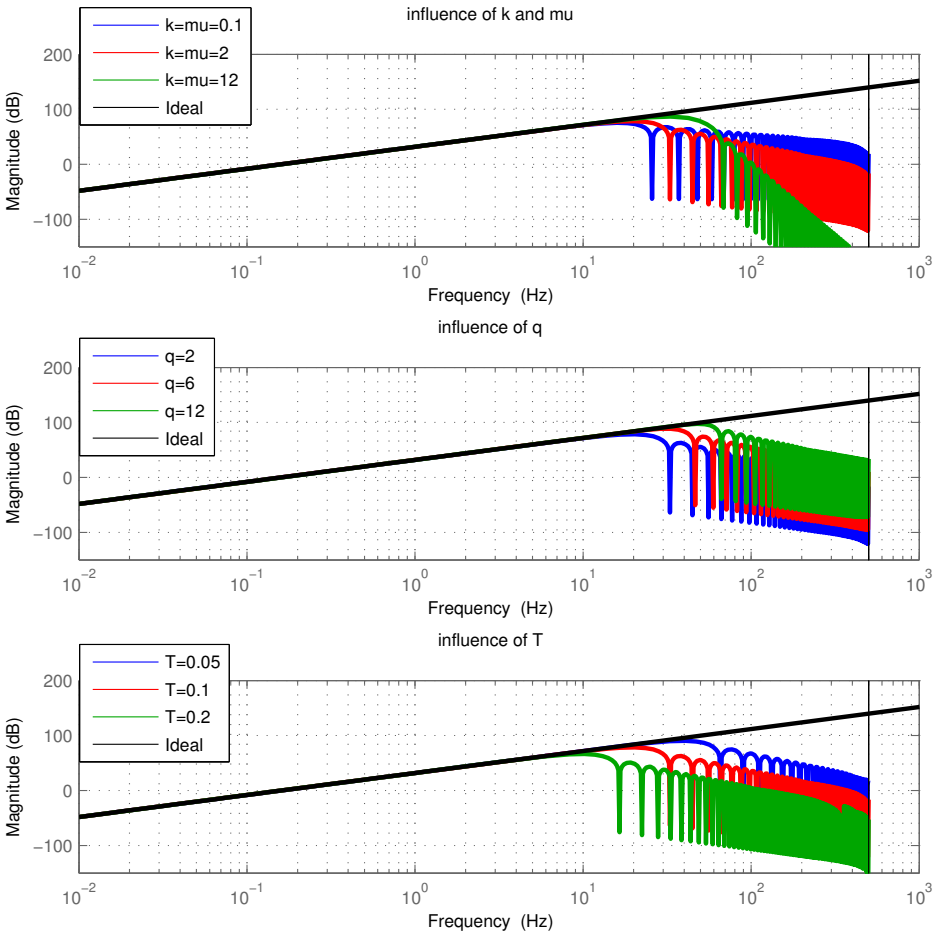


Figure 3.7 – Central Jacobi differentiator parameters influence on bode plot when $T_s = 0.001s$

3.4 Dynamic parameters identification of 2R robot

In this part, we apply the causal and central Jacobi differentiators in robot identification issue. Here the IDIM-LS model is implemented. First we consider the simulation test. The simulation tests use the 2R robot model described in chapter (1.1.1), without considering the friction offset parameter Γ_{off}. The eight minimal dynamic parameters $\mathbf{X}$ are

$$\mathbf{X} = [ZZ_{1R}\ ZZ_2\ LMX_2\ LMY_2\ F_{v1}\ F_{s1}\ F_{v2}\ F_{s2}].$$

3.4.1 Iterative learning identification and computed torque control

Iterative learning control is an efficient method to compensate the variation of system dynamics during the operation. Lots of researches are dedicated to this subject, such as in (Bao et al., 1996; Bristow et al., 2006; Bukkems et al., 2005; Wang, Gao, et al., 2009). Moreover, in recent literature (Gautier, Jubien, et al., 2013) presented the structure of iterative learning identification and computed torque control (IDIM-ILIC) in robot issues, where the computed torque control uses a proportional-Derivative (PD) controller and the IDIM is calculated with noise-free data from the trajectory generator. With noise-free data, it avoids using the noisy derivatives of the actual joint position measurement.

Here, we consider the iterative estimation of the robot dynamic parameters and apply IDIM-ILIC in on-line application, where the robot dynamic parameters are periodically calculated over a moving time window to update the inverse dynamic model of the computed torque controller. And different from (Gautier, Jubien, et al., 2013), we implement a Proportional-Integral-Derivative control for errors in the controller and IDIM is calculated from measurements.

The IDIM-ILIC scheme is shown in Fig. (3.8). The feedback velocity $\dot{\mathbf{q}}_{computed}$ is obtained using the backward difference algorithm of the measured joint position $\mathbf{q}_measure$, in order to ensure real time feedback.

The CTC computes the force/torque control input τ, which is defined such that:

$$\tau = \hat{\mathbf{M}}(\mathbf{q}_{measure})\mathbf{w} + \hat{\mathbf{N}}(\mathbf{q}_{measure}, \dot{\mathbf{q}}_{derivative}), \tag{3.18}$$

where $\mathbf{q}_{measure}$ is the measurement of robot joint position, $\dot{\mathbf{q}}_{derivative}$ is the joint velocity calculated from joint position measurement, $\hat{\mathbf{M}}(\mathbf{q}_{measure})$ is the estimation of symmetric and positive definite inertia matrix, and $\hat{\mathbf{N}}(\mathbf{q}_{measure}, \dot{\mathbf{q}}_{derivative})$

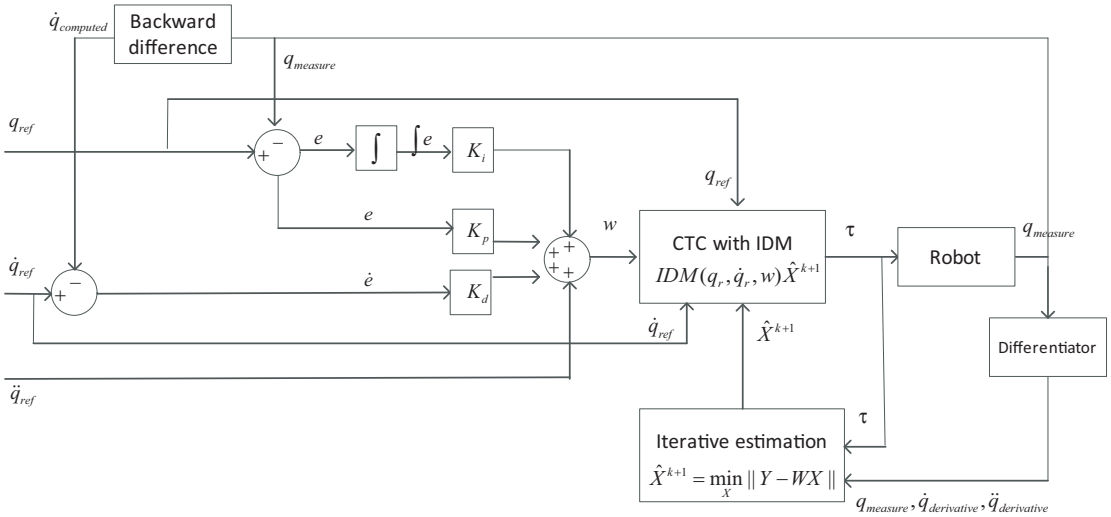


Figure 3.8 – IDIM-ILIC scheme

is the estimations of the Coriolis, centrifugal and gravity force/torque,

$$\hat{\mathbf{N}}(\mathbf{q}_{measure}, \dot{\mathbf{q}}_{derivative}) = C(\mathbf{q}_{measure}, \dot{\mathbf{q}}_{derivative})\dot{\mathbf{q}}_{derivative} + Q(\mathbf{q}_{measure}) + \mathbf{\Gamma}_f$$

The control input $\mathbf{w}$ is dominated by the desired reference acceleration, plus a Proportional-Integral-Derivative control input of errors, where

$$\mathbf{w} = \ddot{\mathbf{q}}_{ref} + K_p\mathbf{e} + K_d\dot{\mathbf{e}} + K_i \int \mathbf{e},$$

with $\mathbf{e} = \mathbf{q}_{ref} - \mathbf{q}_{measure}$ and $\dot{\mathbf{e}} = \dot{\mathbf{q}}_{ref} - \dot{\mathbf{q}}_{derivative}$. K_p, K_i, K_d are positive diagonal matrices of proportional, integral and derivative gains.

Then, the closed-loop system response is determined by the following error equation:

$$\begin{aligned}
&\hat{\mathbf{M}}(\mathbf{q}_{measure})(\ddot{\mathbf{e}} + K_p\mathbf{e} + K_d\dot{\mathbf{e}} + K_i \int \mathbf{e}) \\
&= \mathbf{M}(\mathbf{q}_{measure})\ddot{\mathbf{q}}_{derivative} + \mathbf{N}(\mathbf{q}_{measure}, \dot{\mathbf{q}}_{derivative}) \\
&-\hat{\mathbf{M}}(\mathbf{q}_{measure})\ddot{\mathbf{q}}_{derivative} - \hat{\mathbf{N}}(\mathbf{q}_{measure}, \dot{\mathbf{q}}_{derivative}),
\end{aligned} \tag{3.19}$$

with $\ddot{\mathbf{e}} = \ddot{\mathbf{q}}_{ref} - \ddot{\mathbf{q}}_{derivative}$.

Assume the right part of the equation is 0, which means without modelling error nor error on the measurement or parameter value, the error equation (3.19) becomes

$$\ddot{\mathbf{e}} + K_p\mathbf{e} + K_d\dot{\mathbf{e}} + K_i \int \mathbf{e} = 0. \tag{3.20}$$

The solution $\mathbf{e}$ is the free response of a third order differential equation if we

take the derivative operation on (3.20):

$$\mathbf{e}^{(3)} + K_d\ddot{\mathbf{e}} + K_p\dot{\mathbf{e}} + K_i\mathbf{e} = 0. \tag{3.21}$$

The method for selection of gains are presented in appendix (A.4). The error dynamics depend on the tuning gains. Usually, these gains are selected high enough to get fast dynamics and good robustness to error modelling. While unfortunately, the perfect model assumption is implausible in practice. In fact, the values of parameters are not perfectly known and there are always small errors in the model. Thus, the assumption that right part of (3.19) tends 0 may not hold.

As specified in (Gautier, Jubien, et al., 2013), for strong nonlinear system such as robots, it is impossible to analyse the errors effects. However, we can make some well founded approximations. As shown in (Gautier et al., 2013), the crucial component for the right part is $\mathbf{M}(\mathbf{q}_{measure})\ddot{\mathbf{q}}_{derivative} - \hat{\mathbf{M}}(\mathbf{q}_{measure})\ddot{\mathbf{q}}$. For the vectors of centrifugal and friction force/torque $\mathbf{N}$ and $\hat{\mathbf{N}}$ can be considered as a perturbation. Then, in order to show the error e is bounded, we consider in the one degree of freedom case, where there is only one parameter M, which leads to:

$$\mathbf{e}^{(3)} + K_d\ddot{\mathbf{e}} + K_p\dot{\mathbf{e}} + K_i\mathbf{e} = (\alpha - 1)\ddot{q}_{derivative},\ with\alpha = \frac{M}{\hat{M}}. \tag{3.22}$$

Thus, with α close to 1, with proper gains and because $\ddot{q}_{derivative}$ is bounded, e is bounded. Consequently, we need a robust identification of the inertia parameters and accurate modelling.

3.4.2 On-line Identification

The simulation runs with $\mathbf{X}$ all in SI Units:

$$\mathbf{X} = [3.5\ 0.06\ 0.12\ 0.005\ 0.05\ 0.5\ 0.01\ 0.1]$$

.

The on-line dynamic parameters identification is implemented with IDIM-ILIC, 2R robot simulation model and causal Jacobi differentiator. Here, the gains are selected identical for both joints with $K_{p1} = 49$, $K_{p2} = 580$, $K_{d1} = 31$, $K_{d2} = 46$, $K_{i1} = 30$, $K_{i2} = 3000$ (see appendix A.4). The reference trajectory is composed of a successive point to point trajectories using a classical 5^{th} order polynomial trajectory generator. The simulation is noise free in the measure-

ment. The IDIM-LS is carried out every 3 seconds with estimation time window $T_{est} = 4.5s$, where the estimation window should be large in order to ensure good excitation in the observation matrix and get robust estimation. For the previous 4.5 seconds the dynamic parameters initial condition is set as $ZZ_{1R} = 1$, $ZZ_2 = 0.5$ and the others equal to 0. The feedback velocity is calculated using two steps backward difference of the current measured position.

With causal Jacobi differentiator, the parametrization are chosen as $\kappa = 0.94$, $\mu = 0$, $q = 2$ and $T = 0.1s$ for estimating joint position and torque with delay $0.035s$, $T = 0.5s$ for estimating joint velocity with delay $0.183s$, $T = 0.9s$ for estimating joint acceleration with delay $0.339s$. The joint position, velocity, acceleration and torque are shifted to the same time-line using the known time delay. In Fig. (3.9) it shows the identification results. From the simulation results, the IDIM-LS keeps robustness after the first period of sampling. Tracking error is shown in Fig. (3.10), as well the computed torque is shown in Fig. (3.11). During the robot motion, for the first 6 second, the robot is driven with initial condition, where the controller induces larger error; while aftre 6 second, the estimation are carried out and the tracking error in joint position is less than 0.02 radians and the estimation error is less than 0.05.

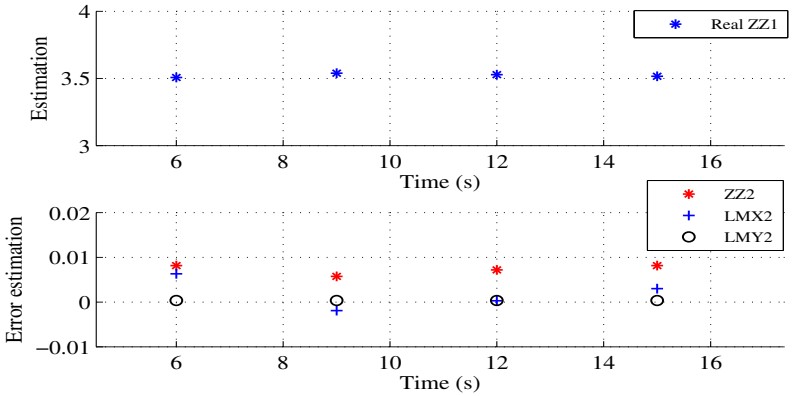


Figure 3.9 – IDIM-ILIC identification results

3.4.3 Non stationary inertial parameter

This part simulates the abrupt change of inertial parameter. The initial dynamic parameters are set to be the same values as those in the previous section. At instant $t = 9s$, ZZ_{1R} changes from 3.9 to 8. The online estimation result is shown in Fig. ()3.12). Notice that from $t = 9s$ to $t = 14s$, there exists a delay of estimation time window T before getting the correction estimation. This delay

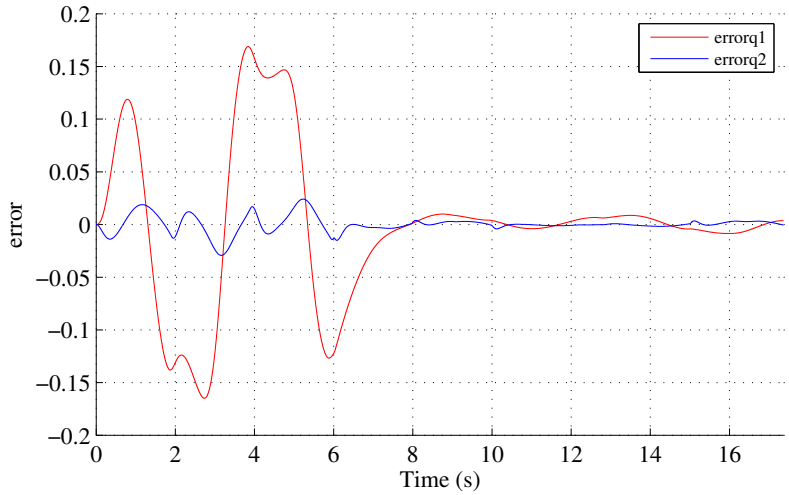


Figure 3.10 – IDIM-ILIC tracking error with simulation

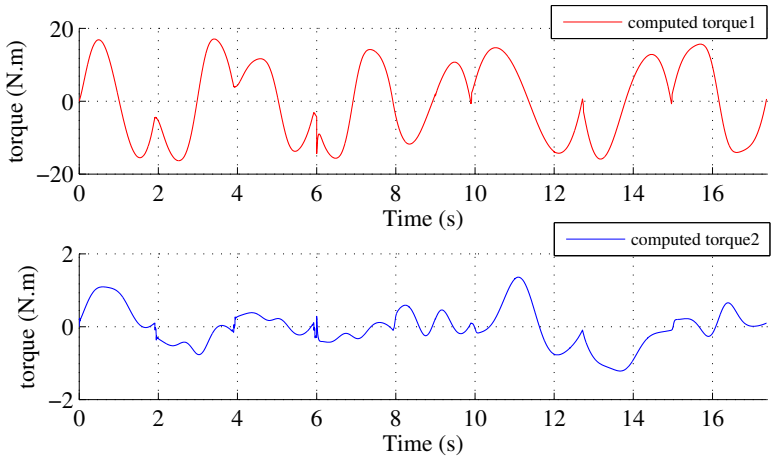


Figure 3.11 – IDIM-ILIC computed torque with simulation

is necessary because it needs enough data to re-estimate the changed parameters. During this transition period all the estimated inertial parameters are varying smoothly to the correct value. During the robot motion, the tracking error in joint position is shown in Fig. (3.13). It can be found, at each iteration, the updating of the parameters will induce a perturbation, resulting in an error in the control.

3.4.4 Offline Identification

The experimental work is done on the two revolute joints planar prototype robot described in chapter (1.1.1). Recall the filtering procedure in chapter (1.1.5), the joint position and torque are pre-filtered to eliminate high frequency noise differentiation. In order to get the filtered $\mathbf{q}$, $\dot{\mathbf{q}}$ and $\ddot{\mathbf{q}}$, set $\kappa = \mu = 2$, $q = 2$

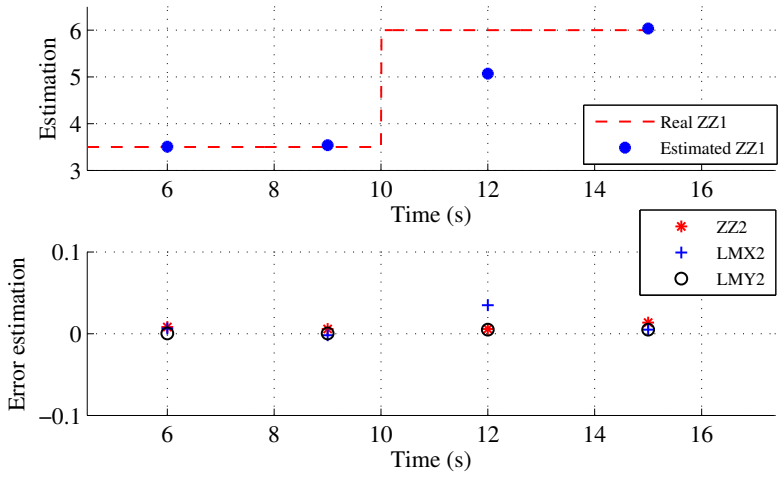


Figure 3.12 – Estimation in ZZ_{1R}, ZZ_2, LMX_2, LMY_2 with variation of ZZ_{1R}

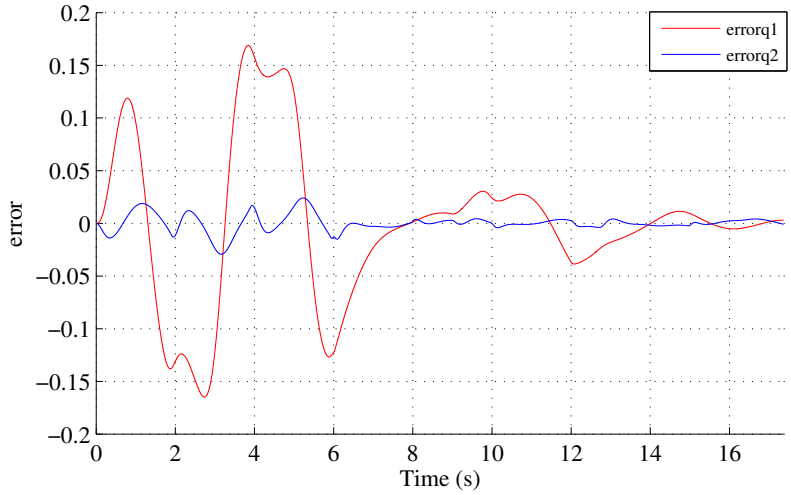


Figure 3.13 – IDIM-ILIC tracking error with simulation and variation of ZZ_{1R}

and $T = 0.16s$ for filtering joint position and torque, $T = 0.24s$ for estimating joint velocity, $T = 0.28s$ for estimating joint acceleration with central Jacobi differentiator. For comparison, also use a low-pass Butterworth filter to eliminate the noise, whose cutoff frequency $\omega_{fq} > 10 \times \omega_{dyn}$ according to the rule of thumb (Gautier, 1996), then apply the Euler central difference algorithm to obtain the time derivative. For the decimate procedure with low-pass filter $F_p(s)$, the decimate ratio n_d can be calculated with $n_d = \frac{\omega_c}{2\omega_{fp}}$ for a FIR filter and $n_d = 0.8 \times \frac{\omega_c}{2\omega_{fp}}$ for an IIR filter, where ω_c is the control rate. Take $\omega_c = 100 \times \omega_{dyn}$ and $\omega_{F_p} = 5 \times \omega_{dyn}$, which gives a value of n_d around 10 for decimation. Then, by using matlab function we implement the decimation on all the model. In Fig. (3.14) it shows the velocity and acceleration estimation of the real trajectory.

Identification results given in table (3.3) are quite similar for both methods.

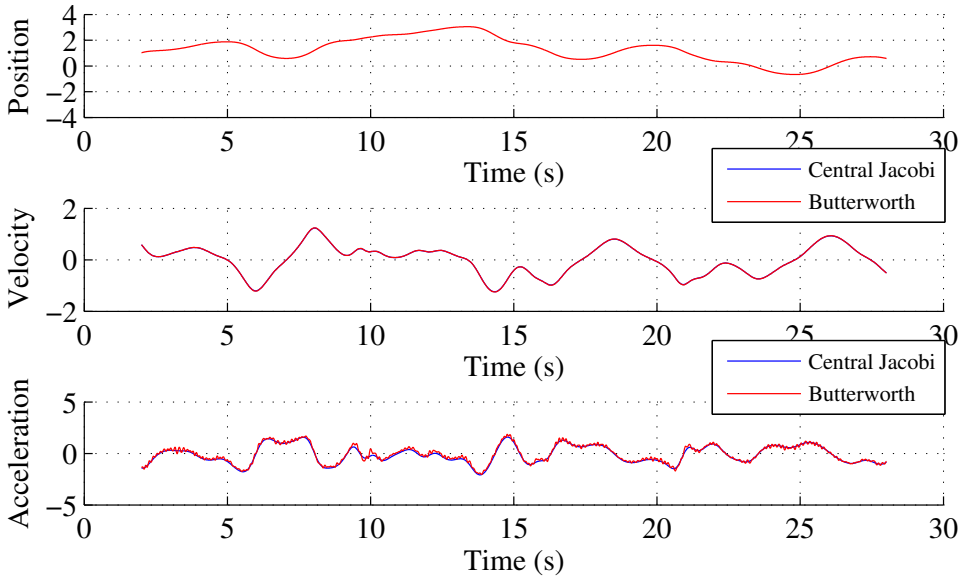


Figure 3.14 – 2R prototype robot trajectory and estimation of velocities, acceleration of q_1

Actually compared to central difference with Butterworth filter approach, the Jacobi differentiator method presents a better precision in identification results on error norm and relative error norm. When the trajectory is not of high frequency, the central Jacobi differentiator is a robust differentiator to get high order derivatives.

3.5 Dynamic parameters identification of EMPS

In the previous section, we discussed the application of causal and central Jacobi differentiators on the 2R scara planar robot. To extend, we look for their applications on an EMPS (which is a high-precision linear Electro-Mechanical Positioning System). In (Alexandre Janot et al., 2011), the authors presented several identification approaches and applied central difference associated with low-pass Butterworth filter technique to obtain the derivatives of EMPS, meanwhile a decimate procedure is implemented on all the models. This application deals with the estimation of high order derivatives, which can be 3^{th} or 4^{th} order, using measured position data. In this sense, it is a good test to examine the robustness of differentiators. Thus, we implement the central Jacobi differentiator during the identification procedure and compare it with the approaches of difference using forward-backward Butterworth filter. In this part, we first introduce the EMPS model and the identification methods, then carry out the

	Jacobi			Butterworth		
Parameters	$\hat{\mathbf{X}}$	$2\sigma_{\hat{\mathbf{X}}}$	$\sigma_{\hat{\mathbf{X}}r}$ %	$\hat{\mathbf{X}}$	$2\sigma_{\hat{\mathbf{X}}}$	$\sigma_{\hat{\mathbf{X}}r}$ %
ZZ_{1R}	3.4776	0.0332	0.4770	3.4836	0.0332	0.4765
F_{v1}	0.1955	0.0894	22.8746	0.2460	0.0895	18.1825
F_{s1}	0.4310	0.0491	5.6951	0.4330	0.0491	5.6677
ZZ_2	0.0593	0.0041	3.4640	0.0596	0.0041	3.4481
LMX_2	0.1253	0.0027	1.0848	0.1253	0.0027	1.0865
LMY_2	0.0007	0.0026	190.1596	0.0006	0.0026	218.2929
F_{v2}	0.0131	0.0153	58.6218	0.0139	0.0153	55.0943
F_{s2}	0.1269	0.0438	17.2724	0.1274	0.0438	17.2103
number of equations= 522						
$cond(W) = 38$ for both cases						

Table 3.3 – Comparison of experimental identification with 2R prototype robot, IDIM-OLS

experimental works to make a comparison.

3.5.1 Presentation of EMPS

An EMPS machine is presented in Fig. (3.15). It is a standard configuration of a drive system for prismatic joint of robots or machine tools. It is connected to a dSPACE digital control system for easy control and data acquisition using Matlab and Simulink software.

Figure 3.15 – EMPS prototype system

The system is composed of several main components:

- A Maxon DC motor equipped with an incremental encoder. This DC motor is position controlled with a PD controller.
- A Star high-precision low-friction ball screw drive positioning unit. An incremental encoder at its extremity supplies information about the angular position of the screw. - A load in translation.
- An accelerometer placed on the load supplies information about the load acceleration.

A brief configuration is presented in Fig. (3.16).

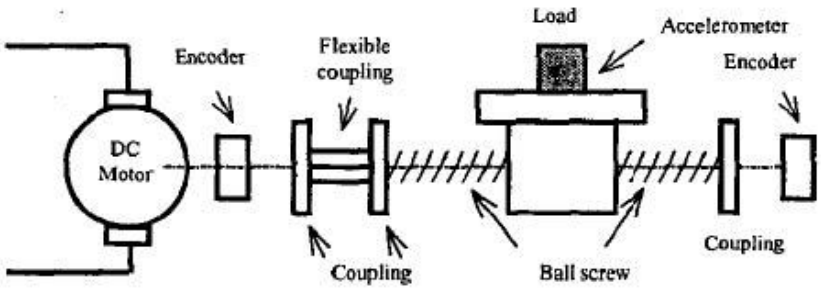


Figure 3.16 – EMPS components

All variables and parameters are given in ISO units on the load side.

3.5.2 Inverse dynamic model of EMPS

When the connection between the motor and the load is fixed, we can express the system dynamics in (3.23) with one inertia parameter and frictions. The inverse dynamic model describes the motor torque by function of the state and derivatives as

$$\tau_1 = ZZ_{1R}\ddot{q}_1 + F_{v1R}\dot{q}_1 + F_{s1R}sign(\dot{q}_1), \tag{3.23}$$

where q_1, $\dot{q}_1$, $\ddot{q}_1$ are respectively the motor position, velocity and acceleration; τ_1 is the motor torque; ZZ_{1R} is the total inertia, F_{v1R} and F_{s1R} are the total viscous and Coulomb friction parameters.

When the connection is flexible, the mechanical system can be modelled with two inertias, a spring and a structural damping, as shown in Fig. (3.17). The inverse dynamic model can be obtained from the Newton-Euler equations (Khalil and Dombre, 2004):

$$\begin{aligned} \tau_1 &= ZZ_1\ddot{q}_1 + F_{v1}\dot{q}_1 + F_{s1}sign(\dot{q}_1) - K_{12}q_2 + F_{off}, \\ 0 &= ZZ_2\ddot{q}_{12} + F_{v2}\dot{q}_{12} + F_{s2}sign(\dot{q}_{12}) + K_{12}q_2, \end{aligned} \tag{3.24}$$

where q_1, $\dot{q}_1$, $\ddot{q}_1$, τ_1 are the same as in the fixed case; q_{12}, $\dot{q}_{12}$, $\ddot{q}_{12}$ are respectively the load position, velocity and acceleration; q_2, $\ddot{q}_2$, $\ddot{q}_2$ are respectively the elastic relative position velocity and acceleration, with $q_{12} = q_1 + q_2$, $\dot{q}_{12} = \dot{q}_1 + \dot{q}_2$, $\dot{q}_{12} = \dot{q}_1 + \dot{q}_2$; ZZ_1 is the motor inertia, F_{v1} and F_{s1} are respectively the viscous and Coulomb motor friction parameters; ZZ_2 is the load inertia, F_{v2} and F_{s2} are respectively the viscous and Coulomb load friction parameters; K_{12} is the stiffness and F_{off} is offset coefficient. Moreover, the flexible inverse dynamic

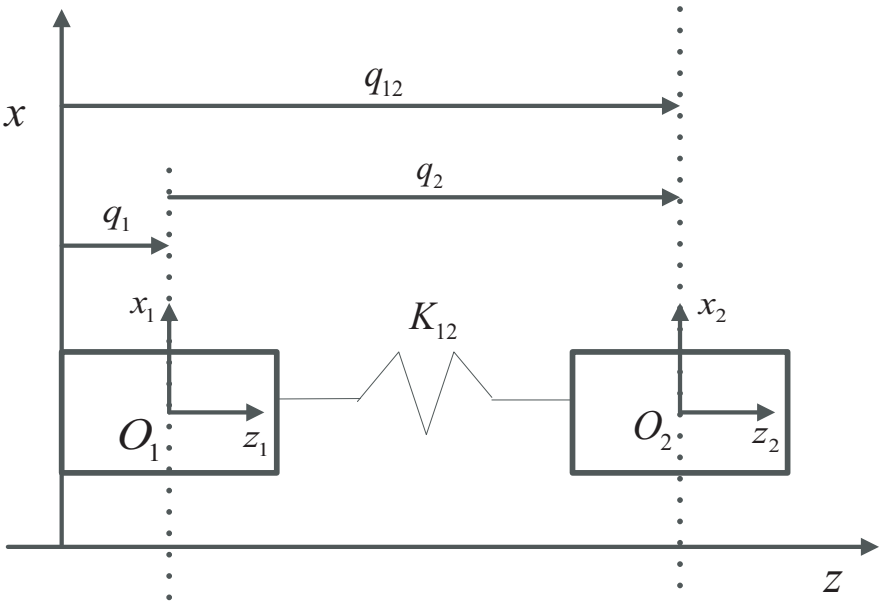


Figure 3.17 – EMPS modeling and DHM frames

model can be formulated as

$$\mathbf{\Gamma} = \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{K}\mathbf{q} + \mathbf{B}, \tag{3.25}$$

with

$$\mathbf{q} = \begin{pmatrix} q_1 \\ q_2 \end{pmatrix}, \dot{\mathbf{q}} = \begin{pmatrix} \dot{q}_1 \\ \dot{q}_2 \end{pmatrix}, \ddot{\mathbf{q}} = \begin{pmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{pmatrix}, \mathbf{\Gamma} = \begin{pmatrix} \tau_1 \\ 0 \end{pmatrix}, \mathbf{M}(\mathbf{q}) = \begin{pmatrix} ZZ_1 & 0 \\ 0 & ZZ_2 \end{pmatrix},$$
$$\mathbf{N}(\mathbf{q}, \dot{\mathbf{q}}) = \begin{pmatrix} F_{v1}\dot{q}_1 + F_{s1}sign(\dot{q}_1) \\ F_{v2}\dot{q}_{12} + F_{s2}sign(\dot{q}_{12}) \end{pmatrix}, \mathbf{K} = \begin{pmatrix} K_{12} & -K_{12} \\ -K_{12} & K_{12} \end{pmatrix}, \mathbf{B} = \begin{pmatrix} F_{off} \\ 0 \end{pmatrix}.$$

Then, the dynamic model can be rewritten as a linear relation with respect to the dynamic parameters as follows:

$$\mathbf{\Gamma} = \mathbf{D_{STD}}\mathbf{X_{STD}}, \tag{3.26}$$

with 8 parameters to be identified called standard parameters

$$\mathbf{X_{STD}} = \left[ZZ_1,\ F_{v1},\ F_{s1},\ K_{12},\ F_{off},\ ZZ_2,\ F_{v2},\ F_{s2}\right]^T$$

,

$$\mathbf{D_{STD}} = \begin{pmatrix} \ddot{q}_1 & \dot{q}_1 & sign(\dot{q}_1) & -q_2 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & q_2 & 0 & \ddot{q}_{12} & \dot{q}_{12} & sign(\dot{q}_{12}) \end{pmatrix}$$

.

Moreover, the direct dynamic model of EMPS is described by:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} = \mathbf{\Gamma} - \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} - \mathbf{K}\mathbf{q} - \mathbf{B}. \tag{3.27}$$

3.5.3 Identification model using motor and load positions

This is the idealistic case, where we have access to joint position measurement and motor torque. Thus, the minimal model corresponds to the standard model is given in (3.26) as:

$$\mathbf{D_1} = \mathbf{D_{STD}},\ \mathbf{X_1} = \mathbf{X_{STD}} \textbf{ and } \mathbf{\Gamma_1} = \mathbf{\Gamma}. \tag{3.28}$$

3.5.4 Identification model with only motor position and torque

This is the common case in the industrial application where only motor position and torque are known. The flexible load position q_2 need to be expressed in terms of q_1 and its derivatives. We remove F_{v2} and F_{s2} from the inverse dynamic model because F_{v2} is poorly identified and F_{s2} is not linear function. Then, the rewritten inverse dynamic model is given as:

$$\tau_1 = ZZ_1\ddot{q}_1 + F_{v1}\dot{q}_1 + F_{s1}sign(\dot{q}_1) - K_{12}q_2 + F_{off}, \tag{3.29}$$

$$0 = ZZ_2\ddot{q}_{12} + +K_{12}q_2. \tag{3.30}$$

We can resolve q_2, $\dot{q}_2$ and $\ddot{q}_2$ in function of motor position q_1 from equation (3.29):

$$\begin{aligned}
q_2 &= \frac{1}{ZZ_2}\left(-\tau_1 + ZZ_1\ddot{q}_1 + F_{v1}\dot{q}_1 + F_{s1}sign(\dot{q}_1) + F_{off}\right) \\
\dot{q}_2 &= \frac{1}{ZZ_2}\left(-\dot{\tau_1} + ZZ_1 q_1^{(3)} + F_{v1}\ddot{q}_1\right),\ sign(\dot{q}_1)\ is\ not\ differentiable \\
\ddot{q}_2 &= \frac{1}{ZZ_2}\left(-\ddot{\tau_1} + ZZ_1 q_1^{(4)} + F_{v1}q_1^{(3)}\right).
\end{aligned} \tag{3.31}$$

Rewrite equation (3.30) using the above expressions:

$$\begin{aligned}
0 = ZZ_1\ddot{q}_1 + ZZ_2\left(-\frac{\ddot{\tau_1}}{K_{12}} + \frac{ZZ_1}{K_{12}}q_1^{(4)} + \frac{F_{v1}}{K_{12}}q_1^{(3)}\right) \\
+ K_{12}\left(\frac{\tau_1}{K_{12}} + \frac{ZZ_1}{K_{12}}\ddot{q}_1 + \frac{F_{v1}}{K_{12}}\dot{q}_1 + \frac{F_{s1}}{K_{12}}sign(\dot{q}_1) + \frac{F_{off}}{K_{12}}\right).
\end{aligned} \tag{3.32}$$

Then we have:

$$\begin{aligned}
\Gamma_2 = \tau_1,\ \mathbf{D}_2 = \left(-\ddot{\tau}_1\ q_1^{(4)}\ q_1^{(3)}\ \ddot{q}_1\ \dot{q}_1\ sign(\dot{q}_1)\ 1\right), \\
X_2 = (o_1\ o_2\ o_3\ o_4\ o_5\ o_6\ o_7)^T.
\end{aligned} \tag{3.33}$$

with

$$o_1 = \frac{ZZ_2}{K_{12}},\ o_2\frac{ZZ_1 \cdot ZZ_2}{K_{12}},\ o_3 = \frac{ZZ_2 \cdot F_{v1}}{K_{12}},$$
$$o_4 = (ZZ_1 + ZZ_2),\ o_5 = F_{v1},\ o_6 = F_{s1},\ o_7 = F_{off}.$$

Finally, the dynamic parameters are calculated from the complete form:

$$ZZ_1 = o_2/o_1,\ ZZ_2 = o_4 - ZZ_1, \tag{3.34}$$
$$F_{v1} = o_5,\ F_{s1} = o_6, \tag{3.35}$$
$$K_{12} = ZZ_2/o_1, F_{off} = o_7. \tag{3.36}$$

We will apply the OLS technique on the two identification model, where derivatives are computed by central Jacobi differentiator, as well as central difference associated forward-backward Butterworth filter in order to make a comparison.

3.5.5 Data acquisition

The comparison is carried out in the differentiator part. On one hand, time derivatives are estimated using the central Jacobi differentiator; on the other hand, time derivatives are estimated by means of a band pass filtering of the position. This band-pass filtering is obtained with the product of a low-pass forward-backward Butterworth filter and from a derivative filter obtained by Euler central difference algorithm, which in all denotes no phase shift. The magnitude of the frequency response is given by:

$$|H(j\omega)| = \frac{1}{(1 + (\frac{j\omega}{\omega_{butter}})^{2n_{butter}})}, \tag{3.37}$$

where n_{butter} is the filter order and ω_{butter} is the cutoff frequency of the filter. n_{butter} is fixed according to the maximum derivatives order in the identification model. The cutoff frequency ω_{butter} of the low-pass filter must be chosen to avoid any magnitude distortion on the filtered signals in the range $[0, \omega_{dyn}]$ defined by the dynamics to be identified. More details about filtering can be found in (Pham et al., 2002, 2001).

To eliminate high frequency noises and torque ripples, a parallel decimation is performed on all identification model. This low-pass decimate filter resamples each signal at a lower rate. It keeps one sample over d_n because no information is contained in the range $[\omega_{dyn}, \omega_s/2]$, where ω_s is sampling frequency. Details about data decimation can be found in (Gautier, 1997).

Motor and load positions are measured by means of high precision encoders working in quadrature count mode and with an accuracy of 100000 counts per revolution. The sample acquisition frequency for joint position and current reference (drive force) is 1 kHz. We calculate the motor torque using the relation:

$$\tau_1 = G_\tau v_\tau, \tag{3.38}$$

where v_τ is the current reference of the amplifier current loop, and G_τ is the gain of the joint drive chain, which is taken as a constant in the frequency range of the robot because of the large bandwidth (700 Hz) of the current loop.

The first natural frequency, is of 30 Hz. This was verified with appropriate mechanical experiments such as blocked output test (see (Janot et al., 2007)). The cutoff frequency of the Butterworth filter is fixed at $60s$ Hz and the decimate ratio $n_d = 10$. It should be noticed in the region where the reference trajectory reaches the pulse, the data is not considered.

The system is position controlled with a PD controller, the bandwidth of the closed loop is tuned at 30 Hz to identify the dynamic parameters. Exciting trajectories consist of trapezoidal velocity with pulses, such that trapezoidal velocity excites very well inertia and friction parameters while pulses excite flexibility. We have condition number of observation matrix is around 30, implying that the dynamic parameters are well excited and can be identified (Gautier and Khalil, 1992; Presse and Gautier, 1993) with a good accuracy.

3.5.6 Experimental Validation

• identification model using motor and load positions: with the model described in (3.26), the maximum derivative order is 2. According to (Pham et al., 2001), the order of Butterworth filter $n_{butter} = 4$, which is a combination of an order 4 forward Butterworth filter and and order 4 backward one. For central Jacobi differentiator, we take $\mu = \kappa = 2$ and $q = 2$, for the estimation time window, we take $0.1s$, $0.2s$, $0.24s$ for derivatives of order 0, 1, 2 respectively. With the same reference trajectory and configuration as first identification method, we solve the LS problem $Y = WX + \rho$. Estimated value, standard deviation $\sigma_{\hat{\mathbf{X}}}$, relative standard deviation $\sigma_{\hat{\mathbf{X}}r}$, relative norm of the residue $\frac{\|(Y-WX_{est})\|}{\|Y\|}$ are shown in table (3.4). Cross tests validations have been performed. They consist in simulating the EMPS with the identified values and in integrating the direct dynamic model (3.27). In Fig. (3.18), we can see for both methods, the estimated torque follows closely the measured one.

• Identification model using motor position and torque: with the model described

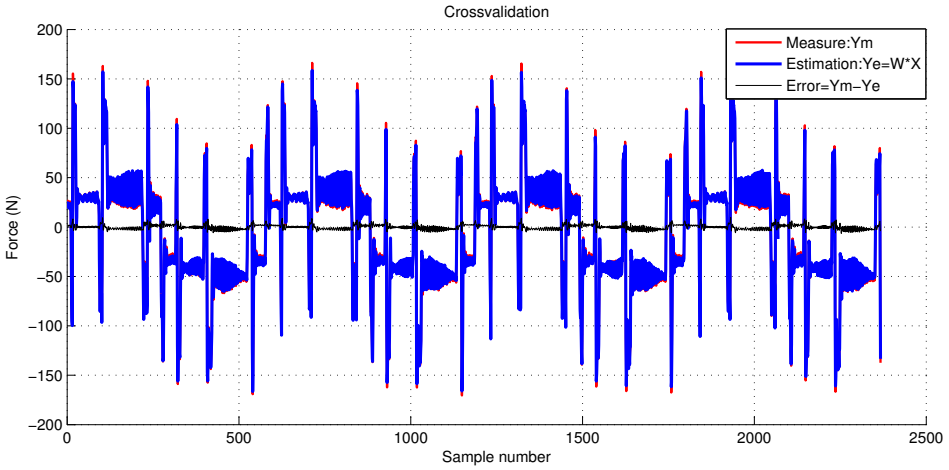
	Central Jacobi			Difference + Butterworth		
Parameters	$\hat{\mathbf{X}}$	$2\sigma_{\hat{\mathbf{X}}}$	$\sigma_{\hat{\mathbf{X}}r}$ %	$\hat{\mathbf{X}}$	$2\sigma_{\hat{\mathbf{X}}}$	$\sigma_{\hat{\mathbf{X}}r}$ %
ZZ_{1R}	71.4	0.46	0.32	72.0	0.46	0.32
F_{v1}	72.3	2.78	1.93	96.8	2.86	1.48
F_{s1}	8.70	0.27	1.56	8.83	0.28	1.56
ZZ_2	34.4	0.43	0.63	34.1	0.43	0.64
F_{v2}	89.7	0.26	1.26	100	2.32	1.16
F_{s2}	10.8	0.22	1.02	10.6	0.22	1.03
K_{12}	$7.52\ 10^5$	8750	0.58	$7.45\ 10^5$	8790	0.59
F_{off}	−7.09	0.01	0.70	0.125	0.01	0.70
$\frac{\|(Y-WX_{est})\|}{\|Y\|}$	0.0591			0.0597		
number of equations= 4734						

Table 3.4 – Results with identification model using motor and load positions

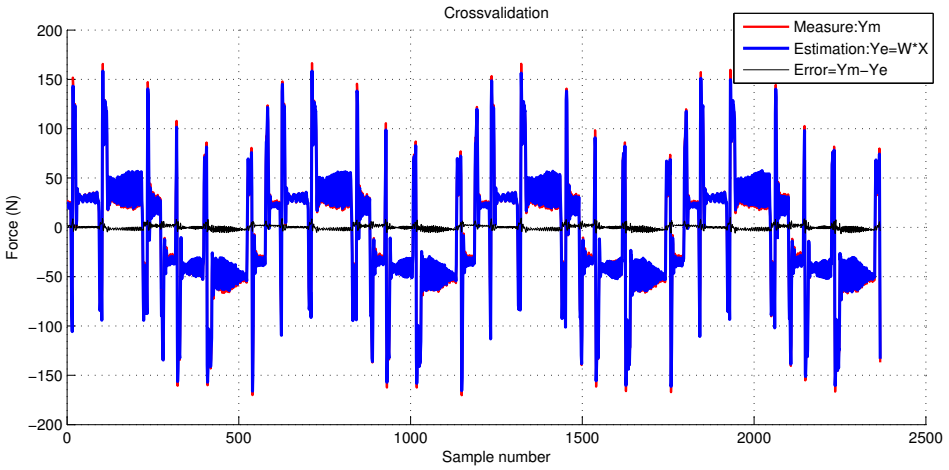
in (3.33), the maximum derivative order is 4. According to (Pham et al., 2001), the order of Butterworth filter $n_{butter} = 6$. For central Jacobi differentiator, we take $\mu = \kappa = 2$ and $q = 2$; for the joint position, $2, 3, 4$ order derivatives, we take sliding time window of $0.1s$, $0.2s$, $0.24s$, $0.4s$, $0.52s$ respectively; for the first, second order derivatives of torque, we take sliding time window of $0.16s$, $0.32s$ respectively. Estimated value, standard deviation $\sigma_{\hat{\mathbf{X}}}$, relative standard deviation $\sigma_{\hat{\mathbf{X}}r}$, relative norm of the residue $\frac{\|(Y-WX_{est})\|}{\|Y\|}$ are shown in table (3.5). After resolution, the identified model parameters are given in table (3.6). The cross validation is also given in Fig. (3.19). It can be found from the results, that the identification results using central Jacobi have smaller relative norm of the residue compared to that using difference with Butterworth filter. This means that the simulated force fits better to the real one.

	Central Jacobi			Difference + Butterworth		
Parameters	$\hat{\mathbf{X}}$	$2\sigma_{\hat{\mathbf{X}}}$	$\sigma_{\hat{\mathbf{X}}r}$ %	$\hat{\mathbf{X}}$	$2\sigma_{\hat{\mathbf{X}}}$	$\sigma_{\hat{\mathbf{X}}r}$ %
o_1	$4.59\ 10^{-5}$	$6.82\ 10^{-7}$	0.74	$4.53\ 10^{-5}$	$6.98\ 10^{-7}$	0.77
o_2	$3.12\ 10^{-3}$	$3.39\ 10^{-5}$	0.54	$3.17\ 10^{-3}$	$3.57\ 10^{-5}$	0.56
o_3	$-2.20\ 10^{-2}$	$3.74\ 10^{-3}$	8.50	$-2.60\ 10^{-2}$	$3.90\ 10^{-3}$	7.49
o_4	$1.06\ 10^2$	0.244	0.11	$1.06\ 10^2$	0.25	0.12
o_5	$1.91\ 10^2$	2.63	0.69	$1.97\ 10^2$	2.52	0.64
o_6	19.4	0.22	0.56	19.4	0.23	0.59
o_7	−5.84	0.09	0.77	−5.84	0.09	0.80
$\frac{\|(Y-WX_{est})\|}{\|Y\|}$	0.0386525			0.0401961		
number of equations= 2367						

Table 3.5 – Results with identification model using motor position and torque



(a) Cross validation with central Jacobi differentiator



(b) Cross validation with Butterworth approach

Figure 3.18 – Cross validation with identification model using motor and load positions

3.5.7 Comparison between two identification model with EMPS

The errors force with cross validation of these two identification methods are presented in Fig. (3.20). From the figure, we can see the identification models using motor and load positions have relatively more errors in cross validation than those using only motor position and torque. Comparing only the factor relative norm of the residue $\frac{||(Y-WX_{est})||}{||Y||}$, the identification model using only motor position and torque with central Jacobi differentiators presents the smallest values, which means the model is well built and the parameters are well identified to correspond the real measurement.

Meanwhile, we can see that with the same identification model, the one us-

Parameters	Central Jacobi	Difference + Butterworth
ZZ_1	68.0	69.9
F_{v1}	191.1	197.0
F_{s1}	19.4	19.4
ZZ_2	38.4	36.1
K_{12}	$8.37\ 10^5$	$7.96\ 10^5$
F_{off}	−5.84	−5.84

Table 3.6 – Identified parameters with identification model using motor position and torque

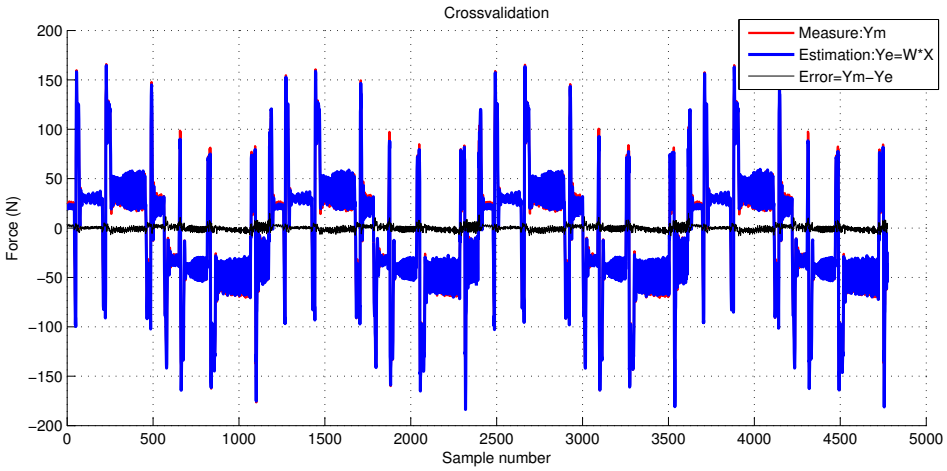
ing central Jacobi differentiators has small residue with cross validation. This proves the robustness of the central Jacobi differentiators in off-line applications.

In conclusion, the central Jacobi differentiator is an efficient off-line high order derivatives differentiator. Compared to central difference with Butterworth filter, it has the same or even better performance with proper parametrization. Tests on simulation and EMPS prove that it can be applied in real application.

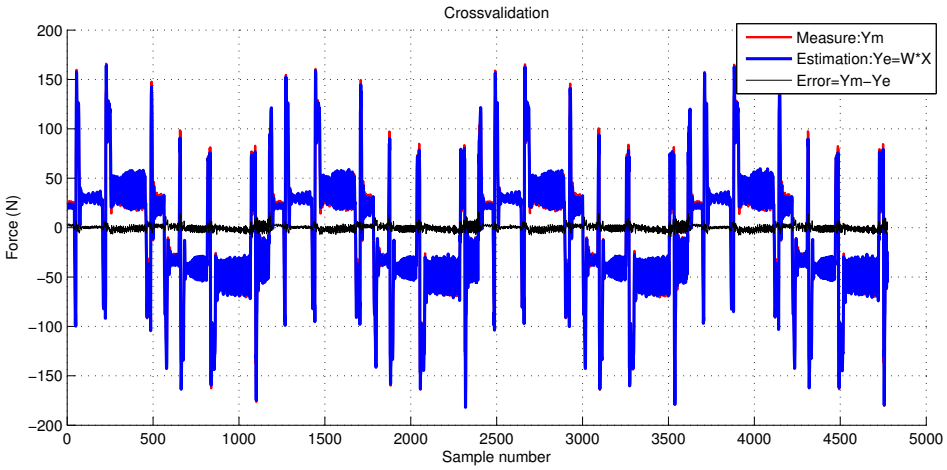
3.6 Conclusion

In this chapter, the recently developed Jacobi differentiators are analysed in both time domain and frequency domain. The frequency domain property corresponds well with time domain analysis. The important fact is that via the frequency analysis, we can evaluate the Jacobi differentiator performance from the bode plot, which for the first time brings to the user a criterion or measurement to design the differentiator. Furthermore, by the frequency analysis, Jacobi differentiator is able to make a comparison with the other differentiators, such as difference with filtered data approaches.

Jacobi differentiators are in nature a low-pass differentiators and comparisons are done to prove that Jacobi differentiators presents accuracy with good linear phase property. Meanwhile in discrete case, the computation of Jacobi differentiators always induce a numerical error when dealing with integration, which causes bias dealing with low frequency signal. Thus, the elimination of deviation is necessary. In the end, several robotic identification tests are done with Jacobi differentiators, which show their performances in applications.

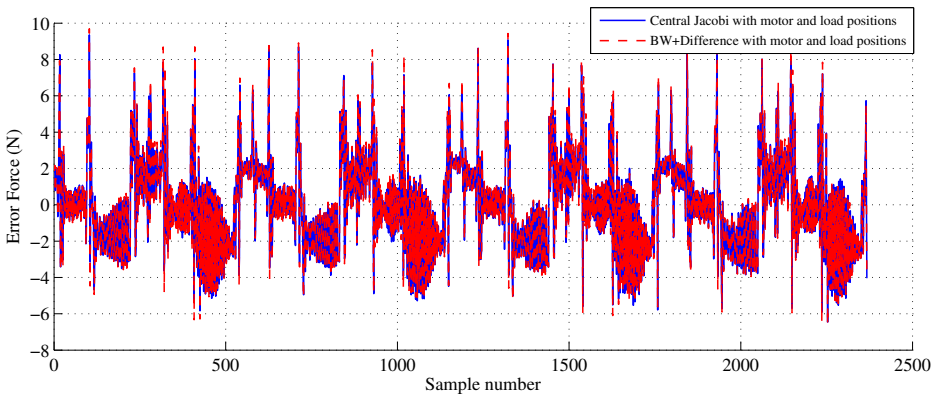


(a) Cross validation with central Jacobi differentiator

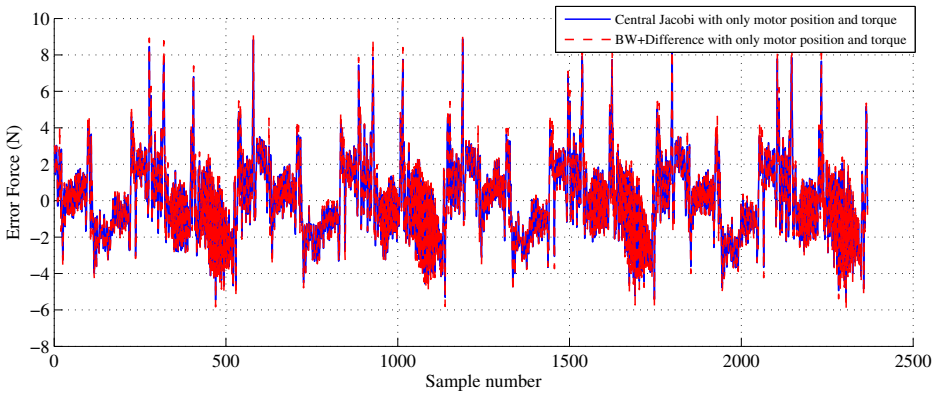


(b) Cross validation with Butterworth approach

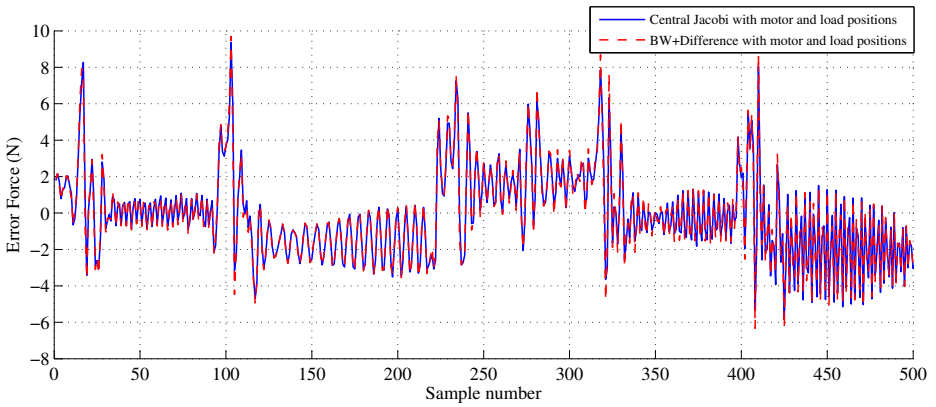
Figure 3.19 – Cross validation with identification model using only motor position and torque



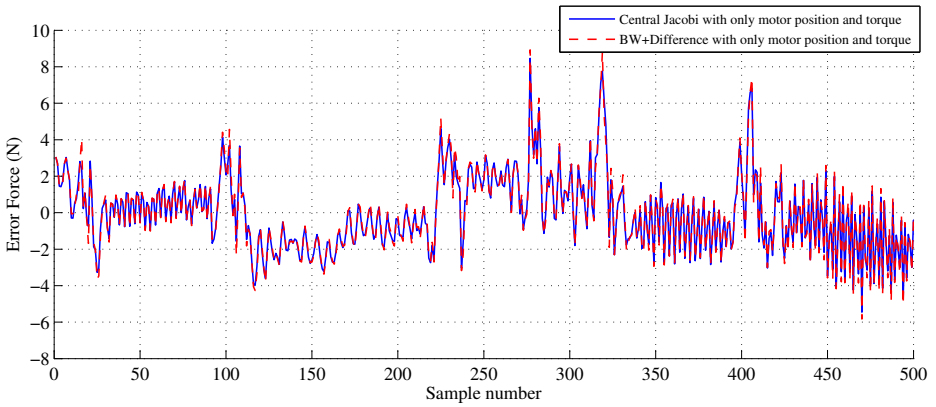
(a) Cross validation error with identification model using motor and load positions



(b) Cross validation error with identification model using only motor position and torque



(c) Zoomed cross validation error with identification model using motor and load positions



(d) Zoomed cross validation error with identification model using only motor position and torque

Figure 3.20 – Cross validation error with two identification model

Chapter 4

Comparison of different identification techniques

Recall in the previous chapters that for series robot identification issues, there exist different techniques mainly fall into three categories

- robot identification model,
- differentiation methods,
- least square techniques.

It should be noticed that different numerical decomposition tools, such as inverse, pseudo-inverse, QR factorization or SVD decomposition, can make difference when the observation matrix is of rank deficiency or near singularity. When the observation matrix has full rank and have good condition number, their performances are the same in precision of results. Thus, here we do not compare them. For the other techniques, we show in this chapter the comparison of the corresponding identification results. In the following table (4.1), we conclude the existing identification methods.

Identification models	Differentiators	LS techniques
dynamic model	difference & Butterworth	OLS
power model	causal Jacobi differentiator	WLS
energy model	central Jacobi differentiator	ILS
power model & modulating function		

Table 4.1 – Different identification approaches

Four robot identification model: in the following part, we will apply four identification model: dynamic identification model, energy identification model, power identification model, identification based on modulating functions with power model. The implementation procedures are discussed in chapter 1.

Three least square techniques: in each identification model, we will apply different LS techniques: OLS, WLS, ILS (see appendix).

Two differentiators: the central difference with forward-backward Butterworth filter is a zero phase off-line differentiator, which is widely used in various applications; the central Jacobi differentiator is a recently developed algebraic differentiator which is robust to noise and also applied off-line. Thus, the comparison between these two differentiators are interesting.

.

4.1 Tests on 2R scara planar robot

Simulation tests are carried out at sampling rate $\omega_c = 2\pi \times 100\ rad/s$, using the robot model described in chapter (1.1.1), without considering the friction offset parameter Γ_{off}. There are eight base dynamic parameters to be identified:

$$\mathbf{X} = [ZZ_{1R}\ ZZ_2\ LMX_2\ LMY_2\ F_{v1}\ F_{s1}\ F_{v2}\ F_{s2}],$$

using the measured joint position and torque. The simulation tests are running with value $\mathbf{X}$ all in SI Units: $\mathbf{X}$ = [3.5 0.06 0.15 0.1 0.3 0.4 0.2 0.12]. The reference trajectory is generated point to point using a classic 5^{th} order polynomial trajectory generator, where a successive 30 points are chosen within interval $[-\pi, \pi]$, which offers au total $n_e = 5000$ sampling points.

4.1.1 Simulation for 2R robot identification

The following tests are carried with the same condition. OLS, ILS are implemented in every cases. For ILS, the regularizing parameter $\lambda = 0.001$, the stopping error tolerance is 10^{-8} and maximum iteration is 10^8 times.

WLS is applied only with dynamic identification model, because other models have only one dimensional information that cannot be weighted. WLS technique is applied so that certain regions of the observation are valued more important or less important in the regression, according to their deviation. Here, weights are given to dynamic equations for each joint respectively, where their value is the inverse of the standard deviation σ_ρ^j. with j the joint number. We

can regroup the dynamic identification model (1.15) by putting the terms concerning the same torque together, and it is easily obtained in the form

$$\begin{bmatrix} \mathbf{\Gamma_{m1}} \\ \vdots \\ \mathbf{\Gamma_{mn}} \end{bmatrix} = \begin{bmatrix} \mathbf{D^{11}} & \dots & \mathbf{D^{1N_b}} \\ \vdots & \dots & \vdots \\ \mathbf{D^{n1}} & \dots & \mathbf{D^{nN_b}} \end{bmatrix} \mathbf{X}, \tag{4.1}$$

where $\mathbf{\Gamma_{mj}}$ is the $N_s \times 1$ torque observation for joint j, $\mathbf{D^{ji}}$ is the $N_s \times 1$ observation element of parameter X_i in the torque dynamic equation of joint j. Then, we can calculate the standard deviation

$$\sigma_\rho^j = \frac{\|\mathbf{Y}^j - \mathbf{\Phi X}^j\|}{\sqrt{r-c}}, \tag{4.2}$$

where

$$\mathbf{Y}^j = \begin{bmatrix} \mathbf{\Gamma_{mj}}(1) \\ \vdots \\ \mathbf{\Gamma_{mj}}(n_s) \end{bmatrix}, \ \mathbf{\Phi}^j = \begin{bmatrix} \mathbf{D^{j1}}(1) & \dots & \mathbf{D^{jN_b}}(1) \\ \vdots & \vdots & \vdots \\ \mathbf{D^{j1}}(n_s) & \dots & \mathbf{D^{jN_b}}(n_s) \end{bmatrix}$$

with r is the total number of equations, c is the number of unknown parameters, n_s is the number of sampling points.

Then, the weights $\frac{1}{\sigma_\rho^j}$ are adding to the corresponding part in (4.1), then the dynamic identification model becomes

$$\begin{bmatrix} \frac{1}{\sigma_\rho^1}\mathbf{\Gamma_{m1}} \\ \vdots \\ \frac{1}{\sigma_\rho^n}\mathbf{\Gamma_{mn}} \end{bmatrix} = \begin{bmatrix} \frac{1}{\sigma_\rho^1}\mathbf{D^{11}} & \frac{1}{\sigma_\rho^1}\dots & \frac{1}{\sigma_\rho^1}\mathbf{D^{1N_b}} \\ \vdots & \dots & \vdots \\ \frac{1}{\sigma_\rho^n}\mathbf{D^{n1}} & \dots & \frac{1}{\sigma_\rho^n}\mathbf{D^{nN_b}} \end{bmatrix} \mathbf{X}. \tag{4.3}$$

Considering the band-pass filtering, the Euler difference algorithm with forward-backward Butterworth filter is of order 12 with cutoff frequency $\omega_{fq} = 0.8\frac{\omega_c/2}{5} = 8$ Hz, where the forward or backward Butterworth filter is the same order 6 with opposite poles in the transfer function. The joint position and torque data are pre-filtered, then the joint velocity and acceleration are computed using Euler central difference algorithm.

For the selection of band-pass filtering using the central Jacobi differentiator, the cutoff frequency should be regulated at 8 Hz as the Butterworth filter. As mentioned in chapter 3, we can analyse the central Jacobi differentiator in frequency domain, and the cutoff property is easily shown. For tuning the parameters, we fix $\kappa = \mu = 2$, $q = 2$, then the only parameters tunable is the time window T. After several plots, we choose the best T as shown in Fig. (4.1) and

set $T = 0.16s$ for filtering joint position and torque, $T = 0.24s$ for joint velocity and $T = 0.28s$ for joint acceleration. From the analysis in chapter 3, if we increase the estimation time window, the cutoff frequency will move to low frequency. It should be noticed that there are bad estimation of the derivatives in the beginning and the end of the trajectory. In case the wrong estimation, we remove the estimation of the first and last $2s$.

Then, for IDIM-LS, power identification model, a low-pass filtering decimate procedure is implemented with decimate ratio $n_d = 10$.

Definite integrals $[t_a,\ t_b]$ are considered in the robot energy identification model and modulating function using power identification model. The rules for selection of integrals are presented in chapter (1.2), where the decimate procedure with the ratio n_d is equivalent to integration definite with $t_b(i) - t_a(i) = n_d * 2\pi/\omega_c$. Because the sampling frequency $\omega_c = 2\pi \times 100\ rad/s$ and n_d is set to 10, thus the definite integral $t_b(i) - t_a(i) = 0.01s$. While for modulating function using power identification model, the definite integral $[t_a,\ t_b]$ can be selected larger because modulating functions with integration change the filtering property. Thus, we test two type of integrals for modulating function using power identification model, the first with $t_b(i) - t_a(i) = 0.01s$, and the second with $t_b(i) - t_a(i) = 0.5$ plus $t_b(i) - t_a(i) = 1$.

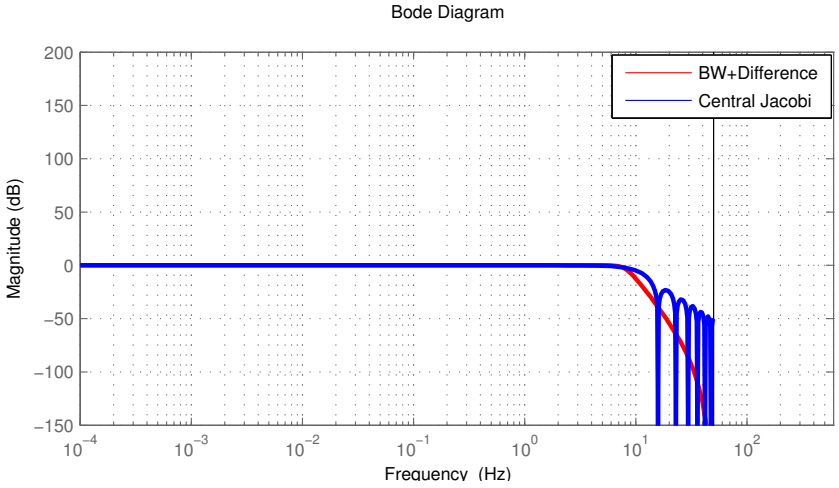
In the modulating function based approach, we implement the Jacobi modulating functions described in (2.9) with $\ell = \{\ell \in \mathbb{N} | \ell = 10, 11, \ldots, 20\}$ and sinusoid based modulating functions $g_\ell(t) = \sin^\ell\left(\frac{\pi}{T}t\right)$ with $\ell = \{\ell \in \mathbb{N} | \ell = 10, 11, \ldots, 20\}$. In the following part, we present the simulation results.

4.1.2 Filtering systematic error

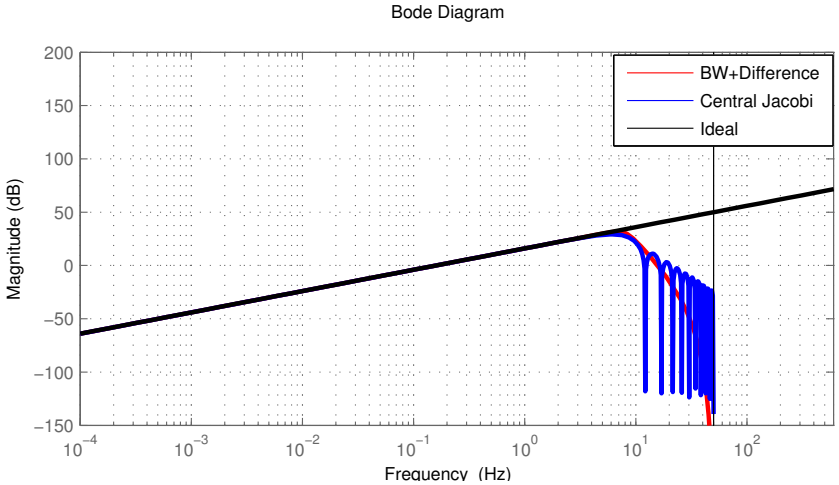
First we carry out the simulation test without considering the measurement noise, in order to investigate the systematic error due to the filtering implementation. The results are shown in table (4.2) with band-pass filter of central difference with Butterworth filter, and in table (4.3) with the central Jacobi differentiator. In order to see the systematic error for each parameter, we define $e_{\hat{X}ir}\ \% = 100 \times |\frac{Xi - \hat{X}i}{Xi}|$.

The conclusion can be drawn in two aspects:

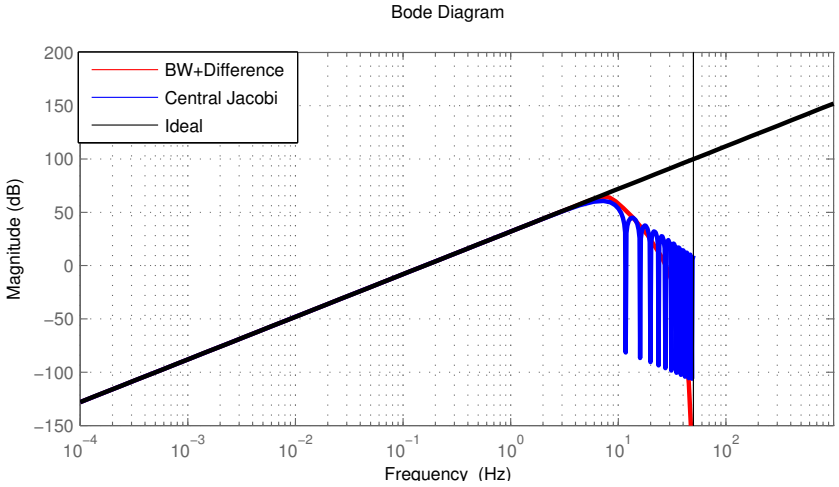
- comparing the band-pass filters, we find that the central Jacobi differentiator approach obtains smaller systematic error than those of the band-pass filter of central difference with Butterworth filter;
- comparing the identification model, form the relative error $|\frac{\mathbf{X} - \hat{\mathbf{X}}}{\mathbf{X}}|\%$, we see that energy identification model has the smallest systematic error, as



(a) Filtering for position



(b) First order differentiator



(c) Second order differentiator

Figure 4.1 – Bode plot cutoff frequency at 8 Hz of band-pass filtering for posiiton, velocity and acceleration

well the modulating function with power model has larger relative error because there are more variation caused by modulating functions, while for the other three methods, the errors are relatively larger.

Difference + Butterworth							
		IDIM-OLS		IDIM-WLS		Power	
Parameters	$\mathbf{X}$	$\hat{\mathbf{X}}$	$e_{\hat{X}ir}\}$ %	$\hat{\mathbf{X}}$	$e_{\hat{X}ir}$ %	$\hat{\mathbf{X}}$	$e_{\hat{X}ir}$ %
ZZ_{1R}	3.5000	3.5015	0.0431	3.5015	0.0425	3.5009	0.0271
F_{v1}	0.0500	0.0854	70.8051	0.0853	70.6945	0.0483	-3.4186
F_{s1}	0.5000	0.4278	-14.4479	0.4278	-14.4397	0.5036	0.7106
ZZ_2	0.0600	0.0602	0.4056	0.0600	0.0762	0.0600	-0.0560
LMX_2	0.1200	0.1201	0.0461	0.1203	0.2619	0.1201	0.0896
LMY_2	0.0050	0.0052	3.4312	0.0049	-1.5058	0.0052	4.1414
F_{v2}	0.0100	0.0143	43.2609	0.0144	44.1980	0.0104	4.1441
F_{s2}	0.1000	0.0896	-10.4330	0.0894	-10.6053	0.0982	-1.7738
$\lvert\frac{\mathbf{X}-\hat{\mathbf{X}}}{\mathbf{X}}\rvert$%		0.8494		0.8529		0.0705	
		Energy		Modulating&Power(1)[1]		Modulating&Power(2)	
Parameters	$\mathbf{X}$	$\hat{\mathbf{X}}$	$e_{\hat{X}ir}$ %	$\hat{\mathbf{X}}$	$e_{\hat{X}ir}$ %	$\hat{\mathbf{X}}$	$e_{\hat{X}ir}$ %
ZZ_{1R}	3.5000	3.5000	0.0013	3.5005	0.0134	3.5004	0.0119
F_{v1}	0.0500	0.0500	-0.0597	0.0499	-0.2168	0.0500	-0.0512
F_{s1}	0.5000	0.5001	0.0142	0.5001	0.0154	0.5001	0.0159
ZZ_2	0.0600	0.0600	0.0504	0.0600	0.0627	0.0600	0.0548
LMX_2	0.1200	0.1199	-0.0462	0.1200	-0.0088	0.1200	-0.0080
LMY_2	0.0050	0.0050	-0.2241	0.0050	-0.2173	0.0050	-0.2457
F_{v2}	0.0100	0.0100	0.0263	0.0100	0.1180	0.0100	0.3561
F_{s2}	0.1000	0.1000	-0.0082	0.0999	-0.0534	0.0999	-0.1183
$\lvert\frac{\mathbf{X}-\hat{\mathbf{X}}}{\mathbf{X}}\rvert$%		0.0024		0.0034		0.0046	

Table 4.2 – Filtering systematic error of central difference with Butterworth filter

4.1.3 Identification results on 2R prototype robot

In this part, the experimental data are acquired from a 2R scara planar prototype robot manufactured in the laboratory (IRCCyN) described in chapter (1.1.1). The joint position $\mathbf{q}$ and the current reference $\mathbf{V}_T$ (the control input) are collected at a 100 Hz sample rate while the robot is tracking a fifth order polynomial trajectory. This trajectory has been calculated in order to obtain a good condition number. This means that it is an exciting trajectory taking the whole trajectory all over at the time of the test. All methods are performed using the same joint position $\mathbf{q}$ and torque $\mathbf{\Gamma}_m$, where each torque Γ_m is calculated

[1] 1 with $t_b(i) - t_a(i) = 0.01s$, 2 with $t_b(i) - t_a(i) = 0.5$ plus $t_b(i) - t_a(i) = 1$.

Central Jacobi differentiator							
		IDIM-OLS		IDIM-WLS		Power	
Parameters	$\mathbf{X}$	$\hat{\mathbf{X}}$	$e_{\hat{X}ir}$ %	$\hat{\mathbf{X}}$	$e_{\hat{X}ir}$ %	$\hat{\mathbf{X}}$	$e_{\hat{X}ir}$ %
ZZ_{1R}	3.5000	3.4994	-0.0185	3.4994	-0.0169	3.4991	-0.0243
F_{v1}	0.0500	0.0293	-41.3506	0.0293	-41.4265	0.0483	-3.4650
F_{s1}	0.5000	0.4290	-14.1965	0.4290	-14.1910	0.5037	0.7360
ZZ_2	0.0600	0.0602	0.3534	0.0600	-0.0165	0.0599	-0.1137
LMX_2	0.1200	0.1199	-0.0920	0.1201	0.0811	0.1200	0.0207
LMY_2	0.0050	0.0052	3.1983	0.0050	-0.5698	0.0052	4.2329
F_{v2}	0.0100	0.0133	32.7090	0.0133	33.4821	0.0104	4.0420
F_{s2}	0.1000	0.0893	-10.7227	0.0891	-10.8555	0.0983	-1.7399
$\lvert\frac{\mathbf{X}-\hat{\mathbf{X}}}{\mathbf{X}}\rvert$%		0.5574		0.5619		0.0706	
		Energy		Modulating&Power(1)		Modulating&Power(2)	
Parameters	$\mathbf{X}$	$\hat{\mathbf{X}}$	$e_{\hat{X}ir}$ %	$\hat{\mathbf{X}}$	$e_{\hat{X}ir}$ %	$\hat{\mathbf{X}}$	$e_{\hat{X}ir}$ %
ZZ_{1R}	3.5000	3.4982	-0.0501	3.4987	-0.0382	3.4986	-0.0398
F_{v1}	0.0500	0.0499	-0.1140	0.0499	-0.2043	0.0500	-0.0363
F_{s1}	0.5000	0.5002	0.0410	0.5001	0.0237	0.5001	0.0149
ZZ_2	0.0600	0.0600	-0.0095	0.0600	-0.0024	0.0600	-0.0040
LMX_2	0.1200	0.1199	-0.1137	0.1199	-0.0749	0.1199	-0.0718
LMY_2	0.0050	0.0050	-0.1093	0.0050	-0.0786	0.0050	-0.1836
F_{v2}	0.0100	0.0100	-0.0599	0.0100	-0.0346	0.0100	-0.4543
F_{s2}	0.1000	0.1000	0.0192	0.1000	0.0087	0.1001	0.1272
$\lvert\frac{\mathbf{X}-\hat{\mathbf{X}}}{\mathbf{X}}\rvert$%		0.0021		0.0024		0.0051	

Table 4.3 – Filtering systematic error of central Jacobi differentiator

as

$$\Gamma_{mj} = G_{Tj} V_{Tj},$$

where G_{Tj} is the drive chain gain which is considered as a constant in the frequency range of the robot dynamics. The measurement joint position is shown in Fig. (2.4) and the motor torques are presented in Fig. (2.5). Note that all the identification methods are implemented with the same data.

Similarly as in the simulation, the joint position and torque data are prefiltered using a 12 order forward-backward Butterworth with a cutoff frequency $\omega_{fq} = 0.8\frac{\omega_c/2}{5} = 2\pi \times 8\ rad/s$ or 8 Hz. Then, the joint velocity and acceleration are computed using Euler central difference algorithm.

To compare, the central Jacobi differentiator is with configuration that $\kappa = \mu = 2$, $q = 2$ and $T = 0.16s$ for filtering joint position and torque, $T = 0.24s$ for joint velocity and $T = 0.28s$ for joint acceleration. Notice that in case the bad estimation in the beginning and the end due to the lack of information, we remove the related data for the first and last $2s$.

For dynamic and power identification model, the data are down-sampled through a decimate filter at 4 Hz, with decimate rate $n_d = 10$, in order to avoid aliasing (Gautier, 1997). For energy identification model and the modulating functions with power model approach, the procedure equivalent to decimate procedure is implemented, so that the integrals are selected with $t_b(i) - t_a(i) = n_d * 2\pi/\omega_c$. Because the sampling frequency $\omega_c = 2 * pi \times 100\ rad/s$ and n_d is set to 10, thus the definite integral $t_b(i) - t_a(i) = 0.01s$. While we try a second configuration $t_b(i) - t_a(i) = 0.5s$ plus $t_b(i) - t_a(i) = 1s$ for modulating functions with power model approach, because integrals associated with modulating functions can change the low-pass filtering property.

In the modulating function based approach, we implement again the Jacobi modulating functions described in (2.9) with $\ell = \{\ell \in \mathbb{N} | \ell = 10, 11, \ldots, 20\}$ and sinusoid based modulating functions $g_\ell(t) = \sin^\ell\left(\frac{\pi}{T}t\right)$ with $\ell = \{\ell \in \mathbb{N} | \ell = 10, 11, \ldots, 20\}$.

Finally, identification results are listed in table (4.4, 4.5, 4.6, 4.7, 4.8), 4.9.

Since the real values of dynamic parameters of the prototype robot are unknown, we list the results with residue norm $\|Y - W\hat{\mathbf{X}}\|$, relative residue norm $\frac{\|Y-W\hat{\mathbf{X}}\|}{\|Y\|}$, root mean square of residue $\frac{\|Y-W\hat{\mathbf{X}}\|}{\|\sqrt{r-c}\|}$, maximum relative residue norm $\max(\left|\frac{(Y-W\hat{\mathbf{X}})_j}{Y_j}\right|)$, and their condition numbers. Among them, the relative residue norm $\frac{\|Y-W\hat{\mathbf{X}}\|}{\|Y\|}$ represent more the model error.

It should be noticed that ILS technique has the same precision with OLS technique, thus here we do not list the results. From the results, the IDIM-OLS and power identification model have the better condition number for the observation matrix. Compared to inertia parameters, the friction parameters are less identified, since their values are small and their relative standard deviation $\sigma_{\hat{\mathbf{X}}r}$ % are relatively large. This come from the fact that viscous friction parameters are small and have little influence on the dynamic model; and Coulomb friction parameters need to be excited with trapezoid velocity in the trajectory.

Conclusion can be drawn as :

Firstly, modulating functions using power model method with definite integral $t_b(i) - t_a(i) = 0.5s, 1s$ has very small relative standard deviation $\sigma_{\hat{\mathbf{X}}r}$ % for each dynamic parameters, and small relative residue norm $\frac{\|Y-W\hat{\mathbf{X}}\|}{\|Y\|}$. Although its condition number is relative lager, it can be regarded as the best identification method. When with definite integral $t_b(i) - t_a(i) = 0.01s$, the relative residue norm $\frac{\|Y-W\hat{\mathbf{X}}\|}{\|Y\|}$ becomes large. In general, for modulating functions using power model method, the relative standard deviation is much improved than the other methods, especially in the inertia dynamic parameters. In this

sense, it is the best identification methods among all the techniques.

Secondly, although not as good as the modulating functions using power model method, the IDIM-WLS obtains also small relative residue norm $\frac{\|Y-W\hat{\mathbf{X}}\|}{\|Y\|}$, but the relative standard deviation $\sigma_{\hat{\mathbf{X}}r}$ % of each parameters are higher. Considering these two elements, IDIM-WLS is the good solution.

While the energy and power identification model also presents small relative residue norm, its relative standard deviations $\frac{\|Y-W\hat{\mathbf{X}}\|}{\|Y\|}$ are high than IDIM-WLS. And IDIM-OLS has the worst performance except that IDIM-OLS has the best condition number of the observation matrix.

For differentiators, the results using central Jacobi differentiator are relatively better than using central difference with Butterworth filter, or at least they are similar. The results show central Jacobi differentiator is a robust off-line differentiator.

So, with the prototype 2R scara planar robot, the experimental works prove that:
modulating function with power model ≥ IDIM-WLS ≥ power identification model ≥ energy identification model ≥ IDIM-OLS .

4.2 Conclusion

From the comparison of different identification techniques, we have the following conclusion:
• comparing the identification models, the modulating function with power model method and IDIM-WLS are the best solution; secondly the power model and energy identification model; while IDIM-OLS has largest error in the relative standard deviation $\frac{\|Y-W\hat{\mathbf{X}}\|}{\|Y\|}$;
• comparing the differentiators, the central Jacobi differentiator is a robust off-line differentiator, which has the better or same performance than band-pass filtering of central difference with Butterworth filter approach;
• comparing the LS techniques, WLS offers better solution than OLS and ILS, but WLS only can be applied in IDIM and power identification model, besides, OLS and ILS have the same solution when the observation matrix has full rank.

	Central Jacobi			Difference + Butterworth		
Parameters	$\hat{\mathbf{X}}$	$2\sigma_{\hat{\mathbf{X}}}$	$\sigma_{\hat{\mathbf{X}}r}$ %	$\hat{\mathbf{X}}$	$2\sigma_{\hat{\mathbf{X}}}$	$\sigma_{\hat{\mathbf{X}}r}$ %
ZZ_{1R}	3.4776	0.0332	0.4770	3.4836	0.0332	0.4765
F_{v1}	0.1955	0.0894	22.8746	0.2460	0.0895	18.1825
F_{s1}	0.4310	0.0491	5.6951	0.4330	0.0491	5.6677
ZZ_2	0.0593	0.0041	3.4640	0.0596	0.0041	3.4481
LMX_2	0.1253	0.0027	1.0848	0.1253	0.0027	1.0865
LMY_2	0.0007	0.0026	190.1596	0.0006	0.0026	218.2929
F_{v2}	0.0131	0.0153	58.6218	0.0139	0.0153	55.0943
F_{s2}	0.1269	0.0438	17.2724	0.1274	0.0438	17.2103
$\|Y - W\hat{\mathbf{X}}\|$	4.9152			4.9148		
$\frac{\|Y-W\hat{\mathbf{X}}\|}{\|Y\|}$	0.0905			0.0905		
$\frac{\|Y-W\hat{\mathbf{X}}\|}{\sqrt{r-c}}$	0.2168			0.2168		
$\max(\left\|\frac{(Y-W\hat{\mathbf{X}})_j}{Y_j}\right\|)$	127.5344			62.1104		
Nb equation	522			522		
Cond(W)	38.5679			38.5368		

Table 4.4 – Results with IDIM-OLS, 2R prototype robot

	Central Jacobi			Difference + Butterworth		
Parameters	$\hat{\mathbf{X}}$	$2\sigma_{\hat{\mathbf{X}}}$	$\sigma_{\hat{\mathbf{X}}r}$ %	$\hat{\mathbf{X}}$	$2\sigma_{\hat{\mathbf{X}}}$	$\sigma_{\hat{\mathbf{X}}r}$ %
ZZ_{1R}	3.4702	0.0459	0.6618	3.4766	0.0460	0.6609
F_{v1}	0.1969	0.1240	31.4964	0.2503	0.1240	24.7741
F_{s1}	0.4320	0.0684	7.9145	0.4325	0.0683	7.9006
ZZ_2	0.0628	0.0005	0.4093	0.0628	0.0005	0.4063
LMX_2	0.1250	0.0026	1.0248	0.1253	0.0026	1.0193
LMY_2	0.0030	0.0028	47.3083	0.0029	0.0028	48.5163
F_{v2}	0.0131	0.0014	5.4010	0.0140	0.0014	5.0296
F_{s2}	0.1261	0.0040	1.5983	0.1266	0.0040	1.5812
$\|Y - W\hat{\mathbf{X}}\|$	22.6716			22.6604		
$\frac{\|Y-W\hat{\mathbf{X}}\|}{\|Y\|}$	0.0655			0.0651		
$\frac{\|Y-W\hat{\mathbf{X}}\|}{\sqrt{r-c}}$	1.0000			0.9995		
$\max(\left\|\frac{(Y-W\hat{\mathbf{X}})_j}{Y_j}\right\|)$	114.4855			27.2200		
Nb equation	522			522		
Cond(W)	268.2370			269.9631		

Table 4.5 – Results with IDIM-WLS, 2R prototype robot

	Central Jacobi			Difference + Butterworth		
Parameters	$\hat{\mathbf{X}}$	$2\sigma_{\hat{\mathbf{X}}}$	$\sigma_{\hat{\mathbf{X}}r}$ %	$\hat{\mathbf{X}}$	$2\sigma_{\hat{\mathbf{X}}}$	$\sigma_{\hat{\mathbf{X}}r}$ %
ZZ_{1R}	3.4459	0.0517	0.7498	3.4482	0.0523	0.7581
F_{v1}	0.2273	0.1815	39.9232	0.2309	0.1836	39.7542
F_{s1}	0.4651	0.2014	21.6476	0.4607	0.2035	22.0882
ZZ_2	0.0642	0.0018	1.4373	0.0642	0.0019	1.4525
LMX_2	0.1261	0.0032	1.2691	0.1260	0.0032	1.2846
LMY_2	0.0016	0.0029	88.8326	0.0014	0.0029	102.2334
F_{v2}	0.0170	0.0068	19.9722	0.0170	0.0069	20.1323
F_{s2}	0.1143	0.0404	17.6803	0.1141	0.0408	17.8885
$\|Y - W\hat{\mathbf{X}}\|$	2.7419			2.7695		
$\frac{\|Y-W\hat{\mathbf{X}}\|}{\|Y\|}$	0.0613			0.0620		
$\frac{\|Y-W\hat{\mathbf{X}}\|}{\sqrt{r-c}}$	0.1724			0.1741		
$\max(\left\|\frac{(Y-W\hat{\mathbf{X}})_j}{Y_j}\right\|)$	131.2531			6.6362		
Nb equation	261			261		
Cond(W)	181.4647			181.3684		

Table 4.6 – Results with power identification model, OLS, 2R prototype robot

	Central Jacobi			Difference + Butterworth		
Parameters	$\hat{\mathbf{X}}$	$2\sigma_{\hat{\mathbf{X}}}$	$\sigma_{\hat{\mathbf{X}}r}$ %	$\hat{\mathbf{X}}$	$2\sigma_{\hat{\mathbf{X}}}$	$\sigma_{\hat{\mathbf{X}}r}$ %
ZZ_{1R}	3.4346	0.0554	0.8070	3.4333	0.0570	0.8301
F_{v1}	0.2413	0.1947	40.3448	0.2369	0.2004	42.2891
F_{s1}	0.4497	0.2163	24.0532	0.4529	0.2225	24.5573
ZZ_2	0.0643	0.0020	1.5335	0.0644	0.0020	1.5761
LMX_2	0.1257	0.0035	1.3789	0.1256	0.0036	1.4211
LMY_2	0.0015	0.0032	107.6274	0.0014	0.0032	117.3415
F_{v2}	0.0167	0.0073	21.7823	0.0169	0.0075	22.0881
F_{s2}	0.1167	0.0434	18.5825	0.1156	0.0446	19.2806
$\|Y - W\hat{\mathbf{X}}\|$	0.2919			0.2999		
$\frac{\|Y-W\hat{\mathbf{X}}\|}{\|Y\|}$	0.0656			0.0675		
$\frac{\|Y-W\hat{\mathbf{X}}\|}{\sqrt{r-c}}$	0.0184			0.0189		
$\max(\left\|\frac{(Y-W\hat{\mathbf{X}})_j}{Y_j}\right\|)$	40.6304			21.3571		
Nb equation	260			260		
Cond(W)	182.3196			182.2672		

Table 4.7 – Results with energy identification model method, OLS, 2R prototype robot

	Central Jacobi			Difference + Butterworth		
Parameters	$\hat{\mathbf{X}}$	$2\sigma_{\hat{\mathbf{X}}}$	$\sigma_{\hat{\mathbf{X}}r}$ %	$\hat{\mathbf{X}}$	$2\sigma_{\hat{\mathbf{X}}}$	$\sigma_{\hat{\mathbf{X}}r}$ %
ZZ_{1R}	3.3767	0.0100	0.1478	3.3897	0.0105	0.1542
F_{v1}	0.3854	0.0354	4.5960	0.3040	0.0370	6.0833
F_{s1}	0.3436	0.0394	5.7277	0.3860	0.0410	5.3142
ZZ_2	0.0655	0.0004	0.2740	0.0651	0.0004	0.2878
LMX_2	0.1251	0.0006	0.2511	0.1250	0.0007	0.2616
LMY_2	0.0014	0.0006	19.8856	0.0009	0.0006	34.1670
F_{v2}	0.0166	0.0013	3.9917	0.0161	0.0014	4.2923
F_{s2}	0.1244	0.0079	3.1757	0.1253	0.0082	3.2828
$\|Y - W\hat{\mathbf{X}}\|$	24.0450			25.0002		
$\frac{\|Y-W\hat{\mathbf{X}}\|}{\|Y\|}$	0.0809			0.0843		
$\frac{\|Y-W\hat{\mathbf{X}}\|}{\sqrt{r-c}}$	0.2249			0.2338		
$\max(\left\|\frac{(Y-W\hat{\mathbf{X}})_j}{Y_j}\right\|)$	94.9759			112.0807		
Nb equation	11440			11440		
Cond(W)	181.0155			181.2747		

Table 4.8 – Results with modulating function using power model method, OLS, 2R prototype robot

	Central Jacobi			Difference + Butterworth		
Parameters	$\hat{\mathbf{X}}$	$2\sigma_{\hat{\mathbf{X}}}$	$\sigma_{\hat{\mathbf{X}}r}$ %	$\hat{\mathbf{X}}$	$2\sigma_{\hat{\mathbf{X}}}$	$\sigma_{\hat{\mathbf{X}}r}$ %
ZZ_{1R}	3.4517	0.0144	0.2091	3.4513	0.0149	0.2152
F_{v1}	0.2362	0.0527	11.1535	0.2334	0.0542	11.6167
F_{s1}	0.4699	0.0598	6.3585	0.4715	0.0614	6.5160
ZZ_2	0.0643	0.0005	0.4096	0.0644	0.0005	0.4208
LMX_2	0.1210	0.0011	0.4666	0.1209	0.0012	0.4806
LMY_2	0.0040	0.0008	10.6176	0.0035	0.0009	12.2959
F_{v2}	0.0155	0.0020	6.4041	0.0154	0.0020	6.6560
F_{s2}	0.1173	0.0121	5.1544	0.1177	0.0124	5.2783
$\|Y - W\hat{\mathbf{X}}\|$	10.0353			10.3118		
$\frac{\|Y-W\hat{\mathbf{X}}\|}{\|Y\|}$	0.0613			0.0630		
$\frac{\|Y-W\hat{\mathbf{X}}\|}{\sqrt{r-c}}$	0.1715			0.1762		
$\max(\left\|\frac{(Y-W\hat{\mathbf{X}})_j}{Y_j}\right\|)$	15.3945			11.5327		
Nb equation	3432			3432		
Cond(W)	205.2649			205.2168		

Table 4.9 – Results with modulating function using power model method, OLS, 2R prototype robot with $t_b(i) - t_a(i) = 0.5s, 1s$

Chapter 5

Simplified model with real time estimation

In all the questions discussed above, the accurate dynamic models are needed for control issue. While some inconveniences come with the accurate identification:

- it requires modelling the robot dynamics and identifying the unknown parameters in the model, which is an extra work;
- the building of dynamic models are theoretical and there always exist some bias compared to the ideal case, resulting in inaccuracy;
- in order to accurately identify of the parameters, identification procedures are usually carried out over a long time window, and need a sufficient information from the sampling data, in this sense, the traditional identification approaches respond not fast enough to corrupt change of the system dynamics.

From control point of view, we can compromise between the estimation accuracy for the system dynamics and the estimation time. In order to get fast response with respect to corrupt dynamic changes, here we proposes a real time estimation iterative learning structure for robot manipulators without knowing the dynamic model of the system, as well it is robust to corrupt payload change and initial conditions. The interesting point of this method is that it estimate the parameters in short time window. It considers a simplified model to describe the robot dynamics, instead of the explicit dynamic model, commonly used for the simplicity of identification model. The simplification allows to reduce the number of parameters to be updated. Moreover the simplified model

should represent the current system dynamics, which can be ensured by a real time estimation of the model states. In this case, the corrupt change of payload will be detected within short time window such as 0.1 second, and the system dynamics will be adjusted quickly to real values. Modulating functions techniques are also applied in the real time estimation process, to decrease the order of input via integration by part method, which avoids using joint velocities and accelerations. Based on filtering property of modulating functions, groups of modulating functions are selected in order to eliminate the high frequency noise influence. In the end, simulation results on a two degrees of freedom planar robot prove the efficiency of the control structure.

5.1 Introduction

Iterative learning control is an efficient method for on-line robot application. Lots of researches are dedicated to this iterative learning subject, such as in (Bao et al., 1996; Bristow et al., 2006; Bukkems et al., 2005; Wang, Gao, et al., 2009). In recent literature (Gautier, Jubien, et al., 2013) presented the structure of iterative learning identification and computed torque control (IDIM-ILIC) in robot issues. It estimates or adjusts on-line the dynamic parameter values and constructs the computed torque with the updated parameters.

However these methods are somehow based on the awareness of robot model and the identification of the model parameters, which is complex and hard to implement in real time. Starting from an algebraic point of view, we proposed an extremely simplified model to represent the manipulator's dynamics, where the model parameters are time-varying. The number of parameters to be updated in this model is small. This gives simplicity to robot model and advantages in parameters updating process, because with simpler structure, it requires less time consumption to get robust estimation. The validity of the simplified model is ensured by real time parameters estimation, where the time-varying parameters are approximated as constant or linear varying component in short time interval, according to their dynamics. The reconstruction of system dynamics is ensured once the estimation time is reduced to $0.1s$ as it is tested in simulation of sampling frequency 100 Hz, so that it allows the estimation responds quickly to the dynamics variation and makes the iterative learning control robust to corrupt change and initial conditions.

In the estimation process we utilize modulating function approach to avoid using joint velocities and accelerations. Commonly these two derivatives are computed from joint position, which causes problem in robot identification

process because small error in measurement can induce large error in the computed derivatives, specially for high order derivatives. Thus it is better to use only joint position. The modulating function property plays an important role in reducing the order of inputs in the estimation model as in (Liu, Laleg-Kirati, et al., 2013). The modulating function identification theories are proposed in several literatures, while in robotics identification field, the modulating function identification application is new and can be found in (Guo et al., 2014). There exist all kinds of modulating functions, we will study their filtering property and select certain groups of modulating functions which have low-pass filtering property.

This chapter is organized as follows: section 2 deduces the simplified robot model from the robot explicit dynamic model and presents the design of the adaptive controller; section 3 gives precise description on real time estimation of model parameters using modulating functions; in section 4 simulation is carried out with a two degrees of freedom planar robot model, the simulation result shows that the adaptive control structure has good tracking precision and is robust to high frequency noise, corrupt change of system dynamics and initial conditions; and in last section it comes to a conclusion.

5.2 Simplified model and iterative learning control

In this section, we first provide the rigid-body dynamic model of manipulator and change it to a simplified model with time-varying parameters. Then an iterative learning controller is designed for this model.

Recall thee general form of the inverse dynamic model (1.2):

$$\tau = \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{Q}(\mathbf{q}) + \tau_{\mathbf{f}},$$

The analytical expression of the inverse dynamic model is complex and the unknown dynamic parameters are numerous. This brings difficulty to estimation because it need rich measurements to well identify each value of the parameters, where some of them usually are coupled and need long time to identify. Here, we propose a simplified model with fewer parameters to reconstruct system dynamics:

$$\tau = \mathbf{M}(\mathbf{t})\ddot{\mathbf{q}} + \mathbf{N}(\mathbf{t}), \tag{5.1}$$

where $\mathbf{M}(\mathbf{t}) = \mathbf{M}(\mathbf{q}(\mathbf{t}))$ is a $n \times n$ symmetric and positive definite inertia matrix, and $\mathbf{N}(\mathbf{t})$ is a $n \times 1$ vector contains other components of the manipulator dynam-

ics. The simplicity holds under condition that model parameters $\mathbf{M}(\mathbf{t})$ and $\mathbf{N}(\mathbf{t})$ are considered time-varying.

5.2.1 Controller design

Given the reference trajectories $\mathbf{q_{ref}}(t)$, $\dot{\mathbf{q}}_{\mathbf{ref}}(t)$ and $\ddot{\mathbf{q}}_{\mathbf{ref}}(t)$ of position, velocity and acceleration respectively, without knowing the robot model, derive an adaptive control law for the actuator torques, and a real time estimation scheme for the adaptive components, such that the manipulator joint position $\mathbf{q}(t)$ precisely tracks $\mathbf{q_{ref}}(t)$ after an initial adaptation process. In order to design such a controller, we use the following controller:

$$\tau = \mathbf{M}(\mathbf{t})\left(\ddot{\mathbf{q}}_{ref} - \lambda_d\dot{\mathbf{e}} - \lambda_p\mathbf{e} - \lambda_i\int_0^T \mathbf{e}\right) + \mathbf{N}(\mathbf{t}), \tag{5.2}$$

with proper gains λ_p, λ_i and λ_d (diagonal matrix, see chapter (A.4)), it is sufficient to asymptotically stabilize the tracking error $\mathbf{e} = \mathbf{q} - \mathbf{q_{ref}}$ for the system modelled by (5.1).

At instant t, we can replace the computed motor torque τ in equation (5.2) by robot model (5.1), and it deduces a state equation for tracking error $\mathbf{e}$

$$\mathbf{M}(\mathbf{t})\left(\ddot{\mathbf{e}} + \lambda_d\dot{\mathbf{e}} + \lambda_p\mathbf{e} + \lambda_i\int_0^T \mathbf{e}\right) = 0. \tag{5.3}$$

Since $\mathbf{M}(\mathbf{t})$ is always invertible (positive definite), the state equation becomes

$$\ddot{\mathbf{e}} + \lambda_d\dot{\mathbf{e}} + \lambda_p\mathbf{e} + \lambda_i\int_0^T \mathbf{e} = 0. \tag{5.4}$$

Derive equation (5.4) with respect to time and we get a third order differential equation

$$\mathbf{e}^{(3)} + \lambda_d\ddot{\mathbf{e}} + \lambda_p\dot{\mathbf{e}} + \lambda_i\mathbf{e} = 0. \tag{5.5}$$

The asymptotically stability and convergence rate of tracking error $\mathbf{e}$ can be ensured and are tunable by selecting the gains (see appendix A.4) λ_p, λ_i and λ_d .

$\mathbf{M}(\mathbf{t})$ and $\mathbf{N}(\mathbf{t})$ should be estimated as $\overline{\mathbf{M}}$ and $\overline{\mathbf{N}}$ within small time interval, which will be discussed in the next section. This control scheme is easy to implement for robot manipulators without knowing the dynamic model of the robot, and simplicity in the model contributes to realize real time estimation. In return real time estimation offers quick response to variation of system dynamics as well as the initial conditions. Fig. (5.1) shows the structure of the

modulating functions based iterative learning controller.

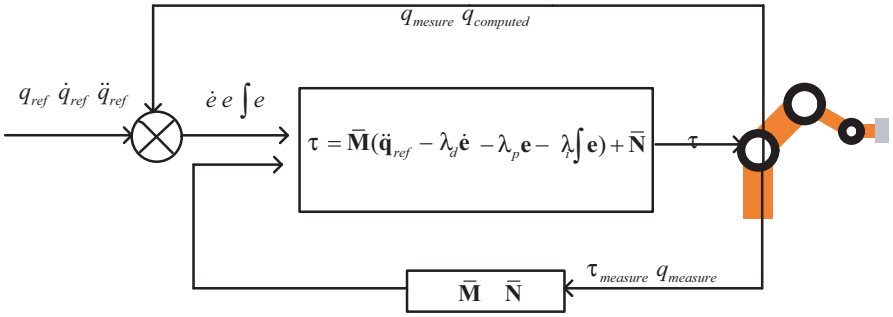


Figure 5.1 – Structure of iterative learning controller with real time estimation

5.3 Real time estimation

According to the model description (5.1), the conventional estimation approach needs the joint torques τ and accelerations $\ddot{\mathbf{q}}$. Usually τ are calculated from the current reference of the amplifier current loop and the gain of each joint drive chain. And $\ddot{\mathbf{q}}$ are computed from discrete joint position measurement via robot sensor, whose sampling rate must be large enough to avoid high frequency noise aliasing the bandwidth of the joint position closed loop, see (Gautier, 1996). But reconstruction of high order derivatives from noisy data is long standing problem because noise component will be enlarged exponentially with increasing order during the numerical computation.

Above all, we propose a modulating functions based structure regarding to the simplified model, where only the joint position data is needed in the estimation process. Meanwhile, the integration with definite integrals is actual a low-pass filtering, which attenuates the high frequency noise influence.

5.3.1 Estimation model

Recall the simplified model (5.1). To update the parameters we need to discretize them. Regarding to small time window, we assume that $\mathbf{M}(\mathbf{t})$ and $\mathbf{N}(\mathbf{t})$ are approximated as a constant $\overline{\mathbf{M}}$ and a linear relation $\overline{\mathbf{N}} = \overline{\mathbf{N}}_\mathbf{0} + \overline{\mathbf{N}}_\mathbf{1}\mathbf{t}$. More precisely, the inertia matrix $\mathbf{M}(\mathbf{t})$ is a function of $\mathbf{q}$ which can be considered constant in short time interval; while the vector $\mathbf{N}(\mathbf{t})$ is a quadratic functions of $\mathbf{q}$ and $\dot{\mathbf{q}}$ whose variation cannot be ignored even in small time interval. In this case it can be treated as linear component. Thus the model rewrites as:

$$\tau = \overline{\mathbf{M}}\ddot{\mathbf{q}} + \overline{\mathbf{N}}_\mathbf{0} + \overline{\mathbf{N}}_\mathbf{1}\mathbf{t}. \tag{5.6}$$

In this case these parameters can be update in real time to reconstruct system dynamics. The number of parameters is greatly reduced so that for a n-link manipulator, the simplified model contains $\frac{n^2+5n}{2}$ parameters. With respect to the 2R scara planar robot, the simplified model contains 7 parameters to identify, while the IDIM contains 8 minimal parameters.

Then, modulating functions are applied to decrease the order of input variables. Let g be a k^{th} order modulating function on $[0, T]$ where $k \geq 2$. Multiply g with acceleration $\ddot{\mathbf{q}}$ and integrate on $[0, T]$. By partial integration, input $\ddot{\mathbf{q}}$ decrease its order to position input $\mathbf{q}$ and modulating function g increase to $\ddot{g}$ which is analytically known (because $g(0) = g(T) = \dot{g}(0) = \dot{g}(T) = 0$). Recall the property (5.7):

$$\int_0^T g\ddot{\mathbf{q}} = \int_0^T \dot{g}\dot{\mathbf{q}} = \int_0^T \ddot{g}\mathbf{q}.$$

Multiply equation (5.6) by modulating function g and integrate on $[0, T]$, using equation (5.7) we formulate the estimation model as:

$$\int_0^T g\tau = \overline{M}\int_0^T \ddot{g}\mathbf{q} + \overline{N}_0\int_0^T g + \overline{N}_1\int_0^T gt \tag{5.7}$$

Notice that equation (5.7) contains n equations. To solve the unknowns it need additional data from multi-equations whose number must not be smaller than that of the unknowns. This can be realized by adding a variable ℓ to modulating function g where $\ell \in \mathbb{R}$, and a combination of different group of modulating functions. With enough sequence of ℓ, the estimator forms an overdetermined observation matrix and it can be solved by least square techniques. Finally we get the linear estimation model

$$\int_0^T g_\ell\tau = \left[\int_0^T \ddot{g}_\ell\mathbf{q} \quad \int_0^T g_\ell \quad \int_0^T g_\ell t\right]\left[\overline{M}\ \overline{N}_0\ \overline{N}_1\right]. \tag{5.8}$$

These scalar equations give an overdetermined system which is linear with respect to unknown parameters $X_s = [\overline{M}\ \overline{N}_0\ \overline{N}_1]$, or can be expressed as $\mathbf{B} = \mathbf{A}\mathbf{X}_\mathbf{s}$. This kind of problem can be solved by minimizing the Euclidian length of the residual vector $\min_{\mathbf{X_s}} \|\mathbf{A}\mathbf{X}_\mathbf{s} - \mathbf{B}\|$, which gives a unique optimal $\hat{\mathbf{X}}_\mathbf{s}$ as solution. There exists a lot of least square (LS) techniques such as OLS, WLS, ILS and so on, and we apply the OLS.

5.4 Simulation

This simulation part utilizes the robot model described in chapter (1.1.1), without considering the friction offset parameter Γ_{off}. Thus, there are eight minimal dynamic parameters $\mathbf{X}$ and recall (1.12)

$$\mathbf{X} = [ZZ_{1R},\ ZZ_2,\ LMX_2,\ LMY_2,\ F_{v1},\ F_{s1},\ F_{v2},\ F_{s2}].$$

The simulation tests are running with value $\mathbf{X}$ which is all in SI Units: $\mathbf{X}$ = [3.5 0.06 0.12 0.005 0.05 0.5 0.01 0.1].

Recall the robot energy model in chapter (1.2) and the we calculate each component as:

$\mathbf{H}(1,1) = ZZ_{1R} + ZZ_2 + 2(C2LMX_2 - S2LMY_2)$,
$\mathbf{H}(1,2) = ZZ_2 + (C2LMX_2 - S2LMY2)$,
$\mathbf{H}(2,2) = ZZ_2$,
$\mathbf{C}(1,1) = -\dot{q}_2(C2LMY_2 + S2LMX_2)$,
$\mathbf{C}(1,2) = -(\dot{q}_1 + \dot{q}_2)(C2LMY_2 + S2LMX_2)$,
$\mathbf{C}(2,1) = \dot{q}_1(C2LMY_2 + S2LMX_2)$,
$\mathbf{C}(2,2) = 0$,
$\mathbf{Q} = \mathbf{0}$,
$\tau_{\mathbf{f}}(1) = F_{V1}\dot{q}_1 + F_{C1}sign(\dot{q}_1)$,
$\tau_{\mathbf{f}}(2) = F_{V2}\dot{q}_1 + F_{C2}sign(\dot{q}_2)$,
with $C1 = \cos(q_1)$ and $C2 = \cos(q_2)$.

Consider the simplified robot model (5.1), we have

$$\mathbf{M}(\mathbf{t}) = \mathbf{H}(\mathbf{q}),\ \mathbf{N}(\mathbf{t}) = \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{Q}(\mathbf{q}) + \tau_{\mathbf{f}}.$$

And $\overline{M}$ and $\overline{N}$ are to be estimated at instant t which is approximately equal to the value of $\mathbf{M}(\mathbf{t})$ and $\mathbf{N}(\mathbf{t})$. The simulation task is to track the desired trajectories using the proposed adaptive control and real time estimation associated with JMF modulating functions. The modulating set is a combination of JMF $g_\ell(v)$:

$$\begin{aligned} g_{\ell 1}(v) &= \left(\frac{v-t+T}{T}\right)^{\ell}\left(\frac{v-t}{T}\right)^{\ell}, \\ g_{\ell 2}(v) &= \left(\frac{v-t+T}{T}\right)^{\ell}\left(\frac{v-t}{T}\right)^{\ell+1}, \end{aligned} \tag{5.9}$$

where $v \in [t-T, t]$ and $\ell = \{\ell \in \mathbb{N} | \ell = 10, 11, \ldots, 30\}$ for JMF.

The trajectories are defined from point to point and between each two points the joint position, velocity, acceleration trajectories are planned as high order polynomials, which can offer good excitation. The simulation sampling frequency is 1000 Hz. First we investigate the noise free case in the simulation.

The gains are selected with $\lambda_{p1} = 49$, $\lambda_{p2} = 580$, $\lambda_{d1} = 31$, $\lambda_{d2} = 46$, $\lambda_{i1} = 30$, $\lambda_{i2} = 3000$ (see appendix A.4). In the feedback loop, the joint position feedback considers the current joint position measurement; the joint velocity feedback is computed using one step backward difference algorithm with the joint position measurement; the integration feedback is the integration of the error position. In order to test the controller, we update M and N by their real value and we show the reference trajectories for joint position and velocity in Fig (5.2) and the computed torque in Fig. (5.3). The tracking error is less than $6 \cdot 10^{-1}$ radiance from the simulation.

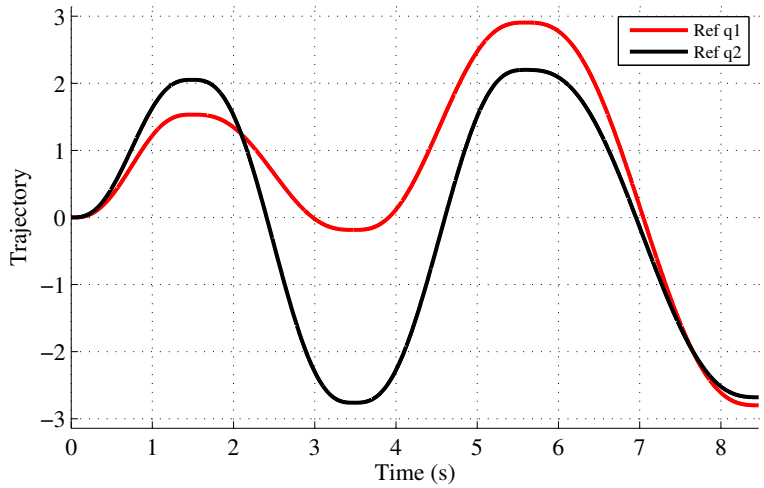
5.4.1 Simulation results with real time estimation

In the estimation process we bound the estimation increase step in order to attenuate the influence the wrong estimation due to the ill excitation at some points. We use QR factorization method to solve the least square problem. The estimation time window can be as small as $0.1s$ with adaptation frequency of 100 Hz, because the computation is small and can be implemented real time.

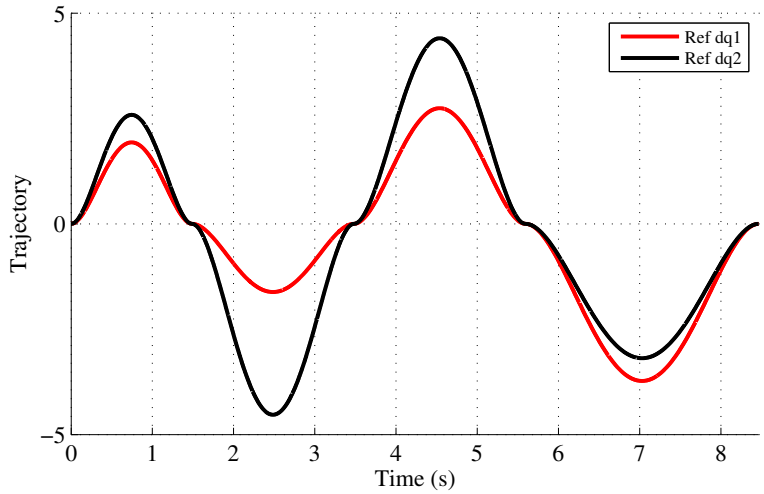
Using the above configuration, the simulation results are good with tracking error less than 0.02. And estimation value of $\overline{M}$ and $\overline{N}$ are floating around the real value. The estimation time is 0.1 second so that the time varying parameters $M(t)$ and $N(t)$ can be estimated. From the Fig. (5.7) it can be found that M is quasi constant but N is varying fast with respect to estimation window. The estimated $\overline{N}$ are reconstructed from two estimation $\overline{N}_0$ and $\overline{N}_1$. The proposed simplified model which consider $\overline{M}$ is constant and $\overline{N} = \overline{N}_0 + \overline{N}_1 t$ is a linear component is reasonable from this result and it can be extended to common manipulator applications because most of their trajectories dynamic properties are similar as those in this case. The computed torque using the estimation of $\overline{M}$ and $\overline{N}$ is given in Fig. (5.4).

5.4.2 Results Robust To Variation Payload

In real applications, sometimes the payload changes during the manipulator operation. Adjusting them on-line is necessary for robust control. This iterative learning adaptive controller is a solution to variation of system dynamics, and in our case the real time estimation ensures the quick response to corrupt



(a) Reference position



(b) Reference velocity

Figure 5.2 – Reference trajectory

change. In simulation, at instant $t = 2s$, ZZ_{1R} changes from 3.5 to 5, and at instant $t = 5s$, ZZ_{1R} changes from 5 to 4. This can simulate the corrupt change of payload. Apply the adaptive control, and result is good with tracking error less than 0.02. The estimation of $\overline{M}$ can be found in Fig. (5.8). Notice that there is a delay of about $0.5s$ before getting the correct estimation of M. This delay is caused by the bounded estimation increase step and the estimation window $0.1s$, as well it needs some time to recover from variation of system dynamics to re-estimate the changed parameter. During this transition period, the estimated parameters are varying smoothly to the correct value. The computed torque is shown in Fig. (5.5).

However, the real time estimation depends on the reference trajectory dy-

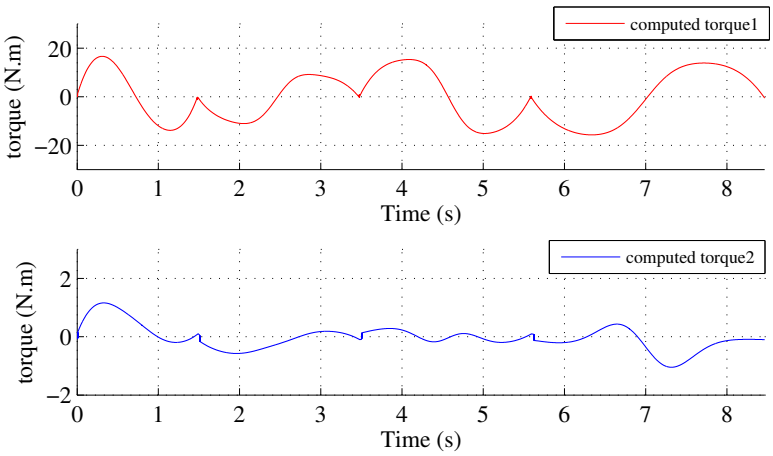


Figure 5.3 – Computed torque with updation of accurate M and N

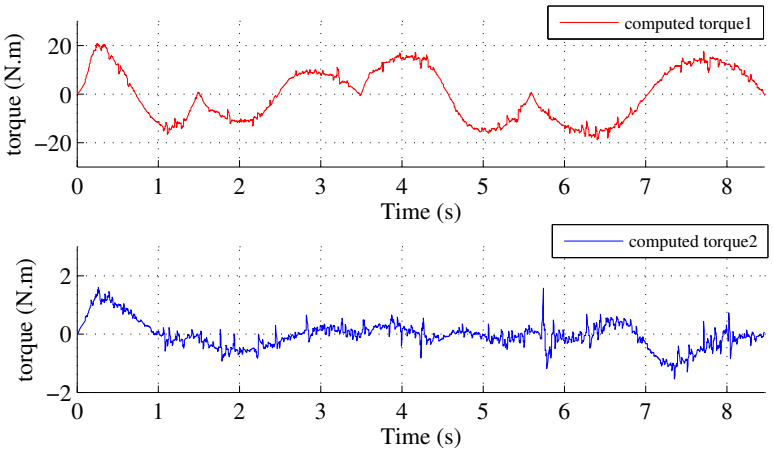


Figure 5.4 – Computed torque in normal tracking task

namics. To illustrate this, we consider from two aspects by:

- reducing the dynamics of the reference trajectory by ten times, by regulating the maximum velocity and maximum acceleration;
- increasing the dynamics of the reference trajectory by ten times.

When the reference trajectory dynamics reduce, the robot motion becomes slower locally, which means $M(t)$ and $N(t)$ is varying slowly. In this case, small time estimation window does not offer advantages because trivial trajectory contains less information for estimation. Thus, in this case, the estimation window is enlarged two times in order to have a good estimation.

When the reference trajectory dynamics increase, it is more difficult to apply the real time estimation, because of the conflict between the requisition for enough sampling data and the estimation time window. In this case, the esti-

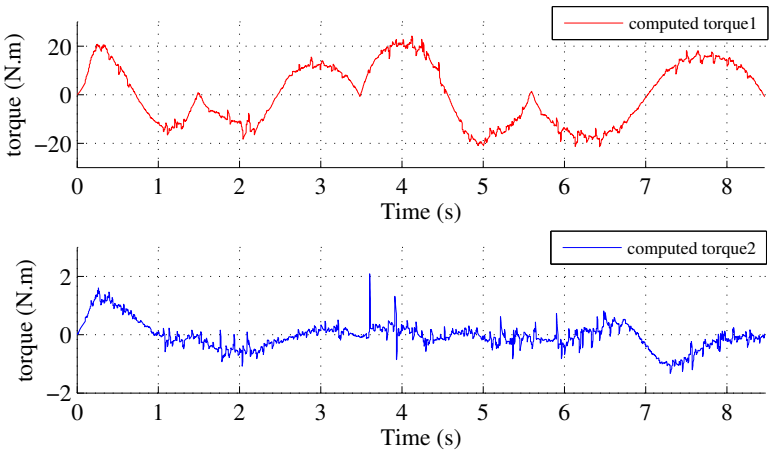


Figure 5.5 – Computed torque when ZZ_{1R} varies

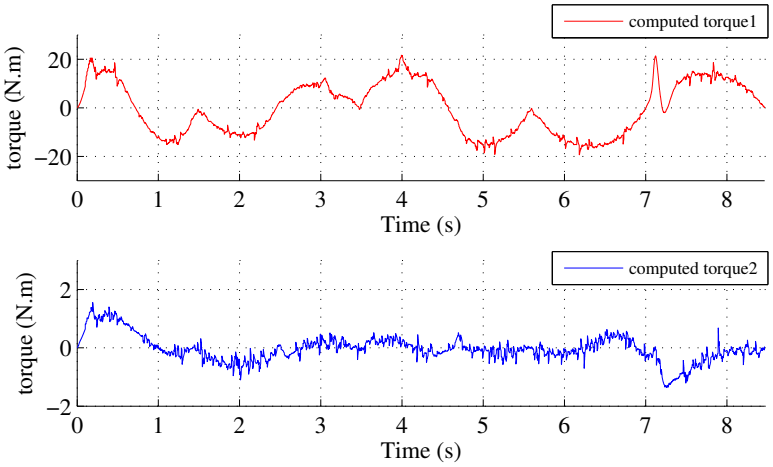


Figure 5.6 – Computed torque in normal tracking task with noise

mation time window should be reduced in order to ensure the assumption that M is constant and N is a linear segment locally in the estimation window. But with less sampling data, the estimation is hard to implemented.

5.4.3 Simulation with noise

We simulate the measured joint position and joint torques with high frequency normally distributed random noise (low frequency part is filtered), the signal to noise ratio is 30dB. Notice that we apply the modulating functions based approach without pre-filtering of the measurement. In this case, the noise influence in the feedback derivative and in the estimation part are non-negligible so that with the previous configuration of gains, the controller obtains large bias in the tracking error. In order to overcome this, we select high gains for the error regulation: with $\omega_1 = \omega_2 = 20\ rad/s$, $\xi_1 = \xi_2 = 0.8$, $T_1 = T_2 = 30\ s^{-1}$,

which gives $\lambda_{p1} = \lambda_{p2} = 1360$, $\lambda_{d1} = \lambda_{d2} = 62$, $\lambda_{i1} = \lambda_{i2} = 12000$. The estimated parameters and tracking error are given in Fig. (5.9), where the tracking error is bounded less than 0.04 radiance. The computed torque is shown in Fig. (5.6), where there exist more disturbance in the computed torque.

5.5 Conclusion

We propose an iterative learning control structure associated with a robust real time estimation module for robot manipulator tracking task. The robot IDIM is replaced by a simplified differential model, which reduces largely the number of parameters to be estimated and decreases the complexity of estimation process. With the simple structure, the estimation time is sharply reduced which means it can be real time estimation and it responds faster to variation of system dynamics. Meanwhile modulating function approach is considered in estimation process. Modulating functions allow to decrease the order of model input via integration. This can avoid the numerical computation of high order derivatives of measured signal, where noise usually induces large error in derivatives calculation. Finally, only the joint position and joint torque are needed in the estimation . The contribution to investigate the frequency domain response of different modulating functions is discussed in chapter 2. The selected modulating functions have a low pass filtering property. This gives simplicity to estimation module because it is not necessary to pre-process the signal to filter the noise component. And compared to to common filter, the modulating function approach needs only the causal data and calculates a scalar without considering phase shift.

But it should be noticed that the modulating function method will lower the identifiability of the parameters because it loses excitation by combining vectors to scalar. And when noise is aliased with signal, modulating functions is sensitive to noise and usually enlarge the noise contribution. For future, experimental work should be carried out on robot manipulator and test should be applied on robot with more links. At the same time, different approaches should be investigated in the real time estimation process in order to deal with more kinds of noise.

At the end, the simulation results show that the estimation can be done in short time and the tracking error is small with respect to the corrupt change in the dynamic parameters.

However, there exist some limits of this method:

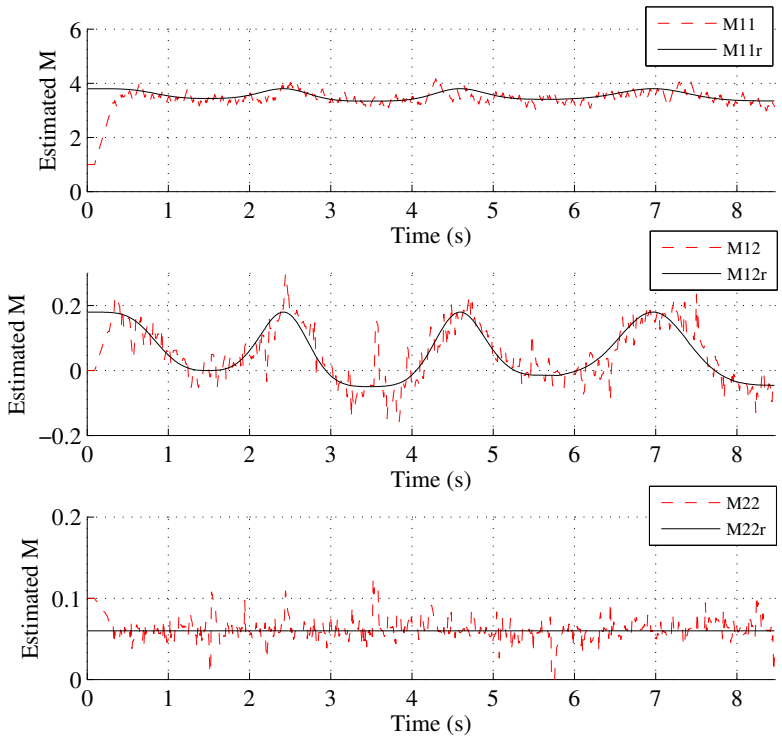
- First, because it reconstruct the local system dynamics, thus the data acquisition frequency should be high enough in order to ensure the identifiability of the real time estimation. In the simulation case, we set sampling frequency of 1000 Hz, which offers 100 sampling data within the estimation time window $0.1s$, which is good for estimation. While in practice, when system sampling frequency is low which means with less data, this estimation procedure is lack of accuracy.

- Second, the real time estimation procedure relies on the reference trajectory dynamics, where the estimation time window should be properly selected according to the reference trajectory dynamics. For example, if the bandwidth of the reference trajectory is high, the estimation time window should be reduced in order to satisfy the local assumption that M is constant, N is linear segment, which gives difficulty to estimation procedure; instead, if the bandwidth is low, the estimation time window can be selected larger because trivial data can not offer good estimation.

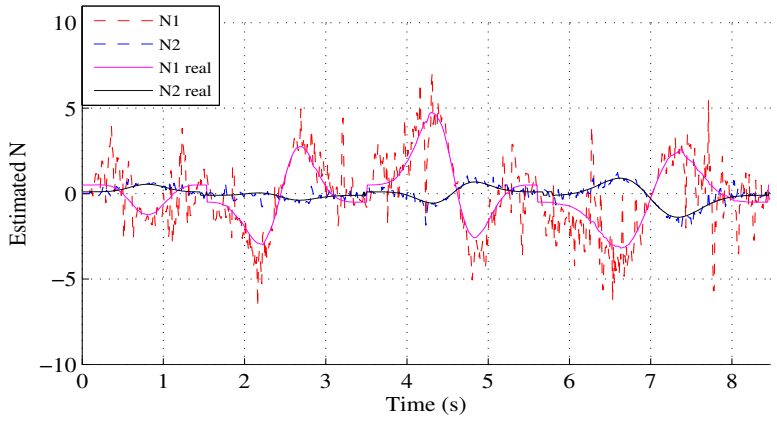
- Third, for complex mechanical systems, some parameters of the simplified model become un-identifiable, with increase of the dimension of M and N.

- Fourth, the computed torques are not as smooth as those generated with off-line identified parameters.

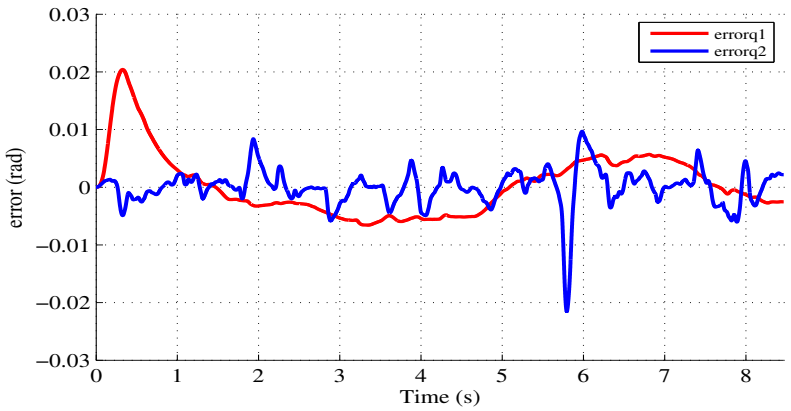
Though it has some drawbacks, this method is still interesting because the system dynamics are reconstructed in short time interval without large numerical computation nor joint velocity and acceleration data. Besides, this approach can be extended to a method to detect system changes.



(a) Estimated $\overline{M}$

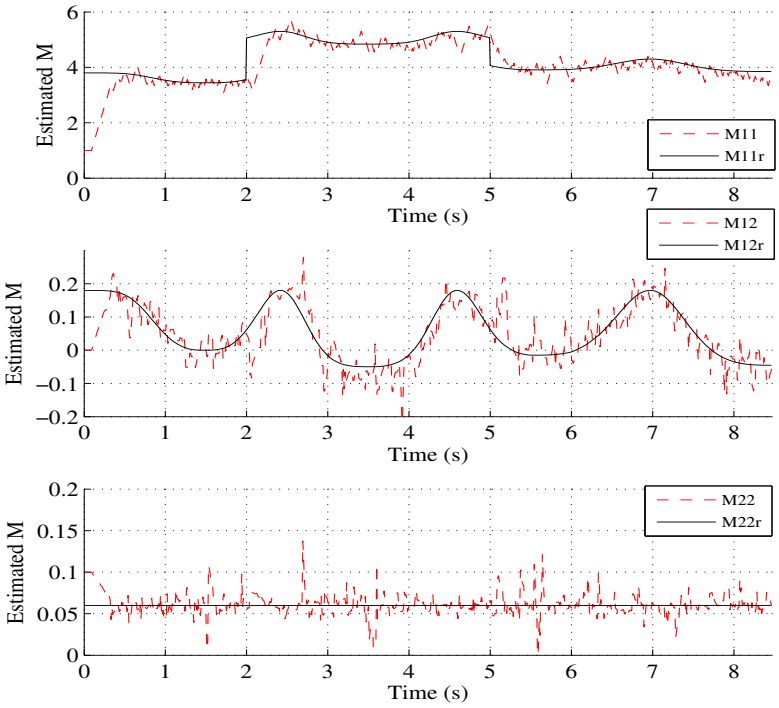


(b) Estimated $\overline{N}$

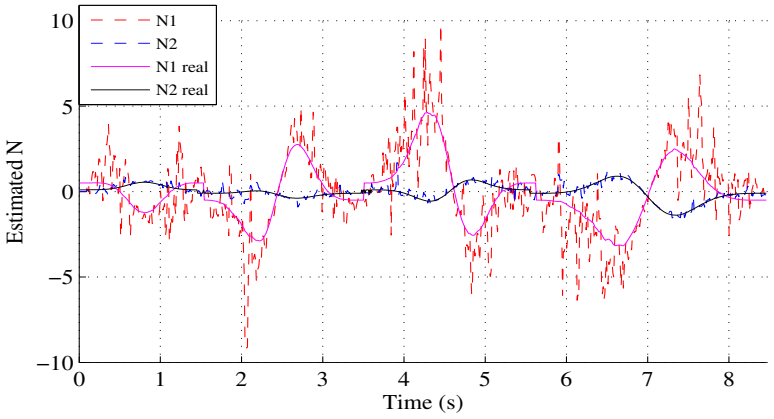


(c) Tracking error

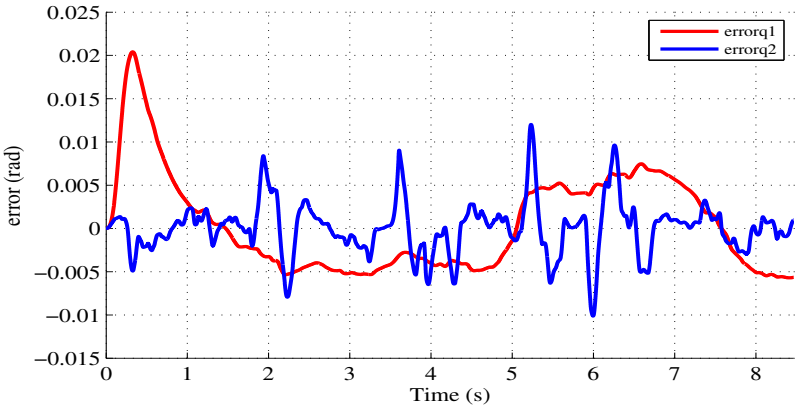
Figure 5.7 – Estimated parameters and tracking error in normal tracking task



(a) Estimated $\overline{M}$

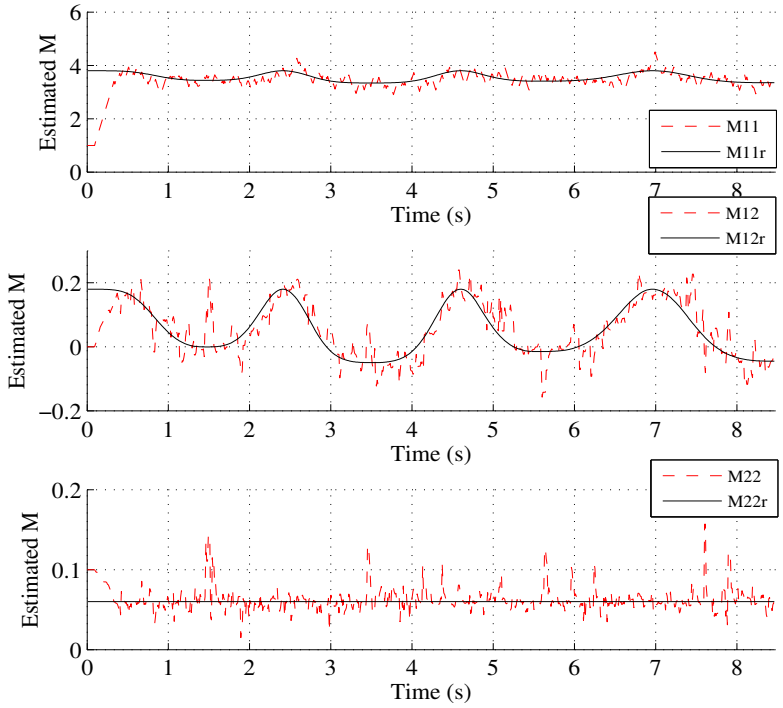


(b) Estimated $\overline{N}$

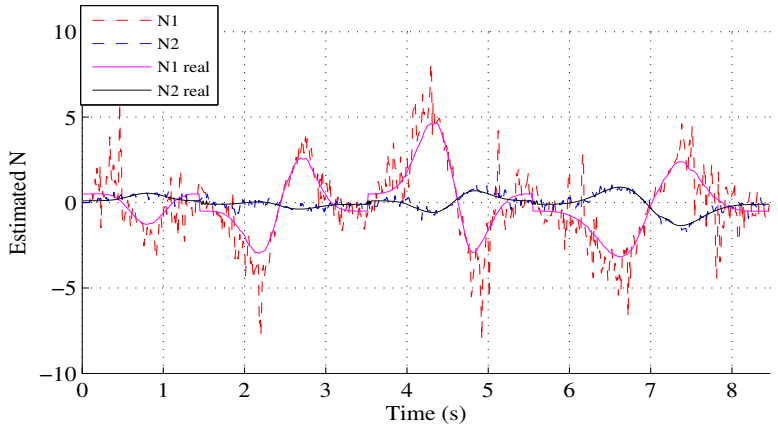


(c) Tracking error

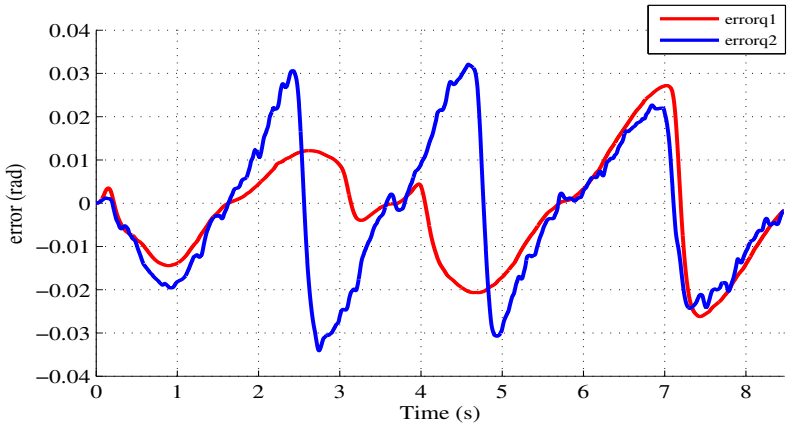
Figure 5.8 – Estimation of $\overline{M}$, $\overline{N}$ and tracking error when ZZ_{1R} varies



(a) Estimated $\overline{M}$



(b) Estimated $\overline{N}$



(c) Tracking error

Figure 5.9 – Estimation of $\overline{M}$, $\overline{N}$ and tracking error in normal tracking task with noise

Conclusion

This work is dedicated to the robot identification issues. It has the following contributions:

- propose the **modulating functions with power model approach** on robot identification;
- analyse the **frequency domain property** of integration with modulating function in order to select good groups of modulating functions for identification use;
- introduce the **Jacobi differentiators** and for the first time, analyse in frequency domain to understand their filtering properties, thus **evaluation and comparisons** with other differentiators are possible;
- make **comparisons** among different identification model, differentiators, LS techniques for robot identification issues and draw conclusion;
- from the simplified model of robot manipulator, we **estimate the states in real time** using modulating functions approach for adaptation of states in the controller.

In chapter 1, it reviews in detail the robot IDIM, energy and power identification models. In chapter 2, a new identification method is proposed as modulating functions with power model approach. This methods identify the dynamic parameters without considering the joint acceleration data. It integrates the power model with modulating functions, and apply integration by part which can turn the derivative operation into an integration operation according to the property of modulating functions. It is a similar approach to energy identification model, while the variations of the modulating functions in this method offers better rank efficiency and condition number of the observation matrix. As well, the integration effect with modulating functions are investigated in frequency domain so that certain groups of modulating functions

are selected with low-pass filtering property for identification use. Comparisons are done on a 2R prototype scara planar robot with IDIM-OLS method, the modulating functions with power model identification proves better precision.

In chapter 3, we introduce the Jacobi differentiators. Analysis in both time domain and frequency domain present the influences of each parameter. Especially, analysis in frequency domain indicate the cutoff property of the differentiator, which make the design of Jacobi differentiators possible. In nature, Jacobi differentiators are low-pass differentiators because of its integration operation. Both the causal and central Jacobi differentiators keep good linear phase property at low frequency. Then, the Jacobi differentiators are applied on the 2R prototype scara planar robot and the EMPS for identification tests. The results show that they are robust differentiators and can be adopted in applications.

In chapter 4, we compare the identification approaches in three aspects: robot identification model, differentiation methods, LS techniques. The tests are done with 2R scara planar robot simulation model and prototype robot. The following conclusions are arrived: comparing the identification models, the modulating function with power model and IDIM-WLS method are the best solution, while energy and power identification model also presents good performance, then IDIM-OLS has relatively larger errors; comparing the differentiators, the central Jacobi differentiator is a robust off-line differentiator, which has the same or better performance than central difference with Butterworth filter approach, it should be noticed that both methods are a band-pass filtering procedure; comparing the LS techniques, WLS offers better solution than OLS and ILS, but WLS only can be applied in IDIM and power identification model, besides, OLS and ILS have the same solution when the observation matrix has full rank.

In chapter 5, we simplified the robot manipulator model and estimate directly its states in a short time window of $0.1s$. The observation use the modulating functions approaches, where only the joint position and torque data are required. The estimation is fast so that it can represent the system dynamics and update them in the controller. In the end, the simulation results show that this control structure presents good estimation of model parameters and tracking precision.

In all, the robot identification problems exist for long time and the most important thing is how to eliminate the noise influence. In this work, we propose some methods based on the algebraic set : use the modulating functions with

power model approach to remove the noisy computation of acceleration; and apply the Jacobi differentiators in order to get robust estimation of high order derivatives.

Prospective

The robot identification issues are widely discussed and the identification procedures are necessary in practice for applications. Thus, it is important to associate the identification method with real application. While this work proposes some solutions based on algebraic point of views, and discusses the applications on simple mechanical systems. It is more theoretical work and need to be adapted to robot application.

For future work, interests are listed as:

- the inherent properties of noise should be studied corresponding to real application;
- extend and apply these algebraic methods on complex robot systems to test the performance;
- study the case where insufficient excitation is presented;
- reduce the time window and increase the precision for identification.

Bibliography

Abramowitz, Milton and Irene A Stegun, eds. (1965). *Handbook of mathematical functions*. GPO.

Alexandre Janot Maxime Gautier, A. Jubien and Pierre Olivier Vandanjon (2011). "Experimental joint stiffness identification depending on measurements availability". In: *50th IEEE Conference on Decision and Control and European Control Conference*. IEEE, pp. 5112–5117.

Anderssen, Bob, Frank De Hoog, and Markus Hegland (1998). "A stable finite difference ansatz for higher order differentiation of non-exact data". In: *Bulletin of the Australian Mathematical Society* Vol. 58.No. 02, pp. 223–232.

Antonelli, Gianluca, Fabrizio Caccavale, and Pasquale Chiacchio (1999). "A systematic procedure for the identification of dynamic parameters of robot manipulators". In: *Robotica* Vol. 17.No. 04, pp. 427–435.

Aronszajn, Nachman (1950). "Theory of reproducing kernels". In: *Transactions of the American mathematical society*, pp. 337–404.

Bao, Ping-An, Ping Jiang, and Hui-Tang Chen (1996). "A learning scheme for the parameter identification of robot dynamics". In: *Industrial Technology, 1996. (ICIT '96), Proceedings of The IEEE International Conference on*, pp. 651–655.

Barak, P. (1995). "Smoothing and Differentiation by an Adaptive-Degree Polynomial Filter". In: *Anal. Chem.* Vol. 67, pp. 2758–2762.

Bristow, Douglas, Marina Tharayil, Andrew G Alleyne, et al. (2006). "A survey of iterative learning control". In: *Control Systems, IEEE* Vol. 26.No. 3, pp. 96–114.

Brown, Ronald H, Susan C Schneider, and Michael G Mulligan (1992). "Analysis of algorithms for velocity estimation from discrete position versus time data". In: *Industrial Electronics, IEEE Transactions on* Vol. 39.No. 1, pp. 11–19.

Bukkems, Björn, Dragan Kosti, Bram De Jager, and Maarten Steinbuch (2005). "Learning-based identification and iterative learning control of direct-drive robots". In: *Control Systems Technology, IEEE Transactions on* Vol. 13.No. 4, pp. 537–549.

Canudas de Wit, C and A Aubin (1991). "Robot parameter identification via sequential hybrid estimation algorithm". In: *Robotics and Automation, 1991. Proceedings., 1991 IEEE International Conference on*. IEEE, pp. 952–957.

Chen, Dali, Yang Quan Chen, and Dingyu Xue (2011). "Digital fractional order Savitzky-Golay differentiator". In: *Circuits and Systems II: Express Briefs, IEEE Transactions on* Vol. 58.No. 11, pp. 758–762.

Chitour, Yacine (2002). "Time-varying high-gain observers for numerical differentiation". In: *Automatic Control, IEEE Transactions on* Vol. 47.No. 9, pp. 1565–1569.

Co, TB and BE Ydstie (1990). "System identification using modulating functions and fast Fourier transforms". In: *Computers & chemical engineering* Vol. 14.No. 10, pp. 1051–1066.

Co, Tomas B and Sridhar Ungarala (1997). "Batch scheme recursive parameter estimation of continuous-time systems using the modulating functions method". In: *Automatica* Vol. 33.No. 6, pp. 1185–1191.

Collado, Francisco De Ass Garca, Brigitte D'Andréa-Novel, Michel Fliess, Hugues Mounier, et al. (2009). "Analyse fréquentielle des dérivateurs algébriques". In: *XXIIe Colloque GRETSI*.

Cullum, Jane (1971). "Numerical differentiation and regularization". In: *SIAM J. Numer. Anal.* Vol. 8, pp. 254–265.

Daniel-Berhe, S and Heinz Unbehauen (1998). "Bilinear continuous-time systems identification via Hartley-based modulating functions". In: *Automatica* Vol. 34.No. 4, pp. 499–503.

De Larminat P and Thomas Y (1977). *Automatique des systèmes linéaires : Tome 2: identification*. Paris, France: Editions Flammarion.

Diop, S., J.W. Grizzle, and F. Chaplais (2000). "On numerical differentiation algorithms for nonlinear estimation". In: *Proc. 39th IEEE Conf. Decision Control*. Sidney.

Diop, S., J.W. Grizzle, P.E. Moraal, and A. Stefanopoulou (1994). "Interpolation and Numerical Differentiation for Observer Design". In: *Proc. Amer. Control Conf.* Baltimore, pp. 1329–1333.

Dou, Fang-Fang, Chu-Li Fu, and Yun-Jie Ma (2010). "A wavelet-Galerkin method for high order numerical differentiation". In: *Appl. Math. Comput.* Vol. 215, pp. 3702–3712.

Driels, Morris R and Uday S Pathre (1990). "Significance of observation strategy on the design of robot calibration experiments". In: *Journal of Robotic Systems* Vol. 7.No. 2, pp. 197–223.

Fedele, Giuseppe and Loredana Coluccio (2010). "A recursive scheme for frequency estimation using the modulating functions method". In: *Applied Mathematics and Computation* Vol. 216.No. 5, pp. 1393–1400.

Fedele, Giuseppe, Ciro Picardi, and Domenico Sgro (2009). "A power electrical signal tracking strategy based on the modulating functions method". In: *Industrial Electronics, IEEE Transactions on* Vol. 56.No. 10, pp. 4079–4087.

Fliess, Michel (2006). "Analyse non standard du bruit". In: *C.R. Acad. Sci. Paris Ser. I* Vol. 342, pp. 797–802.

Fliess, Michel, Cédric Join, Mamadou Mboup, and Hebertt Sira-Ramrez (2004). "Compression différentielle de transitoires bruités". In: *Comptes Rendus Mathematique* Vol. 339.No. 11, pp. 821–826.

Fliess, Michel, Mamadou Mboup, Hugues Mounier, and Hebertt Sira-Ramirez (2003). "Questioning some paradigms of signal processing via concrete examples". In: *Algebraic Methods in Flatness, Signal Processing and State Estimation*. Editorial Lagares, pp–1.

Fliess, Michel and Hebertt Sira-Ramirez (2004). "An algebraic framework for linear identification". In: *ESAIM controle optimisation et calcul des variations* Vol. 9, p. 151.

Fliess, Michel and Hebertt Sira-Ramrez (2004). "Reconstructeurs d'état". In: *Comptes Rendus Mathematique* Vol. 338.No. 1, pp. 91–96.

Fong, David Chin-Lung and Michael Saunders (2011). "LSMR: An iterative algorithm for sparse least-squares problems". In: *SIAM Journal on Scientific Computing* Vol. 33.No. 5, pp. 2950–2971.

Fu, Chu-Li, Xiao-Li Feng, and Zhi Qian (2010). "Wavelets and high order numerical differentiation". In: *Appl. Math. Model.* Vol. 34, pp. 3008–3021.

Gautier, M and Ph Poignet (2001). "Extended Kalman filtering and weighted least squares dynamic identification of robot". In: *Control Engineering Practice* Vol. 9.No. 12, pp. 1361–1372.

Gautier, Maxime (1986). "Identification of robots dynamics". In: *IFAC/IFIP/IMACS Symposium on Theory of Robots*, pp. 125–130.

— (1990). "Contribution à la modélisation et à l'identification des robots". PhD thesis. Nantes.

— (1991). "Numerical calculation of the base inertial parameters of robots". In: *Journal of Robotic Systems* Vol. 8.No. 4, pp. 485–506.

— (1996). "A comparison of filtered models for dynamic identification of robots". In: *Decision and Control, 1996., Proceedings of the 35th IEEE*. Vol. 1. IEEE, pp. 875–880.

— (1997). "Dynamic identification of robots with power model". In: *Robotics and Automation, 1997. Proceedings., 1997 IEEE International Conference on*. Vol. 3. IEEE, pp. 1922–1927.

Gautier, Maxime, Alexandre Janot, and Pierre Olivier Vandanjon (2008). "DIDIM: A new method for the dynamic identification of robots from only torque data". In: *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*. IEEE, pp. 2122–2127.

Gautier, Maxime, Alexandre Janot, and P-O. Vandanjon (2013). "A New Closed-Loop Output Error Method for Parameter Identification of Robot Dynamics". In: *Control Systems Technology, IEEE Transactions on* Vol. 21.No. 2, pp. 428–444.

Gautier, Maxime, Anthony Jubien, and Alexandre Janot (2013). "Iterative learning identification and computed torque control of robots". In: *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*. IEEE, pp. 3419–3424.

Gautier, Maxime and Wisama Khalil (1988). "On the identification of the inertial parameters of robots". In: *Decision and Control, 1988., Proceedings of the 27th IEEE Conference on*. IEEE, pp. 2264–2269.

— (1990). "Direct calculation of minimum set of inertial parameters of serial robots". In: *Robotics and Automation, IEEE Transactions on* Vol. 6.No. 3, pp. 368–373.

— (1992). "Exciting trajectories for the identification of base inertial parameters of robots". In: *The International journal of robotics research* Vol. 11.No. 4, pp. 362–375.

Gautier, Maxime, Wisama Khalil, and PP Restrepo (1995). "Identification of the dynamic parameters of a closed loop robot". In: *Robotics and Automation, 1995. Proceedings., 1995 IEEE International Conference on*. Vol. 3, pp. 3045–3050.

Gautier, Maxime, P-O Vandanjon, and Alexandre Janot (2011). "Dynamic identification of a 6 dof robot without joint position data". In: *Robotics and Automation (ICRA), 2011 IEEE International Conference on*. IEEE, pp. 234–239.

Golub, Gene H and Charles F Van Loan (2012). *Matrix computations*. Vol. 3. JHU Press.

Gorry, Peter A (1990a). "General least-squares smoothing and differentiation by the convolution (Savitzky-Golay) method". In: *Anal. Chem.* Vol. 62, pp. 570–573.

— (1990b). "General least-squares smoothing and differentiation by the convolution (Savitzky-Golay) method". In: *Analytical Chemistry* Vol. 62.No. 6, pp. 570–573.

Guo, Qi, Wilfrid Perruquetti, and Maxime Gautier (2014). "On-line robot dynamic identification based on power model, modulating functions and causal Jacobi estimator". In: *Advanced Intelligent Mechatronics (AIM), 2014 IEEE/ASME International Conference on*. IEEE, pp. 494–499.

Hanke, Martin and Otmar Scherzer (2001). "Inverse problems light: numerical differentiation". In: *Am. Math. Monthly* Vol. 108, pp. 512–521.

Hao, Dinh Nho, A Schneiders, and H-J Reinhardt (1995). "Regularization of a non-characteristic Cauchy problem for a parabolic equation". In: *Inverse Probl.* Vol. 11.No. 6, pp. 1247–1264.

Herceg, Dragoslav and Ljiljana Cvetkovic (1986). "On a numerical differentiation". In: *SIAM journal on numerical analysis* Vol. 23.No. 3, pp. 686–691.

Hollerbach, John, Wisama Khalil, and Maxime Gautier (2008). *Model Identification*. Springer, pp. 321–344.

Janot, Alexandre, Catherine Bidard, Florian Gosselin, Maxime Gautier, Delphine Keller, and Yann Perrot (2007). "Modeling and identification of a 3 DOF haptic interface". In: *Robotics and Automation, 2007 IEEE International Conference on*. IEEE, pp. 4949–4955.

Jordan, JR and NE Paterson (1986). "A modulating-function method for on-line fault detection (machinery health monitoring)". In: *Journal of Physics E: Scientific Instruments* Vol. 19.No. 9, p. 681.

Kauppinen, J.K., D.J. Moffatt, H.H. Mantsch, and D.G. Cameron (1981). "Fourier transforms in the computation of self-deconvoluted and first-order derivative spectra of overlapped band contours". In: *Anal. Chem.* Vol. 53, pp. 1454–1457.

Khalil, Wisama and Etienne Dombre (2004). *Modeling, identification and control of robots*. Butterworth-Heinemann.

Khalil, Wisama, Maxime Gautier, and Philippe Lemoine (2007). "Identification of the payload inertial parameters of industrial manipulators". In: *Robotics and Automation, 2007 IEEE International Conference on*. IEEE, pp. 4943–4948.

Khalil, Wisama, Aravindkumar Vijayalingam, Bogdan Khomutenko, Izzatbek Mukhanov, Philippe Lemoine, and Gaël Ecorchard (2014). "OpenSYMORO: An open-source software package for Symbolic Modelling of Robots". In:

Advanced Intelligent Mechatronics (AIM), 2014 IEEE/ASME International Conference on. IEEE, pp. 1206–1211.

Khan, Ishtiaq Rasool and Ryoji Ohba (2000). "New finite difference formulas for numerical differentiation". In: *J. Comput. Appl. Math.* Vol. 126, pp. 269–276.

Khatib, Oussama (1987). "A unified approach for motion and force control of robot manipulators: The operational space formulation". In: *Robotics and Automation, IEEE Journal of* Vol. 3.No. 1, pp. 43–53.

Khosla, Pradeep K and Takeo Kanade (1985). "Parameter identification of robot dynamics". In: *Decision and Control, 1985 24th IEEE Conference on*. Vol. 24. IEEE, pp. 1754–1760.

Lanczos, Cornelius (1956). *Applied Analysis*. Englewood Cliffs, NJ: Prentice-Hall.

Leung, Alexander Kai-man, Foo-tim Chau, and Jun-bin Gao (1998). "Wavelet transform: a method for derivative calculation in analytical chemistry". In: *Analytical Chemistry* Vol. 70.No. 24, pp. 5222–5229.

Levant, Arie (1998). "Robust exact differentiation via sliding mode technique". In: *Automatica* Vol. 34.No. 3, pp. 379–384.

— (2003). "Higher-order sliding modes, differentiation and output-feedback control". In: *International journal of Control* Vol. 76.No. 9-10, pp. 924–941.

Liu, Da-Yan (2011). "Error analysis of a class of derivative estimators for noisy signals and applications". PhD thesis. Université des Sciences et Technologie de Lille-Lille I.

Liu, Da-Yan, Olivier Gibaru, and Wilfrid Perruquetti (2009a). "Error analysis for a class of numerical differentiator: application to state observation". In: *48th IEEE Conf. on Decision and Control*. Shanghai, China.

— (2009b). "Error analysis for a class of numerical differentiator: application to state observation". In: *Decision and Control, 2009 held jointly with the 2009 28th Chinese Control Conference. CDC/CCC 2009. Proceedings of the 48th IEEE Conference on*. IEEE, pp. 8238–8243.

— (2011a). "Error analysis of Jacobi derivative estimators for noisy signals". In: *Numerical Algorithms* Vol. 58.No. 1, pp. 53–83.

— (2011b). "Error analysis of Jacobi derivative estimators for noisy signals". In: *Numerical Algorithms* Vol. 58.No. 1, pp. 53–83.

— (2011c). "Differentiation by integration with Jacobi polynomials". In: *J. Comput. Appl. Math.* Vol. 235, pp. 3015–3032.

— (2011d). "Differentiation by integration with Jacobi polynomials". In: *Journal of Computational and Applied Mathematics* Vol. 235.No. 9, pp. 3015–3032.

— (2012a). "Convergence rate of the causal jacobi derivative estimator". In: *Curves and Surfaces*. Springer, pp. 445–455.

— (2012b). "Convergence rate of the causal jacobi derivative estimator". In: *Curves and Surfaces*. Springer, pp. 445–455.

Liu, Da-Yan, Taous-Meriem Laleg-Kirati, Olivier Gibaru, and Wilfrid Perruquetti (2013). "Identification of fractional order systems using modulating functions method". In: *American Control Conference (ACC), 2013*. IEEE, pp. 1679–1684.

Lu, Ziren, Karunakar B Shimoga, and Andrew A Goldenberg (1993). "Experimental determination of dynamic parameters of robotic arms". In: *Journal of robotic systems* Vol. 10.No. 8, pp. 1009–1029.

Maciejewski, Anthony A and Charles A Klein (1989). "The singular value decomposition: Computation and applications to robotics". In: *The International journal of robotics research* Vol. 8.No. 6, pp. 63–79.

Mayeda, Hirokazu, Koji Yoshida, and Koichi Osuka (1990). "Base parameters of manipulator dynamic models". In: *Robotics and Automation, IEEE Transactions on* Vol. 6.No. 3, pp. 312–321.

Mboup, Mamadou (2007). "Parameter estimation via differential algebra and operational calculus". In:

Mboup, Mamadou, Cédric Join, and Michel Fliess (2007). "A revised look at numerical differentiation with an application to nonlinear feedback control". In: *The 15th Mediterrean Conference on Control and Automation-MED'2007*.

— (2009a). "Numerical differentiation with annihilators in noisy environment". In: *Numerical algorithms* Vol. 50.No. 4, pp. 439–467.

— (2009b). "Numerical differentiation with annihilators in noisy environment". In: *Numerical Algorithms* Vol. 50.No. 4, pp. 439–467.

McClellan, James H, Thomas W Parks, and Lawrence R Rabiner (1973). "A computer program for designing optimum FIR linear phase digital filters". In: *Audio and Electroacoustics, IEEE Transactions on* Vol. 21.No. 6, pp. 506–526.

Murio, DA, Carlos E Meja, and S Zhan (1998). "Discrete mollification and automatic numerical differentiation". In: *Computers & Mathematics with Applications* Vol. 35.No. 5, pp. 1–16.

Murio, Diego A (1993). *The Mollification Method and the Numerical Solution of Ill-Posed Problems*. John Wiley & Sons Inc.

Nakamura, Gen, Shengzhang Wang, and Yanbo Wang (2008). "Numerical differentiation for the second order derivatives of functions of two variables". In: *Journal of Computational and Applied Mathematics* Vol. 212.No. 2, pp. 341–358.

Nie, Lei, Shouguo Wu, Xiangqin Lin, Longzhen Zheng, and Lei Rui (2002). "Approximate derivative calculated by using continuous wavelet transform". In: *Journal of chemical information and computer sciences* Vol. 42.No. 2, pp. 274–283.

Ogata, Katsuhiko and Yanjuan Yang (1970). "Modern control engineering". In:

Olsen, Martin M, Jan Swevers, and Walter Verdonck (2002). "Maximum likelihood identification of a dynamic robot model: Implementation issues". In: *The international Journal of robotics research* Vol. 21.No. 2, pp. 89–96.

Paige, Christopher C and Michael A Saunders (1982). "LSQR: An algorithm for sparse linear equations and sparse least squares". In: *ACM Transactions on Mathematical Software (TOMS)* Vol. 8.No. 1, pp. 43–71.

Parzen, Emanuel (1999). *Stochastic processes*. Vol. 24. SIAM.

Pearson, A and F Lee (1985). "On the identification of polynomial input-output differential systems". In: *Automatic Control, IEEE Transactions on* Vol. 30.No. 8, pp. 778–782.

Pham, Minh Tu, Maxime Gautier, and Ph Poignet (2002). "Accelerometer based identification of mechanical systems". In: *Robotics and Automation, 2002.*

Proceedings. ICRA'02. IEEE International Conference on. Vol. 4. IEEE, pp. 4293–4298.

Pham, Minh Tu, Maxime Gautier, and Philippe Poignet (2001). "Identification of joint stiffness with bandpass filtering". In: *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on*. Vol. 3. IEEE, pp. 2867–2872.

Piltan, Farzin, Mohammad Hossein Yarmahmoudi, Mohammad Shamsodini, Ebrahim Mazlomian, and Ali Hosainpour (2012). "PUMA-560 Robot Manipulator Position Computed Torque Control Methods Using MATLAB/SIMULINK and Their Integration into Graduate Nonlinear Control and MATLAB Courses". In: *International Journal of Robotics and Automation* Vol. 3.No. 3, pp. 167–191.

Polyakov, Andrey, Denis Efimov, and Wilfrid Perruquetti (2014). "Homogeneous differentiator design using implicit lyapunov function method". In: *Control Conference (ECC), 2014 European*. IEEE, pp. 288–293.

Preisig, HA and DWT Rippin (1993). "Theory and application of the modulating function methodI. Review and theory of the method and theory of the spline-type modulating functions". In: *Computers & chemical engineering* Vol. 17.No. 1, pp. 1–16.

Presse, C and Maxime Gautier (1993). "New criteria of exciting trajectories for robot identification". In: *Robotics and Automation, 1993. Proceedings., 1993 IEEE International Conference on*. IEEE, pp. 907–912.

Presse, Claudine (1994). "Identification des paramètres dynamiques des robots". PhD thesis.

Qian, Zhi, Chu-Li Fu, and Xiao-Li Feng (2006). "A modified method for high order numerical derivatives". In: *Applied mathematics and computation* Vol. 182.No. 2, pp. 1191–1200.

Qian, Zhi, Chu-Li Fu, Xiang-Tuan Xiong, and Ting Wei (2006). "Fourier truncation method for high order numerical derivatives". In: *Applied mathematics and computation* Vol. 181.No. 2, pp. 940–948.

Qu, Ruibin (1996). "A new approach to numerical differentiation and integration". In: *Mathematical and computer modelling* Vol. 24.No. 10, pp. 55–68.

Rahul, Kumar and SN Bhattacharyya (2006). "One-sided finite-difference approximations suitable for use with Richardson extrapolation". In: *Journal of Computational Physics* Vol. 219.No. 1, pp. 13–20.

Ramm, Alexander and Alexandra Smirnova (2001). "On stable numerical differentiation". In: *Mathematics of computation* Vol. 70.No. 235, pp. 1131–1153.

Rangarajana, S.K. and S.P. Purushothaman (2005). "Lanczos' generalized derivative for higher orders". In: *J. Comput. Appl. Math.* Vol. 177, pp. 461–465.

Raucent, Benoit et al. (1990). "Identification des paramètres dynamiques des robots manipulateurs". PhD thesis. Université de Louvain, Belgium.

Savitzky, Abraham and Marcel JE Golay (1964a). "Smoothing and differentiation of data by simplified least squares procedures." In: *Analytical chemistry* Vol. 36.No. 8, pp. 1627–1639.

— (1964b). "Smoothing and differentiation of data by simplified least squares procedures." In: *Analytical chemistry* Vol. 36.No. 8, pp. 1627–1639.

Shao, Xueguang and Chaoxiong Ma (2003). "A general approach to derivative calculation using wavelet transform". In: *Chemometrics and Intelligent Laboratory Systems* Vol. 69.No. 1, pp. 157–165.

Shao, Xueguang, Chunyan Pang, and Qingde Su (2000). "A novel method to calculate the approximate derivative photoacoustic spectrum using continuous wavelet transform". In: *Fresenius' journal of analytical chemistry* Vol. 367.No. 6, pp. 525–529.

Siciliano, Bruno and Oussama Khatib (2008). *Springer handbook of robotics*. Springer Science & Business Media.

Slotine, Jean-Jacques E and Weiping Li (1987). "On the adaptive control of robot manipulators". In: *The International Journal of Robotics Research* Vol. 6.No. 3, pp. 49–59.

Swevers, Jan, Chris Ganseman, D Bilgin Tukel, Joris De Schutter, and Hendrik Van Brussel (1997). "Optimal robot excitation and identification". In: *Robotics and Automation, IEEE Transactions on* Vol. 13.No. 5, pp. 730–740.

Swevers, Jan, Walter Verdonck, and Joris De Schutter (2007). "Dynamic model identification for industrial robots". In: *Control Systems, IEEE* Vol. 27.No. 5, pp. 58–71.

Szegö, Gabor (1967). *Orthogonal polynomials*. third. Vol. 23. the United States of America: AMS, Providence, RI.

Tian, Yang, Thierry Floquet, and Wilfrid Perruquetti (2008). "Fast state estimation in linear time-varying systems: an algebraic approach". In: *Decision and Control, 2008. CDC 2008. 47th IEEE Conference on*. IEEE, pp. 2539–2544.

Unbehauen, H and GP Rao (1997). "Identification of continuous-time systems: a tutorial". In: *11th IFAC Symposium on System Identification*. Vol. 3, pp. 1023–1049.

Wang, YB, XZ Jia, and J Cheng (2002). "A numerical differentiation method and its application to reconstruction of discontinuity". In: *Inverse Problems* Vol. 18.No. 6, p. 1461.

Wang, Youqing, Furong Gao, and Francis J Doyle (2009). "Survey on iterative learning control, repetitive control, and run-to-run control". In: *Journal of Process Control* Vol. 19.No. 10, pp. 1589–1600.

Wang, Zewen and Rongsheng Wen (2010). "Numerical differentiation for high orders by an integration method". In: *Journal of Computational and Applied Mathematics* Vol. 234.No. 3, pp. 941–948.

Wei, T, YC Hon, and YB Wang (2005). "Reconstruction of numerical derivatives from scattered noisy data". In: *Inverse Problems* Vol. 21.No. 2, p. 657.

Yang, Lu (2008). "A perturbation method for numerical differentiation". In: *Applied Mathematics and Computation* Vol. 199.No. 1, pp. 368–374.

Appendix A

Appendix

A.1 Least squares techniques

Seen from the above identification models, the robot identification problems turn out to be a regression question in the linear form $\mathbf{B} = \mathbf{A}\mathbf{X} + \rho$, where $\mathbf{X}$ is the $(N_b \times 1)$ base dynamic parameters to be identified, $\mathbf{A}$ is the $(n_s \times N_b)$ observation matrix, $\mathbf{B}$ is the $(n_s \times 1)$ vector of output data, and ρ is assumed to be $(n_s \times N_b)$ vector of unobserved zero mean independent identically distributed (iid) disturbances with $\rho_i \sim N(0, \sigma^2)$.

Different mathematical tools have been developed to solve the LS problems, including the ordinary LS, weighted LS, iterative LS techniques. And methods are proposed to decompose the observation matrix, such as SVD and QR decomposition.

A.1.1 Ordinary LS

In statistics, ordinary least squares (OLS), or called linear least squares, is the basic method for estimating the unknown parameters in a linear regression model. The OLS estimator is consistent when the regressors are exogenous and linearly independent, on other word the regressors should have full column rank. It is the optimal linear unbiased estimator when the errors are of homogeneous variance and are uncorrelated. Under these conditions, the method of OLS provides minimum-variance mean-unbiased estimation when the errors have finite variances. It minimizes the 2-norm Euclidean length of the residual vector:

$$\mathrm{OLS}(\hat{\mathbf{X}}) = \sum_{i=1}^{n_s} (B_i - \mathbf{A}\mathbf{X}_i)^2. \tag{A.1}$$

The optimal solution is given in the matrix form:

$$\mathrm{OLS}(\hat{\mathbf{X}}) = \mathbf{A}^{+}\mathbf{B} = (\mathbf{A}^{T}\mathbf{A})^{-1}\mathbf{A}^{T}\mathbf{B} \tag{A.2}$$

A.1.2 Weighted LS

The weighted least squares (WLS) is an improvement of the OLS method. OLS is actually the maximum likelihood estimator under assumption of homoscedasticity, where the regression curve is measured the same precision everywhere, or in other word the disturbances ρ have constant variance. However when treating the heteroskedastic problem, where the magnitude of the ρ is not constant, OLS is no longer the maximum likelihood estimate and no longer efficient. Instead the WLS techinique is implemented.

Assume that we have explicit knowledge of different variance disturbances $\rho_{\mathrm{i}} \sim \mathrm{N}(0, \sigma_{\mathrm{i}}^2)$, the weights sequence can be reconstructed such that $\omega_{\mathrm{i}} = \frac{1}{\sigma_{\mathrm{i}}^2}, \forall \mathrm{i}$. It indicates that we assign different weight to each observation, so that observations with smaller σ_i^2 are treated as more important.

Compared to OLS, it minimizes the weighted sum of the squares:

$$\mathrm{WLS}(\hat{\mathbf{X}}) = \sum_{\mathrm{i}=1}^{\mathrm{n_s}} \omega_{\mathrm{i}} (\mathrm{B_i} - (\mathbf{AX})_{\mathrm{i}})^2. \tag{A.3}$$

In fact, WLS includes OLS as a special case where all the weight coefficients $\omega_i = 1$. We can solve it by the same algebraic method used in OLS method. The optimization problem has known solution which is better understood in matrix form. The WLS solution is written as

$$\mathrm{WLS}(\hat{\mathbf{X}}) = (\mathbf{A}^{\mathrm{T}}\mathbf{WA})^{-1}\mathbf{A}^{\mathrm{T}}\mathbf{WB}, \tag{A.4}$$

where $\mathbf{W}$ is diagonal $(\mathrm{n_s} \times \mathrm{n_s})$ matrix with $W_{i,i} = \omega_i, \forall i$.

The WLS method has two advantages regarding to OLS approach:

- Focusing accuracy;
- Discounting of imprecision.

We may predict the response for certain values of input, if we give the points big weights near the reliable region and points elsewhere smaller weights, the regression will pulled towards matching the data in the reliable region. On other words, it concerns more about the fitting well where the noise is small, and expect to fit poorly where the noise is big.

A.1.3 Iterative LS

Here we refer an iterative method LSMR (Fong et al., 2011) for solving linear systems $\mathbf{Y} = \mathbf{WX}$. The LSMR is based on the Golub-kahan bidiagonalization process. It is analytically equivalent to the MINRES method applied to the normal equation $\mathbf{W^T WX} = \mathbf{W^T Y}$, so that the quantities $\|\mathbf{W^T r_k}\|$ are monotonically decreasing (where $\mathbf{r_k} = \mathbf{YWX_k}$ is the residual for the current iterate $\mathbf{X_k}$). It is observed that $\|\mathbf{r_k}\|$ also decreases monotonically, so that compared to LSQR (Paige et al., 1982) (for which only $\|\mathbf{r_k}\|$ is monotonic) it is safer to terminate LSMR early.

A.1.4 SVD decomposition

The singular value decomposition (SVD) is a factorization of a real or complex matrix. For a $(m \times n)$ real or complex matrix $\mathbf{M}$, it has a factorization of the form $\mathbf{M} = \mathbf{U}\sum\mathbf{V}^{\star}$, where $\mathbf{U}$ is a $(m \times m)$ real or complex unitary matrix, $\sum$ is a $(m \times n)$ rectangle diagonal matrix with non-negative real numbers on the diagonal and $\mathbf{V}^{\star}$ is a $(n \times n)$ real or complex unitary matrix ($\mathbf{V}^{\star}$ is the conjugate transpose of $\mathbf{V}$). In particular, the diagonals of $\sum_{ii}$ are called the singular values of $\mathbf{M}$. In LS problems, SVD are applied when computing the pseudoinverse for the following threes aspects

- to see if the observation matrix is singular;
- even if not singular, the singular values are used to compute condition number, which tells how stable the solution will be;
- set singular value $\frac{1}{\sum_{ii}}$ to 0, if $\sum_{ii}$ is near 0, for the purpose to avoid bad inverse.

A.1.5 QR factorization

The QR factorization is a decomposition of a $(m \times n)$ matrix $\mathbf{W}$ into a product $\mathbf{W} = \mathbf{QR}$ of an $(m \times m)$ orthogonal matrix $\mathbf{Q}$ and an $(m \times n)$ matrix $\begin{bmatrix} \mathbf{R} \\ \mathbf{0} \end{bmatrix}$, where $\mathbf{R}$ is a $(n \times n)$ upper triangular matrix. QR decomposition is often used to solve the linear least squares problem, and is the basis for a particular eigenvalue algorithm, the QR algorithm.

We can consider the Gram-Schmidt procedure, with the vectors to be considered in the process as columns of the matrix $\mathbf{W}$. That is,

$$\mathbf{W} = [\mathbf{a_1}\ \mathbf{a}_2\ \cdots\ \mathbf{a_n}]$$

. Then

$$\mathbf{u_1} = \mathbf{a_1},\ \mathbf{e_1} = \frac{\mathbf{u_1}}{\|\mathbf{u_1}\|},\ nonumber \tag{A.5}$$

$$\mathbf{u}_2 = \mathbf{a}_2 - (\mathbf{a}_2\mathbf{e_1})\mathbf{e_1},\ \mathbf{e}_2 = \frac{\mathbf{u}_2}{\|\mathbf{u}_2\|},\ nonumber \tag{A.6}$$

$$\mathbf{u_{k+1}} = \mathbf{a_{k+1}} - (\mathbf{a_{k+1}e_1})\mathbf{e_1} - \cdots - (\mathbf{a_{k+1}e_k})\mathbf{e_k},\ \mathbf{e_{k+1}} = \frac{\mathbf{u_{k+1}}}{\|\mathbf{u_{k+1}}\|}, \tag{A.7}$$

where $\|\|$ is the L_2 norm.

From the above, the resulting QR decomposition is

$$\mathbf{M} = [\mathbf{e_1}\ \mathbf{e}_2\ \cdots\ \mathbf{e_n}] \begin{bmatrix} \mathbf{a_1e_1} & \mathbf{a}_2\mathbf{e_1} & \cdots & \mathbf{a_ne_1} \\ \mathbf{0} & \mathbf{a}_2\mathbf{e}_2 & \cdots & \mathbf{a_ne}_2 \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{a_ne_n} \end{bmatrix} = \mathbf{Q} \begin{bmatrix} \mathbf{R} \\ \mathbf{0} \end{bmatrix} \tag{A.8}$$

In order to solve the LS problem $\mathbf{Y} = \mathbf{WX}$, we have

$$\begin{aligned}
&\mathbf{W^T WX} = \mathbf{W^T Y},\\
&\mathbf{R^T Q^T QRX} = \mathbf{R^T Q^T Y},\\
&\mathbf{R^T RX} = \mathbf{R^T Q^T Y},\ (\mathbf{Q^T Q} = \mathbf{I})\\
&\mathbf{RX} = \mathbf{Q^T Y}.
\end{aligned} \tag{A.9}$$

Finally, we solve $\mathbf{R^T RX} = \mathbf{R^T Q^T Y}$ which has a smaller cost for computation (compared to other factorization such as Cholesky factorization).

A.2 Causal Jacobi differentiator

Continuing with the definitions in section 3.2, let us recall two useful formula (Szegö, 1967)

$$P_i^{(\mu,\kappa)}(\tau)\hat{w}_{\mu,\kappa}(\tau) = \frac{(-1)^i}{i!}\frac{d^i}{d\tau^i}[\hat{w}_{\mu+i,\kappa+i}(\tau)] \quad \textit{the Rodrigues formula}, \tag{A.10}$$

$$\frac{d}{d\tau}P_i^{(\mu,\kappa)}(\tau) = (i+\mu+\kappa+1)P_{i-1}^{(\mu+1,\kappa+1)}(\tau) \tag{A.11}$$

Let x is a smooth function belong to $\mathcal{C}^n(I)$. Now we define the q^{th} order truncated Jacobi orthogonal series expansion of the n^{th} order derivative $x^{(n)}(t_0 - T\xi)$ by the following operator: $\forall\ t_0 \in I$

$$D^{(n)}_{\kappa,\mu,T,q}x(t_0 - T\xi) = \sum_{i=0}^{q} \frac{\left\langle P_i^{(\mu+n,\kappa+n)}(\cdot), x^{(n)}(t_0 - T\cdot)\right\rangle^{(0,1)}_{\mu+n,\kappa+n}}{\|P_i^{(\mu+n,\kappa+n)}\|^2_{\mu+n,\kappa+n}} P_i^{(\mu+n,\kappa+n)}(\xi), \tag{A.12}$$

We also define the $(q+n)^{th}$ order truncated Jacobi orthogonal series expansion of $x(t_0 - T\xi)$ by the following: $\forall\ t_0 \in I$

$$D^{(0)}_{\kappa,\mu,T,q}x(t_0 - T\xi) = \sum_{i=0}^{q+n} \frac{\left\langle P_i^{(\mu,\kappa)}(\cdot), x(t_0 - T\cdot)\right\rangle^{(0,1)}_{\mu,\kappa}}{\|P_i^{(\mu,\kappa)}\|^2_{\mu,\kappa}} P_i^{(\mu,\kappa)}(\xi). \tag{A.13}$$

With fixed value t_0, $D^{(0)}_{\kappa,\mu,T,q}x(t_0 - T\xi)$ is actually a polynomial which approximates the function $x(t_0 - T\xi)$. In the next part, we will demonstrate that $D^{(n)}_{\kappa,\mu,T,q}x(t_0 - T\xi)$ is in fact related to the n^{th} order derivative of $D^{(0)}_{\kappa,\mu,T,q}x(t_0 - T\xi)$, which functions as causal Jacobi differentiator.

Lemma 1. *Let $xi \in \mathcal{C}[0,1]$, then we have $\forall\ t_0 \in I$*

$$D^{(n)}_{\kappa,\mu,T,q}x(t_0 - T\xi) = \frac{1}{(-T)^n}\frac{d^n}{d\xi^n}\left[D^{(0)}_{\kappa,\mu,T,q}x(t_0 - T\xi)\right]. \tag{A.14}$$

Proof of Lemma 1. Applying n times derivations to (A.13) and according to

(A.11), we obtain

$$\begin{aligned}
&\frac{d^n}{d\xi^n}\left[D^{(0)}_{\kappa,\mu,T,q}x(t_0-T\xi)\right]\\
&=\sum_{i=0}^{q}\frac{\left\langle P^{(\mu,\kappa)}_{i+n}(\cdot),x(t_0-T\cdot)\right\rangle^{(0,1)}_{\mu,\kappa}}{\|P^{(\mu,\kappa)}_{i+n}\|^2_{\mu,\kappa}}\frac{d^n}{d\xi^n}P^{(\mu,\kappa)}_{i+n}(\xi)\\
&=\sum_{i=0}^{q}\frac{\left\langle P^{(\mu,\kappa)}_{i+n}(\cdot),x(t_0-T\cdot)\right\rangle^{(0,1)}_{\mu,\kappa}}{\|P^{(\mu,\kappa)}_{i+n}\|^2_{\mu,\kappa}}\frac{\Gamma(\mu+\kappa+2n+i+1)}{\Gamma(\mu+\kappa+n+i+1)}P^{(\mu+n,\kappa+n)}_{i}(\xi). \quad \text{(A.15)}
\end{aligned}$$

Then apply the Rodrigues formula given in (A.10) and take n integration by part, the result gives

$$\begin{aligned}
&\left\langle P^{(\mu+n,\kappa+n)}_{i}(\cdot),x^{(n)}(t_0-T\cdot)\right\rangle^{(0,1)}_{\mu+n,\kappa+n}\\
&=\int_0^1\hat{w}_{\mu+n,\kappa+n}(\tau)x^{(n)}(t_0-T\tau)d\tau\\
&=\int_0^1\frac{(-1)^i}{i!}\hat{w}^{(i)}_{\mu+n+i,\kappa+n+i}(\tau)x^{(n)}(t_0-T\tau)d\tau\\
&=\frac{1}{(-T)^n}\int_0^1\frac{(-1)^{i+n}}{i!}\hat{w}^{(n+i)}_{\mu+n+i,\kappa+n+i}(\tau)x(t_0-T\tau)d\tau\\
&=\frac{1}{(-T)^n}\int_0^1\frac{(n+i)!}{i!}\hat{w}_{\mu,\kappa}(\tau)P^{(\mu,\kappa)}_{n+i}(\tau)x(t_0-T\tau)d\tau. \quad \text{(A.16)}
\end{aligned}$$

After some calculation by using (3.5) we can obtain

$$\begin{aligned}
&\frac{\left\langle P^{(\mu+n,\kappa+n)}_{i}(\cdot),x^{(n)}(t_0-T\cdot)\right\rangle^{(0,1)}_{\mu+n,\kappa+n}}{\|P^{(\mu+n,\kappa+n)}_{i}\|^2_{\mu+n,\kappa+n}}\\
&=\frac{\left\langle P^{(\mu,\kappa)}_{i+n}(\cdot),x^{(n)}(t_0-T\cdot)\right\rangle^{(0,1)}_{\mu,\kappa}}{(-T)^n\|P^{(\mu,\kappa)}_{n+i}\|^2_{\mu,\kappa}}\frac{\Gamma(\mu+\kappa+2n+i+1)}{\Gamma(\mu+\kappa+n+i+1)}. \quad \text{(A.17)}
\end{aligned}$$

Finally, from (A.12), (A.15) and (A.17), we achieve the relation

$$\forall\ t_0\in I,\ D^{(n)}_{\kappa,\mu,T,q}x(t_0-T\xi)=\frac{1}{(-T)^n}\frac{d^n}{d\xi^n}\left[D^{(0)}_{\kappa,\mu,T,q}x(t_0-T\xi)\right]. \quad \text{(A.18)}$$

Moreover, after developing (A.18) with (A.15), we have the analytical continuous form of the causal Jacobi differentiator, which calculates the n^{th} derivative

at instant t_0, $\forall \xi \in [0,1], \forall t_0 \in I$,

$$D^{(n)}_{\kappa,\mu,T,q} x(t_0 - T\xi) = \frac{1}{(-T)^n} \int_0^1 Q_{\kappa,\mu,n,q,\xi}(\tau)\, x(t_0 - T\tau) d\tau, \tag{A.19}$$

with $\mu, \kappa \in]-1, +\infty[$,

$$C_{\kappa,\mu,n,i} = \frac{(\mu+\kappa+2n+2i+1)\Gamma(\kappa+\mu+2n+i+1)\Gamma(n+i+1)}{\Gamma(\kappa+n+i+1)\Gamma(\mu+n+i+1)}, \tag{A.20}$$

$$Q_{\kappa,\mu,n,q,\xi}(\tau) = \hat{w}_{\mu,\kappa}(\tau) \sum_{i=0}^{q} C_{\kappa,\mu,n,i} P_i^{(\mu+n,\kappa+n)}(\xi) P_{n+i}^{(\mu,\kappa)}(\tau). \tag{A.21}$$

□

A.3 Central Jacobi differentiator

Proposition 1. *Let x still be a smooth function in $C^n(I)$, then a family of central estimators of $x^{(n)}$ can be given as follows*

$$\forall t_0 \in I,\ \overline{D}^{(n)}_{h,\mu,\kappa} x(t_0) = \frac{1}{h^n} \int_{-1}^{1} \rho_{n,\mu,\kappa}(\tau)\, x(t_0 + h\tau) d\tau,, \tag{A.22}$$

where

$$\rho_{n,\mu,\kappa}(\tau) = \frac{2^{-(n+\mu+\kappa+1)} n!}{B(n+\mu+1, n+\kappa+1)} \overline{P}_n^{(\mu,\kappa)}(\tau)\, \overline{w}_{\mu,\kappa}(\tau)$$

,

$$B(n+\mu+1, n+\kappa+1) = \frac{\Gamma(n+\mu+1)\Gamma(n+\kappa+1)}{\Gamma(2n+\mu+\kappa+2)}$$

.

Moreover, we have $\overline{D}^{(n)}_{h,\mu,\kappa} x(t_0) = x^{(n)}(t_0) + O(h)$.

Remark 1. *In order to compute $\rho_{n,\mu,\kappa}$, we should calculate $\overline{P}_n^{(\mu,\kappa)}$ whose computational complexity is $O(n^2)$. Hence, the computational effort of $\rho_{n,\mu,\kappa}$ is $O(n^2)$.*

Proof. By taking the Taylor expansion of x, we obtain for any $t_0 \in I_h$ that there exists $\theta \in]t_0 - h, t_0 + h[$ such that

$$x(t_0 + h\tau) = x(t_0) + h\tau x'(t_0) + \cdots + \frac{h^n \tau^n}{n!} x^{(n)}(t_0) + \frac{h^{n+1}\tau^{n+1}}{(n+1)!} x^{(n+1)}(\theta). \tag{A.23}$$

Substituting (A.23) in (A.22), we deduce from the classical orthogonal proper-

ties of the Jacobi polynomials (see (Szegö, 1967)) that

$$\int_{-1}^{1} \rho_{n,\mu,\kappa}(\tau)\,\tau^m\,d\tau = 0,\ 0 \le m < n, \tag{A.24}$$

$$\int_{-1}^{1} \rho_{n,\mu,\kappa}(\tau)\,\tau^n\,d\tau = (n!). \tag{A.25}$$

Using (A.23), (A.24) and (A.25), we can conclude that

$$\overline{D}^{(n)}_{h,\mu,\kappa}x(t_0) = \frac{1}{h^n}\int_{-1}^{1} \rho_{n,\mu,\kappa}(\tau)\,x(t_0+h\tau)d\tau = x^{(n)}(t_0) + O(h).$$

Hence, this proof is completed. □

In fact, we have taken an n^{th} order truncation in the Taylor expansion of x in Proposition (1) where n is the order of the estimated derivative. Thus, we call these estimators **minimal estimators** (see (Mboup et al., 2007, 2009a)).

Recall the central Rodrigues formula with central Jacobi orthogonal polynomial (see (Szegö, 1967))

$$\overline{P}_n^{(\mu,\kappa)}(\tau)\overline{w}_{\mu,\kappa}(\tau) = \frac{(-1)^n}{2^n n!}\frac{d^n}{dt^n}[\overline{w}_{\mu+n,\kappa+n}(\tau)]. \tag{A.26}$$

Proposition 2. *Let $x \in C^n(I)$, then the minimal estimators of $x^{(t_0)}$ given in Proposition (1) can be also written as follows*

$$\forall t_0 \in I_h,\ \overline{D}^{(n)}_{h,\mu,\kappa}x(t_0) = \frac{\left\langle \overline{P}_0^{(\mu+n,\kappa+n)}(\tau), x^{(n)}(t_0+h\tau)\right\rangle_{\mu+n,\kappa+n}}{\|\overline{P}_0^{(\mu+n,\kappa+n)}\|^2_{\mu+n,\kappa+n}}\,\overline{P}_0^{(\mu+n,\kappa+n)}(0). \tag{A.27}$$

Moreover, we have

$$\forall t_0 \in I_h,\ \overline{D}^{(n)}_{h,\mu,\kappa}x(t_0) = \overline{D}^{(0)}_{h,\mu+n,\kappa+n}x^{(n)}(t_0). \tag{A.28}$$

Proof. By using the central Rodrigues formula in (A.22) and applying n times integrations by parts we get

$$\begin{aligned}
\overline{D}^{(n)}_{h,\mu,\kappa}x(t_0) &= \frac{1}{h^n}\frac{(-1)^n 2^{-(2n+\mu+\kappa+1)}}{B(n+\mu+1,n+\kappa+1)}\int_{-1}^{1}\frac{d^n}{d\tau^n}[\overline{w}_{\mu+n,\kappa+n}(\tau)]\,x(t_0+h\tau)\,d\tau \\
&= \frac{2^{-(2n+\mu+\kappa+1)}}{B(n+\mu+1,n+\kappa+1)}\int_{-1}^{1}\overline{w}_{\mu+n,\kappa+n}(t)\,x^{(n)}(t_0+h\tau)\,d\tau \\
&= \overline{D}^{(0)}_{h,\mu+n,\kappa+n}x^{(n)}(t_0).
\end{aligned} \tag{A.29}$$

Then, by using $\overline{P}_0^{(\mu+n,\kappa+n)}(t) \equiv 1$ and $\|\overline{P}_0^{(\mu+n,\kappa+n)}\|^2_{\mu+n,\kappa+n} = 2^{2n+\mu+\kappa+1}B(n+\mu+1,n+\kappa+1)$, we can valid the equation (A.27).

□

It is shown in Proposition (2) that the minimal estimators of $x^{(n)}(t_0)$ given in Proposition (1) are equal to the value of the 0 order truncated Jacobi orthogonal series expansion of $x^{(n)}(t_0+h\xi)$ at $\xi=0$. Let us assume that $x\in C^n(I)$, then we define now the q^{th} ($q\in\mathbb{N}$) order truncated Jacobi orthogonal series of $x^{(n)}(t_0+h\xi)$ by the following operator

$$\forall t_0\in I_h,\ \overline{D}^{(n)}_{h,\mu,\kappa,q}x(t_0+h\xi):=\sum_{i=0}^{q}\frac{\left\langle\overline{P}_i^{(\mu+n,\kappa+n)}(\cdot),x^{(n)}(t_0+h\cdot)\right\rangle_{\mu+n,\kappa+n}}{\|\overline{P}_i^{(\mu+n,\kappa+n)}\|^2_{\mu+n,\kappa+n}}\,\overline{P}_i^{(\mu+n,\kappa+n)}(\xi). \tag{A.30}$$

Take $\xi=0$ in (A.30), we obtain a family of estimators of $x^{(n)}(t_0)$ with

$$\forall t_0\in I_h,\ \overline{D}^{(n)}_{h,\mu,\kappa,q}x(t_0)=\sum_{i=0}^{q}\frac{\left\langle\overline{P}_i^{(\mu+n,\kappa+n)}(\cdot),x^{(n)}(t_0+h\cdot)\right\rangle_{\mu+n,\kappa+n}}{\|\overline{P}_i^{(\mu+n,\kappa+n)}\|^2_{\mu+n,\kappa+n}}\,\overline{P}_i^{(\mu+n,\kappa+n)}(0). \tag{A.31}$$

To better explain our method, let us recall some well known facts. We consider the subspace of $C^0([-1,1])$, defined by

$$\mathcal{H}_q=\operatorname{span}\left\{\overline{P}_0^{(\mu+n,\kappa+n)},\overline{P}_1^{(\mu+n,\kappa+n)},\cdots,\overline{P}_q^{(\mu+n,\kappa+n)}\right\}. \tag{A.32}$$

Equipped with the inner product $\langle\cdot,\cdot\rangle_{\mu+n,\kappa+n}$, $\mathcal{H}_q$ is clearly a reproducing kernel Hilbert space (Aronszajn, 1950), with the reproducing kernel

$$\mathcal{K}_q(\tau,\xi)=\sum_{i=0}^{q}\frac{\overline{P}_i^{(\mu+n,\kappa+n)}(\tau)\overline{P}_i^{(\mu+n,\kappa+n)}(\xi)}{\|\overline{P}_i^{(\mu+n,\kappa+n)}\|^2_{\mu+n,\kappa+n}}. \tag{A.33}$$

The reproducing property implies that for any function $x^{(n)}(t_0+h\cdot)$ belonging to $C^0([-1,1])$, we have

$$\left\langle\mathcal{K}_q(\cdot,\xi),x^{(n)}(t_0+h\cdot)\right\rangle_{\mu+n,\kappa+n}=\overline{D}^{(n)}_{h,\mu,\kappa,q}x(t_0+h\xi), \tag{A.34}$$

where $\overline{D}^{(n)}_{h,\mu,\kappa,q}x(t_0+h\cdot)$ stands for the orthogonal projection of $x^{(n)}(t_0+h\cdot)$ on $\mathcal{H}_q$. Thus, the differentiators given in (A.31) can be obtained by taking $\xi=0$.

Then the affine differentiators of $x^{(n)}$, which is the central Jacobi differentiators, can be rewritten as (Liu et al., 2011c):

$$\overline{D}^{(n)}_{\kappa,\mu,h,q}x(t_0)=\frac{1}{h^n}\int_{-1}^{1}O_{\kappa,\mu,n,q}(\tau)\,x(t_0+h\tau)d\tau, \tag{A.35}$$

where $\rho_{n,\kappa,\mu}(\tau) = \frac{2^{-(n+\kappa+\mu+1)} n!\Gamma(2n+\kappa+\mu+2)}{\Gamma(n+\kappa+1)\Gamma(n+\mu+1)}$,

$$O_{\kappa,\mu,n,q}(\tau) = \sum_{i=0}^{q} \overline{P}_i^{\mu+n,\kappa+n}(0) \sum_{j=0}^{i} (-1)^{i+j} \binom{i}{j} \frac{2i+\kappa+\mu+2n+1}{i+\kappa+\mu+2n+1} \rho_{n,\kappa+i-j,\mu+j}(\tau). \tag{A.36}$$

A.4 Gains selection for third order error equation

For a high order linear differential equation with constant coefficient such as:

$$a_n y^{(n)} + a_{n-1} y^{(n-1)} + \cdots + a_2 y^{(2)} + a_1 y^{(1)} + a_0 y = 0, \tag{A.37}$$

it has a general solution in the form

$$y = C_1 y_1 + C_2 y_2 + \cdots + C_{n-1} y_{n-1} + C_n y_n, \tag{A.38}$$

where $y_1,\ y_2, \ldots, y_{n-1}, y_n$, are the n linearly independent solutions of the equation.

A general solution of this equation can be found by solving the differential equation's characteristic equation:

$$a_n r^n + a_{n-1} r^{n-1} + \cdots + a_2 r^2 + a_1 r + a_0 = 0. \tag{A.39}$$

This is a polynomial equation of degree n, therefore it has n real or complex roots. And the solutions can be given in the following law:
• 1: if r is a distinct real root, then $y = \exp_{rt}$ is a solution;
• 2: if $r = \lambda \pm \mu i$ are distinct complex conjugate roots, then, $y = \exp_{\lambda t} \cos \mu t$ and $y = \exp_{\lambda t} \sin \mu t$ are solutions;
• 3: if r is a real root appearing k times, then $y = te^{rt}$, $y = t^2 e^{rt}$, $y = t^{k-1} e^{rt}, \ldots$, and $y = e^{rt}$ are all solutions;
• 4: if $r = \lambda \pm \mu i$ are complex conjugate roots each appears k times, then

$$y = \exp_{\lambda t} \cos \mu t, \ and \ y = \exp_{\lambda t} \sin \mu t,$$

$$y = t \exp_{\lambda t} \cos \mu t, \ and \ y = t \exp_{\lambda t} \sin \mu t,$$

$$y = t^2 \exp_{\lambda t} \cos \mu t, \ and \ y^2 = t \exp_{\lambda t} \sin \mu t,$$

$$\cdots$$

$$y = t^{k-1} \exp_{\lambda t} \cos \mu t, \ and \ y = t^{k-1} \exp_{\lambda t} \sin \mu t$$

are all solutions.

Recall that the closed-loop system response of each joint j is determined by the following third order linear differential error equation (3.21):

$$\mathbf{e}_j^{(3)} + K_{dj} \ddot{\mathbf{e}}_j + K_{pj} \dot{\mathbf{e}}_j + K_{ij} \mathbf{e}_j = 0.$$

The aim is to ensure the error $\mathbf{e}$ converges exponentially to 0, which requires

that the roots of characteristic equation have negative real part. First look at the characteristic equation:

$$r^3 + K_{dj}r^2 + K_{pj}r + K_{ij} = 0. \tag{A.40}$$

We can rewrite the characteristic equation into the product of a second order system and a first order system:

$$r^2 + \omega_j \xi_j r + \omega_j^2 = 0; \tag{A.41}$$

$$r + T_j = 0. \tag{A.42}$$

Where, the poles of the second order system is determined by the damping coefficient ξ_j and the natural frequency ω_j; the pole of the first order system is determined by the coefficient T_j. All the poles should be on the left side of the s-plane in order to keep the error converge. After combining the characteristic equations of (A.41), we can obtain the gains:

$$K_{pj} = 2\xi_j \omega_j T_j + \omega_j^2; \tag{A.43}$$

$$K_{dj} = T_j + 2\xi_j \omega_j; \tag{A.44}$$

$$K_{ij} = T_j \omega_j^2. \tag{A.45}$$

We can tune the settling time by the choice of ω_j, ξ_j and T_j following: for the second order system, the settling time t_{sj} is around $\frac{3}{\xi_j \omega_j}$; for the first order system, the settling time t_{sj} is around $\frac{3}{T_j}$ (Ogata et al., 1970).

In (Gautier et al., 2013), the authors present the way to get the desired natural frequency ω_{di} bandwidth of the controller for this prototype robot: ω_{di} is chosen according to the driving capacity without saturation of the joint drive. In the field of motion control, it is known that the bandwidth of the velocity and position closed-loop are limited by the electro-mechanical cutoff frequency ω_{EM} of the open-loop transfer function between the velocity and the voltage control of the electrical motor, including the case of current controlled motor

$$\omega_{EM} = K_{\tau j}^2 / R_{Aj} \cdot J_j, \ for \ j = 1, 2 \tag{A.46}$$

where $K_{\tau j}$ is the electromagnetic motor torque constant and R_{Aj} is the motor armature resistance. Based on this, the full bandwidth of the prototype scara robot are given in (Gautier et al., 2013) with $\omega_1 = 1 \ rad/s$ and $\omega_2 = 10 \ rad/s$. Then, we select the damping coefficient $\xi_1 = \xi_2 = 0.8$, and the pole for the first order system $T_1 = T_2 = 30 \ s^{-1}$. Finally we get the gains $K_{p1} = 49$, $K_{p2} = 580$, $K_{d1} = 31$, $K_{d2} = 46$, $K_{i1} = 30$, $K_{i2} = 3000$.

Titre en français :
Identification et commande en ligne des robots avec utilisation de différentiateurs algébriques

Résumé en français :
Cette thèse traite de l'identification des paramètres dynamiques des robots, en s'appuyant sur les méthodes d'identification en robotique, qui utilisent le modèle dynamique inverse, ou le modèle de puissance, ou le modèle d'énergie du robot. Ce travail revisite le modèle d'énergie en exploitant le caractère intégral des fonctions modulatrices appliquées au modèle de puissance du robot. En outre, les procédures d'intégration sont analysées dans le domaine fréquentiel, et certains groupes de fonctions modulatrices sont sélectionnés afin d'offrir un bon comportement de filtrage. Ensuite, un différentiateur algèbrique est proposé, nommé différentiateurs de Jacobi. L'analyse est effectuée dans le domaine temporel, et dans le domaine fréquenciel, ce qui met en évidence la propriété de filtrage passe bande. Puis, ces différentiateurs sont appliqués avec succès à l'identification de robot, ce qui prouve leur bonne performance. Les comparaisons entre les différents modèles d'identification, les différenciateurs, les techniques des moindres carrés sont présentées et des conclusions sont tirées dans le domaine de l'identification de robot.

Mots-clefs : identification de robot, fonctions modulatrices, différentiateur de Jacobi, analyse fréquentielle, commande de robot

Titre en anglais :
Online identification and control of robots using algebraic differentiators

Résumé en anglais :
This thesis discusses the identification issues of the robot dynamic parameters. Starting with the well-known inverse dynamic identification model, power and energy identification models for robots, it extends the identification model from an energy point of view, by integrating modulating functions with robot power model. This new identification model avoids the computation of acceleration data. As well, the integration procedures are analyzed in frequency domain so that certain groups of modulating functions are selected in order to offer a good low-pass filtering property. Then, a recently developed high order algebraic differentiator is proposed and studied, named Jacobi differentiators. The analysis are done in both the time domain and in the frequency domain, which gives a clear clue about the differentiator filtering property and about how to select the differentiator parameters. Comparisons among different identification models, differentiators, least square techniques are presented and conclusions are drawn in the robot identification issues.

Mots-clefs : robot identification, modulating functions, Jacobi differentiators, frequency analysis, control of robot