



Identification dynamique des robots à flexibilités articulaires

A. Jubien

► **To cite this version:**

A. Jubien. Identification dynamique des robots à flexibilités articulaires. Physique de l'espace [physics.space-ph]. UNIVERSITÉ DE NANTES, 2014. Français. <tel-01171730>

HAL Id: tel-01171730

<https://hal.archives-ouvertes.fr/tel-01171730>

Submitted on 6 Jul 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

UNIVERSITÉ DE NANTES
FACULTÉ DES SCIENCES ET DES TECHNIQUES

ÉCOLE DOCTORALE STIM, Sciences et Technologies de
l'Information et Mathématiques (503)

Année 2014

Identification dynamique des robots à flexibilités articulaires

THÈSE DE DOCTORAT

Discipline : automatique, productique et robotique (921)
Spécialité : Automatique et Informatique Appliquée

*Présentée
et soutenue publiquement par*

Anthony Jubien

Le 14 novembre 2014 à l'IRCCyN, devant le jury ci-dessous

Président :	M. Hugues GARNIER	Professeur des universités	Université de Lorraine et CRAN, Nancy
Rapporteurs :	M. Gabriel ABBA	Professeur des universités	ENI, Metz
	M. Francisco CARRILLO	Professeur des universités	ENI, Tarbes
Examineurs :	M. Maxime GAUTIER	Professeur des universités	Université de Nantes et IRCCYN, Nantes
	M. Alexandre JANOT	Ingénieur de recherche	ONERA, Toulouse
Directeur de thèse :	M. Maxime GAUTIER		
Encadrant :	M. Alexandre JANOT		

A la mémoire de Gaël ...

Remerciements

Ce travail de thèse est né de la collaboration entre l'*ONERA* de Toulouse et l'*IRCCyN* de Nantes. Je remercie Patrick Fabiani, directeur du Département Commandes des systèmes et Dynamique du vol à l'*ONERA* à mon début de doctorat et Philippe Wenger, responsable de l'équipe Robotique de l'*IRCCyN* de m'avoir accueilli au sein de leurs laboratoires respectifs.

Je remercie sincèrement Maxime Gautier, professeur à l'université de Nantes et chercheur à l'*IRCCyN*, et Alexandre Janot, ingénieur de recherche à l'*ONERA*, pour m'avoir proposé cette thèse. Merci pour leurs confiances, leurs conseils et leurs encouragements durant mes travaux de recherche.

Mes sincères remerciements à M. Gabriel Abba et M. Francisco Carrillo pour m'avoir fait l'honneur de rapporter mes travaux.

Je remercie M. Hugues Garnier d'avoir accepté d'être président du jury de thèse.

Je remercie tous les membres du jury pour le temps passé à lire le présent mémoire et à évaluer les travaux effectués durant mon doctorat.

Mes vifs remerciements à Sébastien Briot pour son aide, ses remarques pertinentes et nos échanges.

Je remercie Philippe Lemoine, Stéphane Jolivet, Fabien Truchet, Arnaud Hamon et Denis Creusot d'avoir toujours répondu à mes demandes techniques et de m'avoir apporté leur aide afin que mes travaux puissent se dérouler dans les meilleures conditions possibles.

Il me tient à cœur de remercier tous les collègues de l'*IRCCyN* et des autres laboratoires situé sur le site de l'Ecole Centrale de Nantes ainsi qu'aux membres de l'*AED* pour nos échanges et pour l'ambiance de travail: Coralie, Vincent, Pierre, Paul-Emile, David, Raphaël, Olivier, Maité, les Marie(s), Jad, Francesc, Maxime, Julien, que ceux que j'oublie me pardonne.

Merci à Mathieu, Louay et Latifa d'avoir partagé mon bureau durant tout ce temps.

Enfin je remercie infiniment ma famille et mes ami(e)s pour m'avoir supporté et encouragé au cours de ces années d'étude.

Table des matières

Table des figures.....	V
Liste des tableaux	IX
Introduction générale.....	1
Chapitre 1 Modélisation des robots	5
1.1 Modélisation des robots rigides.....	7
1.1.1 Définitions.....	7
1.1.2 Modèle géométrique direct des robots à chaîne ouverte simple	8
1.1.3 Modèle dynamique inverse	10
1.1.4 Modèle dynamique direct.....	11
1.1.5 Modèle de frottements	12
1.1.6 Efforts moteurs linéaires par rapport aux paramètres dynamiques	13
1.1.7 Paramètres de base	13
1.2 Modélisation des robots à flexibilités articulaires	14
1.2.1 Modélisation selon la description de Denavit-Hartenberg Modifiée	14
1.2.2 Modèle dynamique inverse	15
1.2.3 Modèle dynamique direct.....	19
1.3 Bras <i>EMPS</i>	20
1.3.1 Modèle dynamique du robot rigide.....	21
1.3.2 Modèle dynamique du robot flexible	23
1.4 Robot Stäubli <i>TX40</i>	25
1.5 Kuka LightWeight Robot (<i>LWR</i>).....	28
1.5.1 Modèle dynamique du robot rigide.....	31
1.5.2 Modèle dynamique du robot flexible	33
1.6 Conclusion.....	36
Chapitre 2 Identification par modèle inverse et moindres carrés (<i>IDIM-LS</i>)	37
2.1 Introduction	39
2.2 Identification par modèle inverse et moindres carrés linéaires (<i>IDIM-LS</i>)	41
2.3 Estimation des écart-types.....	45
2.4 Moindres carrés pondérés.....	46
2.5 Filtrage parallèle de Y et des colonnes de W	47
2.6 Calcul des efforts moteurs	48
2.7 Mouvements excitants.....	49
2.8 Suppression des paramètres mal identifiés et réduction de modèle	49
2.9 Validation des résultats	50
2.10 Identification des paramètres inertiels d'une charge.....	50
2.11 Conclusion.....	52
Chapitre 3 Méthode <i>DIDIM</i> appliquée aux robots rigides.....	53

3.1 Introduction.....	55
3.2 Identification par une méthode <i>CLOE</i>	55
3.3 Méthode <i>DIDIM</i>	59
3.3.1 Théorie.....	59
3.3.2 Initialisation de l'algorithme	63
3.3.3 Structure de l'algorithme.....	68
3.4 Identification des paramètres inertiels d'une charge	68
3.5 Application au robot Stäubli <i>TX40</i>	69
3.6 Effet de la pondération et du filtre Decimate	80
3.7 Simplification du coefficient d'adaptation $k\alpha_j$	83
3.8 Comparaison entre la méthode <i>CLOE</i> à erreur de position et la méthode <i>DIDIM</i>	87
3.8.1 Comparaison analytique	87
3.8.2 Comparaison expérimentale	89
3.8.3 Discussion.....	96
3.9 Comparaison expérimentale entre la méthode <i>CLOE</i> à erreur d'effort et la méthode <i>DIDIM</i>	97
3.10 Conclusion	97
Chapitre 4 Méthode <i>DIDIM</i> appliquée aux robots à flexibilités articulaires	99
4.1 Introduction.....	101
4.2 Etude préliminaire sur le bras <i>EMPS</i>	101
4.2.1 Identification statique et dynamique avec des essais à sortie bloquée.....	101
4.2.2 Identification du modèle dynamique rigide du bras <i>EMPS</i> avec <i>IDIM-LS</i>	105
4.2.3 Identification du modèle dynamique flexible du bras <i>EMPS</i> avec <i>IDIM-LS</i>	108
4.2.4 Etude de la convergence de la méthode <i>DIDIM</i> en fonction des conditions initiales	110
4.3 Extension de la méthode <i>DIDIM</i> aux robots à flexibilités articulaires	113
4.3.1 Introduction	113
4.3.2 Etape 1 : identification du modèle dynamique rigide du robot.....	113
4.3.3 Etape 2 : identification des raideurs et des rapports d'inerties.....	113
4.3.4 Etape 3 : identification globale du modèle dynamique flexible	117
4.3.5 Structure de l'algorithme pour les robots à flexibilités articulaires	118
4.4 Application au bras <i>EMPS</i>	118
4.4.1 Identification du modèle dynamique flexible avec la méthode <i>DIDIM</i>	118
4.4.2 Comparaison avec la méthode <i>CLOE</i> usuelle	123
4.5 Application au robot Kuka <i>LWR</i>	125
4.5.1 Introduction	125
4.5.2 Identification expérimentale du modèle dynamique rigide avec la méthode <i>IDIM-LS</i>	126
4.5.3 Identification expérimentale des raideurs articulaires avec les capteurs de couple	131
4.5.4 Identification du modèle dynamique flexible avec la méthode <i>DIDIM</i>	137

4.6 Conclusion.....	156
Conclusion générale et perspectives.....	159
Bibliographie de l'auteur.....	163
Bibliographie.....	165
Annexes.....	171
A. Energie linéaire par rapport aux paramètres inertiels.....	171
B. Formalisme de Newton-Euler pour le calcul du modèle dynamique.....	177
C. Méthodes formelle et numérique pour le calcul des paramètres de base.....	181
D. Table de Fisher-Snedecor.....	189
E. Test de normalité de Kolmogorov-Smirnov.....	191
F. Test de Durbin-Watson.....	193
G. Paramètres du contrôleur simulé du Kuka <i>LWR</i>	195

Table des figures

Figure 1-1 : Schéma d'un robot à chaîne ouverte simple	7
Figure 1-2 : Paramètres de <i>DHM</i> , passage entre 2 repères successifs R_{j-1} et R_j	9
Figure 1-3 : Modèle de frottements	12
Figure 1-4 : Schéma d'une articulation avec une flexibilité articulaire.....	15
Figure 1-5 : Bras <i>EMPS</i> et charge de 10(kg).....	20
Figure 1-6 : Accouplement flexible du bras <i>EMPS</i>	20
Figure 1-7 : Composants du bras <i>EMPS</i>	21
Figure 1-8 : Modélisation et description <i>DHM</i> du bras <i>EMPS</i> flexible.....	23
Figure 1-9 : Robot Stäubli <i>TX40</i>	26
Figure 1-10 : Description <i>DHM</i> du robot Stäubli <i>TX40</i>	26
Figure 1-11 : Robot Kuka <i>LWR</i>	30
Figure 1-12 : Assemblage mécanique entre les corps 2 et 3 du Kuka <i>LWR</i>	30
Figure 1-13 : Description <i>DHM</i> du Kuka <i>LWR</i> rigide.....	31
Figure 1-14 : Description <i>DHM</i> du Kuka <i>LWR</i> flexible.....	33
Figure 2-1 : Schéma du principe de la méthode <i>IDIM-LS</i>	41
Figure 2-2 : Filtre passe-bande dérivée première	43
Figure 3-1 : Méthode à erreur de sortie en boucle ouverte	56
Figure 3-2 : Méthode à erreur de sortie en boucle fermée (<i>CLOE</i>).....	56
Figure 3-3 : Méthode <i>DIDIM</i>	59
Figure 3-4 : Méthode <i>DIDIM</i> (Direct and Inverse Dynamic Identification Models)	63
Figure 3-5 : Commande en position articulaire, robot réel.....	65
Figure 3-6 : Commande en position articulaire, robot simulé	65
Figure 3-7 : Stäubli <i>TX40</i> , Consignes de positions articulaires	69
Figure 3-8 : Stäubli <i>TX40</i> , Consignes de vitesses articulaires.....	70
Figure 3-9 : Stäubli <i>TX40</i> , Consignes d'accélération articulaires	70
Figure 3-10 : Stäubli <i>TX40</i> , Couples mesurés	71
Figure 3-11 : Stäubli <i>TX40</i> , Zoom sur la consigne d'accélération articulaire de l'axe 1	71
Figure 3-12 : Stäubli <i>TX40</i> , Histogramme des erreurs (en bleu) et Gaussienne (en rouge) avec la méthode <i>IDIM-LS</i>	74
Figure 3-13 : Stäubli <i>TX40</i> , Validation directe avec la méthode <i>IDIM-LS</i> (sans charge), couples mesurés (rouge), couple reconstruits (bleu) et erreur (noir).....	75
Figure 3-14 : Stäubli <i>TX40</i> , simulateur complet avec Matlab/Simulink.....	75
Figure 3-15 : Stäubli <i>TX40</i> , contrôleur avec Matlab/Simulink (Stäubli <i>TX40</i> CS8C controller).....	76
Figure 3-16 : Stäubli <i>TX40</i> , schéma de la simulation du modèle dynamique direct avec Matlab/Simulink (<i>MDD TX40</i>)	76
Figure 3-17 : Stäubli <i>TX40</i> , Histogramme des erreurs (en bleu) et Gaussienne (en rouge) avec la méthode <i>DIDIM</i> à l'itération 3	79

Figure 3-18 : Stäubli <i>TX40</i> , Validation directe avec la méthode <i>DIDIM</i> (sans charge), couples mesurés (rouge), couple reconstruits (bleu) et erreur (noir)	80
Figure 3-19 : Stäubli <i>TX40</i> , Histogramme des erreurs (en bleu) et Gaussienne (en rouge) avec la méthode <i>IDIM-LS</i> , cas avec pondération et utilisation du filtre Decimate	81
Figure 3-20 : Stäubli <i>TX40</i> , Histogramme des erreurs (en bleu) et Gaussienne (en rouge) avec la méthode <i>IDIM-LS</i> , cas avec pondération sans utilisation du filtre Decimate	81
Figure 3-21 : Stäubli <i>TX40</i> , Histogramme des erreurs (en bleu) et Gaussienne (en rouge) avec la méthode <i>IDIM-LS</i> , cas avec utilisation du filtre Decimate sans pondération	82
Figure 3-22 : Stäubli <i>TX40</i> , Histogramme des erreurs (en bleu) et Gaussienne (en rouge) avec la méthode <i>IDIM-LS</i> , cas sans utilisation du filtre Decimate et sans pondération.....	82
Figure 3-23 : Commande en position articulaire avec $\alpha_j=1$	84
Figure 3-24 : Commande en position articulaire.....	87
Figure 3-25 : Stäubli <i>TX40</i> , Positions mesurées (rouge), positions simulées (bleu) et erreur x 200 (noir), algorithme de Gauss-Newton.....	92
Figure 3-26 : Stäubli <i>TX40</i> , Positions mesurées (rouge), positions simulées (bleu) et erreur x 200 (noir), algorithme du Simplex.....	96
Figure 4-1 : Bras <i>EMPS</i> , effort moteur, essai statique	102
Figure 4-2 : Bras <i>EMPS</i> , position moteur, essai statique	102
Figure 4-3 : Bras <i>EMPS</i> , essai statique à sortie bloquée, validation directe, Y (rouge), W_χ (bleu) et erreur (noir)	103
Figure 4-4 : Bras <i>EMPS</i> , effort moteur, essai dynamique.....	104
Figure 4-5 : Bras <i>EMPS</i> , position et vitesse moteur, essai dynamique	104
Figure 4-6 : Bras <i>EMPS</i> , essai à sortie bloquée en dynamique, validation directe, Y (rouge), W_χ (bleu) et erreur (noir)	105
Figure 4-7 : Bras <i>EMPS</i> , consigne de position moteur	106
Figure 4-8 : Bras <i>EMPS</i> , consigne de vitesse moteur	106
Figure 4-9 : Bras <i>EMPS</i> , effort moteur	107
Figure 4-10 : Bras <i>EMPS</i> , modèle rigide, validation directe, Y (rouge), W_χ (bleu) et erreur (noir)	108
Figure 4-11 : <i>EMPS</i> , Histogramme des erreurs (en bleu) et Gaussienne (en rouge) avec la méthode <i>IDIM-LS</i>	110
Figure 4-12: Bras <i>EMPS</i> , modèle flexible, validation directe, Y (rouge), W_χ (bleu) et erreur (noir), sans charge.....	110
Figure 4-13: Simulateur du bras <i>EMPS</i> flexible sous Matlab/Simulink	111
Figure 4-14 : Contrôleur du bras <i>EMPS</i> flexible.....	111
Figure 4-15 : Simulateur du bras <i>EMPS</i> flexible	112
Figure 4-16 : Bras <i>EMPS</i> , modèle rigide, validation directe avec la méthode <i>DIDIM</i> , validation directe, Y (rouge), W_χ (bleu) et erreur (noir)	119
Figure 4-17 : <i>EMPS</i> , Histogramme des erreurs (en bleu) et Gaussienne (en rouge) avec la méthode <i>DIDIM</i>	122
Figure 4-18 : Bras <i>EMPS</i> , modèle flexible, validation directe avec la méthode <i>DIDIM</i> (sans charge)	122

Figure 4-19 : Kuka <i>LWR</i> , Consignes de positions articulaires	126
Figure 4-20 : Kuka <i>LWR</i> , Consignes de vitesses articulaires	127
Figure 4-21 : Kuka <i>LWR</i> , Consignes d'accélération articulaires	127
Figure 4-22 : Kuka <i>LWR</i> , Zoom sur la consigne d'accélération articulaire de l'axe 1	128
Figure 4-23 : Kuka <i>LWR</i> , Efforts moteurs mesurés	128
Figure 4-24 : Kuka <i>LWR</i> , Histogramme des erreurs (en bleu) et Gaussienne (en rouge) avec la méthode <i>IDIM-LS</i>	129
Figure 4-25 : Kuka <i>LWR</i> , Validation directe avec la méthode <i>IDIM-LS</i> , Y (rouge), $W\chi$ (bleu) et erreur (noir)	130
Figure 4-26 : Kuka <i>LWR</i> , configuration pour l'identification de l'articulation $i_c=1$	135
Figure 4-27 : Kuka <i>LWR</i> , configuration pour l'identification de l'articulation $i_c=2$	136
Figure 4-28 : Kuka <i>LWR</i> , configuration pour l'identification de l'articulation $i_c=3$	136
Figure 4-29 : Simulateur du Kuka <i>LWR</i> sous Matlab/Simulink	140
Figure 4-30 : Contrôleur simulé du Kuka <i>LWR</i>	140
Figure 4-31 : Modèle dynamique du Kuka <i>LWR</i>	141
Figure 4-32 : Efforts moteurs simulés pour le Kuka <i>LWR</i> flexible	143
Figure 4-33 : Zoom sur les efforts moteurs simulés pour le Kuka <i>LWR</i> flexible	144
Figure 4-34 : Kuka <i>LWR</i> , Validation directe avec la méthode <i>DIDIM</i> , étape 1, Y (rouge), $W\chi$ (bleu) et erreur (noir), axes 1 à 4	146
Figure 4-35 : Kuka <i>LWR</i> , Validation directe avec la méthode <i>DIDIM</i> , étape 1, Y (rouge), $W\chi$ (bleu) et erreur (noir), axes 5 à 7	147
Figure 4-36 : Kuka <i>LWR</i> , Validation directe avec la méthode <i>DIDIM</i> , étape 3, Y (rouge), $W\chi$ (bleu) et erreur (noir), axes 1 à 4, avec zoom sur les oscillations.....	150
Figure 4-37 : Kuka <i>LWR</i> , Validation directe avec la méthode <i>DIDIM</i> , étape 3, Y (rouge), $W\chi$ (bleu) et erreur (noir), axes 5 à 7	151
Figure 4-38 : Kuka <i>LWR</i> , Efforts moteurs simulés avec bruit.....	151
Figure 4-39 : Kuka <i>LWR</i> , Validation directe avec la méthode <i>DIDIM</i> , étape 1, Y (rouge), $W\chi$ (bleu) et erreur (noir).....	153
Figure 4-40 : Kuka <i>LWR</i> , Validation directe avec la méthode <i>DIDIM</i> , étape 3, Y (rouge), $W\chi$ (bleu) et erreur (noir).....	156

Liste des tableaux

Tableau 1-1 : Paramètres géométriques (<i>DHM</i>) du bras <i>EMPS</i> rigide	21
Tableau 1-2 : Paramètres de base du bras <i>EMPS</i> rigide	22
Tableau 1-3 : Paramètres de frottements et d'offset du bras <i>EMPS</i> rigide.....	22
Tableau 1-4 : Regroupement des paramètres du bras <i>EMPS</i> rigide.....	22
Tableau 1-5 : Paramètres géométriques (<i>DHM</i>) du bras <i>EMPS</i> flexible	23
Tableau 1-6 : Paramètres de base du bras <i>EMPS</i> flexible	23
Tableau 1-7 : Regroupement des paramètres du bras <i>EMPS</i> flexible.....	23
Tableau 1-8 : Paramètres de frottements et d'offset du bras <i>EMPS</i> flexible.....	24
Tableau 1-9 : Paramètres géométriques (<i>DHM</i>) du robot Stäubli <i>TX40</i>	27
Tableau 1-10 : Paramètres de base du robot Stäubli <i>TX40</i>	28
Tableau 1-11 : Regroupement des paramètres du robot Stäubli <i>TX40</i>	28
Tableau 1-12 : Paramètres de frottements et d'offsets du robot Stäubli <i>TX40</i>	28
Tableau 1-13 : Paramètres géométriques (<i>DHM</i>) du Kuka <i>LWR</i> rigide	32
Tableau 1-14 : Paramètres de base du robot Kuka <i>LWR</i> rigide	32
Tableau 1-15 : Regroupement des paramètres du robot Kuka <i>LWR</i> rigide	32
Tableau 1-16 : Paramètres de frottements et d'offset du robot Kuka <i>LWR</i> rigide	33
Tableau 1-17 : Paramètres géométriques (<i>DHM</i>) du modèle flexible du Kuka <i>LWR</i>	34
Tableau 1-18 : Paramètres de base du robot Kuka <i>LWR</i> flexible	34
Tableau 1-19 : Regroupement des paramètres du robot Kuka <i>LWR</i> flexible.....	35
Tableau 1-20 : Paramètres de frottements et d'offset du robot Kuka <i>LWR</i> flexible.....	35
Tableau 1-21 : Equivalence entre les paramètres du robot Kuka <i>LWR</i> rigide et du robot Kuka <i>LWR</i> flexible.....	36
Tableau 3-1 : Stäubli <i>TX40</i> , paramètres essentiels identifiés avec la méthode <i>IDIM-LS</i>	73
Tableau 3-2: Stäubli <i>TX40</i> , valeurs de ${}^{ap}M_{i,j}$	76
Tableau 3-3 : Stäubli <i>TX40</i> , paramètres essentiels identifiés avec la méthode <i>DIDIM</i>	78
Tableau 3-4 : Norme relative du résidu <i>IDIM-LS</i> et <i>DIDIM</i>	80
Tableau 3-5 : Norme relative du résidu <i>DIDIM</i>	84
Tableau 3-6 : Stäubli <i>TX40</i> , paramètres essentiels identifiés avec la méthode <i>DIDIM</i> avec $\alpha=1$	86
Tableau 3-7 : Stäubli <i>TX40</i> , paramètres essentiels identifiés avec les méthodes, <i>IDIM-LS</i> , <i>DIDIM</i> et la méthode <i>CLOE</i> à erreur de position	91
Tableau 3-8 : Stäubli <i>TX40</i> , Gauss-Newton, erreurs relatives entre les positions mesurées et simulées et entre les couples mesurés et simulés	92
Tableau 3-9 : Stäubli <i>TX40</i> , <i>SQP</i> , erreurs relatives entre les positions mesurées et simulées et entre les couples mesurés et simulés	93
Tableau 3-10 : Stäubli <i>TX40</i> , paramètres de base identifiés avec la méthode <i>DIDIM</i> et la méthode <i>CLOE</i>	95

Tableau 4-1 : Bras <i>EMPS</i> , essai à sortie bloquée, paramètres identifiés.....	104
Tableau 4-2 : Bras <i>EMPS</i> , paramètres identifiés du modèle dynamique rigide avec la méthode <i>IDIM-LS</i>	108
Tableau 4-3 : Bras <i>EMPS</i> , paramètres identifiés du modèle dynamique flexible avec la méthode <i>IDIM-LS</i>	109
Tableau 4-4 : Bras <i>EMPS</i> , paramètres identifiés du modèle dynamique rigide avec la méthode <i>DIDIM</i>	119
Tableau 4-5 : Valeurs identifiées de λ_1 et k_1	120
Tableau 4-6 : Bras <i>EMPS</i> , paramètres identifiés du modèle dynamique flexible avec la méthode <i>DIDIM</i>	121
Tableau 4-7 : Valeurs identifiées avec la méthode <i>CLOE</i> à erreur de sortie sur la position moteur	124
Tableau 4-8 : Valeurs identifiées avec la méthode <i>CLOE</i> à erreur de sortie sur l'effort moteur	125
Tableau 4-9 : Kuka <i>LWR</i> , paramètres identifiés du modèle dynamique rigide avec la méthode <i>IDIM-LS</i>	129
Tableau 4-10 : Kuka <i>LWR</i> , paramètres et raideurs identifiées avec la méthode <i>IDIM-LS</i>	137
Tableau 4-11 : Kuka <i>LWR</i> , raideurs a priori	137
Tableau 4-12 : Kuka <i>LWR</i> , paramètres de base (différents de 0) du modèle dynamique flexible utilisés pour la simulation	142
Tableau 4-13 : Kuka <i>LWR</i> , conditions initiales sur les moments d'inerties	144
Tableau 4-14 : Kuka <i>LWR</i> , paramètres de base identifiés pour l'étape 1	145
Tableau 4-15 : Kuka <i>LWR</i> , raideurs, k et λ en fonction de k	148
Tableau 4-16 : Kuka <i>LWR</i> , paramètres de base différents de 0 identifiés pour l'étape 3	149
Tableau 4-17 : Kuka <i>LWR</i> , paramètres identifiés pour l'étape 1	152
Tableau 4-18 : Kuka <i>LWR</i> , raideurs, k et λ en fonction de k	154
Tableau 4-19 : Kuka <i>LWR</i> , paramètres identifiés pour l'étape 3	155

Ce travail de thèse s'inscrit dans une collaboration scientifique entre l'*ONERA* de Toulouse et l'*IRCCyN* de Nantes sur les problématiques d'identification de robots.

Des modèles dynamiques précis des robots sont nécessaires pour l'élaboration de simulateurs précis, pour l'étude et la mise en œuvre de nouvelles commandes et pour la conception intégrée de type mécatronique. De nouvelles applications en robotique comme l'usinage, le soudage par friction, le ponçage, la collaboration homme-robot, nécessitent la prise en compte des flexibilités articulaires dans le contrôle en effort et en position de l'effecteur. De plus, l'apparition de robots légers et flexibles comparés aux robots industriels rigides classiques, tel que le Kuka Light Weight Robot (*LWR*) (Bischoff et al., 2010) et les robots *UR5* et *UR10* de Universal Robot (Universal Robot, 2014) montre qu'il devient nécessaire de prendre en compte les flexibilités articulaires.

La modélisation et l'identification des robots rigides sont aujourd'hui des domaines bien explorés et maîtrisés (Hollerbach et al., 2008), ce qui est moins le cas pour les robots à flexibilités articulaires. La modélisation des robots à flexibilités articulaires peut être vue comme une extension de la modélisation des robots rigides (Khalil et Gautier, 2000), réalisant un compromis entre la simplicité de la description de Denavit Hartenberg Modifiée des robots rigides (Khalil et Kleinfinger, 1986) et la modélisation des robots souples par des flexibilités réparties. Dans cette thèse, on considère que les flexibilités sont localisées dans les transmissions (câble ou courroie de transmission, ou réducteurs de type "harmonic drive", par exemple).

L'identification des raideurs par des mesures statiques à l'aide de capteurs extéroceptifs onéreux tel que les lasers de poursuite (laser tracker) permettent d'identifier uniquement les raideurs statiques d'un robot à flexibilités articulaires (Alici et Shirinzadeh, 2005), (Dumas et al., 2011). Cependant, ces approches ne permettent pas de connaître les raideurs dynamiques nécessaires à l'étude des déformations élastiques et des oscillations en fonctionnement dynamique du robot.

Pour identifier le comportement dynamique, il existe deux approches principales: les techniques issues de l'analyse modale et les techniques issues des robots rigides. Les techniques d'identification issues de l'analyse modale fournissent des résultats intéressants pour des modèles linéaires sous forme d'équations d'état ou de fonctions de transfert. Cependant, pour les systèmes multi-corps non linéaires tels que les robots, les modèles identifiés sont des modèles linéarisés autour de configurations moyennes de fonctionnement et mal adaptés pour une utilisation dans tout l'espace de travail du robot.

Le principal avantage des techniques issues de l'identification des robots rigides est que les paramètres identifiés sont les paramètres physiques du robot. Les résultats sont donc faciles à interpréter. Cependant, ces techniques nécessitent d'avoir accès à toutes les variables d'état du robot, c'est à dire à la fois aux positions des moteurs, aux déformations élastiques et à leurs dérivées. Les déformations élastiques sont rarement accessibles sur les robots industriels.

Récemment, une nouvelle technique d'identification a été proposée: la méthode *DIDIM* (Direct and Inverse Dynamic Identification Model) (Gautier et al., 2012) (Gautier et al., 2011). *DIDIM* est une méthode d'identification hors ligne des paramètres dynamiques des robots qui utilise les modèles dynamiques inverse et direct. C'est une méthode à erreur de sortie en boucle fermée (*CLOE* : Closed-Loop Output Error) qui n'utilise pas les mesures des positions, vitesses et accélérations. Les paramètres optimaux minimisent le carré de la norme de l'erreur entre les efforts moteurs réels et les efforts simulés, avec la même loi de commande en boucle fermée et la même trajectoire de référence pour le robot réel et le robot simulé.

La méthode *DIDIM* a été validée sur plusieurs robots rigides, prototypes et robots industriels multi-axes (Gautier et al., 2012) (Gautier et al., 2011). Cette méthode est robuste vis-à-vis des bruits de mesures et sa convergence est très rapide, contrairement aux méthodes *CLOE* classiques à erreur de sortie sur la position (Janot et al., 2014b). La matrice d'observation étant construite avec des données simulées non bruitées, cette méthode fournit des estimées non biaisées en absence d'erreur de modèle.

La méthode *DIDIM* présente un intérêt majeur pour l'identification des robots à flexibilités articulaires pour lesquels la mesure des déformations élastiques n'est pas disponible en interne et nécessiterait une instrumentation de métrologie externe onéreuse.

Le présent mémoire est composé de quatre chapitres :

Le **chapitre 1** présente la modélisation des robots rigides à chaînes ouvertes simples selon la description de Denavit-Hartenberg Modifiée utilisée en robotique (Khalil et Kleinfinger, 1986). Les corps sont supposés rigides et reliés par des articulations pivots ou prismatiques. La description des modèles géométriques et le calcul des modèles dynamiques nécessaires pour l'identification paramétrique sont présentés. Le modèle rigide des robots est ensuite généralisé au modèle avec flexibilités localisées aux articulations. Les modèles des trois robots utilisés durant les travaux de thèse sont calculés. Le premier robot est le banc d'essai *EMPS* qui est un robot à un axe prismatique avec une flexibilité articulaire, le second est le robot rigide à 6 articulations rotoïdes, le Stäubli *TX40*, et le dernier est le robot flexible à 7 articulations rotoïdes, le Kuka Light-Weight Robot (*LWR*).

Le **chapitre 2** présente la méthode d'identification usuelle utilisée en robotique après une introduction sur les différentes techniques d'identification existantes pour les robots rigides et les problématiques liées à l'identification des robots à flexibilités articulaires. Habituellement, la linéarité du modèle dynamique inverse par rapport aux paramètres dynamiques est exploitée pour l'identification. Les paramètres dynamiques sont calculés par la solution d'un système linéaire surdéterminé au sens des moindres carrés linéaires. Cette méthode qui utilise le modèle dynamique inverse et les moindres carrés linéaires ordinaires ou pondérés est appelée dans la suite *IDIM-LS* (en anglais: Inverse Dynamic Identification Model with Least Squares) (Gautier, 1986) (Gautier, 1990) (Khalil et Dombre, 1999) (Gautier et Poignet, 2002) (Khalil et Dombre, 2002) (Hollerbach et al., 2008).

Le **chapitre 3** introduit la méthode à erreur de sortie en boucle fermée (*CLOE*, Closed-Loop Output Error) puis présente la méthode *DIDIM* et son application sur les robots rigides. Les paramètres optimaux minimisent le carré de la norme de l'erreur entre les efforts articulaires réels et les efforts simulés en boucle fermée. La mesure des positions articulaires n'est plus nécessaire contrairement à la méthode *IDIM-LS* et aux méthodes *CLOE* usuelle à erreur de sortie en position. Une étude comparative approfondie menée sur le robot Stäubli *TX40* avec 61 paramètres à identifier montre la supériorité de la méthode *DIDIM* sur une méthode *CLOE* usuelle à erreur de sortie en position : convergence 100 fois plus rapide et robustesse vis à vis des erreurs sur les conditions initiales largement supérieure.

Le **chapitre 4** présente l'extension de la méthode *DIDIM* aux robots à flexibilités articulaires. La méthode est divisée en trois étapes successives qui aboutissent à l'identification

complète d'un robot à flexibilités articulaires. La première étape permet d'identifier les paramètres du modèle dynamique rigide du robot en basses fréquences au moyen d'un filtrage passe-bas approprié. La deuxième étape utilise un algorithme de programmation non-linéaire, avec des conditions initiales calculées avec les paramètres identifiés à l'étape précédente, pour approcher les valeurs des inerties de part et d'autre des flexibilités et pour approcher les raideurs. La dernière étape permet d'identifier globalement avec précision tous les paramètres du robot avec la méthode *DIDIM* avec pour conditions initiales les valeurs approchées durant les deux premières étapes. Cette procédure est appliquée et validée expérimentalement sur le banc *EMPS*. Une comparaison avec la méthode *IDIM-LS* et une méthode *CLOE* à erreur de sortie en position puis en effort moteur est effectuée. Elle est ensuite appliquée et étudiée sur le robot Kuka *LWR*, uniquement en simulation par manque d'information précise sur la commande implantée dans le robot réel.

Chapitre 1 Modélisation des robots

Sommaire:

1.1.1 Définitions.....	7
1.1.2 Modèle géométrique direct des robots à chaîne ouverte simple	8
1.1.3 Modèle dynamique inverse	10
1.1.4 Modèle dynamique direct.....	11
1.1.5 Modèle de frottements	12
1.1.6 Efforts moteurs linéaires par rapport aux paramètres dynamiques	13
1.1.7 Paramètres de base	13
1.2 Modélisation des robots à flexibilités articulaires	14
1.2.1 Modélisation selon la description de Denavit-Hartenberg Modifiée	14
1.2.2 Modèle dynamique inverse	15
1.2.3 Modèle dynamique direct.....	19
1.3 Bras <i>EMPS</i>	20
1.3.1 Modèle dynamique du robot rigide.....	21
1.3.2 Modèle dynamique du robot flexible	23
1.4 Robot Stäubli <i>TX40</i>	25
1.5 Kuka LightWeight Robot (<i>LWR</i>).....	28
1.5.1 Modèle dynamique du robot rigide.....	31
1.5.2 Modèle dynamique du robot flexible	33
1.6 Conclusion.....	36

Résumé: Ce chapitre présente la modélisation des robots rigides à chaînes ouvertes simples. La description des modèles géométriques et le calcul des modèles dynamiques nécessaires pour l'identification paramétrique sont présentés. Le modèle rigide des robots est ensuite généralisé au modèle avec flexibilités articulaires. La modélisation de trois robots utilisés durant les travaux de thèse est donnée.

1.1 Modélisation des robots rigides

Cette partie présente la modélisation géométrique et dynamique des robots rigides à chaîne ouverte simple selon la description de Denavit-Hartenberg Modifiée (*DHM*) utilisée en robotique (Khalil et Kleinfinger, 1986). Les corps sont supposés rigides et reliés par des articulations pivots ou prismatiques. Ce chapitre emprunte principalement à l'ouvrage (Khalil et Dombre, 2002).

1.1.1 Définitions

On définit les termes utilisés dans cette partie.

Articulations (ou liaisons) : une articulation est la partie reliant deux corps consécutifs. Le nombre de degrés de liberté d'une articulation est compris entre 1 et 6, soit de 1 à 3 degrés de liberté en rotation et de 1 à 3 degrés de liberté en translation. Par convention toute articulation à plusieurs degrés de liberté est décomposée en plusieurs articulations à un seul degré de liberté, de type prismatique (un degré de liberté en translation) ou de type rotoïde appelé aussi pivot (un degré de liberté en rotation). Par exemple la liaison rotule est décomposée en trois liaisons pivots successives d'axes concourants.

Corps : Un corps relie deux articulations successives. Le corps d'indice j est noté C_j .

Base : La base d'un robot est le corps C_0 fixé au sol pour les robots à base fixe.

Effecteur ou organe terminal : L'organe terminal (aussi appelé l'effecteur) d'un robot est le corps le plus éloigné de sa base, en pratique il manipule des objets ou est en contact avec l'environnement extérieur. Ce corps est d'indice n .

Les corps sont numérotés de C_0 pour la base à C_n pour l'organe terminal pour un robot à chaîne ouverte simple (aussi appelé robot série) à n articulations (voir figure 1-1).

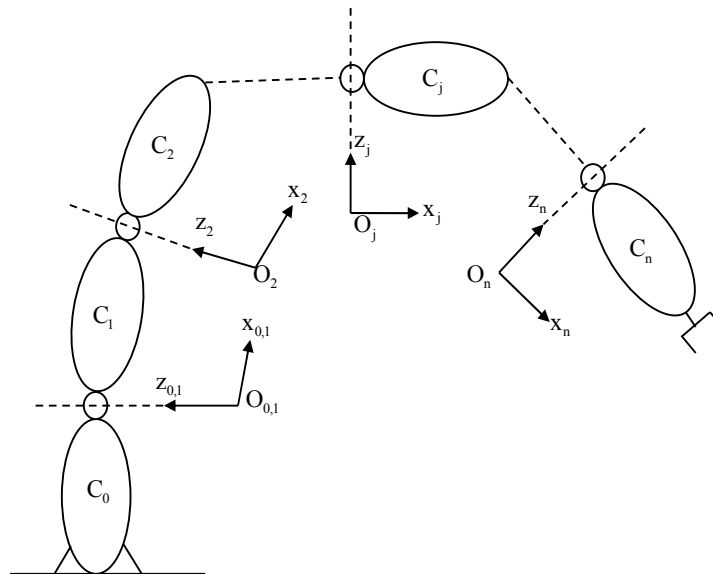


Figure 1-1 : Schéma d'un robot à chaîne ouverte simple

1.1.2 Modèle géométrique direct des robots à chaîne ouverte simple

La description de Denavit-Hartenberg Modifiée utilise un nombre minimum de paramètres géométriques et conduit à un modèle dynamique linéaire par rapport aux paramètres inertiels (Khalil et Kleinfinger, 1986). Elle facilite le calcul symbolique des modèles dynamiques direct et inverse d'un robot (Khalil et Creusot, 1997).

Le repère R_0 est attaché à la base du robot et à la terre pour les robots à base fixe. Un repère $R_j(0_j, x_j, y_j, z_j)$ est attaché à chaque corps C_j . Le repère R_n est attaché à l'organe terminal.

La description de Denavit-Hartenberg Modifiée est la suivante (voir figure 1-2):

- l'axe z_j est porté par l'axe de l'articulation j ,
- l'axe x_j est la perpendiculaire commune à z_j et z_{j+1} ,
- l'axe y_j est choisi de manière à créer un repère orthonormé direct avec $(0_j, x_j, y_j, z_j)$.

Le passage du repère R_{j-1} au repère R_j est défini avec les 4 paramètres géométriques minimaux suivants :

- α_j est l'angle correspondant à une rotation d'axe x_{j-1} entre z_{j-1} et z_j ,
- θ_j est l'angle correspondant à une rotation d'axe z_j entre x_{j-1} et x_j ,
- d_j est la distance portée par x_{j-1} entre z_{j-1} et z_j ,
- r_j est la distance portée par z_j entre x_{j-1} et x_j .

La variable de position articulaire généralisée q_j s'écrit :

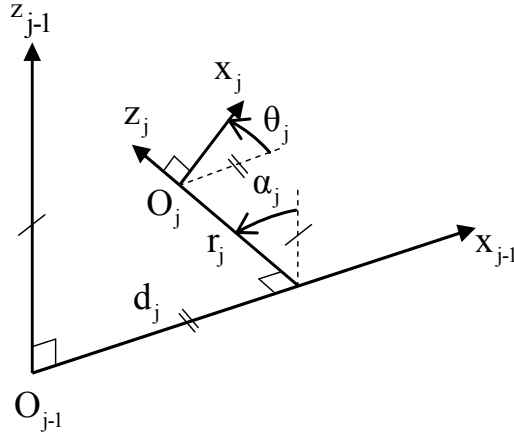
$$q_j = \bar{\sigma}_j \theta_j + \sigma_j r_j \quad (1.1)$$

avec :

- $\sigma_j=1$ si l'articulation d'indice j est prismatique,
- $\sigma_j=0$ si l'articulation d'indice j est un pivot,
- $\bar{\sigma}_j = 1 - \sigma_j$.

D'autres règles de construction s'ajoutent afin d'apporter des simplifications de modélisation :

- si q_1 est nul, on prend R_0 confondu avec R_1 (α_1 , d_1 et q_1 nuls),
- si q_n est nul, on prend x_n colinéaire à x_{n+1} pour rendre q_n nul,
- si l'articulation j est prismatique, on place z_j de manière à rendre d_j ou d_{j-1} nul,
- si z_{j+1} est colinéaire à z_j , on place x_j de manière à rendre r_j ou r_{j+1} nul.


 Figure 1-2 : Paramètres de *DHM*, passage entre 2 repères successifs R_{j-1} et R_j

La matrice homogène ${}^{j-1}\mathbf{T}_j$, appelée aussi matrice de changement de repère, permet de calculer les coordonnées d'un vecteur $\bar{\mathbf{V}}$ dans R_j à partir de ses coordonnées dans R_{j-1} :

$${}^{j-1}\mathbf{V} = {}^{j-1}\mathbf{T}_j {}^j\mathbf{V} \quad (1.2)$$

${}^{j-1}\mathbf{T}_j$ définit le repère R_j dans le repère R_{j-1} avec :

$$\begin{aligned} {}^{j-1}\mathbf{T}_j &= \text{Trans}(x_{j-1}, d_j) \text{Rot}(x_{j-1}, \alpha_j) \text{Trans}(z_j, r_j) \text{Rot}(z_j, \theta_j) \\ {}^{j-1}\mathbf{T}_j &= \begin{bmatrix} \cos(\theta_j) & -\sin(\theta_j) & 0 & d_j \\ \cos(\alpha_j)\sin(\theta_j) & \cos(\alpha_j)\cos(\theta_j) & -\sin(\alpha_j) & -r_j\sin(\alpha_j) \\ \sin(\alpha_j)\sin(\theta_j) & \sin(\alpha_j)\cos(\theta_j) & \cos(\alpha_j) & r_j\cos(\alpha_j) \\ 0 & 0 & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} {}^{j-1}\mathbf{s}_j & {}^{j-1}\mathbf{n}_j & {}^{j-1}\mathbf{a}_j & {}^{j-1}\mathbf{p}_j \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} {}^{j-1}\mathbf{A}_j & {}^{j-1}\mathbf{P}_j \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix} \end{aligned} \quad (1.3)$$

où :

- ${}^{j-1}\mathbf{V}$ est la matrice (3x1) des coordonnées de $\bar{\mathbf{V}}$ exprimées dans de le repère R_{j-1} ,
- ${}^j\mathbf{V}$ est la matrice (3x1) des coordonnées de $\bar{\mathbf{V}}$ exprimées dans de le repère R_j ,
- ${}^{j-1}\mathbf{s}_j$, ${}^{j-1}\mathbf{n}_j$, ${}^{j-1}\mathbf{a}_j$ sont les matrices (3x1) des vecteurs unitaires de la base orthonormée suivant les axes x_j , y_j et z_j du repère R_j , exprimés dans de le repère R_{j-1} ,
- ${}^{j-1}\mathbf{A}_j$ est la matrice orthogonale (3x3) de rotation définissant l'orientation du repère R_j dans le repère R_{j-1} . Les colonnes de ${}^{j-1}\mathbf{A}_j$ définissent les cosinus directeurs d'orientation,
- ${}^{j-1}\mathbf{P}_j$ est la matrice (3x1) des coordonnées du point O_j , dans le repère R_{j-1} .

1.1.3 Modèle dynamique inverse

Le Modèle Dynamique Inverse (*MDI*) d'un robot calcule les efforts moteurs en fonction des positions, vitesses et accélérations articulaires. Pour simplifier les notations, il est toujours admis d'utiliser le terme d'effort pour désigner un couple (Nm) ou une force (N).

$$\boldsymbol{\tau}_{\text{mdi}} = \mathbf{f}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) \quad (1.4)$$

où :

- $\boldsymbol{\tau}_{\text{mdi}}$ est le vecteur (nx1) des efforts moteurs exprimés en unité S.I. (Système International) côté corps; (Nm) pour les couples ou (N) pour les forces,
- $\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}$ sont les vecteurs (nx1) des positions, vitesses et accélérations articulaires. Pour les robots à articulations rigides, les variables articulaires côté corps sont égales aux variables côté moteur exprimées en unité S.I. côté corps.

Le modèle dynamique inverse d'un robot peut être calculé par le formalisme de Lagrange ou le formalisme de Newton-Euler (voir annexe A) (Khalil et Dombre, 2002).

Le formalisme de Lagrange décrit les équations de mouvement à l'aide de l'énergie cinétique et de l'énergie potentielle du système :

$$\boldsymbol{\tau}_{\text{mdi}} = \frac{d}{dt} \left(\frac{\partial \mathbf{L}}{\partial \dot{\mathbf{q}}} \right) - \frac{\partial \mathbf{L}}{\partial \mathbf{q}} + \boldsymbol{\tau}_f \quad \text{avec } \mathbf{L} = \mathbf{E} - \mathbf{U} \quad (1.5)$$

où :

- $\boldsymbol{\tau}_{\text{mdi}}$ est la matrice (nx1) des efforts moteurs,
- $\boldsymbol{\tau}_f$ est la matrice (nx1) des efforts de frottements,
- \mathbf{L} est le lagrangien du système,
- \mathbf{E} est l'énergie cinétique totale du système,
- \mathbf{U} est l'énergie potentielle totale du système.

L'énergie cinétique totale du système est une fonction quadratique des vitesses :

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

Avec $\mathbf{M}(\mathbf{q})$ la matrice (nxn) d'inertie du robot, appelée aussi matrice de l'énergie cinétique, qui est symétrique et définie positive. Ses éléments sont fonction des positions \mathbf{q} .

L'énergie potentielle totale du système $\mathbf{U}(\mathbf{q})$ est fonction des positions \mathbf{q} .

L'équation de Lagrange (1.5) s'écrit sous la forme explicite suivante :

$$\boldsymbol{\tau}_{\text{mdi}} = \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{H}(\mathbf{q}, \dot{\mathbf{q}}) \quad (1.7)$$

$\mathbf{H}(\mathbf{q}, \dot{\mathbf{q}})$ est le vecteur (nx1) qui regroupe les efforts centrifuges et de Coriolis, de gravité et de frottements :

$$\mathbf{H}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{Q}(\mathbf{q}) + \boldsymbol{\tau}_f \quad (1.8)$$

Où :

- $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}}$ est le vecteur (nx1) des efforts centrifuges et de Coriolis :

$$\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} = \frac{d\mathbf{M}(\mathbf{q})}{dt}\dot{\mathbf{q}} - \frac{\partial \mathbf{E}}{\partial \mathbf{q}} \quad (1.9)$$

- $\mathbf{Q}(\mathbf{q}) = \partial \mathbf{U} / \partial \mathbf{q}$ est le vecteur (nx1) des efforts de gravité.

Les coefficients de la matrice $\mathbf{M}(\mathbf{q})$ et des vecteurs $\mathbf{H}(\mathbf{q}, \dot{\mathbf{q}})$ et $\mathbf{Q}(\mathbf{q})$ dépendent des paramètres géométriques et dynamiques du robot. Le calcul de ces coefficients est détaillé en annexe A. Les équations dynamiques d'un robot forment un système de n équations différentielles du second ordre, couplées et non linéaires par rapport à \mathbf{q} et $\dot{\mathbf{q}}$.

1.1.4 Modèle dynamique direct

Le Modèle Dynamique Direct (*MDD*) exprime les accélérations articulaires en fonction des efforts moteurs, des vitesses et positions. Il s'écrit sous la forme d'une équation différentielle non linéaire implicite et peut être obtenu à partir du modèle dynamique inverse.

Le modèle dynamique direct s'écrit sous la forme d'une équation différentielle implicite à partir de (1.7):

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} = \boldsymbol{\tau}_{\text{mdi}} - \mathbf{H}(\mathbf{q}, \dot{\mathbf{q}}) \quad (1.10)$$

L'équation (1.10) peut s'écrire sous une forme d'état:

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{\mathbf{q}} \\ \ddot{\mathbf{q}} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{q} \\ \dot{\mathbf{q}} \end{bmatrix} + \begin{bmatrix} 0 \\ \mathbf{M}(\mathbf{q})^{-1} \end{bmatrix} \mathbf{u} - \begin{bmatrix} 0 \\ \mathbf{M}(\mathbf{q})^{-1} \mathbf{H}(\mathbf{q}, \dot{\mathbf{q}}) \end{bmatrix} \quad (1.11)$$

où :

- $\mathbf{x} = [\mathbf{q}^T \quad \dot{\mathbf{q}}^T]^T$ est le vecteur d'état,
- $\mathbf{u} = \boldsymbol{\tau}_{\text{mdi}}$ est le vecteur commande.

1.1.5 Modèle de frottements

Les valeurs des paramètres de frottements identifiées sur les robots ont montré que les efforts de frottement peuvent atteindre 10 à 20% de l'effort nominal d'actionnement, il est donc nécessaire de prendre ces derniers en compte dans les chaînes de transmission articulaires des robots (Hamon, 2011) (Khalil et al., 2007) (Gautier et al., 2012).

Un modèle simple utilisé en robotique comprend un terme de frottements sec Fc_j (frottement de Coulomb) et un terme de frottement visqueux $Fv_j \dot{q}_j$ (voir figure 1-3) (Gautier et al., 1995) (Gautier, 1997) (Khalil et Dombre, 2002). Ce modèle est suffisant pour les gammes de vitesses utilisées en poursuite de trajectoire par un robot. Cependant en cas de déplacement à très faibles vitesses, d'autres modèles plus complexes doivent être utilisés (Bogdan et Abba, 2009) (Bogdan et Abba, 2012).

A ce modèle est ajouté un terme d'offset Off_j qui prend en compte l'offset de l'amplificateur de puissance qui alimente en courant le moteur de l'articulation j et la dissymétrie du frottement de Coulomb:

$$\tau_{f_j} = Fc_j \text{sign}(\dot{q}_j) + Fv_j \dot{q}_j + \text{Off}_j \quad (1.12)$$

Avec $\text{sign}(\dot{q}_j) = 1$ si $\dot{q}_j > 0$ et $\text{sign}(\dot{q}_j) = -1$ si $\dot{q}_j < 0$.

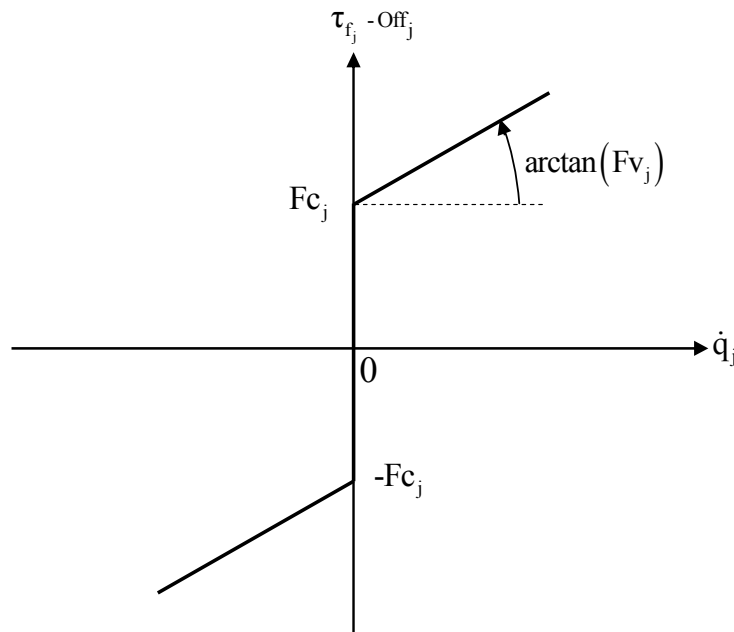


Figure 1-3 : Modèle de frottements

1.1.6 Efforts moteurs linéaires par rapport aux paramètres dynamiques

Le calcul de l'énergie mécanique du robot permet de montrer simplement la linéarité du modèle dynamique inverse (1.7) par rapport aux paramètres inertiels et de frottements (Gautier, 1990). Ces calculs sont rappelés en annexe A.

On définit un nouveau vecteur (14x1) de paramètres appelé vecteur de paramètres standards qui contient les paramètres inertiels standards, l'inertie de la chaîne d'actionnement, les frottements et l'offset de l'articulation j :

$$\boldsymbol{\chi}_{st}^j = \begin{bmatrix} XX_j & XY_j & XZ_j & YY_j & YZ_j & ZZ_j \\ MX_j & MY_j & MZ_j & M_j & Ia_j & Fv_j & Fc_j & Off_j \end{bmatrix}^T \quad (1.13)$$

Le robot comporte ($N_s=14 \times n$) paramètres standards au total :

$$\boldsymbol{\chi}_{st} = \begin{bmatrix} \boldsymbol{\chi}_{st}^{1T} & \boldsymbol{\chi}_{st}^{2T} & \cdots & \boldsymbol{\chi}_{st}^{nT} \end{bmatrix}^T \quad (1.14)$$

Le Lagrangien $L=E-U$ est linéaire par rapport à $\boldsymbol{\chi}_{st}$ car E et U sont linéaires par rapport à $\boldsymbol{\chi}_{st}$ (voir annexe A). Il en résulte que l'équation de Lagrange (1.7) du modèle dynamique inverse est linéaire par rapport $\boldsymbol{\chi}_{st}$:

$$\boldsymbol{\tau}_{mdi} = \mathbf{MDI}_{st}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) \boldsymbol{\chi}_{st} \quad (1.15)$$

où $\mathbf{MDI}_{st} = \partial \boldsymbol{\tau}_{mdi} / \partial \boldsymbol{\chi}_{st}$ est la matrice ($n \times N_s$) jacobienne de $\boldsymbol{\tau}_{mdi}$ par rapport à $\boldsymbol{\chi}_{st}$ fonction des positions, vitesses, accélérations et des paramètres géométriques du robot.

1.1.7 Paramètres de base

Les paramètres inertiels de base (ou paramètres minimaux) sont obtenus en éliminant les paramètres inertiels standards qui n'interviennent pas dans le modèle dynamique et en regroupant d'autres paramètres sous forme de relations linéaires. Ces éliminations et regroupements peuvent se faire avec une méthode formelle calculée à partir de l'énergie des corps du robot ou une méthode numérique basée sur la factorisation QR ou SVD (Gautier et Khalil, 1990) (Mayeda et al., 1990). Ces méthodes sont rappelés en annexe B.

Ces paramètres de base sont les seuls paramètres identifiables avec le modèle (1.15) et constituent un jeu minimal de paramètres nécessaires pour calculer le modèle dynamique inverse.

Le modèle dynamique inverse (1.15) se simplifie donc de la manière suivante :

$$\boldsymbol{\tau}_{\text{mdi}} = \mathbf{MDI}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\boldsymbol{\chi} \quad (1.16)$$

où :

- **MDI** la matrice (nxb) fonction des positions, vitesses, accélérations et des paramètres géométriques du robot,
- $\boldsymbol{\chi}$ le vecteur (bx1) des paramètres de base,
- b est le nombre de paramètres de base.

1.2 Modélisation des robots à flexibilités articulaires

1.2.1 Modélisation selon la description de Denavit-Hartenberg Modifiée

La modélisation des robots à flexibilités articulaires a été proposée comme une extension de la modélisation des robots rigides (Khalil et Gautier, 2000) (Pham, 2002). Pour les robots industriels dont les corps sont rigides, les déformations élastiques sont localisées dans les chaînes de transmission articulaires. Elles sont considérées comme des articulations rotoïdes ou prismatiques qui ajoutent des degrés de liberté flexibles aux degrés de liberté du robot rigide.

Une raideur de traction-compression introduit une liaison prismatique. Une raideur de torsion introduit une liaison rotoïde. Le calcul des modèles dynamiques direct et inverse est semblable à celui développé pour les robots rigides.

Cette modélisation permet de décrire un robot à flexibilités articulaires avec la description de *DHM* comme pour un robot rigide à chaîne ouverte simple. Pour chaque articulation i_{art} , on définit un degré de liberté $q_{m_{i_{\text{art}}}}$, côté moteur par rapport à la flexibilité, d'indice *DHM* impair $j_m=2i_{\text{art}}-1$ et un degré de liberté élastique $q_{e_{i_{\text{art}}}}$, d'indice *DHM* pair $j_e=2i_{\text{art}}$. Un robot à flexibilités articulaires avec n_{art} articulations possède $2n$ degrés de liberté. Cette modélisation peut être utilisée pour tous les robots où chaque articulation possède une flexibilité.

Pour l'écriture formelle explicite des modèles dynamiques, on utilise aussi les variables de positions côté corps (L=Link) :

$$q_{L_{i_{\text{art}}}} = q_{m_{i_{\text{art}}}} + q_{e_{i_{\text{art}}}} \quad (1.17)$$

où :

- $q_{m_{i_{\text{art}}}}$ est la position moteur de l'articulation i_{art} exprimée en unité S.I. côté corps,
- $q_{e_{i_{\text{art}}}}$ est la position du degré de liberté élastique (aussi appelé degré de liberté flexible) de l'articulation i_{art} exprimée en unité S.I. côté corps,
- $q_{L_{i_{\text{art}}}}$ est la position côté corps de l'articulation i_{art} exprimée en unité S.I. côté corps.

Ces notations sont illustrées figure 1-4 et sur les deux cas d'étude du bras *EMPS* (voir partie 1.3.2) et du Kuka LightWeight Robot (voir partie 1.5.2).

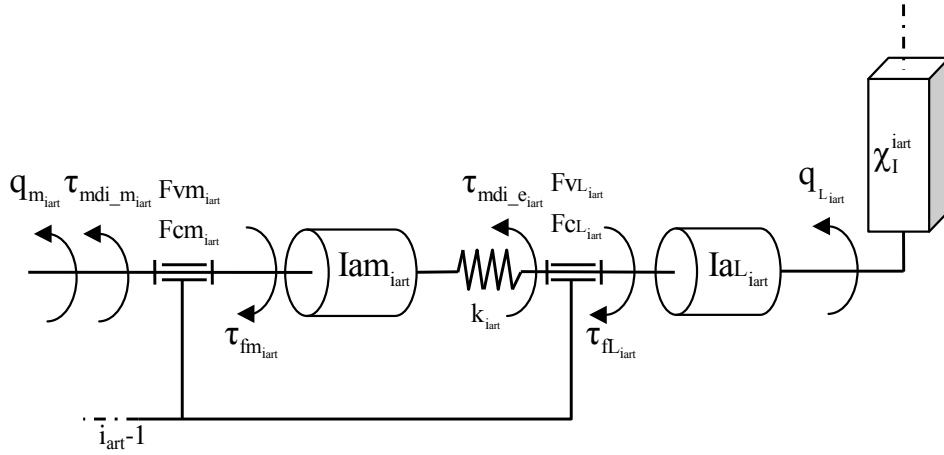


Figure 1-4 : Schéma d'une articulation avec une flexibilité articulaire

1.2.2 Modèle dynamique inverse

Le modèle dynamique inverse d'un robot à flexibilités articulaires s'exprime de la manière suivante (voir figure 1-4) :

$$\begin{aligned}
 \tau_{mdi_m} &= \mathbf{Iam} \ddot{\mathbf{q}}_m + \tau_{mdi_e} + \tau_{fm} \\
 \tau_{mdi_e} &= \mathbf{Ial} \ddot{\mathbf{q}}_L + \mathbf{M}(\mathbf{q}_L) \ddot{\mathbf{q}}_L + \mathbf{H}(\mathbf{q}_L, \dot{\mathbf{q}}_L) \\
 \tau_{mdi_e} &= -\mathbf{k} \mathbf{q}_e
 \end{aligned} \tag{1.18}$$

où :

- \mathbf{k} est la matrice ($n_{art} \times n_{art}$) diagonale des raideurs articulaires :

$$\mathbf{k} = \begin{bmatrix} k_1 & 0 & \dots & 0 \\ 0 & k_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & k_{nart} \end{bmatrix} \tag{1.19}$$

- \mathbf{Iam} est la matrice ($n_{art} \times n_{art}$) diagonale des moments d'inertie des éléments des chaînes d'actionnement côté moteur, par rapport aux flexibilités :

$$\mathbf{Iam} = \begin{bmatrix} Iam_1 & 0 & \dots & 0 \\ 0 & Iam_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & Iam_{nart} \end{bmatrix} \tag{1.20}$$

- \mathbf{IaL} est la matrice ($n_{art} \times n_{art}$) diagonale des moments d'inertie des éléments des chaînes d'actionnement, côté corps du robot ($L=Link$) :

$$\mathbf{IaL} = \begin{bmatrix} \mathbf{Ia}_{L_1} & 0 & \cdots & 0 \\ 0 & \mathbf{Ia}_{L_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{Ia}_{L_{nart}} \end{bmatrix} \quad (1.21)$$

- $\boldsymbol{\tau}_{mdi_m}$ est le vecteur ($n_{art} \times 1$) des efforts moteurs :

$$\boldsymbol{\tau}_{mdi_m} = \begin{bmatrix} \tau_{mdi_m_1} & \tau_{mdi_m_2} & \cdots & \tau_{mdi_m_{nart}} \end{bmatrix}^T \quad (1.22)$$

- $\boldsymbol{\tau}_{mdi_e}$ est le vecteur ($n_{art} \times 1$) des efforts articulaires élastiques :

$$\boldsymbol{\tau}_{mdi_e} = \begin{bmatrix} \tau_{mdi_e_1} & \tau_{mdi_e_2} & \cdots & \tau_{mdi_e_{nart}} \end{bmatrix}^T \quad (1.23)$$

Ces efforts ne sont pas indépendants comme les efforts moteurs $\boldsymbol{\tau}_{mdi_m}$. Ils dépendent de \mathbf{q}_e selon la troisième ligne de l'équation (1.18).

- $\mathbf{q}_m, \dot{\mathbf{q}}_m, \ddot{\mathbf{q}}_m$ sont respectivement les vecteurs ($n_{art} \times 1$) des positions, vitesses et accélérations moteur exprimées en unité S.I. côté corps avec :

$$\mathbf{q}_m = \begin{bmatrix} q_{m_1} & q_{m_2} & \cdots & q_{m_{nart}} \end{bmatrix}^T \quad (1.24)$$

- $\mathbf{q}_e, \dot{\mathbf{q}}_e, \ddot{\mathbf{q}}_e$ sont respectivement les vecteurs ($n_{art} \times 1$) des positions, vitesses et accélérations des degrés de liberté flexibles exprimées en unité S.I. côté corps avec :

$$\mathbf{q}_e = \begin{bmatrix} q_{e_1} & q_{e_2} & \cdots & q_{e_{nart}} \end{bmatrix}^T \quad (1.25)$$

- $\mathbf{q}_L, \dot{\mathbf{q}}_L, \ddot{\mathbf{q}}_L$ sont respectivement les vecteurs ($n_{art} \times 1$) des positions, vitesses et accélérations articulaires, côté corps :

$$\mathbf{q}_L = \mathbf{q}_m + \mathbf{q}_e, \quad \dot{\mathbf{q}}_L = \dot{\mathbf{q}}_m + \dot{\mathbf{q}}_e, \quad \ddot{\mathbf{q}}_L = \ddot{\mathbf{q}}_m + \ddot{\mathbf{q}}_e \quad (1.26)$$

- $\boldsymbol{\tau}_{fm}$ est le vecteur ($n_{art} \times 1$) des efforts de frottement côté moteur :

$$\boldsymbol{\tau}_{fm} = \begin{bmatrix} \tau_{fm_1} & \tau_{fm_2} & \cdots & \tau_{fm_{n_{art}}} \end{bmatrix}^T \quad (1.27)$$

- $\mathbf{M}(\mathbf{q}_L)$ la matrice ($n_{art} \times n_{art}$) d'inertie du robot, ses éléments sont fonction des positions \mathbf{q}_L .

$\mathbf{H}(\mathbf{q}_L, \dot{\mathbf{q}}_L)$ est le vecteur ($n_{art} \times 1$) qui regroupe les efforts centrifuges et de Coriolis, de gravité et de frottements côté corps:

$$\mathbf{H}(\mathbf{q}_L, \dot{\mathbf{q}}_L) = \mathbf{C}(\mathbf{q}_L, \dot{\mathbf{q}}_L) \dot{\mathbf{q}}_L + \mathbf{Q}(\mathbf{q}_L) + \boldsymbol{\tau}_{fL} \quad (1.28)$$

Où :

- $\mathbf{C}(\mathbf{q}_L, \dot{\mathbf{q}}_L) \dot{\mathbf{q}}_L$ est le vecteur ($n_{art} \times 1$) des efforts centrifuges et de Coriolis,
- $\mathbf{Q}(\mathbf{q}_L)$ est le vecteur ($n_{art} \times 1$) des efforts de gravité,
- $\boldsymbol{\tau}_{fL}$ est le vecteur ($n \times 1$) des efforts de frottement côté corps :

$$\boldsymbol{\tau}_{fL} = \begin{bmatrix} \tau_{fL_1} & \tau_{fL_2} & \cdots & \tau_{fL_{n_{art}}} \end{bmatrix}^T \quad (1.29)$$

A chaque partie de la chaîne d'actionnement côté moteur est associé le modèle de frottement suivant :

$$\tau_{fm_{iart}} = Fvm_{iart} \dot{q}_{m_{iart}} + Fcm_{iart} \text{sign}(\dot{q}_{m_{iart}}) + Offm_{iart} \quad (1.30)$$

où :

- Fvm_{iart} est le coefficient de frottement visqueux de la chaîne d'actionnement de l'articulation i_{art} ,
- Fcm_{iart} est le coefficient de frottement de Coulomb de la chaîne d'actionnement de l'articulation i_{art} ,
- $Offm_{iart}$ est l'offset de l'effort moteur qui contient l'offset de l'alimentation en courant des moteurs et la dissymétrie du frottement de coulomb de l'articulation i_{art} .

A chaque partie de la chaîne d'actionnement côté corps du robot est associé le modèle de frottement suivant :

$$\tau_{fL_{iart}} = FvL_{iart} \dot{q}_{L_{iart}} + FcL_{iart} \text{sign}(\dot{q}_{L_{iart}}) \quad (1.31)$$

où :

- FvL_{iart} est le coefficient de frottement visqueux côté corps de l'articulation i_{art} ,

- $F_{cL_{i_{art}}}$ est le coefficient de frottement de Coulomb côté corps de l'articulation i_{art} .

Tous les termes précédents sont exprimés en unité S.I. côté corps.

Le modèle (1.18) peut s'écrire sous forme matricielle suivante:

$$\begin{bmatrix} \tau_{mdi_m} \\ 0 \end{bmatrix} = \begin{bmatrix} \mathbf{Iam} & 0_{n \times n} \\ \mathbf{M}(\mathbf{q}_L) + \mathbf{IaL} & \mathbf{M}(\mathbf{q}_L) + \mathbf{IaL} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}}_m \\ \ddot{\mathbf{q}}_e \end{bmatrix} + \begin{bmatrix} \tau_{fm} - \mathbf{kq}_e \\ \mathbf{H}(\mathbf{q}_L, \dot{\mathbf{q}}_L) + \mathbf{kq}_e \end{bmatrix} \quad (1.32)$$

En pratique le modèle dynamique inverse est calculé avec le formalisme de Newton-Euler comme pour les robots rigides en utilisant le logiciel Symoro+ sous la forme suivante :

$$\begin{bmatrix} \tau_{mdi_m_1} \\ \tau_{mdi_e_1} \\ \tau_{mdi_m_2} \\ \tau_{mdi_e_2} \\ \vdots \\ \tau_{mdi_m_{nart}} \\ \tau_{mdi_e_{nart}} \end{bmatrix} = \mathbf{M}_e(\mathbf{q}_L) \begin{bmatrix} \ddot{q}_{m_1} \\ \ddot{q}_{e_1} \\ \ddot{q}_{m_2} \\ \ddot{q}_{e_2} \\ \vdots \\ \ddot{q}_{m_{nart}} \\ \ddot{q}_{e_{nart}} \end{bmatrix} + \mathbf{H}_e(\mathbf{q}_L, \dot{\mathbf{q}}_L, \dot{\mathbf{q}}_m) \quad (1.33)$$

Où :

- $\mathbf{M}_e(\mathbf{q}_L)$ est la matrice (nxn) d'inertie du robot. Ses éléments sont fonction des positions \mathbf{q}_L ,
- $\mathbf{H}_e(\mathbf{q}_L, \dot{\mathbf{q}}_L, \dot{\mathbf{q}}_m)$ est le vecteur (nx1) qui regroupe les efforts centrifuges et de Coriolis, de gravité, de frottements et de rappel dus aux raideurs.

Mais à la différence des robots rigides, les efforts articulaires élastiques τ_{mdi_e} ne sont pas des efforts indépendants. Ils dépendent de \mathbf{q}_e selon $\tau_{mdi_e} = -\mathbf{kq}_e$ (voir équation (1.18)).

En faisant passer les efforts articulaires élastiques $\tau_{mdi_e_{iart}}$ dans le membre de droite, le modèle dynamique inverse (1.33) d'un robot à flexibilités articulaires peut s'écrire sous une forme linéaire par rapport à un ensemble de paramètres dynamiques standards χ_{st} :

$$\boldsymbol{\tau}_{\text{mdi}} = \begin{bmatrix} \tau_{\text{mdi}_{m_1}} \\ 0 \\ \tau_{\text{mdi}_{m_2}} \\ 0 \\ \vdots \\ \tau_{\text{mdi}_{m_{\text{nart}}}} \\ 0 \end{bmatrix} = \mathbf{MDI}_{\text{st}} \boldsymbol{\chi}_{\text{st}} \quad (1.34)$$

Où:

- \mathbf{MDI}_{st} la matrice (nxNs) fonction des positions, vitesses, accélérations et des paramètres géométriques du robot,
- $\boldsymbol{\chi}_{\text{st}}$ est le vecteur des paramètres standards du robot.

Pour chaque articulation i_{art} , on définit un vecteur de 18 paramètres standards $\boldsymbol{\chi}_{\text{st}}^{i_{\text{art}}}$ qui comprend les paramètres dynamiques de la chaîne d'actionnement, les paramètres dynamiques du corps et la raideur de l'articulation i_{art} :

$$\boldsymbol{\chi}_{\text{st}}^{i_{\text{art}}} = \begin{bmatrix} \text{Iam}_{i_{\text{art}}} & \text{Fvm}_{i_{\text{art}}} & \text{Fcm}_{i_{\text{art}}} & \text{Offm}_{i_{\text{art}}} & k_{i_{\text{art}}} & \text{XX}_{i_{\text{art}}} & \text{XY}_{i_{\text{art}}} & \text{XZ}_{i_{\text{art}}} & \text{YY}_{i_{\text{art}}} \\ \text{YZ}_{i_{\text{art}}} & \text{ZZ}_{i_{\text{art}}} & \text{MX}_{i_{\text{art}}} & \text{MY}_{i_{\text{art}}} & \text{MZ}_{i_{\text{art}}} & \text{M}_{i_{\text{art}}} & \text{IaL}_{i_{\text{art}}} & \text{FvL}_{i_{\text{art}}} & \text{Fcl}_{i_{\text{art}}} \end{bmatrix}^T \quad (1.35)$$

Le vecteur des paramètres standards est composé de (18n) paramètres :

$$\boldsymbol{\chi}_{\text{st}} = \left[\boldsymbol{\chi}_{\text{st}}^1{}^T \quad \boldsymbol{\chi}_{\text{st}}^2{}^T \quad \cdots \quad \boldsymbol{\chi}_{\text{st}}^{\text{nart}}{}^T \right]^T \quad (1.36)$$

Le calcul des paramètres de base s'effectue de la même manière que dans le cas des robots rigides (voir partie 1.1.7 et annexe C).

1.2.3 Modèle dynamique direct

Le modèle dynamique direct d'un robot à flexibilités articulaires s'écrit sous la forme d'une équation différentielle implicite à partir du modèle dynamique inverse (1.33):

$$\mathbf{M}_e(\mathbf{q}_L) \begin{bmatrix} \ddot{\mathbf{q}}_{m_1} \\ \ddot{\mathbf{q}}_{e_1} \\ \ddot{\mathbf{q}}_{m_2} \\ \ddot{\mathbf{q}}_{e_2} \\ \vdots \\ \ddot{\mathbf{q}}_{m_{\text{nart}}} \\ \ddot{\mathbf{q}}_{e_{\text{nart}}} \end{bmatrix} = \begin{bmatrix} \tau_{\text{mdi}_{m_1}} \\ \tau_{\text{mdi}_{e_1}} \\ \tau_{\text{mdi}_{m_2}} \\ \tau_{\text{mdi}_{e_2}} \\ \vdots \\ \tau_{\text{mdi}_{m_{\text{nart}}}} \\ \tau_{\text{mdi}_{e_{\text{nart}}}} \end{bmatrix} - \mathbf{H}_e(\mathbf{q}_L, \dot{\mathbf{q}}_L, \dot{\mathbf{q}}_m) \quad (1.37)$$

1.3 Bras *EMPS*

Trois exemples de modélisation de robots sont présentés. Le premier robot est le bras *EMPS* du fabricant Dspace qui est un robot à un axe prismatique avec une flexibilité articulaire. Le second est le robot rigide à 6 axes *TX40* du fabricant Stäubli. Ce robot rigide est étudié dans le troisième chapitre. Le dernier est le robot flexible à 7 axes LightWeight Robot (*LWR*) du fabricant Kuka. Le bras *EMPS* et le Kuka *LWR* seront étudiés dans le chapitre 4.

Le bras *EMPS* (pour Electro-Mechanical Positioning System) est un bras de robot prismatique (voir figure 1-5).



Figure 1-5 : Bras *EMPS* et charge de 10(kg)

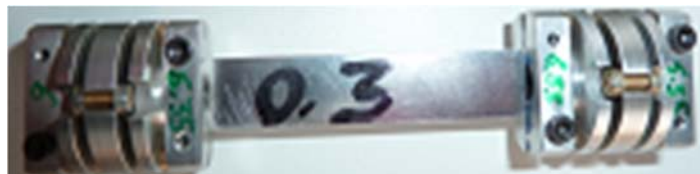


Figure 1-6 : Accouplement flexible du bras *EMPS*

Il est composé d'un moteur à courant continu Maxon qui entraîne un chariot par une vis à billes à faible frottement et à haute précision (voir figure 1-7).

Le chariot se déplace en translation. Une charge peut être fixée sur le chariot. La transmission entre le rotor du moteur et la vis à billes utilise un accouplement flexible à lame (voir figure 1-6). Un codeur incrémental côté moteur mesure la position du moteur et un codeur côté vis à billes mesure la position de la vis et du chariot.

Le moteur à courant continu est alimenté par un amplificateur de courant composé d'un hacheur quatre quadrants qui fournit une tension modulée en largeur d'impulsions à la fréquence de 17 (kHz) et d'une boucle de courant dont la bande passante à -45 degrés est de 630 (Hz), très supérieure à la dynamique mécanique du robot.

Le moteur est asservi en position avec une commande proportionnelle-dérivée (*PD*) de bande passante fixée à 30 (Hz).

Un contrôleur dSPACE permet de commander le robot et d'avoir accès aux différentes mesures en temps réel ou différé (via les logiciels ControlDesk de dSPACE (dSPACE, 2014) et Matlab de Mathworks (Mathworks, 2014)).

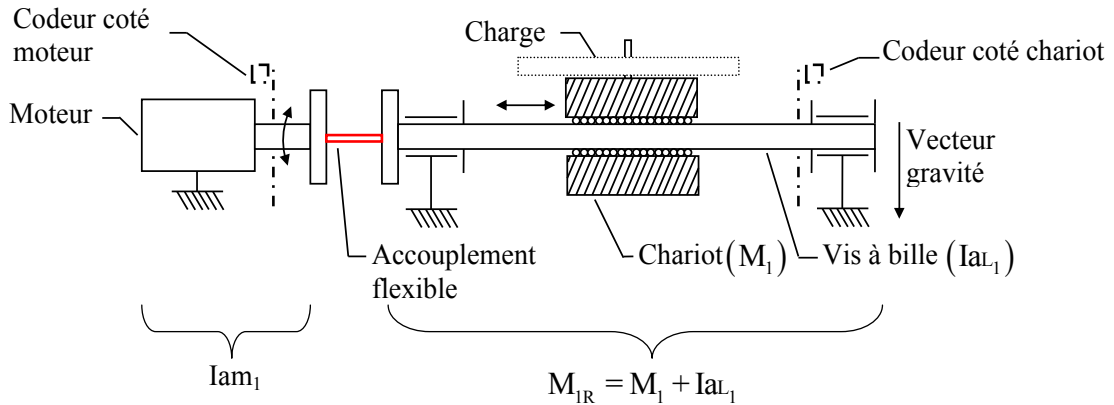


Figure 1-7 : Composants du bras *EMPS*

Toutes les positions, vitesses et accélérations et les paramètres sont exprimés en unités S.I. côté chariot (mouvement linéaire).

L'effort moteur (N) (unité S.I. côté chariot) est proportionnel au couple moteur (Nm) (unité S.I. côté moteur) selon le rapport de réduction de la vis à billes $N_{r1}=2513,3$ (rd/m) :

$$\tau(N) = N_{r1} \tau(N.m) \quad (1.38)$$

1.3.1 Modèle dynamique du robot rigide

Le robot est considéré comme une structure rigide dans une analyse de la dynamique aux basses fréquences inférieures au premier mode flexible.

Les paramètres géométriques définissant le bras *EMPS* rigide sont donnés dans le tableau 1-1.

Tableau 1-1 : Paramètres géométriques (*DHM*) du bras *EMPS* rigide

j	σ_j	α_j	d_j	θ_j	r_j
1	1	0	0	0	q_{m1}

Pour un corps mobile, il y a $N_s=14$ paramètres dynamiques. Après simplification du modèle, les paramètres de base sont au nombre de 4 (voir tableau 1-2 et tableau 1-3). 2 paramètres sont regroupés et 9 paramètres sont supprimés. Les regroupements des paramètres sont donnés dans le tableau 1-4.

Tableau 1-2 : Paramètres de base du bras *EMPS* rigide

j	XX	XY	XZ	YY	YZ	ZZ	MX	MY	MZ	M	Ia
1											M _{1R}

Tableau 1-3 : Paramètres de frottements et d'offset du bras *EMPS* rigide

j	Fv	Fc	Off
1	Fv ₁	Fc ₁	Off ₁

Tableau 1-4 : Regroupement des paramètres du bras *EMPS* rigide

M_{1R} = M₁ + Ia₁
--

Le modèle dynamique inverse du bras *EMPS* rigide est le suivant :

$$\tau_{\text{mdi}} = \tau_{\text{mdi}_{m_1}} = M_{1R} \ddot{q}_{m_1} + Fv_1 \dot{q}_{m_1} + Fc_1 \text{sign}(\dot{q}_{m_1}) + \text{Off}_1 \quad (1.39)$$

Où :

- M_{1R} (kg) est la masse totale des éléments côté moteur (inertie en rotation) et chariot (masse en translation),
- Fv₁ (N/m/s) est le frottement visqueux total,
- Fc₁ (N) est le frottement sec total,
- Off₁ (N) est l'offset.

L'effort moteur du bras rigide peut être exprimé linéairement par rapport aux paramètres dynamiques de la manière suivante :

$$\tau_{\text{mdi}} = \mathbf{MDI} \chi$$

avec : $\mathbf{MDI} = \begin{bmatrix} \ddot{q}_{m_1} & \dot{q}_{m_1} & \text{sign}(\dot{q}_{m_1}) & 1 \end{bmatrix}, \chi = [M_{1R} \quad Fv_1 \quad Fc_1 \quad \text{Off}_1]^T$ (1.40)

L'équation (1.39) peut s'écrire comme suit :

$$\tau_{\text{mdi1}} = M(q_{m_1}) \ddot{q}_{m_1} + H(q_{m_1}, \dot{q}_{m_1}) \quad (1.41)$$

Avec $M(q_{m_1}) = M_{1R}$ et $H(q_{m_1}, \dot{q}_{m_1}) = Fv_1 \dot{q}_{m_1} + Fc_1 \text{sign}(\dot{q}_{m_1}) + \text{Off}_1$.

1.3.2 Modèle dynamique du robot flexible

Le système possède 2 degrés de liberté, un degré de liberté côté moteur q_{m_1} et un degré de liberté flexible q_{e_1} avec: q_{m_1} (m), la position du moteur, q_{e_1} (m), la position du degré de liberté flexible où $q_{L_1} = q_{m_1} + q_{e_1}$ est la position absolue du chariot dans R_0 (voir figure 1-8).

Dans le cas du bras en position horizontale, le mouvement du chariot n'est pas affecté par la gravité et est modélisé, côté chariot, par deux masses, une raideur, et les forces de frottement. La flexibilité des corps n'est pas prise en compte car elle est négligeable par rapport à la souplesse de l'accouplement flexible. Les repères de la notation *DHM* sont représentés figure 1-8 et les paramètres géométriques définissant le bras *EMPS* flexible sont donnés dans le tableau 1-5.

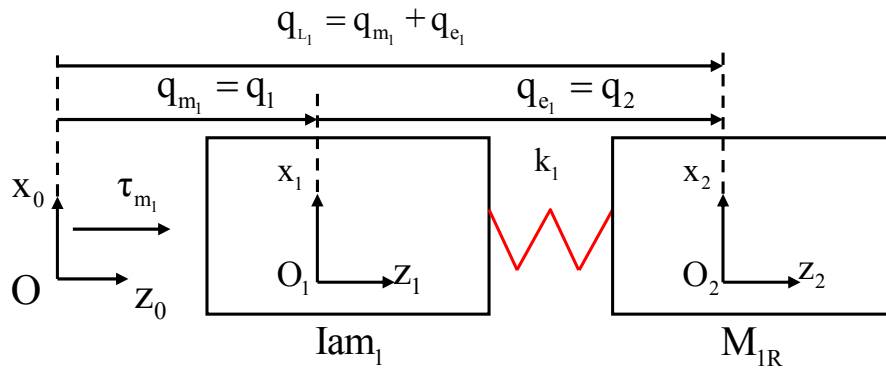


Figure 1-8 : Modélisation et description *DHM* du bras *EMPS* flexible

Tableau 1-5 : Paramètres géométriques (*DHM*) du bras *EMPS* flexible

i_{art}	j_m	j_e	σ_j	α_j	d_j	θ_j	r_j
1	1		1	0	0	0	q_{m_1}
		2	1	0	0	0	q_{e_1}

Le bras *EMPS* flexible possède $N_s=18$ paramètres standards. Les paramètres de base sont au nombre de 8 après simplification (voir tableau 1-6). 9 paramètres sont supprimés et 2 paramètres sont regroupés.

Tableau 1-6 : Paramètres de base du bras *EMPS* flexible

i_{art}	XX	XY	XZ	YY	YZ	ZZ	MX	MY	MZ	M	I_{am}	I_{al}	k
1										M_{1R}	I_{am_1}		k_1

Les regroupements des paramètres sont donnés dans le tableau 1-7.

Tableau 1-7 : Regroupement des paramètres du bras *EMPS* flexible

$M_{1R} = M_1 + I_{al_1}$

Tous les paramètres de frottements et d'offset définis dans le tableau 1-8 sont conservés.

Tableau 1-8 : Paramètres de frottements et d'offset du bras *EMPS* flexible

i_{art}	Fvm	Fcm	Offm	FvL	FcL
1	Fvm ₁	Fcm ₁	Offm ₁	FvL ₁	FcL ₁

Le modèle dynamique inverse du bras *EMPS* flexible permet de calculer l'effort moteur en fonction des positions et de leurs dérivées de la manière suivante :

$$\begin{aligned}
 \tau_{mdi_m_1} &= Iam_1 \ddot{q}_{m_1} + Fvm_1 \dot{q}_{m_1} + Fcm_1 \text{sign}(\dot{q}_{m_1}) + \tau_{mdi_e_1} + Offm_1 \\
 \tau_{mdi_e_1} &= M_{1R} \ddot{q}_{l_1} + FvL_1 \dot{q}_{l_1} + FcL_1 \text{sign}(\dot{q}_{l_1}) \\
 \tau_{mdi_e_1} &= -k_1 q_{e_1}
 \end{aligned} \tag{1.42}$$

Où :

- \dot{q}_{m_1} (m/s) est la vitesse du moteur,
- \ddot{q}_{m_1} (m/s/s) est l'accélération du moteur,
- $\tau_{mdi_m_1}$ (N) est l'effort moteur,
- \dot{q}_{l_1} (m/s) est la vitesse du chariot avec $\dot{q}_{l_1} = \dot{q}_{m_1} + \dot{q}_{e_1}$,
- \ddot{q}_{l_1} (m/s/s) est l'accélération du chariot avec $\ddot{q}_{l_1} = \ddot{q}_{m_1} + \ddot{q}_{e_1}$,
- Iam_1 (kg) est la masse équivalente des éléments côté moteur (rotor, accouplements et codeur),
- Fvm_1 (N/m/s) est le frottement visqueux côté moteur,
- Fcm_1 (N) est le frottement sec côté moteur,
- $Offm_1$ (N) est l'offset,
- M_{1R} (kg) est la masse équivalente des éléments côté chariot (accouplements, codeur, vis en rotation, écrou, chariot et charge en translation),
- FvL_1 (N/m/s) est le frottement visqueux côté chariot,
- FcL_1 (N) est le frottement sec côté chariot,
- k_1 (N/m) est la raideur de l'articulation flexible.

Les équivalences entre les paramètres du modèle à flexibilité articulaire (voir tableau 1-19) et les paramètres du modèle rigide (voir tableau 1-15) sont obtenues en posant $q_{e1}=0$ dans le modèle dynamique flexible (1.42) :

$$\begin{aligned}
 M_{1R} \text{ (rigide)} &= M_{1R} \text{ (flexible)} + Iam_1 \\
 Fv_1 &= Fvm_1 + FvL_1 \\
 Fc_1 &= Fcm_1 + FcL_1 \\
 Off_1 &= Offm_1
 \end{aligned} \tag{1.43}$$

Le modèle dynamique inverse peut s'écrire comme suit :

$$\begin{bmatrix} \tau_{\text{mdi}_{m_1}} \\ \tau_{\text{mdi}_{e_1}} \end{bmatrix} = \mathbf{M}_e(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{H}_e(\mathbf{q}, \dot{\mathbf{q}}) \quad (1.44)$$

Avec:

$$\mathbf{q} = \begin{bmatrix} q_{m_1} \\ q_{e_1} \end{bmatrix}, \quad \dot{\mathbf{q}} = \begin{bmatrix} \dot{q}_{m_1} \\ \dot{q}_{e_1} \end{bmatrix}, \quad \ddot{\mathbf{q}} = \begin{bmatrix} \ddot{q}_{m_1} \\ \ddot{q}_{e_1} \end{bmatrix},$$

$$\mathbf{M}_e(\mathbf{q}) = \begin{bmatrix} I_{am_1} + M_{1R} & M_{1R} \\ M_{1R} & M_{1R} \end{bmatrix},$$

$$\mathbf{H}_e(\mathbf{q}, \dot{\mathbf{q}}) = \begin{bmatrix} F_{vm_1}\dot{q}_{m_1} + F_{cm_1}\text{sign}(\dot{q}_{m_1}) + \text{Off}_{m_1} + F_{vL_1}\dot{q}_{L_1} + F_{cL_1}\text{sign}(\dot{q}_{L_1}) \\ F_{vL_1}\dot{q}_{L_1} + F_{cL_1}\text{sign}(\dot{q}_{L_1}) \end{bmatrix}.$$

L'effort moteur peut être exprimé linéairement par rapport aux paramètres dynamiques comme suit :

$$\boldsymbol{\tau}_{\text{mdi}} = \begin{bmatrix} \tau_{\text{mdi}_{m_1}} \\ 0 \end{bmatrix} = \mathbf{MDI} \boldsymbol{\chi} \quad (1.45)$$

Avec :

$$\bullet \mathbf{MDI} = \begin{bmatrix} \ddot{q}_{m_1} & \dot{q}_{m_1} & \text{sign}(\dot{q}_{m_1}) & 1 & 0 & \ddot{q}_{L_1} & \dot{q}_{L_1} & \text{sign}(\dot{q}_{L_1}) \\ 0 & 0 & 0 & 0 & q_{e_1} & \ddot{q}_{L_1} & \dot{q}_{L_1} & \text{sign}(\dot{q}_{L_1}) \end{bmatrix},$$

$$\bullet \boldsymbol{\chi} = [I_{am_1} \quad F_{vm_1} \quad F_{cm_1} \quad \text{Off}_{m_1} \quad k_1 \quad M_{1R} \quad F_{vL_1} \quad F_{cL_1}]^T.$$

1.4 Robot Stäubli TX40

Le robot Stäubli *TX40* est un robot rigide industriel à 6 degrés de liberté à architecture série (voir figure 1-9). Il possède 6 articulations rotoïdes. Sa charge nominale est de 1.7 (kg) et sa masse est de 27 (kg). Les transmissions des axes 1, 2 et 3 sont des réducteurs Stäubli et celle de l'axe 4 comprend un étage poulie-courroie suivi d'un étage de type réducteur Stäubli. Les deux derniers axes du robot sont couplés cinématiquement. Les mesures des courants et positions moteurs sont disponibles et la commande du robot est fournie par Stäubli.

La cinématique du robot est définie en utilisant la convention *DHM* décrite à la section 1.1.2 et représentée en figure 1-10. Les paramètres géométriques définissant le robot sont donnés dans le tableau 1-9.



Figure 1-9 : Robot Staubli TX40

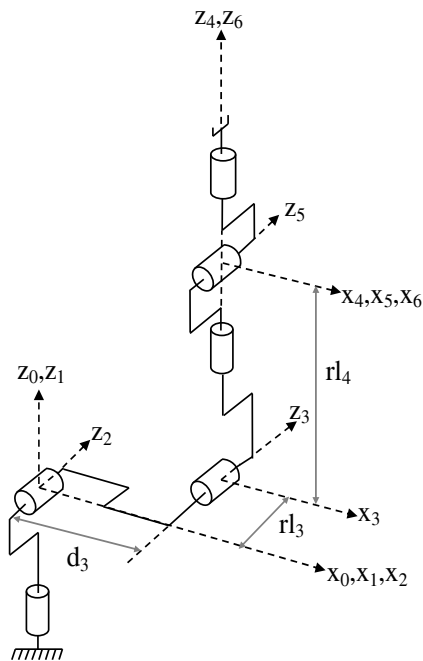


Figure 1-10 : Description *DHM* du robot Staubli TX40

Toutes les articulations sont rotoïdes, la position θ_j de l'articulation j est égale à la position mesurée q_j sauf pour l'articulation 2 ($\theta_2 = q_2 - \pi/2$), l'articulation 3 ($\theta_3 = q_3 + \pi/2$) et l'articulation 6 ($\theta_6 = q_6 + \pi$).

Tableau 1-9 : Paramètres géométriques (DHM) du robot Stäubli TX40

j	α_j	d_j	θ_j	r_j
1	0	0	q_1	0
2	$-\pi/2$	0	$q_2 - \pi/2$	0
3	0	$d_3 (=0.225\text{m})$	$q_3 + \pi/2$	$r_{l3} (=0.035\text{m})$
4	$\pi/2$	0	q_4	$r_{l4} (=0.225\text{m})$
5	$-\pi/2$	0	q_5	0
6	$\pi/2$	0	$q_6 + \pi$	0

Le robot est caractérisé par un couplage cinématique entre les articulations 5 et 6 (Gautier et Briot, 2012) (Khalil et al., 2007):

$$\begin{bmatrix} \dot{q}_{r5} \\ \dot{q}_{r6} \end{bmatrix} = \begin{bmatrix} N_{r5} = 45 & 0 \\ N_{r6} = 32 & N_{r6} = 32 \end{bmatrix} \begin{bmatrix} \dot{q}_5 \\ \dot{q}_6 \end{bmatrix}, \begin{bmatrix} \tau_{c5} \\ \tau_{c6} \end{bmatrix} = \begin{bmatrix} N_{r5} & N_{r6} \\ 0 & N_{r6} \end{bmatrix} \begin{bmatrix} \tau_{r5} \\ \tau_{r6} \end{bmatrix} \quad (1.46)$$

Où :

- \dot{q}_{rj} est vitesse du rotor du moteur de l'articulation j en (rd/s) côté moteur,
- \dot{q}_j est la vitesse articulaire de l'articulation j en (rd/s) côté corps,
- τ_{cj} est le couple moteur prenant en compte le couplage de l'articulation j en (Nm) côté corps,
- τ_{rj} est le couple électromagnétique du rotor du moteur de l'articulation j en (Nm) côté corps,
- N_{r5} et N_{r6} sont respectivement les coefficients de réduction des transmissions des articulations 5 et 6.

L'effet du couplage entre les articulations 5 et 6 ajoute 3 paramètres standards, une inertie I_{am6} , un frottement visqueux F_{vm6} et un frottement sec F_{cm6} au modèle dynamique du Stäubli TX40 :

$$\begin{aligned} \tau_{c5} &= \tau_5 + I_{am6} \ddot{q}_6 + F_{vm6} \dot{q}_6 + F_{cm6} \text{sign}(\dot{q}_5 + \dot{q}_6) \\ \tau_{c6} &= \tau_6 + I_{am6} \ddot{q}_5 + F_{vm6} \dot{q}_5 + F_{cm6} \text{sign}(\dot{q}_5 + \dot{q}_6) \end{aligned} \quad (1.47)$$

Le Stäubli TX40 possède $N_s=87$ paramètres dynamiques standards définis par les (14x6) paramètres standards habituels et les 3 paramètres dus au couplage cinématique.

Le modèle dynamique inverse est automatiquement calculé grâce au logiciel Symoro+ (Khalil et Creusot, 1997) et les équations de Newton-Euler avec un nombre minimal d'équations (Khalil et Dombre, 2002).

Après simplification, les paramètres de base sont au nombre de 61 (voir tableau 1-10), 11 paramètres sont éliminés et 15 paramètres sont regroupés.

Tableau 1-10 : Paramètres de base du robot Stäubli TX40

j	XX	XY	XZ	YY	YZ	ZZ	MX	MY	MZ	M	Ia
1						ZZ _{1R}					
2	XX _{2R}	XY ₂	XZ _{2R}		YZ ₂	ZZ _{2R}	MX _{2R}	MY ₂			
3	XX _{3R}	XY ₃	XZ ₃		YZ ₃	ZZ _{3R}	MX ₃	MY _{3R}			Ia ₃
4	XX _{4R}	XY ₄	XZ ₄		YZ ₄	ZZ _{4R}	MX ₄	MY _{4R}			Ia ₄
5	XX _{5R}	XY ₅	XZ ₅		YZ ₅	ZZ _{5R}	MX ₅	MY _{5R}			Ia ₅
6	XX _{6R}	XY ₆	XZ ₆		YZ ₆	ZZ ₆	MX ₆	MY ₆			Ia ₆

Les regroupements des paramètres sont donnés dans le tableau 1-11.

Tableau 1-11 : Regroupement des paramètres du robot Stäubli TX40

$ZZ_{1R} = ZZ_1 + Ia_1 + YY_2 + YY_3 + 2rl_3MZ_3 + (rl_3^2 + d_3^2)(M_3 + M_4 + M_5 + M_6)$
$XX_{2R} = XX_2 - YY_2 - d_3^2(M_3 + M_4 + M_5 + M_6)$
$XX_{2R} = XX_2 - d_3MZ_3 - rl_3d_3(M_3 + M_4 + M_5 + M_6)$
$ZZ_{2R} = ZZ_2 + Ia_2 + d_3^2(M_3 + M_4 + M_5 + M_6)$
$MX_{2R} = MX_2 + d_3(M_3 + M_4 + M_5 + M_6)$
$XX_{3R} = XX_3 - YY_3 + YY_4 + 2d_3MZ_4 + rl_4^2(M_4 + M_5 + M_6)$
$ZZ_{3R} = ZZ_3 + YY_4 + 2d_3MZ_4 + rl_4^2(M_4 + M_5 + M_6)$
$MY_{3R} = MY_3 - MZ_4 - rl_4(M_4 + M_5 + M_6) ;$
$XX_{4R} = XX_4 - YY_4 + YY_5$
$ZZ_{4R} = ZZ_4 + YY_5$
$MY_{4R} = MY_4 + MZ_5$
$XX_{5R} = XX_5 - YY_5 + YY_6$
$ZZ_{5R} = ZZ_5 + YY_6$
$MY_{5R} = MY_5 - MZ_6$
$XX_{6R} = XX_6 - YY_6$

Tous les paramètres de frottements et d'offset définis dans le tableau 1-12 ainsi que les paramètres dus au couplage cinématique sont conservés.

Tableau 1-12 : Paramètres de frottements et d'offsets du robot Stäubli TX40

j	Fv	Fc	Off
1	Fv ₁	Fc ₁	Off ₁
2	Fv ₂	Fc ₂	Off ₂
3	Fv ₃	Fc ₃	Off ₃
4	Fv ₄	Fc ₄	Off ₄
5	Fv ₅	Fc ₅	Off ₅
6	Fv ₆	Fc ₆	Off ₆

Les paramètres de base permettent de définir le modèle dynamique minimal identifiable.

1.5 Kuka LightWeight Robot (LWR)

Le robot LightWeight Robot (*LWR* ou *LBR*, en version 4+) du fabricant Kuka est un robot à architecture série (Albu-Schäffer et al., 2007) (Bischoff et al., 2010) (Rackl et al., 2012). C'est un robot à 7 articulations rotoïdes. Contrairement aux robots industriels rigides, il supporte une charge

nominale importante par rapport à son poids, soit 7 (kg) pour un poids 14 (kg) (ratio de 1/2 - ratio de 1/16 pour le Stäubli *TX40-*). La transmission de chaque axe utilise un réducteur à engrenages à onde de déformation (harmonic drive) avec un coefficient de réduction élevé (>100) mais caractérisé par une flexibilité articulaire importante.

Ce robot a été conçu principalement à des fins d'interaction avec l'homme pour des applications de co-manipulation et d'apprentissage de trajectoire avec et sans charge. Pour cela, le robot possède une cinématique redondante proche de celle d'un bras humain avec 7 articulations. Plusieurs lois de commande sont implémentées dans la baie de commande du robot: commande en position, compensation de gravité et commande en impédance. Les deux dernières commandes nécessitent la mesure des couples articulaires côté corps. Pour cela, le Kuka *LWR* est équipé d'un capteur de couple sur chaque articulation en sortie de réducteur côté corps (figure 1-12). Les détails sur les lois de commande implémentées dans le robot ne sont pas fournis par le Kuka.

Les mesures des couples moteurs, des positions moteurs \mathbf{q}_m (mesurées par des codeurs côté moteur à l'entrée du réducteur) et des couples côté corps élastiques $\boldsymbol{\tau}_e$ (mesurés par les capteurs de couple côté corps en sortie de réducteur) sont disponibles. Des potentiomètres sont placés en sortie des réducteurs pour mesurer les positions absolues côté corps \mathbf{q}_L . On a vérifié que leur précision est trop faible pour distinguer significativement \mathbf{q}_L des positions moteurs \mathbf{q}_m et permettre le calcul des degrés de liberté flexibles \mathbf{q}_e .

La position du degré de liberté flexible de l'articulation i_{art} est estimée à partir du couple côté corps $\tau_{e_{i_{art}}}$ et de l'estimation de la raideur articulaire ${}^{ap}k_{i_{art}}$. Un terme d'offset $Offl_{i_{art}}$ sur le couple côté corps est rajouté par rapport à l'équation (1.18) pour prendre en compte l'offset sur la mesure par le capteur de couple articulaire :

$$\mathbf{q}_{e_{i_{art}}} = -\frac{\tau_{e_{i_{art}}} - Offl_{i_{art}}}{{}^{ap}k_{i_{art}}} \quad (1.48)$$

Où ${}^{ap}k_{i_{art}}$ est la raideur articulaire de l'articulation i_{art} estimée par Kuka.

La position articulaire, côté corps, de l'articulation i_{art} est estimée de la manière suivante :

$$\mathbf{q}_{L_{i_{art}}} = \mathbf{q}_{m_{i_{art}}} + \mathbf{q}_{e_{i_{art}}} = \mathbf{q}_{m_{i_{art}}} - \frac{\tau_{e_{i_{art}}} - Offl_{i_{art}}}{{}^{ap}k_{i_{art}}} \quad (1.49)$$

Une photo du robot est montrée en figure 1-11 et le détail d'une articulation est représenté figure 1-12.



Figure 1-11 : Robot Kuka *LWR*

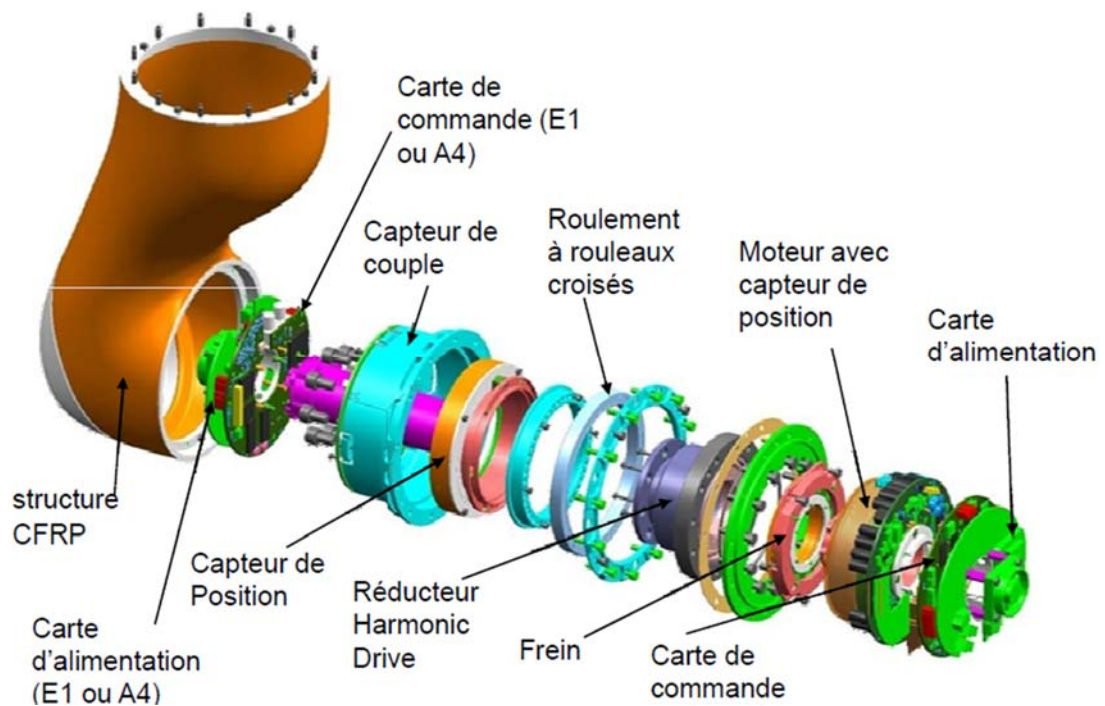


Figure 1-12 : Assemblage mécanique entre les corps 2 et 3 du Kuka *LWR*

1.5.1 Modèle dynamique du robot rigide

Le Kuka *LWR* peut être considéré comme une structure rigide dans l'analyse en basses fréquences, inférieures aux premiers modes flexibles articulaires.

Les repères de la représentation *DHM* du robot rigide sont représentés figure 1-13 et les paramètres géométriques sont donnés dans le tableau 1-13.

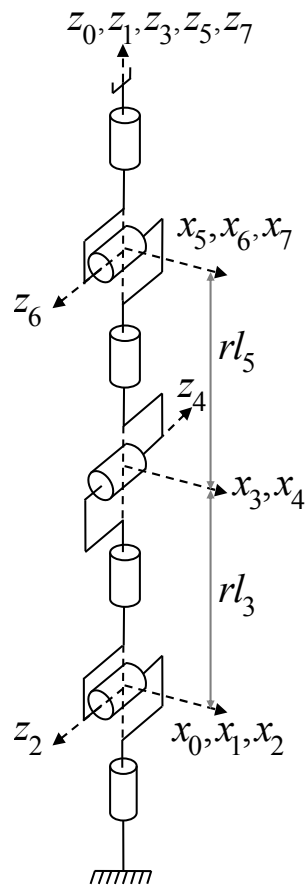


Figure 1-13 : Description *DHM* du Kuka *LWR* rigide

Tableau 1-13 : Paramètres géométriques (DHM) du Kuka LWR rigide

j	a(j)	α_j	d_j	θ_j	r_j
1	0	0	0	q_{m_1}	0
2	0	$\pi/2$	0	q_{m_2}	0
3	0	$-\pi/2$	0	q_{m_3}	$rl_3(= 0.400 \text{ m})$
4	0	$-\pi/2$	0	q_{m_4}	0
5	0	$\pi/2$	0	q_{m_5}	$rl_5(= 0.390 \text{ m})$
6	0	$\pi/2$	0	q_{m_6}	0
7	0	$-\pi/2$	0	q_{m_7}	0

A partir des $N_s=14 \times 7=98$ paramètres standards, 29 paramètres sont regroupés ou supprimés pour donner les 69 paramètres de base. Les paramètres de base et leurs regroupements sont donnés dans le tableau 1-14 et le tableau 1-15.

Tableau 1-14 : Paramètres de base du robot Kuka LWR rigide

j	XX	XY	XZ	YY	YZ	ZZ	MX	MY	MZ	M	Ia
1						ZZ_{1R}					
2	XX_{2R}	XY_2	XZ_2		YZ_2	ZZ_{2R}	MX_2	MY_{2R}			
3	XX_{3R}	XY_3	XZ_3		YZ_3	ZZ_{3R}	MX_3	MY_{3R}			Ia_3
4	XX_{4R}	XY_4	XZ_4		YZ_4	ZZ_{4R}	MX_4	MY_{4R}			Ia_4
5	XX_{5R}	XY_5	XZ_5		YZ_5	ZZ_{5R}	MX_5	MY_{5R}			Ia_5
6	XX_{6R}	XY_6	XZ_6		YZ_6	ZZ_{6R}	MX_6	MY_{6R}			Ia_6
7	XX_{7R}	XY_7	XZ_7		YZ_7	ZZ_7	MX_7	MY_7			Ia_7

Tableau 1-15 : Regroupement des paramètres du robot Kuka LWR rigide

$$\begin{aligned}
 ZZ_{1R} &= ZZ_1 + Ia_1 + YY_2 \\
 ZZ_{2R} &= ZZ_2 + Ia_2 + YY_3 + 2rl_3MZ_3 + rl_3^2 (M_3 + M_4 + M_5 + M_6 + M_7) \\
 XX_{2R} &= XX_2 - YY_2 + YY_3 + 2rl_3MZ_3 + rl_3^2 (M_3 + M_4 + M_5 + M_6 + M_7) \\
 MY_{2R} &= MY_2 + MZ_3 + rl_3(M_3 + M_4 + M_5 + M_6 + M_7) \\
 XX_{3R} &= XX_3 - YY_3 + YY_4 \\
 ZZ_{3R} &= ZZ_3 + YY_4; \\
 MY_{3R} &= MY_3 + MZ_4 \\
 XX_{4R} &= XX_4 - YY_4 + YY_5 + 2rl_5MZ_5 + rl_5^2(M_5 + M_6 + M_7) \\
 ZZ_{4R} &= ZZ_4 + YY_5 + 2rl_5MZ_5 + rl_5^2(M_5 + M_6 + M_7) \\
 MY_{4R} &= MY_4 - MZ_5 - rl_5(M_5 + M_6 + M_7) \\
 XX_{5R} &= XX_5 - YY_5 + YY_6 \\
 ZZ_{5R} &= ZZ_5 + YY_6 \\
 MY_{5R} &= MY_5 - MZ_6 \\
 XX_{6R} &= XX_6 - YY_6 + YY_7 \\
 ZZ_{6R} &= ZZ_6 + YY_7 \\
 MY_{6R} &= MY_6 + MZ_7 \\
 XX_{7R} &= XX_7 - YY_7
 \end{aligned}$$

Tous les paramètres de frottements et d'offset définis dans le tableau 1-16 sont conservés.

Tableau 1-16 : Paramètres de frottements et d'offset du robot Kuka LWR rigide

j	Fv	Fc	Off
1	Fv ₁	Fc ₁	Off ₁
2	Fv ₂	Fc ₂	Off ₂
3	Fv ₃	Fc ₃	Off ₃
4	Fv ₄	Fc ₄	Off ₄
5	Fv ₅	Fc ₅	Off ₅
6	Fv ₆	Fc ₆	Off ₆
7	Fv ₇	Fc ₇	Off ₇

1.5.2 Modèle dynamique du robot flexible

Le robot flexible possède 14 degrés de liberté, 7 degrés de liberté côté moteur et 7 degrés de liberté flexibles qui sont les 7 flexibilités de torsion articulaires localisées dans les réducteurs de type harmonic drive.

Les repères de la représentation *DHM* du robot flexible sont représentés figure 1-15. Les paramètres géométriques sont donnés dans le tableau 1-17 où la colonne j_m contient les indices impairs des degrés de liberté moteur et la colonne j_e contient les indices pairs des degrés de liberté flexibles.

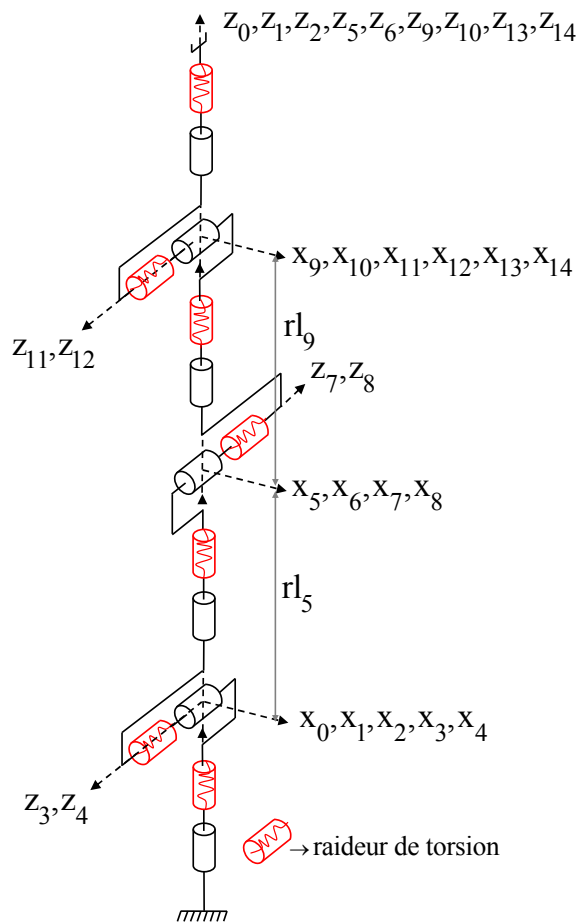


Figure 1-14 : Description DHM du Kuka LWR flexible

Tableau 1-17 : Paramètres géométriques (DHM) du modèle flexible du Kuka LWR

i_{art}	j_m	j_e	α_j	d_j	θ_j	r_j
1	1		0	0	q_{m_1}	0
		2	0	0	q_{e_1}	0
2	3		$\pi/2$	0	q_{m_2}	0
		4	0	0	q_{e_2}	0
3	5		$-\pi/2$	0	q_{m_3}	$rl_3(= 0.400 \text{ m})$
		6	0	0	q_{e_3}	0
4	7		$-\pi/2$	0	q_{m_4}	0
		8	0	0	q_{e_4}	0
5	9		$\pi/2$	0	q_{m_5}	$rl_5(= 0.390 \text{ m})$
		10	0	0	q_{e_5}	0
6	11		$\pi/2$	0	q_{m_6}	0
		12	0	0	q_{e_6}	0
7	13		$-\pi/2$	0	q_{m_7}	0
		14	0	0	q_{e_7}	0

Le modèle dynamique flexible du Kuka LWR contient $N_s=18 \times 7=126$ paramètres dynamiques standards. Les paramètres de base sont au nombre de 97 après regroupement de 18 paramètres et suppression de 11 paramètres (voir tableau 1-18). Les regroupements de paramètres sont donnés dans le tableau 1-19.

Tableau 1-18 : Paramètres de base du robot Kuka LWR flexible

i_{art}	XX	XY	XZ	YY	YZ	ZZ	MX	MY	MZ	M	Ia	Ia _L	k
1						ZZ_{1R}					Ia _{m1}		k ₁
2	XX _{2R}	XY ₂	XZ ₂		YZ ₂	ZZ _{2R}	MX ₂	MY _{2R}			Ia _{m2}		k ₂
3	XX _{3R}	XY ₃	XZ ₃		YZ ₃	ZZ _{3R}	MX ₃	MY _{3R}			Ia _{m3}	Ia _{L3}	k ₃
4	XX _{4R}	XY ₄	XZ ₄		YZ ₄	ZZ _{4R}	MX ₄	MY _{4R}			Ia _{m4}	Ia _{L4}	k ₄
5	XX _{5R}	XY ₅	XZ ₅		YZ ₅	ZZ _{5R}	MX ₅	MY _{5R}			Ia _{m5}	Ia _{L5}	k ₅
6	XX _{6R}	XY ₆	XZ ₆		YZ ₆	ZZ _{6R}	MX ₆	MY _{6R}			Ia _{m6}	Ia _{L6}	k ₆
7	XX _{7R}	XY ₇	XZ ₇		YZ ₇	ZZ _{7R}	MX ₇	MY _{7R}			Ia _{m7}	Ia _{L7}	k ₇

Tableau 1-19 : Regroupement des paramètres du robot Kuka LWR flexible

$$\begin{aligned}
 \mathbf{ZZ}_{1R} &= \mathbf{ZZ}_1 + \mathbf{Ia}_{L_1} + \mathbf{YY}_2 \\
 \mathbf{ZZ}_{2R} &= \mathbf{ZZ}_2 + \mathbf{Ia}_{L_2} + \mathbf{YY}_3 + 2r_{l_3}M_{Z_3} + r_{l_3}^2 (M_3 + M_4 + M_5 + M_6 + M_7) \\
 \mathbf{XX}_{2R} &= \mathbf{XX}_2 - \mathbf{YY}_2 + \mathbf{YY}_3 + 2r_{l_3}M_{Z_3} + r_{l_3}^2 (M_3 + M_4 + M_5 + M_6 + M_7) \\
 \mathbf{MY}_{2R} &= \mathbf{MY}_2 + M_{Z_3} + r_{l_3}(M_3 + M_4 + M_5 + M_6 + M_7) \\
 \mathbf{XX}_{3R} &= \mathbf{XX}_3 - \mathbf{YY}_3 + \mathbf{YY}_4 \\
 \mathbf{ZZ}_{3R} &= \mathbf{ZZ}_3 + \mathbf{YY}_4; \\
 \mathbf{MY}_{3R} &= \mathbf{MY}_3 + M_{Z_4} \\
 \mathbf{XX}_{4R} &= \mathbf{XX}_4 - \mathbf{YY}_4 + \mathbf{YY}_5 + 2 r_{l_5} M_{Z_5} + r_{l_5}^2 (M_5 + M_6 + M_7) \\
 \mathbf{ZZ}_{4R} &= \mathbf{ZZ}_4 + \mathbf{YY}_5 + 2r_{l_5}M_{Z_5} + r_{l_5}^2 (M_5 + M_6 + M_7) \\
 \mathbf{MY}_{4R} &= \mathbf{MY}_4 - M_{Z_5} - r_{l_5}(M_5 + M_6 + M_7) \\
 \mathbf{XX}_{5R} &= \mathbf{XX}_5 - \mathbf{YY}_5 + \mathbf{YY}_6 \\
 \mathbf{ZZ}_{5R} &= \mathbf{ZZ}_5 + \mathbf{YY}_6 \\
 \mathbf{MY}_{5R} &= \mathbf{MY}_5 - M_{Z_6} \\
 \mathbf{XX}_{6R} &= \mathbf{XX}_6 - \mathbf{YY}_6 + \mathbf{YY}_7 \\
 \mathbf{ZZ}_{6R} &= \mathbf{ZZ}_6 + \mathbf{YY}_7 \\
 \mathbf{MY}_{6R} &= \mathbf{MY}_6 + M_{Z_7} \\
 \mathbf{XX}_{7R} &= \mathbf{XX}_7 - \mathbf{YY}_7
 \end{aligned}$$

Les paramètres du modèle de frottement sont donnés dans le tableau 1-20.

Tableau 1-20 : Paramètres de frottements et d'offset du robot Kuka LWR flexible

i_{art}	Fvm	Fcm	Offm	FvL	FcL
1	Fvm ₁	Fcm ₁	Offm ₁	FvL ₁	FcL ₁
2	Fvm ₂	Fcm ₂	Offm ₂	FvL ₂	FcL ₂
3	Fvm ₃	Fcm ₃	Offm ₃	FvL ₃	FcL ₃
4	Fvm ₄	Fcm ₄	Offm ₄	FvL ₄	FcL ₄
5	Fvm ₅	Fcm ₅	Offm ₅	FvL ₅	FcL ₅
6	Fvm ₆	Fcm ₆	Offm ₆	FvL ₆	FcL ₆
7	Fvm ₇	Fcm ₇	Offm ₇	FvL ₇	FcL ₇

Les paramètres du modèle à flexibilités articulaires (voir tableau 1-19) sont les mêmes paramètres que le modèle rigide (voir tableau 1-15) hormis pour les paramètres donnés dans le tableau 1-21.

Tableau 1-21 : Equivalence entre les paramètres du robot Kuka *LWR* rigide et du robot Kuka *LWR* flexible

j	Ia	Fv	Fc	Off
1	$Ia_1 = Iam_1 + Ial_1$	$Fv_1 = Fvm_1 + Fvl_1$	$Fc_1 = Fcm_1 + Fcl_1$	$Off_1 = Offm_1$
2	$Ia_2 = Iam_2 + Ial_2$	$Fv_2 = Fvm_2 + Fvl_2$	$Fc_2 = Fcm_2 + Fcl_2$	$Off_2 = Offm_2$
3	$Ia_3 = Iam_3 + Ial_3$	$Fv_3 = Fvm_3 + Fvl_3$	$Fc_3 = Fcm_3 + Fcl_3$	$Off_3 = Offm_3$
4	$Ia_4 = Iam_4 + Ial_4$	$Fv_4 = Fvm_4 + Fvl_4$	$Fc_4 = Fcm_4 + Fcl_4$	$Off_4 = Offm_4$
5	$Ia_5 = Iam_5 + Ial_5$	$Fv_5 = Fvm_5 + Fvl_5$	$Fc_5 = Fcm_5 + Fcl_5$	$Off_5 = Offm_5$
6	$Ia_6 = Iam_6 + Ial_6$	$Fv_6 = Fvm_6 + Fvl_6$	$Fc_6 = Fcm_6 + Fcl_6$	$Off_6 = Offm_6$
7	$Ia_7 = Iam_7 + Ial_7$	$Fv_7 = Fvm_7 + Fvl_7$	$Fc_7 = Fcm_7 + Fcl_7$	$Off_7 = Offm_7$

1.6 Conclusion

Les outils de modélisation géométriques et dynamiques des robots à corps rigides et à flexibilités articulaires ont été présentés.

La contribution de ce chapitre est l'extension de la description et de la modélisation des robots rigides à chaîne ouverte simple au cas des robots à flexibilités articulaires avec la description *DHM*. Elle permet un calcul automatique des modèles géométriques, cinématiques, dynamiques, et d'identification en utilisant le logiciel de calcul symbolique Symoro+ pour les robots complexes, sans limite du nombre d'articulations.

Les modèles dynamiques des robots étudiés, nécessaires à leur identification, ont été calculés. Le chapitre suivant présente les techniques d'identification utilisées en robotique.

Chapitre 2 Identification par modèle inverse et moindres carrés (*IDIM-LS*)

Sommaire:

2.1 Introduction	39
2.2 Identification par modèle inverse et moindres carrés linéaires (<i>IDIM-LS</i>)	41
2.3 Estimation des écart-types	45
2.4 Moindres carrés pondérés	46
2.5 Filtrage parallèle de Y et des colonnes de W	47
2.6 Calcul des efforts moteurs	48
2.7 Mouvements excitants	49
2.8 Suppression des paramètres mal identifiés et réduction de modèle	49
2.9 Validation des résultats	50
2.10 Identification des paramètres inertiels d'une charge	50
2.11 Conclusion	52

Résumé : Cette partie présente les principales techniques d'identification utilisées en robotique. La méthode d'identification par modèle inverse et moindres carrés est présentée en détail. Les problématiques liées à l'identification des robots à flexibilités articulaires sont abordées.

2.1 Introduction

La connaissance des valeurs numériques des paramètres dynamiques des robots est nécessaire que ce soit pour la simulation qui utilise le modèle dynamique direct ou pour la commande par anticipation qui utilise le modèle dynamique inverse.

Habituellement, le modèle dynamique inverse linéaire par rapport aux paramètres est utilisé pour l'identification hors ligne. Le modèle dynamique inverse est échantillonné en utilisant les positions et les efforts moteurs enregistrés sur le robot en poursuite de trajectoires excitantes.

Les paramètres sont alors estimés comme la solution des moindres carrés linéaires ordinaires ou pondérés du système linéaire surdéterminé obtenu. Cette méthode appelée *IDIM-LS* (en anglais : Inverse Dynamic Identification Model with Least Squares) est présentée en première partie de ce chapitre. Cette méthode est aussi utilisée dans la méthode *DIDIM* (en anglais : Direct and Inverse Dynamic Identification Models) présentée et étudiée au prochain chapitre. La méthode *IDIM-LS* a été utilisée avec succès pour l'identification expérimentale de nombreux robots industriels et prototypes et constitue une référence pour d'autres approches hors ligne ou en ligne.

D'autres méthodes ont été utilisées en automatique (souvent appliquées aux systèmes avec peu de paramètres) et en robotique. Par exemple, le filtre de Kalman étendu a été appliqué sur un robot 2 axes avec 8 paramètres dans (Guglielmi et al., 1987). Il a aussi été utilisé dans (Gautier et al., 1993) (Gautier et Poignet, 2001). L'état du système est estimé grâce au modèle dynamique direct. Cette méthode est consommatrice en temps de calcul. De plus, elle nécessite de bonnes conditions initiales sur les paramètres, sur l'état et sur la matrice de variance-covariance pour converger correctement (Gautier et Poignet, 2001).

La méthode du maximum de vraisemblance qui revient à faire les moindres carrés pondérés, peut aussi être utilisée si la variance de l'erreur n'est pas normalisée (Swevers et al., 1997) (Olsen et Petersen, 2001) (Olsen et al., 2002).

La méthode de la variable instrumentale peut être utilisée pour identifier les paramètres d'un robot (Janot et al., 2014a), elle permet d'éviter un biais dans l'estimation des paramètres en présence de données corrompues dans la matrice d'observation grâce à l'utilisation de données simulées.

L'identification de modèles linéaires paramétriques à temps continu, à partir de données échantillonnées sous formes de modèles d'état ou de fonctions de transfert, est étudiée dans (Garnier et al., 2008). Cependant les paramètres identifiés ne sont pas directement les paramètres physiques du système. Il est nécessaire de linéariser le comportement du système autour d'un point de fonctionnement.

L'identification des robots à flexibilités articulaires est plus complexe que l'identification des robots rigides. En effet, souvent, les degrés de libertés flexibles ne sont pas mesurés. Il n'est donc pas possible d'utiliser directement les méthodes de régression (Spong, 1987). Cette partie présente un état de l'art des différentes méthodes utilisées pour identifier un robot à flexibilités articulaires en fonction des mesures disponibles.

Mesures des degrés de liberté côté moteur et élastiques

Si les degrés de libertés flexibles sont mesurés, il est possible d'utiliser la méthode *IDIM-LS* pour identifier le modèle dynamique complet d'un robot flexible (Pham, 2002).

Dans (Pfeiffer et Holzl, 1995), le modèle dynamique rigide du robot est identifié en premier lieu. Ensuite les raideurs sont identifiées une par une dans le domaine fréquentiel grâce à un capteur d'effort avec des essais statiques et dynamiques et en bloquant le corps lié à l'articulation flexible courante.

L'approche est similaire dans (Albu-Schäffer et Hirzinger, 2001), mais ici les corps du robot ne sont pas bloqués pour identifier les raideurs. De plus des capteurs d'effort sont présents dans le robot en sortie de chaque réducteur.

Les degrés de libertés flexibles du robot peuvent être reconstruits grâce à la capture de mouvement à une fréquence de 250 (Hz) et grâce à une méthode d'optimisation non-linéaire (Lightcap et Banks, 2007) mais cela nécessite un système onéreux et difficile à installer. L'identification est effectuée en deux étapes. Premièrement les masses, inerties et frottements sont identifiés. Ces derniers permettent de calculer le couple côté corps et d'identifier les raideurs durant la deuxième étape grâce aux estimations des positions côté corps. Les positions des pastilles du système de capture de mouvement sont estimées avec une erreur d'environ 1mm *RMS*.

Dans (Pham et al., 2001), (Pham et al., 2002) et (Janot et al., 2011), les auteurs démontrent que la mesure de l'accélération du degré de liberté flexible permet l'identification du modèle dynamique d'un robot un axe.

Ces méthodes d'identification permettent de connaître le comportement dynamique du robot et son comportement transitoire oscillant. Cependant, il est aussi possible d'identifier uniquement les raideurs d'un robot par des études statiques (Alici et Shirinzadeh, 2005) (Dumas et al., 2011) en mesurant avec un laser-tracker les déformations statiques de l'effecteur sous l'effet de différentes charges.

Mesure des degrés de liberté côté moteur seuls

Les paramètres flexibles (raideurs et coefficients de frottements visqueux moteur et articulaire) des deux premiers axes d'un robot sont identifiés avec une approche fréquentielle et un modèle partiellement découplé dans (Makarov et al., 2012). Cependant les paramètres identifiés dépendent fortement de la configuration du robot.

Dans (Östring, 2003), l'identification d'un modèle à 3 masses et 2 raideurs appliqué sur le premier axe d'un robot à 6 degrés de liberté est effectuée grâce à la boîte à outils System Identification du logiciel matriciel Matlab (Ljung, 2000). L'approche est intéressante car les paramètres inertiels et les raideurs sont bien identifiés mais les paramètres de frottements visqueux varient en fonction des données utilisées car les frottements secs ne sont pas pris en compte.

Dans (Janot et al., 2011), basé sur les travaux de (Pham 2002), la dérivation des équations mécaniques permet l'identification d'un robot un axe avec seulement la mesure de la position moteur, cependant la dérivée seconde de l'effort moteur et la dérivée d'ordre 4 de la position moteur sont nécessaires. Cela nécessite un codeur de grande précision et un filtrage adapté.

Le modèle dynamique complet d'un servomécanisme à flexibilité localisée (avec plusieurs modèles de frottements non-linéaires) est identifié grâce à un algorithme de programmation non linéaire qui minimise l'écart entre le courant moteur et le courant simulé dans (Bogdan et Abba, 2009) et (Bogdan et Abba, 2012). Pour converger, l'algorithme nécessite 150 itérations et de bonnes conditions initiales.

Après identification du modèle dynamique rigide du robot, l'identification de la raideur est effectuée à l'aide d'une boîte noire dans (Flacco et De Luca, 2011) à partir du résidu calculé à la première identification.

Les méthodes précédentes utilisent soit les mesures disponibles avec les capteurs internes soit les mesures obtenues avec plusieurs capteurs externes. Certaines méthodes prennent en compte les non-linéarités du modèle dynamique du robot ou utilisent des approximations linéaires autour d'un point de fonctionnement, ce qui permet d'identifier seulement une partie des paramètres. D'autres méthodes nécessitent des dérivées d'ordre élevé qui nécessitent des capteurs de grande précision pour mesurer les positions moteurs et un filtrage adapté. D'autres méthodes utilisent des algorithmes de programmation non linéaire qui nécessitent de bonnes conditions initiales sur les paramètres et un temps de calcul conséquent avant convergence.

2.2 Identification par modèle inverse et moindres carrés linéaires (*IDIM-LS*)

La méthode *IDIM-LS* est la plus populaire pour l'identification des paramètres des robots. Elle a été appliquée avec succès sur plusieurs robots industriels et prototypes (Gautier, 1986) (Gautier, 1990) (Khalil et Dombre, 1999) (Gautier et Khalil, 2001) (Khalil et Dombre, 2002) (Gautier et Poignet, 2002) (Hollerbach et al., 2008) (Gautier et al. 2012). La méthode est illustrée figure 2-1.

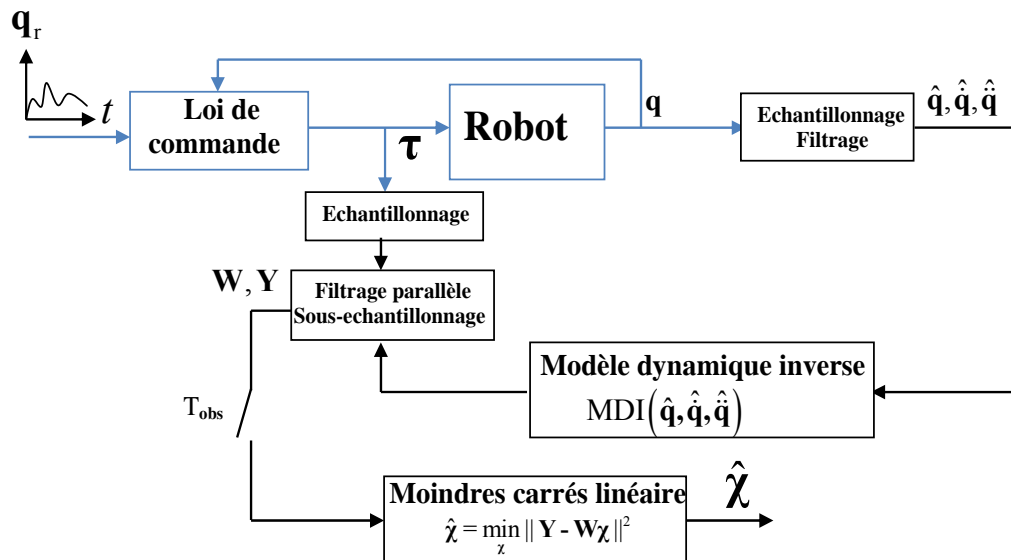


Figure 2-1 : Schéma du principe de la méthode *IDIM-LS*

On rappelle le modèle dynamique inverse qui est linéaire par rapport aux paramètres (voir chapitre 1 et équation (1.16)):

$$\tau_{\text{mdi}} = \mathbf{MDI}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\chi \quad (2.1)$$

Le couple moteur réel τ diffère du modèle τ_{mdi} d'une erreur e due aux bruits de mesure et aux erreurs de modèle. Nous obtenons donc :

$$\boldsymbol{\tau} = \mathbf{MDI}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\boldsymbol{\chi} + \mathbf{e} \quad (2.2)$$

Les vitesses et accélérations articulaires sont calculées sous forme discrète hors ligne. Les dérivées première et seconde des positions sont calculées à l'aide d'un filtre dérivateur à bande passante limitée, produit d'un filtre dérivateur par type différence centrée et d'un filtre passe-bas non causal de type Butterworth aller-retour (Gautier, 1997) (Pham et al., 2001) (Pham, 2002). Ce filtre est appliqué après suppression des échantillons aberrants via un filtre médian glissant sur 3 échantillons. Le gain du filtre Butterworth aller-retour est égal au carré du gain du filtre aller :

$$\left| \frac{\hat{\dot{\mathbf{q}}}}{\dot{\mathbf{q}}} \right| = \frac{1}{1 + \left(\frac{\omega}{\omega_c} \right)^{2n_{\text{but}}}} \quad (2.3)$$

Où:

- $\hat{\mathbf{q}}$ est l'estimée de \mathbf{q} obtenue par filtrage,
- ω est la pulsation,
- ω_c est la pulsation de coupure à -6dB du filtre aller-retour,
- n_{but} est l'ordre du filtre de Butterworth.

La dérivée est calculée par différence centrée de la position échantillonnée filtrée. L'accélération est calculée par différence centrée de la vitesse:

$$\hat{\dot{\mathbf{q}}}_k = \frac{\hat{\mathbf{q}}_{k+1} - \hat{\mathbf{q}}_{k-1}}{2T_e}, \quad \hat{\ddot{\mathbf{q}}}_k = \frac{\hat{\dot{\mathbf{q}}}_{k+1} - \hat{\dot{\mathbf{q}}}_{k-1}}{2T_e} \quad (2.4)$$

Avec :

- $\hat{\dot{\mathbf{q}}}_k$ est la valeur de la dérivée de $\hat{\mathbf{q}}$ à l'instant t_k ,
- $\hat{\ddot{\mathbf{q}}}_k$ est la valeur de la dérivée de $\hat{\dot{\mathbf{q}}}$ à l'instant t_k ,
- $\hat{\mathbf{q}}_{k+1}$ et $\hat{\mathbf{q}}_{k-1}$ sont respectivement les valeurs de $\hat{\mathbf{q}}$ à l'instant t_{k+1} et t_{k-1} ,
- $\hat{\dot{\mathbf{q}}}_{k+1}$ et $\hat{\dot{\mathbf{q}}}_{k-1}$ sont respectivement les valeurs de $\hat{\dot{\mathbf{q}}}$ à l'instant t_{k+1} et t_{k-1} ,
- T_e est la période d'échantillonnage.

La version discrète du filtre passe-bande de calcul de la vitesse produit une distorsion d'amplitude due au gain du filtre Butterworth et au gain du filtre dérivateur par différence centrée (Pham, 2002) :

$$A(\omega) = \frac{|\hat{\dot{q}}|}{|\dot{q}|} = \frac{1}{1 + \left(\frac{\omega}{\omega_c}\right)^{2n_{but}}} \left(\frac{\sin(\omega T_c)}{\omega T_c}\right)^{n_d} \quad (2.5)$$

Soit n_d l'ordre de la dérivée, avec $n_{but} = n_d + 2$.

La pulsation de coupure ω_c dépend de la dynamique du système à identifier caractérisée par ω_{dyn} . Une règle pratique est de choisir $\omega_c = 10 \omega_{dyn}$ (Gautier, 1997).

La mise en œuvre du filtrage avec Matlab utilise les fonctions 'butter' et 'filtfilt'.

Les diagrammes de Bode des filtres aller-retour continu, de la dérivée centrée ($n_d=1$), du filtre passe-bas Butterworth discret avec $n_{but}=3$, $\omega_c=2\pi/T_c=4000$ (rd/s) et $\omega_c=100$ (rd/s), et du passe-bande sont représentés figure 2-2.

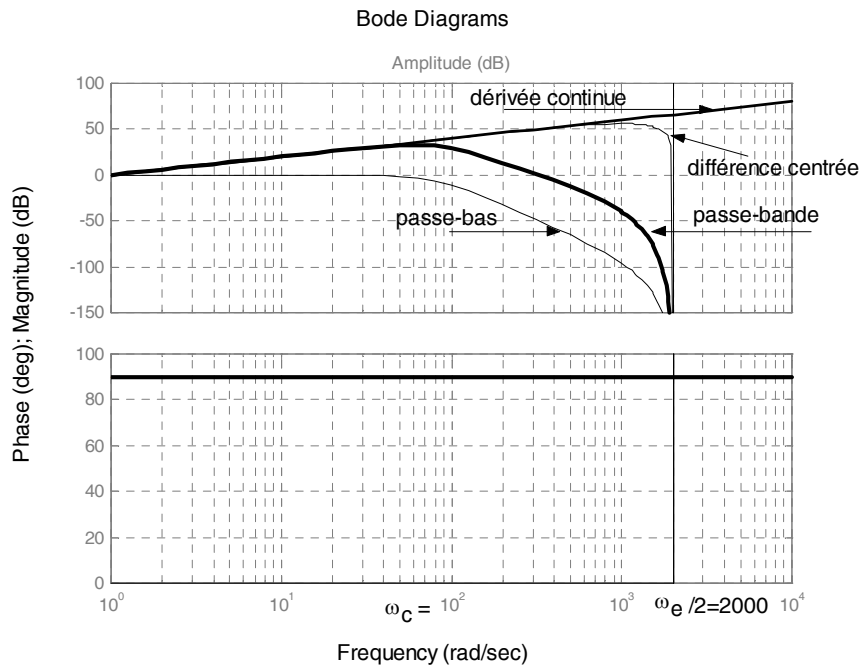


Figure 2-2 : Filtre passe-bande dérivée première

Après échantillonnage de (2.2) à la fréquence de mesure f_m , et regroupement des échantillons axe par axe, on obtient le système surdéterminé suivant :

$$\mathbf{Y}_{fm}(\tau) = \mathbf{W}_{fm}(\hat{q}, \hat{\dot{q}}, \hat{\ddot{q}})\boldsymbol{\chi} + \boldsymbol{\rho}_{fm} \quad (2.6)$$

avec :

- \mathbf{Y}_{fm} est le vecteur ($n_{fm} \times 1$) :

$$\mathbf{Y}_{fm} = \begin{bmatrix} \mathbf{Y}_{fm}^1 \\ \mathbf{Y}_{fm}^2 \\ \vdots \\ \mathbf{Y}_{fm}^n \end{bmatrix}, \quad \mathbf{Y}_{fm}^j = \begin{bmatrix} \tau_j(t_1) \\ \tau_j(t_2) \\ \vdots \\ \tau_j(t_{n_e}) \end{bmatrix} \quad (2.7)$$

- \mathbf{W}_{fm} est la matrice ($n_{fm} \times b$) :

$$\mathbf{W}_{fm}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) = \begin{bmatrix} \mathbf{W}_{fm}^1 \\ \mathbf{W}_{fm}^2 \\ \vdots \\ \mathbf{W}_{fm}^n \end{bmatrix}, \mathbf{W}_{fm}^j = \begin{bmatrix} \mathbf{MDI}^j(\hat{\mathbf{q}}(t_1), \hat{\dot{\mathbf{q}}}(t_1), \hat{\ddot{\mathbf{q}}}(t_1)) \\ \mathbf{MDI}^j(\hat{\mathbf{q}}(t_2), \hat{\dot{\mathbf{q}}}(t_2), \hat{\ddot{\mathbf{q}}}(t_2)) \\ \vdots \\ \mathbf{MDI}^j(\hat{\mathbf{q}}(t_{n_e}), \hat{\dot{\mathbf{q}}}(t_{n_e}), \hat{\ddot{\mathbf{q}}}(t_{n_e})) \end{bmatrix} \quad (2.8)$$

- n_e est le nombre d'échantillons enregistrés sur une fenêtre d'observation T_{obs} avec $n_e = T_{obs} \times f_m$,
- $\boldsymbol{\rho}_{fm}$ est le vecteur colonne ($n_{fm} \times 1$) des résidus avec $n_{fm} = n \times n_e$,
- $\tau_j(t_k)$ est l'échantillon à l'instant t_k de τ_j (2.2),
- $\mathbf{MDI}^j(\hat{\mathbf{q}}(t_k), \hat{\dot{\mathbf{q}}}(t_k), \hat{\ddot{\mathbf{q}}}(t_k))$ est l'échantillon à l'instant t_k de la ligne j de (2.2),
- $(\hat{\mathbf{q}}, \hat{\dot{\mathbf{q}}}, \hat{\ddot{\mathbf{q}}})$ est l'estimation de $(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$.

Le système surdéterminé suivant est obtenu après filtrage parallèle de la relation (2.6) (voir partie 2.5) :

$$\mathbf{Y} = \mathbf{W}\boldsymbol{\chi} + \boldsymbol{\rho} \quad (2.9)$$

Avec :

- \mathbf{Y} est le vecteur ($r \times 1$) des mesures avec $r = n_{fm}/n_p$ (voir partie 2.5),
- \mathbf{W} est la matrice ($r \times b$) d'observation,
- $\boldsymbol{\rho}$ est le vecteur colonne ($r \times 1$) des résidus.

Sous la condition que $\boldsymbol{\chi}$ soit le vecteur des b paramètres de base et que les échantillons de (2.6) soient obtenus sur des trajectoires excitantes (Gautier et Khalil, 1991), \mathbf{W} est de rang maximum (Gautier, 1997). La solution $\hat{\boldsymbol{\chi}}$ des moindres carrés ordinaires de (2.9) est unique et s'exprime par :

$$\hat{\boldsymbol{\chi}} = \underset{\boldsymbol{\chi}}{\operatorname{argmin}} \|\boldsymbol{\rho}\|^2 = \mathbf{W}^+ \mathbf{Y} \quad (2.10)$$

Avec \mathbf{W}^+ la matrice pseudo-inverse de \mathbf{W} :

$$\mathbf{W}^+ = (\mathbf{W}^T \mathbf{W})^{-1} \mathbf{W}^T \quad (2.11)$$

2.3 Estimation des écart-types

On suppose que \mathbf{W} est déterministe et que \mathbf{p} est un bruit additif indépendant à moyenne nulle, de composantes indépendantes, d'écart-type σ_p et de matrice de variance-covariance \mathbf{C}_{pp} :

$$\mathbf{C}_{pp} = E(\mathbf{p}\mathbf{p}^T) = \sigma_p^2 \mathbf{I}_r \quad (2.12)$$

Avec :

- E est l'espérance mathématique,
- \mathbf{I}_r est la matrice $(r \times r)$ identité.

L'écart-type σ_p peut être calculé par l'estimateur non biaisé suivant:

$$\hat{\sigma}_p^2 = \frac{\|\mathbf{Y} - \mathbf{W}\hat{\boldsymbol{\chi}}\|^2}{(r - b)} \quad (2.13)$$

La matrice de variance-covariance des estimées au sens des moindres carrés ordinaires a alors pour expression:

$$\mathbf{C}_{xx} = \hat{\sigma}_p^2 (\mathbf{W}^T \mathbf{W})^{-1} \quad (2.14)$$

Les écart-types des paramètres de base sont les racines carrées de la diagonale de \mathbf{C}_{xx} :

$$\hat{\sigma}_{\hat{\chi}_i} = \sqrt{\mathbf{C}_{xx}(i,i)} \quad (2.15)$$

Où $\mathbf{C}_{xx}(i,i)$ est le $i^{\text{ème}}$ élément diagonal de \mathbf{C}_{xx} .

L'écart type relatif $\% \hat{\sigma}_{\hat{\chi}_i}$ du paramètre de base d'indice i est donné par :

$$\% \hat{\sigma}_{\hat{\chi}_i} = 100 \frac{\hat{\sigma}_{\hat{\chi}_i}}{|\hat{\chi}_i|} \text{ pour } |\hat{\chi}_i| \neq 0 \quad (2.16)$$

Le calcul des écart-types relatifs fournit un indicateur sur l'incertitude de l'estimation des paramètres. Cet indicateur est à utiliser avec prudence car les hypothèses statistiques utilisées pour le calcul ne sont pas garanties dans notre cas. Les coefficients de \mathbf{W} sont perturbés par des mesures et des erreurs de modèle, en conséquence \mathbf{W} n'est pas déterministe.

Cependant l'utilisation du test de Kolmogorov-Smirnov sur les 3 cas d'études de la thèse a montré que le résidu \mathbf{p} est un bruit gaussien (avec les moindres carrés pondérés, voir partie suivante). Ce point est développé dans les chapitres 3 et 4 et dans l'annexe E.

2.4 Moindres carrés pondérés

Le calcul précédent suppose que \mathbf{p} est un bruit additif indépendant à moyenne nulle de matrice de variance-covariance $\sigma_p^2 \mathbf{I}_r$ ce qui traduit la même confiance à chaque ligne du système (2.9). L'hypothèse n'est pas judicieuse car la précision des équations dans (2.9) dépend de chaque articulation j . L'introduction des moindres carrés pondérés (Gautier, 1997) permet d'améliorer l'estimation des paramètres et des écart-types.

Cette méthode consiste à calculer un nouveau système global en attribuant à chaque équation de (2.9) un facteur de confiance inversement proportionnel à l'écart-type calculé avec une résolution par moindres carrés linéaires de chaque sous-système linéaire j construit avec les échantillons de l'articulation j . La méthode des moindres carrés pondérés permet de normaliser la variance de l'erreur \mathbf{p} dans (2.9) après pondération.

En partant de chaque sous-système (j) définis en (2.7) et (2.8) et après filtrage parallèle:

$$\mathbf{Y}^j = \mathbf{W}^j \boldsymbol{\chi} + \boldsymbol{\rho}^j \quad (2.17)$$

Avec :

- \mathbf{Y}^j est le vecteur ($r_j \times 1$) des mesures du sous-système (j),
- \mathbf{W}^j est la matrice ($r_j \times b$) d'observation du sous-système (j),
- $\boldsymbol{\rho}^j$ est le vecteur ($r_j \times 1$) des erreurs du sous-système (j).

On calcule l'écart-type de chaque sous-système résolu par moindres carrés (Lawson et Hanson, 1974) (Gautier, 1997):

$$\hat{\sigma}_{\rho_j}^2 = \frac{\|\mathbf{Y}^j - \mathbf{W}^j \hat{\boldsymbol{\chi}}\|^2}{(r_j - b)} \quad (2.18)$$

Avec :

- $\hat{\sigma}_{\rho_j}^2$ est la variance calculée avec le sous-système j ,
- $\|\mathbf{Y}^j - \mathbf{W}^j \hat{\boldsymbol{\chi}}\|$ est la norme minimale de l'erreur entre l'effort moteur mesuré et sa valeur calculée par le modèle avec la solution des moindres carrés $\hat{\boldsymbol{\chi}}$,
- r_j est le nombre de lignes de la matrice d'observation \mathbf{W}^j du sous-système j ,
- b est le nombre de colonnes de la matrice d'observation \mathbf{W}^j du sous-système j .

Afin d'effectuer la pondération, on attribue à chaque sous-système j un facteur de confiance Ω_j^{-1} fonction de son écart-type. Ω_j^{-1} est factorisé sous la forme

$$\Omega_j^{-1} = \mathbf{S}_j^T \mathbf{S}_j \text{ avec } \begin{cases} \mathbf{S}_j = \frac{1}{\hat{\sigma}_{\rho_j}} \mathbf{I}_{r_j} \\ \mathbf{S}_j = \Omega_j^{-1/2} \end{cases} \quad (2.19)$$

Où \mathbf{I}_{r_j} est une matrice identité de taille $(r_j \times r_j)$.

On factorise Ω^{-1} sous la forme $\Omega^{-1} = \mathbf{S}^T \mathbf{S}$ avec :

$$\mathbf{S} = \begin{bmatrix} \mathbf{S}_1 & 0 & \dots & 0 \\ 0 & \mathbf{S}_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \mathbf{S}_n \end{bmatrix} \quad (2.20)$$

On obtient (2.9) pondéré par \mathbf{S} sous la forme suivante:

$$\mathbf{Y}_p = \mathbf{W}_p \boldsymbol{\chi} + \boldsymbol{\rho}_p \text{ avec } \begin{cases} \mathbf{Y}_p = \mathbf{S} \mathbf{Y} \\ \mathbf{W}_p = \mathbf{S} \mathbf{W} \\ \boldsymbol{\rho}_p = \mathbf{S} \boldsymbol{\rho} \end{cases} \quad (2.21)$$

Les opérations de pondération normalisent la variance de l'erreur $\boldsymbol{\rho}$. La solution des moindres carrés pondérés est la solution des moindres carrés ordinaires du système (2.21) :

$$\begin{aligned} \hat{\boldsymbol{\chi}} &= \underset{\boldsymbol{\chi}}{\operatorname{argmin}} (\boldsymbol{\rho}_p^T \boldsymbol{\rho}_p) \\ &= \underset{\boldsymbol{\chi}}{\operatorname{argmin}} (\boldsymbol{\rho}^T \Omega^{-1} \boldsymbol{\rho}) \end{aligned} \quad (2.22)$$

2.5 Filtrage parallèle de \mathbf{Y} et des colonnes de \mathbf{W}

La matrice \mathbf{W} et le vecteur \mathbf{Y} sont construits à partir de données bruitées car les mesures sont expérimentales et les estimations des vitesses et accélérations sont bruitées. En conséquence, l'estimation $\hat{\boldsymbol{\chi}}$ peut présenter un biais (Gautier, 1997) (Pham, 2002).

Un filtrage permet d'éliminer les perturbations hautes fréquences présentes dans les efforts moteurs. Chaque colonne de \mathbf{W} et le vecteur \mathbf{Y} sont filtrés en parallèle par un filtre passe-bas de fréquence de coupure ω_p de type Chebyshev de type 1 et d'ordre 8. Ce filtre, appelé filtre parallèle,

n'affecte pas la solution des moindres carrés car la distorsion introduite par le filtre est identique dans chaque membre du système linéaire.

Cependant, comme pour le filtre Butterworth, la pulsation de coupure ω_p doit permettre de garder la dynamique du système à identifier. Enfin, comme il n'y a plus d'informations pour les fréquences supérieures à ω_p , un sous-échantillonnage est appliqué sur chaque colonne de \mathbf{W} et sur le vecteur \mathbf{Y} . Un échantillon sur n_p est conservé, n_p étant l'ordre de décimation choisi:

$$n_p = 0.8 \frac{\omega_c}{2\omega_p} \quad (2.23)$$

Ce filtrage et ce sous-échantillonnage permettent de diminuer la taille du système surdéterminé (2.21), d'améliorer l'indépendance statistique entre les échantillons et d'améliorer l'estimation des écart-types. Il est possible d'effectuer un test statistique de Durbin-Watson pour vérifier l'indépendance des échantillons du résidu de ρ_p , le détail du test est donné en annexe F.

2.6 Calcul des efforts moteurs

Dans la plupart des cas, seules les consignes de courant des boucles de courant des amplificateurs de puissance qui alimentent les moteurs sont fournies par le contrôleur du robot.

La boucle de courant à une dynamique très supérieure (bande passante supérieure à 500 (Hz)) à la dynamique mécanique du robot (bande passante inférieure à 50 (Hz)), sa fonction de transfert pour les fréquences inférieures à 10 fois sa bande passante est donc approchée par un gain statique (Restrepo et Gautier, 1995) (Gautier et al., 2012).

L'effort moteur de l'articulation j se calcule de la manière suivante :

$$\tau_j = g_{\tau j} v_{\tau j} \quad (2.24)$$

Avec :

- τ_j est le l'effort moteur de l'articulation j ,
- $v_{\tau j}$ est la consigne de courant de l'articulation j .
- $g_{\tau j}$ est le gain d'actionnement de l'articulation j :

$$g_{\tau j} = N_{\tau j} G_j k_{\tau j} \quad (2.25)$$

Avec :

- $N_{\tau j}$ est le coefficient de réduction de la transmission mécanique de l'articulation j ,
- G_j est le gain statique de la boucle de courant de l'articulation j .
- $k_{\tau j}$ est la constante de couple électromagnétique de l'articulation j :

Les gains d'actionnement peuvent être fournis par le fabricant du robot. Il est possible de les identifier à l'aide d'une méthode se basant sur les moindres carrés totaux (Gautier et Briot, 2012) (Gautier et Briot, 2014).

2.7 Mouvements excitants

La résolution du système (2.21) au sens des moindres carrés nécessite l'utilisation de trajectoires qui excitent au mieux les paramètres de base pour que la matrice \mathbf{W} soit de rang plein (Gautier et Khalil, 1991). Une trajectoire excitante se caractérise par un bon conditionnement de \mathbf{W} .

Ainsi, le calcul de trajectoires excitantes peut s'effectuer à l'aide d'une méthode d'optimisation non linéaire dont le critère à minimiser est une fonction de $\text{cond}(\mathbf{W})$ avec:

$$\text{cond}(\mathbf{W}) = \frac{\sigma_{\max}}{\sigma_{\min}} \quad (2.26)$$

où σ_{\max} et σ_{\min} sont respectivement les valeurs singulières maximale et minimale de la matrice \mathbf{W} (Presse et Gautier, 1993) (Hollerbach et al., 2008).

Un critère qui traduit un bon conditionnement de \mathbf{W} et un bon rapport signal/bruit dans \mathbf{Y} est le suivant :

$$f = \text{cond}(\mathbf{W}) + \frac{1}{\sigma_{\min}} = \frac{\sigma_{\max} + 1}{\sigma_{\min}} \quad (2.27)$$

Où le terme $1/\sigma_{\min}$ permet d'éviter d'avoir un bon conditionnement de \mathbf{W} avec des valeurs singulières faibles.

Il est également possible de construire différentes trajectoires pour exciter spécifiquement certains paramètres. Pour exciter les paramètres de frottement, des paliers de vitesse peuvent être utilisés. Pour exciter les paramètres de gravité, des allers-retours à faible vitesse peuvent être effectués dans l'ensemble de l'espace de travail du robot. Enfin pour les inerties, des trajectoires à forte accélération sont utilisées (Janot, 2007). Dans le cas des systèmes flexibles, le sinus à fréquence variable (sinus wobulé) ou l'utilisation d'impulsions répétées de faible amplitude permet d'exciter les raideurs (Pham, 2002).

2.8 Suppression des paramètres mal identifiés et réduction de modèle

Comme certains paramètres ont une contribution négligeable dans la dynamique du robot, ils restent mal identifiés. Ils sont très faibles et possèdent un grand écart-type relatif malgré l'utilisation de bonnes trajectoires excitantes. Ils peuvent être éliminés du modèle dynamique inverse en conservant la précision du modèle.

Récemment, une méthode de réduction basée sur le test de Fischer (ou F-test) a été proposée dans (Janot et al., 2014a).

On utilise le résidu pondéré \mathbf{p}_p . A partir des b paramètres de base, b_c paramètres peuvent constituer un jeu de paramètres essentiels.

La réduction de modèle avec le F-test se déroule de la manière suivante :

- la norme de l'erreur $\|\mathbf{p}_p\|$ est calculée avec le système (2.21) qui contient les b paramètres de base,
- la norme de l'erreur $\|\mathbf{p}_{p_c}\|$ est calculée avec le système (2.21) avec \mathbf{W} réduite à b_c colonnes correspondant à un jeu de b_c paramètres essentiels,
- \hat{F} est calculé:

$$\hat{F} = \frac{\frac{\|\mathbf{p}_{p_c}\|^2 - \|\mathbf{p}_p\|^2}{(b - b_c)}}{\frac{\|\mathbf{p}_p\|^2}{(n_c - b)}} \quad (2.28)$$

Si \hat{F} est inférieur ou égal à $F_{(1-\alpha)}$ alors le F-test accepte la réduction de modèle sinon il le rejette (voir tableau de Fisher-Snedecor en annexe E).

L'algorithme commence avec les paramètres ayant les écart-types relatifs les plus grands et progresse pas à pas en décroissant jusqu'à que le F-test échoue.

L'utilisation du test de Fisher n'est pertinente que si le résidu \mathbf{p}_{p_c} a une distribution normale. Il est donc nécessaire d'effectuer un test de Kolmogorov-Smirnov pour vérifier cette condition (Davidson et MacKinnon, 1993) (Massey, 1951) (Janot et al. 2014a), ce test est décrit en annexe E.

2.9 Validation des résultats

Plusieurs méthodes de validation des résultats sont possibles :

- les paramètres estimés peuvent être comparés à ceux du constructeur du robot ou à des mesures directes,
- une validation directe permet de comparer le vecteur des mesures \mathbf{Y} avec le vecteur des mesures estimé par le modèle $\mathbf{W}\hat{\boldsymbol{\chi}}$,
- une validation croisée permet de comparer le vecteur des mesures \mathbf{Y} avec le vecteur des mesures estimé par le modèle $\mathbf{W}\hat{\boldsymbol{\chi}}$ calculé sur des trajectoires différentes de celles utilisées pour l'identification,
- une validation par simulation qui compare les efforts moteurs mesurés aux efforts calculés par simulation du modèle dynamiques direct en boucle fermée, ce qui nécessite la connaissance précise de la commande du robot.

2.10 Identification des paramètres inertiels d'une charge

L'identification des paramètres inertiels d'une charge fixée sur l'effecteur d'un robot peut être nécessaire pour adapter sa loi de commande afin d'améliorer la précision dynamique.

Elle permet aussi de comparer les paramètres inertiels identifiés de la charge avec tout ou partie des paramètres *a priori* de cette dernière. Un test simple compare la masse de la charge

identifiée avec sa masse mesurée par une balance de précision. Cela permet de vérifier la précision d'une identification.

Une des méthodes proposée dans (Khalil et al. 2007) permet l'identification globale des paramètres du robot et des paramètres inertiels de la charge.

La charge est considérée comme un corps d'indice $n+1$ attaché au corps d'indice n du robot (le repère R_{n+1} est fixe par rapport au repère R_n). Elle est caractérisée par les 10 paramètres inertiels définis dans le vecteur (A.13), où l'indice $n+1$ est remplacé par l'indice L (Load) pour plus de lisibilité :

$$\chi_L = \begin{bmatrix} XX_L & XY_L & XZ_L & YY_L & YZ_L & ZZ_L & MX_L & MY_L & MZ_L & M_L \end{bmatrix}^T \quad (2.29)$$

Le modèle dynamique inverse (2.2) avec la charge devient :

$$\tau = \begin{bmatrix} \mathbf{MDI}(q, \dot{q}, \ddot{q}) & \mathbf{MDI}_L(q, \dot{q}, \ddot{q}) \end{bmatrix} \begin{bmatrix} \chi \\ \chi_L \end{bmatrix} + \mathbf{e} \quad (2.30)$$

où :

- \mathbf{MDI}_L est la matrice ($n \times 10$) correspondant à la charge, fonction des positions, vitesses, accélérations et des paramètres géométriques du robot,
- χ_L est le vecteur (10×1) des paramètres inertiels de la charge.

Pour identifier séparément les paramètres inertiels de la charge et les paramètres du robot, il est nécessaire que le robot parcoure deux trajectoires : une trajectoire sans la charge et une trajectoire avec la charge. Ces deux trajectoires peuvent être différentes.

Le système surdéterminé suivant est obtenu après échantillonnage et filtrage parallèle de la relation (2.30) :

$$\mathbf{Y} = \begin{bmatrix} \mathbf{Y}_a \\ \mathbf{Y}_b \end{bmatrix} = \begin{bmatrix} \mathbf{W}_a & 0 \\ \mathbf{W}_b & \mathbf{W}_L \end{bmatrix} \begin{bmatrix} \chi \\ \chi_L \end{bmatrix} + \mathbf{p} \quad (2.31)$$

où :

- \mathbf{Y}_a est le vecteur des mesures (2.7) construit avec le robot sans la charge,
- \mathbf{Y}_b est le vecteur des mesures (2.7) construit avec le robot avec la charge,
- \mathbf{W}_a est la matrice d'observation (2.8) construite avec le robot sans la charge,
- \mathbf{W}_b est la matrice d'observation (2.8) construite avec le robot avec la charge,
- \mathbf{W}_L est la matrice d'observation liée à la charge (construite avec le robot avec la charge):

$$\mathbf{W}_L = \begin{bmatrix} \mathbf{W}_{L_1} \\ \mathbf{W}_{L_2} \\ \vdots \\ \mathbf{W}_{L_n} \end{bmatrix}, \mathbf{W}_{L_j} = \begin{bmatrix} \mathbf{MDI}_{L_j}(t_1) \\ \mathbf{MDI}_{L_j}(t_2) \\ \vdots \\ \mathbf{MDI}_{L_j}(t_{n_e}) \end{bmatrix} \quad (2.32)$$

Les paramètres du robot $\hat{\boldsymbol{\chi}}$ et de la charge $\hat{\boldsymbol{\chi}}_L$ sont la solution au sens des moindres carrés pondérés de (2.31).

Cette méthode d'identification des paramètres inertiels d'une charge, fixée sur l'effecteur d'un robot, utilise une procédure d'identification globale qui évite l'accumulation d'erreurs des méthodes d'identification séquentielles.

2.11 Conclusion

Dans ce chapitre, la méthode d'identification *IDIM-LS* a été présentée. Les avantages de cette méthode sont les suivants :

- le calcul du modèle dynamique inverse est réalisé automatiquement avec des logiciels de calcul formel tel que Symoro+ (Khalil et Creusot, 1997),
- aucune condition initiale ni sur les variables articulaires ni sur les paramètres n'est nécessaire,
- le temps de calcul est très faible (aucune simulation du modèle dynamique direct du robot),
- les outils numériques de l'algèbre linéaire (factorisation *QR*, *SVD*) sont utilisés,
- les paramètres de base sont calculés automatiquement.

La difficulté de cette méthode réside sur l'estimation précise des vitesses et accélérations articulaires ($\hat{\mathbf{q}}, \hat{\dot{\mathbf{q}}}, \hat{\ddot{\mathbf{q}}}$) intervenant dans le modèle dynamique inverse. Ces dernières sont estimées par un filtre dérivateur passe-bande à partir des positions mesurées. La fréquence d'échantillonnage doit être compatible avec la bande passante de ce filtre dérivateur. La précision est sensible au réglage du filtre et nécessite une connaissance *a priori* des dynamiques à identifier. Si la fréquence d'échantillonnage est trop faible ou si le filtre est mal réglé, une distorsion d'amplitude apparaît sur les vitesses et accélérations et dans la matrice d'observation \mathbf{W} (Gautier et al., 2012), ce qui biaise la solution des moindres carrés.

Chapitre 3 Méthode *DIDIM* appliquée aux robots rigides

Sommaire:

3.2	Identification par une méthode <i>CLOE</i>	55
3.3	Méthode <i>DIDIM</i>	59
3.3.1	Théorie.....	59
3.3.2	Initialisation de l'algorithme.....	63
3.3.3	Structure de l'algorithme.....	68
3.4	Identification des paramètres inertiels d'une charge.....	68
3.5	Application au robot Stäubli <i>TX40</i>	69
3.6	Effet de la pondération et du filtre Decimate.....	80
3.7	Simplification du coefficient d'adaptation k_{α_j}	83
3.8	Comparaison entre la méthode <i>CLOE</i> à erreur de position et la méthode <i>DIDIM</i>	87
3.8.1	Comparaison analytique.....	87
3.8.2	Comparaison expérimentale.....	89
3.8.3	Discussion.....	96
3.9	Comparaison expérimentale entre la méthode <i>CLOE</i> à erreur d'effort et la méthode <i>DIDIM</i>	97
3.10	Conclusion.....	97

Résumé: Ce chapitre présente la méthode *DIDIM* appliquée aux robots rigides. La méthode *CLOE* classique à erreur de position est présentée afin d'introduire la méthode *DIDIM*. Une amélioration de la méthode est proposée par la suite. Enfin, une comparaison entre une méthode *CLOE* classique à erreur de position et la méthode *DIDIM* est effectuée sur le robot Stäubli *TX40*, elle montre la supériorité de la méthode *DIDIM* par rapport aux méthodes *CLOE* usuelles à erreur de position.

3.1 Introduction

La méthode *DIDIM* est une méthode d'identification récente qui ne nécessite pas la mesure des positions articulaires (Janot, 2007) (Gautier et al. 2012). Les paramètres optimaux minimisent le carré de la norme de l'erreur entre les efforts moteurs réels et les efforts moteurs simulés en boucle fermée ce qui classe la méthode *DIDIM* dans la famille des méthodes à erreur de sortie.

3.2 Identification par une méthode *CLOE*

La méthode à erreur de sortie (en anglais: *OE* pour Output Error) minimise l'erreur quadratique entre les sorties réelles \mathbf{y} et les sorties simulées \mathbf{y}_s du système. Le système réel et le système simulé sont soumis à la même entrée. La méthode peut être implémentée en boucle ouverte (Walter et Pronzato, 1997) ou en boucle fermée (Carrillo et al., 2009) (Landau et al., 1999). Cette méthode est aussi appelée méthode du modèle car on cherche les paramètres du modèle qui permettent de s'approcher au mieux du comportement du robot réel (Richalet et al., 1971).

Dans le cas d'une simulation en boucle ouverte (figure 3-1), l'entrée est composée des efforts moteurs $\mathbf{u}=\boldsymbol{\tau}$. Dans le cas d'une simulation en boucle fermée (figure 3-2), l'entrée est composée des consignes de positions, et/ou de vitesses et/ou d'accélération ($\mathbf{u}=\mathbf{q}_r$).

Généralement pour les systèmes mécaniques, la sortie réelle est la position $\mathbf{y}=\mathbf{q}$ et la sortie simulée est la position simulée du système $\mathbf{y}_s=\mathbf{q}_s$. Pour un robot, les positions articulaires simulées sont les solutions du modèle dynamique direct donné sous une forme implicite par :

$$\mathbf{M}(\mathbf{q}_s)\ddot{\mathbf{q}}_s = \boldsymbol{\tau}_s - \mathbf{H}(\mathbf{q}_s, \dot{\mathbf{q}}_s) \quad (3.1)$$

Comme les robots ont un comportement de système double intégrateur, la simulation en boucle ouverte est sensible aux conditions initiales sur les positions et vitesses ainsi qu'aux erreurs de calculs numériques dans les résolutions d'équations différentielles. La méthode à erreur de sortie en boucle fermée est donc à privilégier pour l'identification des robots.

La solution optimale $\hat{\boldsymbol{\chi}}$ minimise le critère quadratique suivant :

$$J(\boldsymbol{\chi}) = \|\mathbf{Y} - \mathbf{Y}_s\|^2 = (\mathbf{Y} - \mathbf{Y}_s)^T (\mathbf{Y} - \mathbf{Y}_s) \quad (3.2)$$

Avec :

- \mathbf{Y} est le vecteur (rx1) obtenu après échantillonnage et filtrage des mesures de $\mathbf{y}=\mathbf{q}$ à la fréquence de mesure f_m sur une fenêtre d'observation T_{obs} ,
- \mathbf{Y}_s est le vecteur (rx1) obtenu après échantillonnage et filtrage des mesures de $\mathbf{y}_s=\mathbf{q}_s$ à la fréquence de mesure f_m sur une fenêtre d'observation T_{obs} .

Comme le modèle dynamique direct est non linéaire par rapport aux paramètres, la minimisation de $J(\boldsymbol{\chi})$ (3.2) est un problème de moindres carrés non-linéaires. L'estimation des paramètres peut être effectuée par différentes méthodes d'optimisation non-linéaire telles que la méthode du gradient, la méthode de Gauss-Newton, la méthode *SQP* (Sequential Quadratic

Programming) ou la méthode de Levenberg-Marquardt. Ces méthodes se basent sur un développement limité du critère $J(\chi)$ au premier ou deuxième ordre (Minoux, 1983).

Il est aussi possible d'utiliser des méthodes heuristiques telle que la méthode du Simplex de Nelder-Mead qui évite le calcul des dérivées du critère (Lagarias et al., 1998).

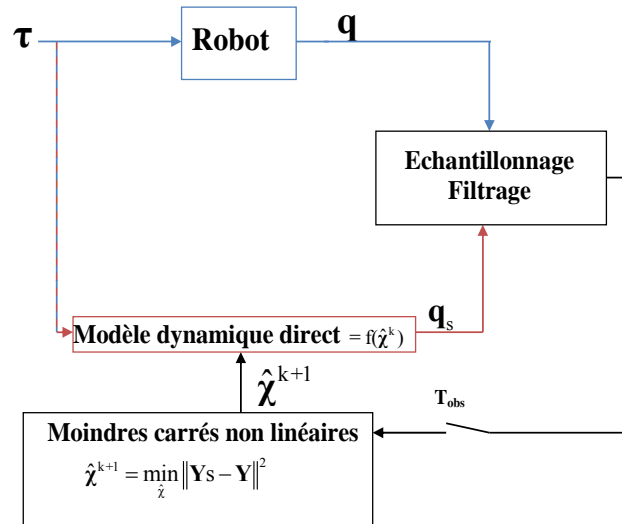


Figure 3-1 : Méthode à erreur de sortie en boucle ouverte

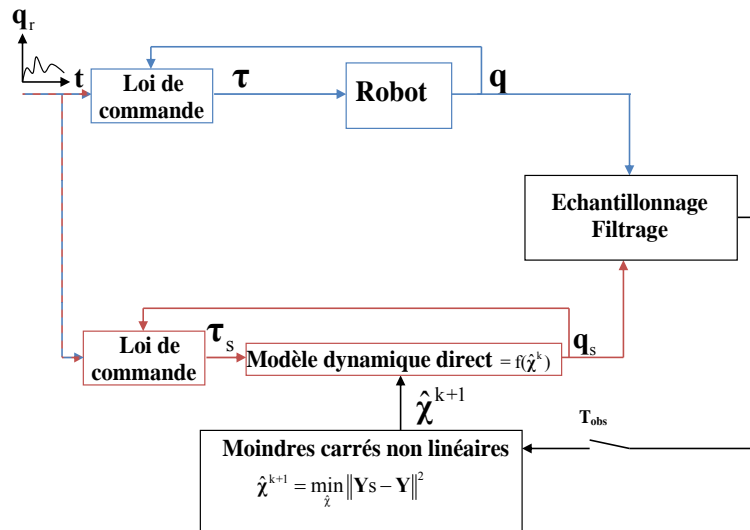


Figure 3-2 : Méthode à erreur de sortie en boucle fermée (*CLOE*)

La méthode de Gauss-Newton est adaptée à la résolution des moindres carrés non linéaires (Davidson et MacKinnon, 1993) (Janot, 2007) (Gautier et al., 2008) (Gautier et al., 2012).

En posant $\varepsilon = \mathbf{Y} - \mathbf{Y}_s$, (3.2) s'écrit :

$$J(\boldsymbol{\chi}) = \boldsymbol{\varepsilon}^T \boldsymbol{\varepsilon} \quad (3.3)$$

C'est un problème des moindres carrés non linéaires qui peut être résolu par la méthode de Newton du fait de sa convergence quadratique :

$$\hat{\boldsymbol{\chi}}^{k+1} = \hat{\boldsymbol{\chi}}^k - (\nabla^2 J(\hat{\boldsymbol{\chi}}^k))^{-1} \nabla J(\hat{\boldsymbol{\chi}}^k) \quad (3.4)$$

Dans ce cas, le gradient de $J(\boldsymbol{\chi})$ est donné par :

$$\nabla J(\boldsymbol{\chi}) = 2(\nabla \boldsymbol{\varepsilon})^T \boldsymbol{\varepsilon} \quad (3.5)$$

Et l'approximation de Gauss-Newton du Hessien de $J(\boldsymbol{\chi})$ s'écrit :

$$\nabla^2 J(\boldsymbol{\chi}) \approx 2(\nabla \boldsymbol{\varepsilon})^T \nabla \boldsymbol{\varepsilon} \quad (3.6)$$

Avec :

$$\nabla \boldsymbol{\varepsilon} = -\nabla \mathbf{Y}_s \quad (3.7)$$

$\nabla \mathbf{Y}_s$ est obtenu après échantillonnage et filtrage de $\nabla \mathbf{y}_s(\boldsymbol{\chi})$:

$$\nabla \mathbf{y}_s(\boldsymbol{\chi}) = \frac{\partial \mathbf{y}_s(\boldsymbol{\chi})}{\partial \boldsymbol{\chi}} = \boldsymbol{\delta}_{y_s/\boldsymbol{\chi}} \quad (3.8)$$

Ainsi dans la méthode de Gauss-Newton, le calcul des dérivées du critère, gradient et Hessien, nécessite le calcul des fonctions de sensibilité $\boldsymbol{\delta}_{y_s/\boldsymbol{\chi}} = (\partial \mathbf{y}_s(\boldsymbol{\chi}) / \partial \boldsymbol{\chi})$.

Cependant un calcul simple de la solution optimale est possible sans passer par les dérivées explicites du critère. C'est la régression de Gauss-Newton qui utilise directement un développement limité de \mathbf{y}_s autour de $\hat{\boldsymbol{\chi}}^k$ à l'itération k (Davidson et MacKinnon, 1993):

$$\mathbf{y}_s(\boldsymbol{\chi}^{k+1}) = \mathbf{y}_s(\hat{\boldsymbol{\chi}}^k) + \boldsymbol{\delta}_{y_s/\boldsymbol{\chi}}(\boldsymbol{\chi}^{k+1} - \hat{\boldsymbol{\chi}}^k) + \mathbf{o} \quad (3.9)$$

Où:

- $\boldsymbol{\delta}_{y_s/\boldsymbol{\chi}} = (\partial \mathbf{y}_s(\boldsymbol{\chi}) / \partial \boldsymbol{\chi})_{\hat{\boldsymbol{\chi}}^k}$ est la matrice jacobienne de \mathbf{y}_s par rapport à $\boldsymbol{\chi}$ estimée à l'itération k ,
- \mathbf{o} est le résidu de la série de Taylor.

Chaque coefficient de la matrice jacobienne définit une fonction de sensibilité. Chacune de ces fonctions de sensibilité définit une variation de \mathbf{y}_s par rapport à une variation d'un paramètre de $\boldsymbol{\chi}$. Les fonctions de sensibilité sont les solutions d'un système différentiel calculé à partir du modèle dynamique direct du système. Le calcul analytique de ce système est possible pour les systèmes linéaires (Richalet et al., 1971) (Janot, 2007) mais est très difficile pour les systèmes non linéaires tels que les robots.

La sortie simulée diffère de la sortie réelle par un vecteur d'erreur \mathbf{e} :

$$\mathbf{y} = \mathbf{y}_s(\boldsymbol{\chi}^{k+1}) + \mathbf{e} \quad (3.10)$$

A partir de (3.9), l'équation (3.10) devient:

$$\mathbf{y} - \mathbf{y}_s(\hat{\boldsymbol{\chi}}^k) = \boldsymbol{\delta}_{\mathbf{y}_s/\boldsymbol{\chi}}(\boldsymbol{\chi}^{k+1} - \hat{\boldsymbol{\chi}}^k) + (\mathbf{o} + \mathbf{e}) \quad (3.11)$$

Un système linéaire surdéterminé est obtenu par échantillonnage et filtrage de (3.11) sur une fenêtre d'observation T_{obs} :

$$\Delta \mathbf{Y} = \mathbf{W}_\delta \Delta \boldsymbol{\chi}^{k+1} + \boldsymbol{\rho} \quad (3.12)$$

Où:

- $\Delta \boldsymbol{\chi}^{k+1}$ est le vecteur (bx1) $\Delta \boldsymbol{\chi}^{k+1} = \boldsymbol{\chi}^{k+1} - \hat{\boldsymbol{\chi}}^k$,
- $\Delta \mathbf{Y}$ est le vecteur (rx1) $(\mathbf{y} - \mathbf{y}_s(\hat{\boldsymbol{\chi}}^k))$ après échantillonnage et filtrage,
- \mathbf{W}_δ est la matrice (rxb) $\boldsymbol{\delta}_{\mathbf{y}_s/\boldsymbol{\chi}}$ après échantillonnage et filtrage,
- $\boldsymbol{\rho}$ est le vecteur (rx1) $(\mathbf{o} + \mathbf{e})$ après échantillonnage et filtrage.

$\Delta \hat{\boldsymbol{\chi}}^{k+1}$ est la solution au sens des moindres carrés de (3.12):

$$\Delta \hat{\boldsymbol{\chi}}^{k+1} = (\mathbf{W}_\delta^T \mathbf{W}_\delta)^{-1} \mathbf{W}_\delta^T \Delta \mathbf{Y} \quad (3.13)$$

La solution à l'itération $k+1$ s'écrit à partir de l'équation (3.13) et de la solution à l'itération k :

$$\hat{\boldsymbol{\chi}}^{k+1} = \hat{\boldsymbol{\chi}}^k + \Delta \hat{\boldsymbol{\chi}}^{k+1} \quad (3.14)$$

Dans le cas où les expressions analytiques des fonctions de sensibilité ne sont pas connues, les méthodes classiques de programmation non linéaire calculent numériquement les dérivées du critère par différence finie, ce qui nécessite un grand nombre de calculs du critère par simulation de la sortie. Par exemple avec la méthode *SQP* (Sequential Quadratic Programming) utilisée dans la fonction 'fmincon' de Matlab, pour un robot avec 61 paramètres à identifier, il faut 123 simulations

pour une itération de l'algorithme *SQP*. Il en résulte qu'en l'absence de connaissance des fonctions de sensibilité, les méthodes à erreur de sortie sont très consommatrices de temps de calcul pour les systèmes avec de nombreux paramètres. De plus la convergence vers des minima locaux multiples est très sensible aux conditions initiales (Walter et Pronzato, 1997).

Dans la suite, nous montrons comment la méthode *DIDIM* a résolu ces problèmes de programmation non linéaire dans le cas des robots rigides.

3.3 Méthode *DIDIM*

3.3.1 Théorie

Dans la méthode *CLOE* classique dont le schéma est représenté figure 3-2, la sortie est la position $\mathbf{y}=\mathbf{q}$, $\mathbf{y}_s=\mathbf{q}_s$.

Dans la suite, on remplace les positions \mathbf{q} et \mathbf{q}_s par les efforts moteurs $\boldsymbol{\tau}$ et les efforts moteurs simulés $\boldsymbol{\tau}_s$.

Donc nous prenons $\mathbf{y}=\boldsymbol{\tau}$ et $\mathbf{y}_s=\boldsymbol{\tau}_s$ (voir figure 3-3).

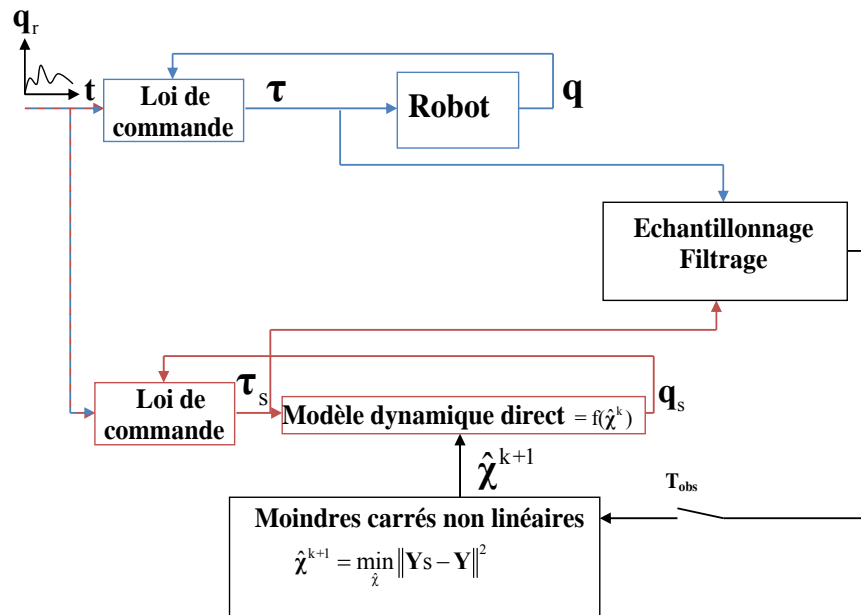


Figure 3-3 : Méthode *DIDIM*

La solution optimale $\hat{\chi}$ minimise le critère quadratique $J(\chi)$ (3.2) où \mathbf{Y} et \mathbf{Y}_s sont respectivement des vecteurs obtenus après échantillonnage et filtrage des efforts moteurs.

Ce problème de moindres carrés non-linéaires est résolu par une régression de Gauss-Newton en utilisant (3.11). Les fonctions de sensibilité sont calculées en exprimant $\boldsymbol{\tau}_s$ avec le modèle dynamique inverse (1.16):

$$\begin{aligned} \mathbf{y}_s(\boldsymbol{\chi}) &= \boldsymbol{\tau}_s(\boldsymbol{\chi}) = \boldsymbol{\tau}_{\text{mdi}}(\boldsymbol{\chi}) \\ &= \text{MDI}(\mathbf{q}_s(\boldsymbol{\chi}), \dot{\mathbf{q}}_s(\boldsymbol{\chi}), \ddot{\mathbf{q}}_s(\boldsymbol{\chi}))\boldsymbol{\chi} \end{aligned} \quad (3.15)$$

Il vient :

$$\begin{aligned}\delta_{y_s/\chi} &= \left(\frac{\partial \boldsymbol{\tau}_s}{\partial \boldsymbol{\chi}} \right)_{\hat{\boldsymbol{\chi}}^k} = \left(\frac{\partial \boldsymbol{\tau}_{\text{mdi}}}{\partial \boldsymbol{\chi}} \right)_{\hat{\boldsymbol{\chi}}^k} \\ &= \frac{\partial}{\partial \boldsymbol{\chi}} \left(\mathbf{MDI}(\mathbf{q}_s(\hat{\boldsymbol{\chi}}^k), \dot{\mathbf{q}}_s(\hat{\boldsymbol{\chi}}^k), \ddot{\mathbf{q}}_s(\hat{\boldsymbol{\chi}}^k)) \right) \hat{\boldsymbol{\chi}}^k\end{aligned}\quad (3.16)$$

avec :

$$\begin{aligned}&\frac{\partial}{\partial \boldsymbol{\chi}} \left(\mathbf{MDI}(\mathbf{q}_s(\hat{\boldsymbol{\chi}}^k), \dot{\mathbf{q}}_s(\hat{\boldsymbol{\chi}}^k), \ddot{\mathbf{q}}_s(\hat{\boldsymbol{\chi}}^k)) \right) \hat{\boldsymbol{\chi}}^k \\ &= \mathbf{MDI}(\mathbf{q}_s(\hat{\boldsymbol{\chi}}^k), \dot{\mathbf{q}}_s(\hat{\boldsymbol{\chi}}^k), \ddot{\mathbf{q}}_s(\hat{\boldsymbol{\chi}}^k)) + \frac{\partial}{\partial \boldsymbol{\chi}} \left(\mathbf{MDI}(\mathbf{q}_s(\hat{\boldsymbol{\chi}}^k), \dot{\mathbf{q}}_s(\hat{\boldsymbol{\chi}}^k), \ddot{\mathbf{q}}_s(\hat{\boldsymbol{\chi}}^k)) \right) \hat{\boldsymbol{\chi}}^k\end{aligned}\quad (3.17)$$

Le calcul du second terme de (3.17) nécessite de calculer l'expression suivante :

$$\begin{aligned}&\frac{\partial}{\partial \boldsymbol{\chi}} \left(\mathbf{MDI}(\mathbf{q}_s(\hat{\boldsymbol{\chi}}^k), \dot{\mathbf{q}}_s(\hat{\boldsymbol{\chi}}^k), \ddot{\mathbf{q}}_s(\hat{\boldsymbol{\chi}}^k)) \right) \\ &= \frac{\partial}{\partial \mathbf{q}_s} \left(\mathbf{MDI}(\mathbf{q}_s(\hat{\boldsymbol{\chi}}^k), \dot{\mathbf{q}}_s(\hat{\boldsymbol{\chi}}^k), \ddot{\mathbf{q}}_s(\hat{\boldsymbol{\chi}}^k)) \right) \frac{\partial \mathbf{q}_s}{\partial \boldsymbol{\chi}} + \dots \\ &\quad + \frac{\partial}{\partial \dot{\mathbf{q}}_s} \left(\mathbf{MDI}(\mathbf{q}_s(\hat{\boldsymbol{\chi}}^k), \dot{\mathbf{q}}_s(\hat{\boldsymbol{\chi}}^k), \ddot{\mathbf{q}}_s(\hat{\boldsymbol{\chi}}^k)) \right) \frac{\partial \dot{\mathbf{q}}_s}{\partial \boldsymbol{\chi}} + \dots \\ &\quad + \frac{\partial}{\partial \ddot{\mathbf{q}}_s} \left(\mathbf{MDI}(\mathbf{q}_s(\hat{\boldsymbol{\chi}}^k), \dot{\mathbf{q}}_s(\hat{\boldsymbol{\chi}}^k), \ddot{\mathbf{q}}_s(\hat{\boldsymbol{\chi}}^k)) \right) \frac{\partial \ddot{\mathbf{q}}_s}{\partial \boldsymbol{\chi}}\end{aligned}\quad (3.18)$$

L'effort moteur $\mathbf{y}=\boldsymbol{\tau}$ est enregistré pendant que le robot poursuit une trajectoire de référence $(\mathbf{q}_r, \dot{\mathbf{q}}_r, \ddot{\mathbf{q}}_r)$ en boucle fermée. La simulation en boucle fermée utilise le modèle dynamique direct, la même loi de commande et la même trajectoire de référence $(\mathbf{q}_r, \dot{\mathbf{q}}_r, \ddot{\mathbf{q}}_r)$ que le robot réel pour calculer l'effort moteur simulé $\boldsymbol{\tau}_s$.

Dans la partie suivante (3.3.2), on montre que le simulateur conserve des performances en boucle fermée très proches de celles du robot réel pour tout $\hat{\boldsymbol{\chi}}^k$ obtenu à l'itération k .

Ainsi l'erreur de poursuite simulée reste proche de l'erreur de poursuite réelle pour tout $\hat{\boldsymbol{\chi}}^k$. On peut donc écrire :

$$\left(\mathbf{q}_s(\hat{\boldsymbol{\chi}}^k), \dot{\mathbf{q}}_s(\hat{\boldsymbol{\chi}}^k), \ddot{\mathbf{q}}_s(\hat{\boldsymbol{\chi}}^k) \right) \simeq (\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) \text{ pour tout } \hat{\boldsymbol{\chi}}^k \quad (3.19)$$

Cela veut dire que les positions, vitesses et accélérations simulées $(\mathbf{q}_s(\hat{\boldsymbol{\chi}}^k), \dot{\mathbf{q}}_s(\hat{\boldsymbol{\chi}}^k), \ddot{\mathbf{q}}_s(\hat{\boldsymbol{\chi}}^k))$ dépendent peu de $\boldsymbol{\chi}$ de telle sorte que :

$$\frac{\partial \mathbf{q}_s}{\partial \boldsymbol{\chi}} \approx \frac{\partial \dot{\mathbf{q}}_s}{\partial \boldsymbol{\chi}} \approx \frac{\partial \ddot{\mathbf{q}}_s}{\partial \boldsymbol{\chi}} \approx 0 \quad (3.20)$$

L'équation (3.18) se simplifie :

$$\frac{\partial}{\partial \boldsymbol{\chi}} \left(\mathbf{MDI}(\mathbf{q}_s(\hat{\boldsymbol{\chi}}^k), \dot{\mathbf{q}}_s(\hat{\boldsymbol{\chi}}^k), \ddot{\mathbf{q}}_s(\hat{\boldsymbol{\chi}}^k)) \right) \approx 0 \quad (3.21)$$

En prenant en compte l'approximation (3.21) dans l'équation (3.17), on obtient l'approximation suivante de la matrice Jacobienne :

$$\frac{\partial}{\partial \boldsymbol{\chi}} \left(\mathbf{MDI}(\mathbf{q}_s(\hat{\boldsymbol{\chi}}^k), \dot{\mathbf{q}}_s(\hat{\boldsymbol{\chi}}^k), \ddot{\mathbf{q}}_s(\hat{\boldsymbol{\chi}}^k)) \hat{\boldsymbol{\chi}}^k \right) \approx \mathbf{MDI}(\mathbf{q}_s(\hat{\boldsymbol{\chi}}^k), \dot{\mathbf{q}}_s(\hat{\boldsymbol{\chi}}^k), \ddot{\mathbf{q}}_s(\hat{\boldsymbol{\chi}}^k)) \quad (3.22)$$

La simulation en boucle fermée réduit fortement la variation des positions, vitesses et accélérations simulées par rapport à la variation des paramètres selon l'équation (3.19). Par contre les variations des efforts simulés (3.15) sont très sensibles aux variations des paramètres (voir partie 3.8).

Ceci montre l'apport majeur de la méthode *DIDIM* qui est d'utiliser comme sortie les efforts moteurs à la place des positions pour l'identification en boucle fermée.

L'autre apport majeur est l'utilisation du modèle dynamique inverse pour approcher les fonctions de sensibilité de la matrice Jacobienne par l'équation (3.22). Ceci est plus simple que pour les méthodes à erreur de sortie usuelles où les fonctions de sensibilité sont des solutions d'équations différentielles compliquées à calculer pour les systèmes non linéaires.

En prenant l'approximation (3.22) de la matrice Jacobienne pour le développement de la série de Taylor (3.11), nous avons l'équation suivante :

$$\mathbf{y} = \boldsymbol{\tau} = \mathbf{y}_s(\hat{\boldsymbol{\chi}}^k) + \mathbf{MDI}(\mathbf{q}_s(\hat{\boldsymbol{\chi}}^k), \dot{\mathbf{q}}_s(\hat{\boldsymbol{\chi}}^k), \ddot{\mathbf{q}}_s(\hat{\boldsymbol{\chi}}^k)) (\boldsymbol{\chi}^{k+1} - \hat{\boldsymbol{\chi}}^k) + (\mathbf{o} + \mathbf{e}) \quad (3.23)$$

En remarquant que (3.15) s'écrit :

$$\mathbf{y}_s(\hat{\boldsymbol{\chi}}^k) = \boldsymbol{\tau}_{\text{idm}}(\hat{\boldsymbol{\chi}}^k) = \mathbf{MDI}(\mathbf{q}_s(\hat{\boldsymbol{\chi}}^k), \dot{\mathbf{q}}_s(\hat{\boldsymbol{\chi}}^k), \ddot{\mathbf{q}}_s(\hat{\boldsymbol{\chi}}^k)) \hat{\boldsymbol{\chi}}^k \quad (3.24)$$

L'équation (3.23) se simplifie suivant l'expression :

$$\mathbf{y} = \boldsymbol{\tau} = \mathbf{MDI}(\mathbf{q}_s(\hat{\boldsymbol{\chi}}^k), \dot{\mathbf{q}}_s(\hat{\boldsymbol{\chi}}^k), \ddot{\mathbf{q}}_s(\hat{\boldsymbol{\chi}}^k)) \boldsymbol{\chi}^{k+1} + (\mathbf{o} + \mathbf{e}) \quad (3.25)$$

C'est le modèle dynamique inverse d'identification (2.2) dans lequel $(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$ sont estimés avec $(\mathbf{q}_s, \dot{\mathbf{q}}_s, \ddot{\mathbf{q}}_s)$, simulés avec le modèle dynamique direct (3.1). A chaque itération k , la méthode *IDIM-LS* est appliquée suivant la procédure décrite au second chapitre.

En reprenant les notations du chapitre 2, (2.6) à (2.9), l'échantillonnage de (3.25) à la fréquence de mesure f_m donne le système surdéterminé suivant :

$$\mathbf{Y}_{f_m}(\boldsymbol{\tau}) = \mathbf{W}_{\delta f_m}(\mathbf{q}_s, \dot{\mathbf{q}}_s, \ddot{\mathbf{q}}_s, \hat{\boldsymbol{\chi}}^k) \boldsymbol{\chi} + \boldsymbol{\rho}_{f_m} \quad (3.26)$$

avec $\mathbf{W}_{\delta f_m}$ la matrice ($n_{f_m} \times b$) :

$$\mathbf{W}_{\delta f_m}(\mathbf{q}_s, \dot{\mathbf{q}}_s, \ddot{\mathbf{q}}_s, \hat{\boldsymbol{\chi}}^k) = \begin{bmatrix} \mathbf{W}_{\delta f_m}^1 \\ \mathbf{W}_{\delta f_m}^2 \\ \vdots \\ \mathbf{W}_{\delta f_m}^n \end{bmatrix}, \mathbf{W}_{\delta f_m}^j = \begin{bmatrix} \mathbf{MDI}^j(\mathbf{q}_s(t_1), \dot{\mathbf{q}}_s(t_1), \ddot{\mathbf{q}}_s(t_1), \hat{\boldsymbol{\chi}}^k) \\ \mathbf{MDI}^j(\mathbf{q}_s(t_2), \dot{\mathbf{q}}_s(t_2), \ddot{\mathbf{q}}_s(t_2), \hat{\boldsymbol{\chi}}^k) \\ \vdots \\ \mathbf{MDI}^j(\mathbf{q}_s(t_{n_e}), \dot{\mathbf{q}}_s(t_{n_e}), \ddot{\mathbf{q}}_s(t_{n_e}), \hat{\boldsymbol{\chi}}^k) \end{bmatrix} \quad (3.27)$$

Le filtrage parallèle de (3.26) donne :

$$\mathbf{Y} = \mathbf{W}(\mathbf{q}_s, \dot{\mathbf{q}}_s, \ddot{\mathbf{q}}_s, \hat{\boldsymbol{\chi}}^k) \boldsymbol{\chi} + \boldsymbol{\rho} \quad (3.28)$$

La solution au sens des moindres carrés de (3.28) donne $\hat{\boldsymbol{\chi}}^{k+1}$, à l'itération k .

Deux critères d'arrêt sont proposés. Le premier est un critère calculé sur la variation relative de la norme du résidu entre les itérations k et $k+1$:

$$\frac{\|\boldsymbol{\rho}_{k+1}\| - \|\boldsymbol{\rho}_k\|}{\|\boldsymbol{\rho}_k\|} \leq \text{tol}_1 \quad (3.29)$$

Le second critère d'arrêt utilise la variation relative maximale des paramètres entre les itérations k et $k+1$:

$$\max_{i=1,2,\dots,b} \left| \frac{\hat{\chi}_i^{k+1} - \hat{\chi}_i^k}{\hat{\chi}_i^k} \right| \leq \text{tol}_2 \quad (3.30)$$

Les seuils de convergence tol_1 et tol_2 sont choisis suivant un compromis entre le niveau de bruits de $\boldsymbol{\tau}$, la rapidité et la précision de convergence, généralement inférieurs à 1% (Gautier et al. 2008) (Gautier et al., 2012).

La méthode *DIDIM* est basée sur une simulation en boucle fermée utilisant le modèle dynamique direct. Les paramètres optimaux minimisent l'erreur quadratique entre les efforts moteurs réels et les efforts moteurs simulés durant un temps d'observation T_{obs} . L'utilisation du modèle dynamique inverse dans la méthode de Gauss-Newton résout les problèmes liés aux algorithmes de programmation non-linéaire utilisés dans les méthodes à erreur de sortie. La méthode est appelée *DIDIM* pour l'acronyme anglais Direct and Inverse Dynamic Identification

Models, car elle utilise les modèles dynamiques direct et inverse dans une procédure globale et unique d'identification.

Les avantages de la méthode *DIDIM* (figure 3-4) sont les suivants :

- elle ne nécessite que la mesure des efforts moteurs et des consignes (ou la connaissance du générateur de trajectoire),
- les vitesses et accélérations articulaires sont calculées par simulation et ne sont pas sensibles au réglage des filtres passe-bande utilisés dans la méthode *IDIM-LS*,
- elle utilise le modèle dynamique inverse et direct et permet de valider dans une même procédure d'identification les deux modèles pour la simulation et la commande dynamique,
- elle simplifie fortement le calcul des fonctions de sensibilité qui sont approchées par des équations algébriques. Le modèle dynamique inverse d'identification remplace la résolution d'un système compliqué d'équations différentielles (quand il existe) dans les méthodes à erreur de sortie classiques.

L'inconvénient de cette méthode est qu'elle nécessite la connaissance précise de la loi de commande implémentée dans le contrôleur du robot réel afin de simuler le robot en boucle fermée. Une erreur entre la commande réelle et la commande simulée entraîne un biais sur l'identification des paramètres en boucle fermée.

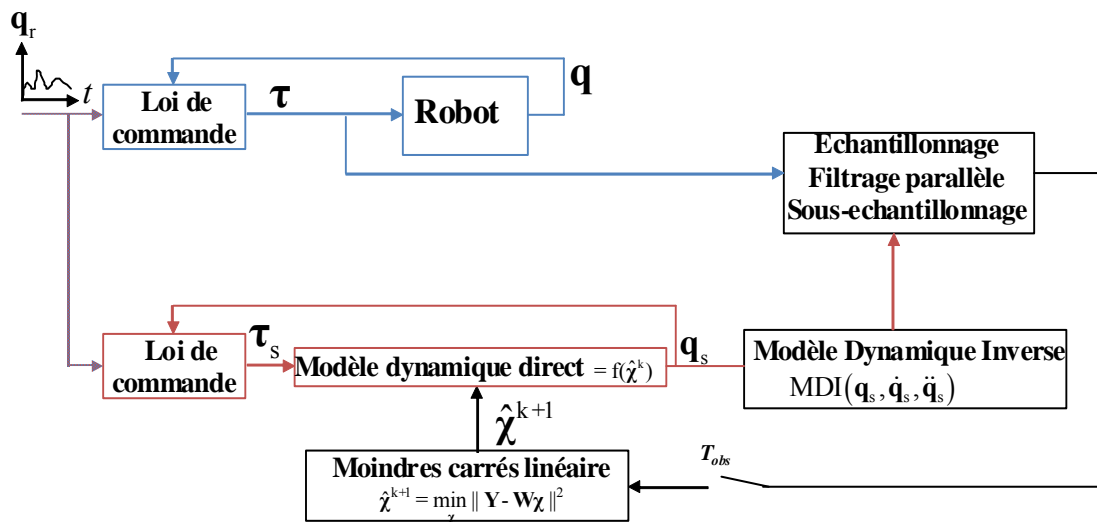


Figure 3-4 : Méthode DIDIM (Direct and Inverse Dynamic Identification Models)

3.3.2 Initialisation de l'algorithme

Pour choisir le vecteur des paramètres initiaux $\hat{\chi}^0$, il est possible d'utiliser des valeurs *CAO* ou des valeurs identifiées avec la méthode *IDIM-LS*. Cependant nous allons voir qu'il n'est pas nécessaire de connaître toutes les valeurs *a priori*.

Un algorithme peu sensible aux conditions initiales a été proposé dans (Janot, 2007) (Gautier et al. 2008) et (Gautier et al. 2012). Il permet de respecter la condition

$(\mathbf{q}_s(\hat{\chi}^k), \dot{\mathbf{q}}_s(\hat{\chi}^k), \ddot{\mathbf{q}}_s(\hat{\chi}^k)) \approx (\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$ pour toute itération k , et en particulier pour l'itération $k=0$ des conditions initiales.

Cela est possible en ayant les mêmes performances en boucle fermée en termes de bande passante et de marge de stabilité entre le robot réel et le robot simulé.

Le modèle dynamique inverse (1.7) de l'articulation j peut être écrit sous la forme d'un double intégrateur découplé avec un terme de perturbation (couplage et frottements):

$$\begin{aligned}\tau_j &= \tau_{\text{idm}_j} = \sum_{i=1}^n M_{j,i}(\mathbf{q}) \ddot{q}_i + H_j(\mathbf{q}, \dot{\mathbf{q}}) \\ &= M_{j,j}(\mathbf{q}) \ddot{q}_j + \sum_{i \neq j}^n M_{j,i}(\mathbf{q}) \ddot{q}_i + H_j(\mathbf{q}, \dot{\mathbf{q}}) \\ &= M_{j,j}(\mathbf{q}) \ddot{q}_j - p_j\end{aligned}\quad (3.31)$$

où p_j est considéré comme un terme de perturbation donné par :

$$p_j = - \sum_{i \neq j}^n M_{j,i}(\mathbf{q}) \ddot{q}_i - H_j(\mathbf{q}, \dot{\mathbf{q}}) \quad (3.32)$$

$M_{j,i}(\mathbf{q})$ qui dépend de \mathbf{q} est approché par un moment d'inertie constant donné par :

$$M_{I_j} = \max_{\mathbf{q}} (M_{j,j}(\mathbf{q})) \quad (3.33)$$

M_{I_j} est idéalement la valeur maximale, par rapport \mathbf{q} , du moment d'inertie autour de l'axe z_j qui garantit le maximum de marge de stabilité lorsque $M_{j,j}(\mathbf{q})$ varie.

La valeur M_{I_j} peut être calculée à partir de valeurs *CAO* des paramètres inertiels. Pour les robots industriels, équipés de réducteur mécaniques ayant un coefficient de réduction élevé ($N_{rj} > 50$), l'inertie du rotor de l'actionneur I_{rj} ramenée côté corps est largement dominante dans $M_{j,j}(\mathbf{q})$. On peut donc choisir M_{I_j} égal à l'inertie du rotor exprimée côté corps :

$${}^{\text{ap}} M_{I_j} = I_{rj} N_{rj}^2 \quad (3.34)$$

Où :

- I_{rj} est l'inertie du rotor de l'actionneur (exprimée côté moteur) de l'articulation j . Cette valeur est disponible dans les fiches de caractéristiques des moteurs.
- N_{rj} est le coefficient de réduction de la transmission mécanique de l'articulation j .

En considérant p_j comme une perturbation, le modèle non linéaire du robot peut s'écrire comme un système de n doubles intégrateurs découplés. Pour chaque articulation j , on obtient :

$$\ddot{q}_j = \frac{1}{M_{j,j}(\mathbf{q})}(\tau_j + p_j) \approx \frac{1}{M_{Ij}}(\tau_j + p_j) \quad (3.35)$$

En conséquence, il est possible d'utiliser les techniques de contrôle linéaire pour régler la boucle-fermée de chaque double-intégrateur j .

Considérons une commande en position de l'articulation j , représentée figure 3-5 avec ${}^a M_{Ij}$ la valeur réelle de M_{Ij} utilisée pour le réglage du correcteur $C(s)$:

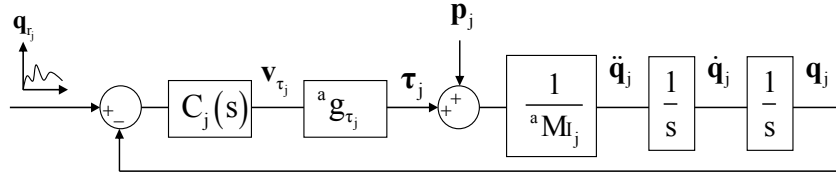


Figure 3-5 : Commande en position articulaire, robot réel

La référence de courant calculée par le contrôleur du robot est donnée par :

$$v_{\tau_j} = C_j(s)(q_r - q_j) \quad (3.36)$$

où

- v_{τ_j} est la référence de courant de l'amplificateur de courant du moteur j (2.24),
- $C_j(s)$ est le correcteur de l'articulation j de type *PD* ou *PID*,
- ${}^a g_{\tau_j}$ est le gain d'actionnement réel.

Le même correcteur $C_j(s)$ est utilisé en simulation suivant la figure 3-6 dans laquelle la valeur réelle de ${}^a g_{\tau_j}$ inconnue est remplacée par sa valeur *a priori* ${}^{ap} g_{\tau_j}$.

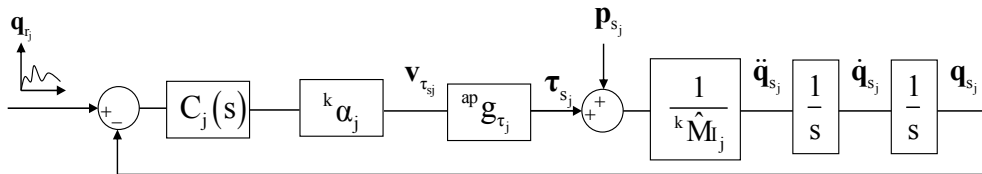


Figure 3-6 : Commande en position articulaire, robot simulé

Les variables $(v_{\tau_{sj}}, \tau_{sj}, q_{sj}, \dot{q}_{sj}, \ddot{q}_{sj})$ de la figure 3-6 sont calculées par intégration numérique du modèle dynamique direct (3.1) qui dépend de $\hat{\chi}^k$.

- ${}^{ap} g_{\tau_j}$ est la valeur *a priori* de ${}^a g_{\tau_j}$,

- ${}^k\hat{M}_{I_j}$ est la valeur de M_{I_j} (3.33) calculée avec l'estimation $\hat{\chi}^k$ à l'itération k ,
- ${}^k\alpha_j$ est un coefficient à déterminer.

Pour obtenir les mêmes performances en boucle fermée en termes de bande passante et de marge de stabilité, les fonctions de transfert en boucle ouverte des deux schémas (figure 3-5 et figure 3-6) doivent être égales :

$$\begin{aligned} {}^aTbo(s) &= C_j(s) \frac{{}^a g_{\tau_j}}{{}^a M_{I_j}} \frac{1}{s^2} \\ {}^kTbo(s) &= C_j(s) {}^k\alpha_j \frac{{}^{ap} g_{\tau_j}}{{}^k\hat{M}_{I_j}} \frac{1}{s^2} \end{aligned} \quad (3.37)$$

Où:

- ${}^aTbo(s)$ est la fonction de transfert en boucle ouverte associée à l'articulation j du robot réel (figure 3-5),
- ${}^kTbo(s)$ est la fonction de transfert en boucle ouverte associée à l'articulation j du robot simulé (figure 3-6).

Il en résulte la relation suivante :

$${}^a g_{\tau_j} \frac{1}{{}^a M_{I_j}} = {}^k\alpha_j \frac{{}^{ap} g_{\tau_j}}{{}^k\hat{M}_{I_j}} \quad (3.38)$$

Soit :

$${}^k\alpha_j = \frac{{}^a g_{\tau_j}}{{}^{ap} g_{\tau_j}} \frac{{}^k\hat{M}_{I_j}}{{}^a M_{I_j}} \quad (3.39)$$

Les valeurs inconnues de ${}^a g_{\tau_j}$ et ${}^a M_{I_j}$ sont remplacées par leurs valeurs *a priori* ${}^{ap} g_{\tau_j}$ et ${}^{ap} M_{I_j}$. Sous condition que les valeurs *a priori* de ${}^{ap} g_{\tau_j}$ et ${}^{ap} M_{I_j}$ soient proches des valeurs réelles, on obtient une valeur approchée de ${}^k\alpha_j$:

$${}^k\alpha_j = \frac{{}^k\hat{M}_{I_j}}{{}^{ap} M_{I_j}} \quad (3.40)$$

Le coefficient ${}^k\alpha_j$ varie donc à chaque itération k selon ${}^k\hat{M}_{I_j}(\hat{\chi}^k)$, avec ${}^{ap} M_{I_j} = I_{r_j} N_{r_j}^2$. Dans (Janot, 2007), le même résultat est utilisé pour identifier un robot à 2 degrés de liberté de type

SCARA avec la méthode *DIDIM*. Cependant les moteurs étant à entraînements directs ($N_{rj} = 1$), ${}^{ap}M_{I_j}$ est pris égal à ${}^{ap}ZZ_{R_j}$ (voir équation (3.44)).

(Gautier et al., 2012) ont proposé de prendre une initialisation $\hat{\chi}^0$ qui définit une matrice d'inertie régulière définie positive $\mathbf{M}(\mathbf{q}_s, \hat{\chi}^0)$ pour avoir une bonne initialisation pour l'intégration numérique du modèle dynamique direct (3.1). Elle est appelée "initialisation régulière".

Elle est obtenue avec :

$$\hat{\chi}^0 = 0 \text{ sauf pour } Ia_j^0 = 1 \text{ pour } j = 1, 2, \dots, n \quad (3.41)$$

Le moment d'inertie de la chaîne d'actionnement est pris en compte dans le modèle dynamique inverse (1.7) sous la forme :

$$\tau_{r_j} = Ia_j \ddot{q}_j \quad (3.42)$$

En conséquence, la matrice d'inertie à l'itération $k=0$ devient la matrice identité, qui est la meilleure matrice régulière :

$$\mathbf{M}(\mathbf{q}_s, \hat{\chi}^0) = \mathbf{I}_n \quad (3.43)$$

Pour les robots à entraînement direct ou avec des rapports de réduction $N_{rj}^2 < 10$, l'initialisation suivante est proposée :

$$\hat{\chi}^0 = 0 \text{ sauf pour } ZZ_j^0 = 1 \text{ pour } j = 1, 2, \dots, n \quad (3.44)$$

La matrice d'inertie à l'itération $k=0$ n'est plus la matrice identité mais elle reste régulière et définie positive.

Un autre point est de choisir les états initiaux du vecteur d'état $(\mathbf{q}_s(0), \dot{\mathbf{q}}_s(0))$ pour l'intégration du modèle dynamique direct (3.1). Comme la méthode *DIDIM* ne nécessite pas les mesures des positions, les valeurs des positions mesurées à $t=0$ $(\mathbf{q}(0), \dot{\mathbf{q}}(0))$ sont supposées inconnues. En conséquence les états initiaux $(\mathbf{q}_s(0), \dot{\mathbf{q}}_s(0))$ sont choisis égaux aux valeurs de consigne calculées par le générateur de mouvement $(\mathbf{q}_r(0), \dot{\mathbf{q}}_r(0))$ qui sont proches de $(\mathbf{q}(0), \dot{\mathbf{q}}(0))$. La réponse transitoire simulée de la boucle fermée est différente de celle du robot réel à cause des conditions initiales différentes sur une durée transitoire de l'ordre $5/\omega_{nj}$. Les échantillons correspondants sont éliminés des données d'identification dans (3.26).

3.3.3 Structure de l'algorithme

L'algorithme *DIDIM* peut être résumé comme suit :

Etape 0: initialisation de l'algorithme avec l'initialisation régulière (3.41) ou (3.44).
 tant que $\left(\left(\frac{\|\rho_{k+1}\| - \|\rho_k\|}{\|\rho_k\|} > \text{tol}_1 \right) \text{ et } \left(\max_{i=1,2,\dots,b} \left| \frac{\hat{\chi}_i^{k+1} - \hat{\chi}_i^k}{\hat{\chi}_i^k} \right| > \text{tol}_2 \right) \right)$
 - étape k+1 : mise à jours des $n^k \alpha_j$ et exécution de la simulation avec $\hat{\chi}^k$
 - exécution de l'identification avec la méthode *IDIM-LS* et la matrice d'observation (3.27) après filtrage parallèle,
 - calcul de la solution au sens des moindres carrés de (3.28) qui donne $\hat{\chi}^{k+1}$.
 fin tant que

3.4 Identification des paramètres inertiels d'une charge

Dans ce paragraphe, je propose d'adapter la méthode décrite à la partie (2.10) et dans (Khalil et al. 2007) à la méthode *DIDIM*.

Le système surdéterminé (3.28) devient :

$$\mathbf{Y} = \begin{bmatrix} \mathbf{Y}_a \\ \mathbf{Y}_b \end{bmatrix} = \begin{bmatrix} \mathbf{W}_a(\mathbf{q}_{s_a}, \dot{\mathbf{q}}_{s_a}, \ddot{\mathbf{q}}_{s_a}, \hat{\chi}^k) & 0 \\ \mathbf{W}_b(\mathbf{q}_{s_b}, \dot{\mathbf{q}}_{s_b}, \ddot{\mathbf{q}}_{s_b}, \hat{\chi}^k, \hat{\chi}_L^k) & \mathbf{W}_L(\mathbf{q}_{s_b}, \dot{\mathbf{q}}_{s_b}, \ddot{\mathbf{q}}_{s_b}, \hat{\chi}^k, \hat{\chi}_L^k) \end{bmatrix} \begin{bmatrix} \chi \\ \chi_L \end{bmatrix} + \rho \quad (3.45)$$

Où :

- \mathbf{W}_a est la matrice d'observation du robot construite avec les positions, vitesses et accélérations simulées $(\mathbf{q}_{s_a}, \dot{\mathbf{q}}_{s_a}, \ddot{\mathbf{q}}_{s_a})$ avec les paramètres du robot fixés à $\hat{\chi}^k$, les 10 paramètres de la charge fixés à 0 et la même trajectoire utilisée sur le robot sans charge,
- \mathbf{W}_b et \mathbf{W}_L sont les matrices d'observation du robot construites avec les positions, vitesses et accélérations simulées $(\mathbf{q}_{s_b}, \dot{\mathbf{q}}_{s_b}, \ddot{\mathbf{q}}_{s_b})$ avec les paramètres du robot fixés à $\hat{\chi}^k$, les paramètres de la charge fixé à $\hat{\chi}_L^k$ et la même trajectoire utilisée sur le robot avec charge.

Il est nécessaire d'effectuer deux simulations du robot par itération de l'algorithme, une avec les paramètres de la charge fixés à 0 et une avec les paramètres de la charge fixés à $\hat{\chi}_L^k$.

Pour les robots industriels équipés de réducteurs à coefficient de réduction élevé ($N_{ij} > 50$), on conserve l'initialisation de *DIDIM* avec ${}^{ap}M_{ij} = I_{a_{ij}} N_{ij}^2$ qui est le terme prépondérant de $M_{j,j}(\mathbf{q})$ quelle que soit la charge du robot.

3.5 Application au robot Stäubli *TX40*

Cette partie traite de l'identification expérimentale de la méthode *DIDIM* sur le robot Stäubli *TX40* avec une charge de 2.370 (kg) (± 5 (g)). A des fins de comparaison, une identification avec la méthode *IDIM-LS* est effectuée avant d'appliquer la méthode *DIDIM* présentée auparavant.

La fréquence de coupure du filtre Butterworth est réglée à 50 (Hz) et la fréquence de coupure du filtre Decimate est réglée à 10 (Hz). Afin d'identifier la charge, le robot effectue des trajectoires sans charge et des trajectoires avec la charge fixée sur l'effecteur.

Les mouvements excitants sont obtenus avec le générateur de mouvement point à point à vitesse nulle et interpolateur trapézoïdal en accélération implanté dans le contrôleur CS8C du robot. En balayant tout l'espace de travail sur une douzaine de points avec des vitesses et accélérations élevées, il est possible d'obtenir un bon critère d'excitation f (voir partie 2.7 et équation (2.27)) sans recourir à l'optimisation de trajectoire (Khalil et al. 2007).

La fréquence d'échantillonnage des mesures est de 5 (kHz). Les consignes articulaires sont représentées figure 3-7, figure 3-8 et figure 3-9 et les couples moteurs sans charge (ramené côté articulaire) sont représentés figure 3-10. Un zoom sur la consigne d'accélération de l'axe 1 est représenté figure 3-11.

Les résultats de l'identification avec la méthode *IDIM-LS* sont donnés dans le tableau 3-1. Les paramètres essentiels sont calculés avec la méthode décrite dans la partie (2.8). Pour des raisons de lisibilité, $[\hat{\chi}^T \quad \hat{\chi}_L^T]^T$ s'écrit $\hat{\chi}$ dans tous les résultats d'identification présentés par la suite.

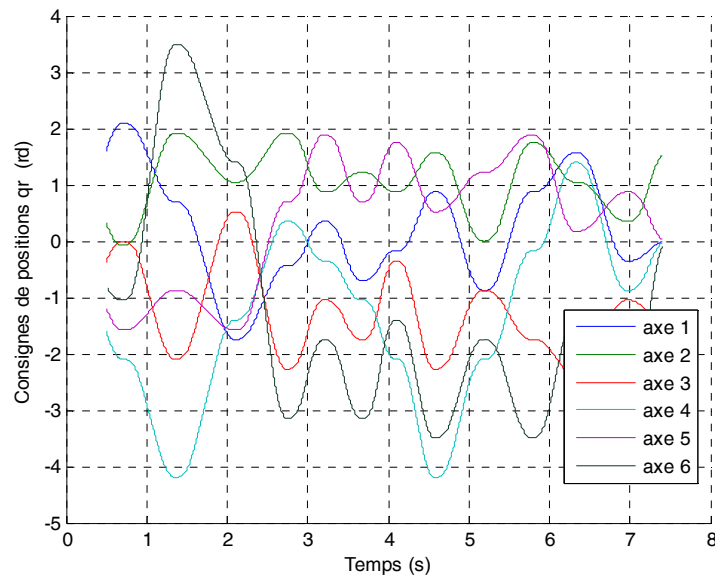


Figure 3-7 : Stäubli *TX40*, Consignes de positions articulaires

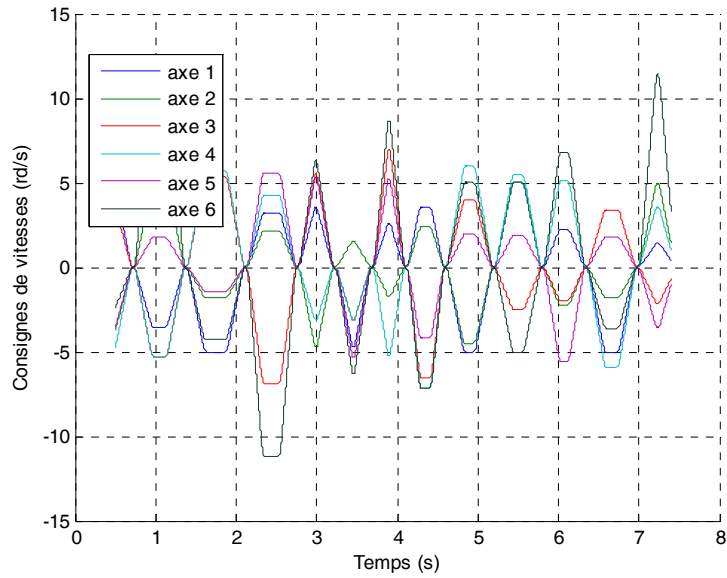


Figure 3-8 : Staubli *TX40*, Consignes de vitesses articulaires

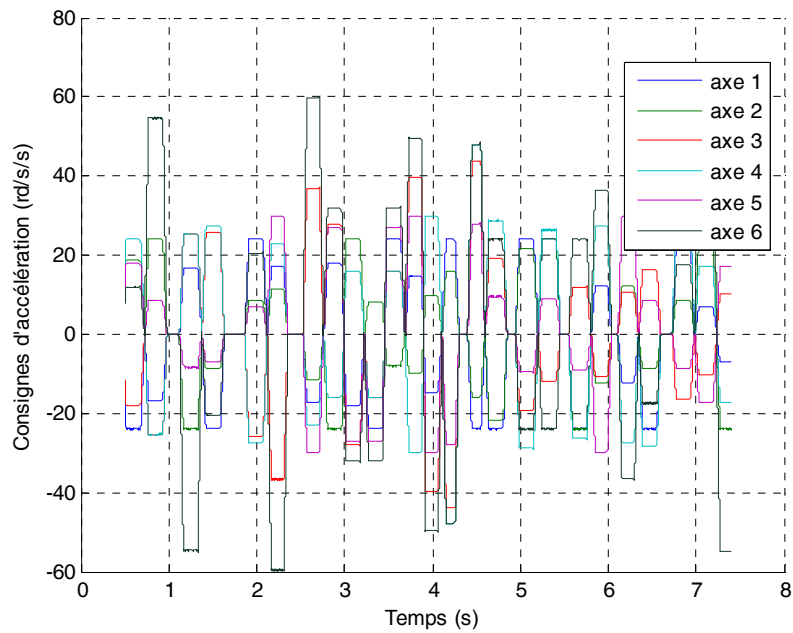


Figure 3-9 : Staubli *TX40*, Consignes d'accélération articulaires

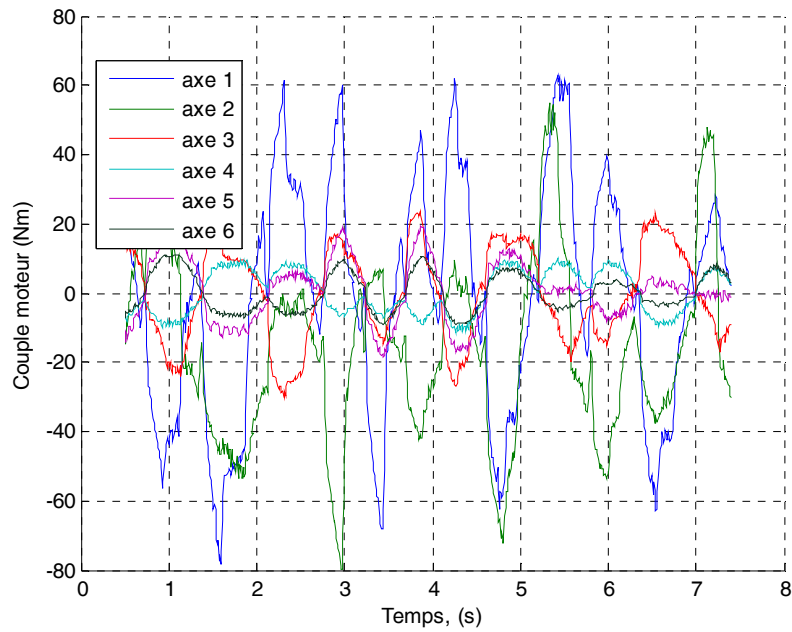


Figure 3-10 : Stäubli *TX40*, Couples mesurés

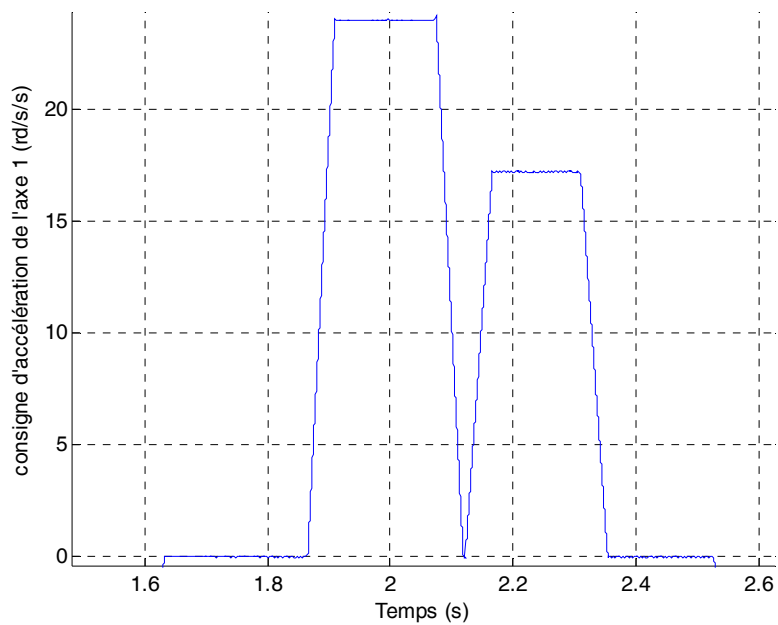


Figure 3-11 : Stäubli *TX40*, Zoom sur la consigne d'accélération articulaire de l'axe 1

Sur les 61 paramètres de base, seuls 33 paramètres essentiels sont conservés, 28 pour les paramètres du robot et 5 pour les paramètres de la charge (sur ses 10 paramètres inertiels).

Les paramètres sont bien identifiés dans l'ensemble car leurs écart-types relatifs sont faibles. La masse de la charge est estimée à 2.31 (kg) pour une valeur pesée de 2.37 (kg) soit une erreur de 0.06 (kg) (inférieure à $2 \hat{\sigma}_{M_L}$, seuil qui permet d'estimer la confiance dans le paramètre à 95.4%).

L'histogramme de l'erreur est représenté figure 3-12, le test de normalité de Kolmogorov-Smirnov vérifie bien que l'erreur suit une loi de distribution gaussienne pour $D_{obs}=0.022$ inférieur à $D_{\alpha}=0.0281$ (avec $\alpha=0.5$, voir annexe E). Les courbes de la validation directe pour l'essai sans charge (voir partie 2.9) sont représentées figure 3-13. Les vecteurs des mesures \mathbf{Y}_j sont proches des vecteurs des mesures estimés par les équations $\mathbf{W}_j \hat{\boldsymbol{\chi}}$ pour j de 1 à 6.

Tableau 3-1 : Stäubli *TX40*, paramètres essentiels identifiés avec la méthode *IDIM-LS*

Paramètre	$\hat{\chi}$	% $\hat{\sigma}_{\hat{\chi}}$
<i>ZZ</i> _{1R}	1.27	0.766
<i>FV</i> ₁	8.17	0.99
<i>Fc</i> ₁	7.45	3.06
<i>XX</i> _{2R}	-0.5	2.98
<i>XZ</i> _{2R}	-0.156	5.47
<i>ZZ</i> _{2R}	1.07	0.812
<i>MX</i> _{2R}	2.16	0.758
<i>FV</i> ₂	5.5	1.51
<i>Fc</i> ₂	8.71	2.26
<i>XX</i> _{3R}	0.11	13.1
<i>ZZ</i> _{3R}	0.125	6.5
<i>MY</i> _{3R}	-0.587	2.55
<i>Ia</i> ₃	0.0856	7.37
<i>FV</i> ₃	1.96	2.82
<i>Fc</i> ₃	6.87	2.69
<i>Ia</i> ₄	0.033	5.29
<i>FV</i> ₄	1.11	1.61
<i>Fc</i> ₄	2.7	2.42
<i>MY</i> _{5R}	-0.0287	18.1
<i>Ia</i> ₅	0.0398	9.01
<i>FV</i> ₅	1.87	2.05
<i>Fc</i> ₅	3.26	3.07
<i>Ia</i> ₆	0.0114	11.2
<i>Iam</i> ₆	0.0116	12.3
<i>FV</i> ₆	0.685	1.84
<i>Fs</i> ₆	0.444	21.7
<i>Fvm</i> ₆	0.654	1.81
<i>Fcm</i> ₆	1.79	4.34
<i>XX</i> _L	0.0296	7.45
<i>YY</i> _L	0.0301	7.41
<i>ZZ</i> _L	0.00551	20.6
<i>MZ</i> _L	0.272	3.06
<i>M_L</i>	2.31	3.13

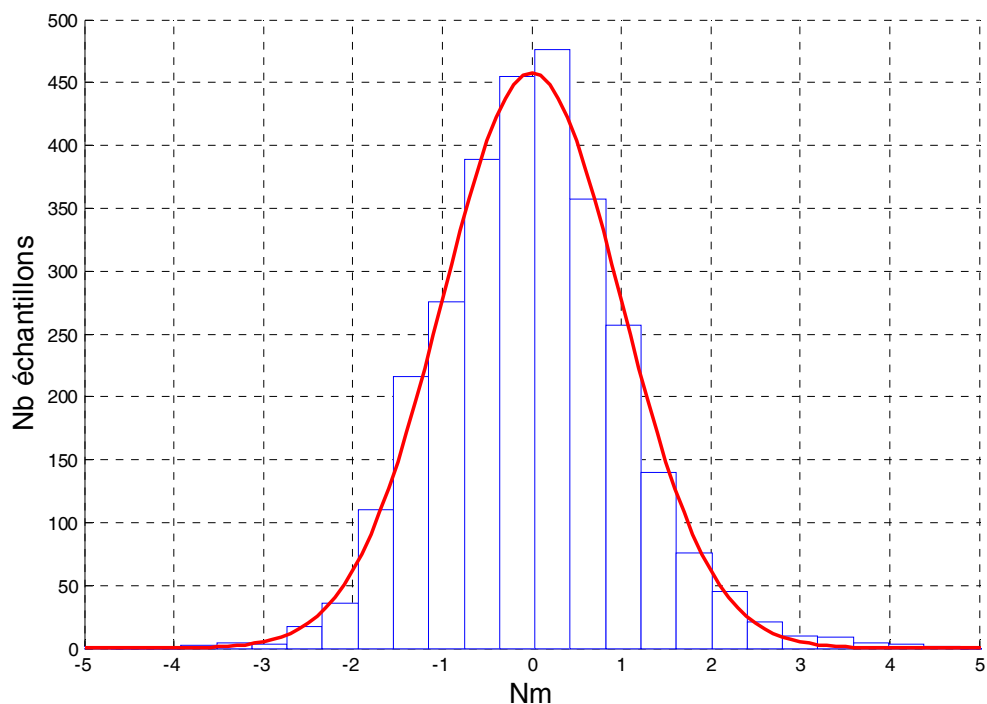
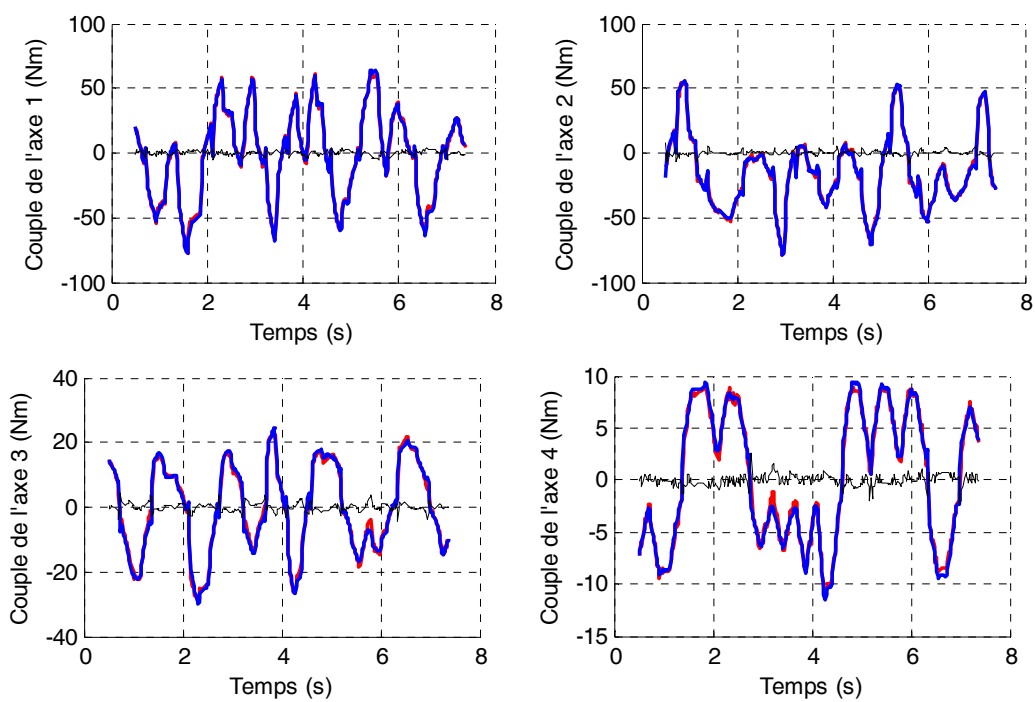


Figure 3-12 : Stäubli *TX40*, Histogramme des erreurs (en bleu) et Gaussienne (en rouge) avec la méthode *IDIM-LS*



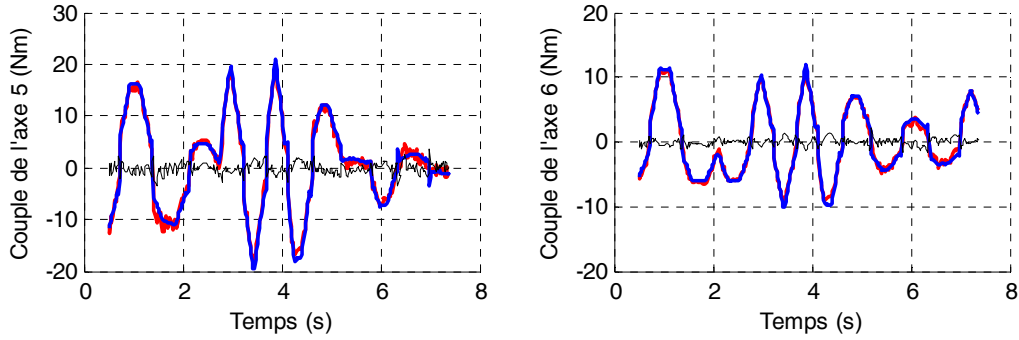


Figure 3-13 : Stäubli *TX40*, Validation directe avec la méthode *IDIM-LS* (sans charge), couples mesurés (rouge), couple reconstruits (bleu) et erreur (noir)

La méthode *DIDIM* est appliquée sur le robot Stäubli *TX40* avec la même charge et les mêmes trajectoires qui sont utilisées avec *IDIM-LS*. L'initialisation régulière est utilisée (voir équation (3.41)). Le robot possède un couplage cinématique entre les axes 5 et 6, il est nécessaire de prendre $I_{a_5}=2$ pour que la matrice d'inertie du robot simulée soit régulière à l'itération $k=0$.

Le simulateur Simulink/Matlab utilisé est représenté figure 3-14. Le contrôleur simulé du robot est représenté figure 3-15, il est de type Proportionnel Intégral Dérivé (*PID*) numérique. Le schéma de la simulation du modèle dynamique direct du robot est représenté figure 3-16. Ce dernier est codé en langage C et est compilé sous forme de 'C-Mex function' pour diminuer le temps de simulation. Le système $\hat{\mathbf{M}}(\mathbf{q}_s)\ddot{\mathbf{q}}_s = \boldsymbol{\tau}_s - \hat{\mathbf{H}}(\mathbf{q}_s, \dot{\mathbf{q}}_s)$ est résolu avec un algorithme de Cholesky qui permet de diminuer le temps de calcul et de vérifier que la matrice d'inertie, calculée avec les paramètres identifiés, est définie positive sur toute la trajectoire. La matrice d'inertie et le vecteur des efforts Coriolis, centrifuges et de frottements sont calculés numériquement à partir du modèle dynamique inverse calculé avec Symoro+.

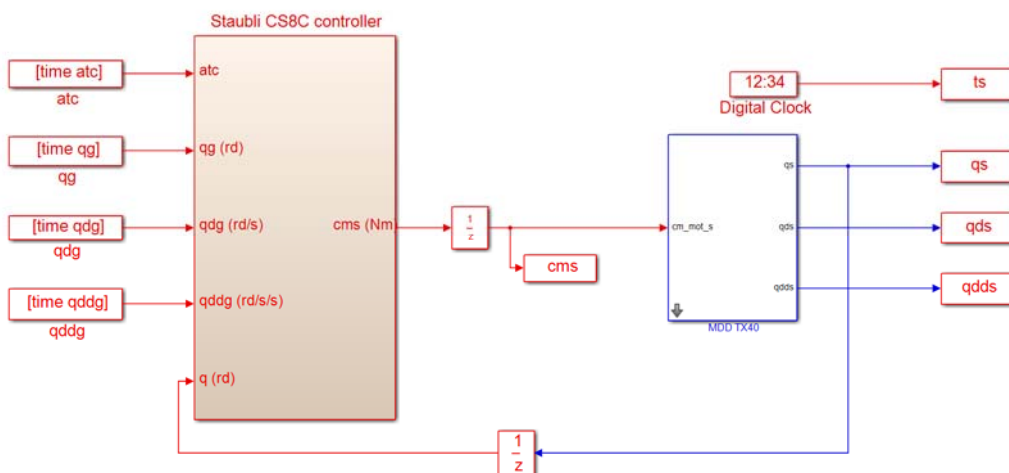


Figure 3-14 : Stäubli *TX40*, simulateur complet avec Matlab/Simulink

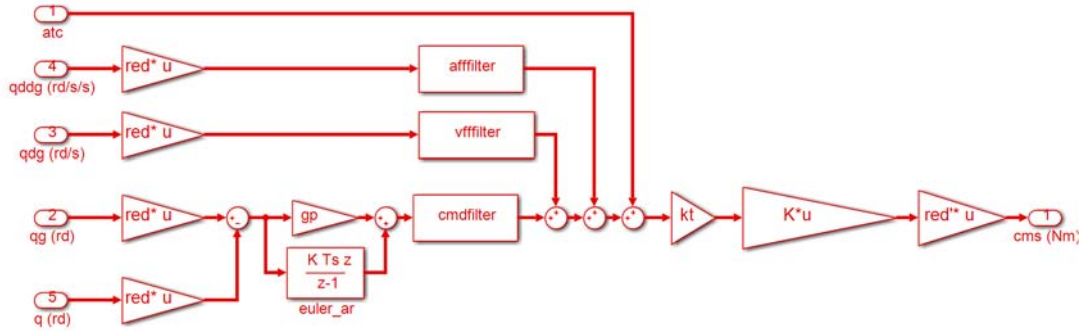


Figure 3-15 : Stäubli *TX40*, contrôleur avec Matlab/Simulink (Stäubli *TX40* CS8C controller)

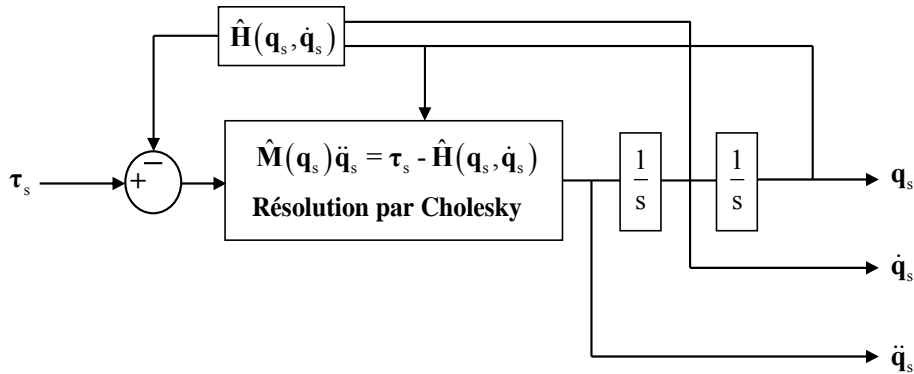


Figure 3-16 : Stäubli *TX40*, schéma de la simulation du modèle dynamique direct avec Matlab/Simulink (MDD *TX40*)

Dans la figure 3-15, K est la matrice ($n \times n$) des coefficients ${}^k\alpha_j$ calculés avec les valeurs de ${}^{ap}M_{I_j}$ (voir tableau 3-2) prises égales aux moments d'inertie des rotors des moteurs fournis par Stäubli selon l'équation (3.34). Les valeurs ${}^{ap}M_{I_j}$ sont légèrement inférieures aux valeurs de ${}^aM_{I_j}$ mais cela permet de tester la robustesse de l'algorithme de *DIDIM* vis à vis des erreurs sur ${}^{ap}M_{I_j}$.

Tableau 3-2: Stäubli *TX40*, valeurs de ${}^{ap}M_{I_j}$

Articulation	${}^{ap}M_{I_j}$
1	0.3625
2	0.3625
3	0.0808
4	0.0139
5	0.0316
6	0.0062

La fréquence de coupure du filtre decimate reste fixée à 10 (Hz). Les résultats de l'identification avec la méthode *DIDIM* sont donnés dans le tableau 3-3. L'algorithme converge en 3 itérations et on remarque que les valeurs identifiées à la première itération sont très proches des valeurs finales. Les paramètres identifiables sont les mêmes que précédemment. Les écart-types relatifs des valeurs identifiées avec la méthode *DIDIM* sont proches des écart-types calculés avec la méthode *IDIM-LS*. Les différences relatives entre les paramètres identifiés avec les deux méthodes sont données dans la dernière colonne du tableau 3-3 et sont calculées avec la formule suivante pour chaque paramètre :

$$\%e_{\hat{\chi}_i} = 100 \left| \frac{\hat{\chi}_{i(DIDIM)} - \hat{\chi}_{i(IDIM-LS)}}{\hat{\chi}_{i(IDIM-LS)}} \right| \quad (3.46)$$

Où :

- $\%e_{\hat{\chi}_i}$ est la différence relative entre le paramètre identifié avec la méthode *IDIM-LS* et le paramètre identifié avec la méthode *DIDIM*,
- $\hat{\chi}_{i(DIDIM)}$ est le paramètre identifié avec la méthode *DIDIM*,
- $\hat{\chi}_{i(IDIM-LS)}$ est le paramètre identifié avec la méthode *IDIM-LS*.

Les différences relatives sont faibles et inférieures à $2\hat{\sigma}_{\hat{\chi}}$, ce qui montre que la méthode *DIDIM* permet d'identifier avec précision les paramètres d'un robot (et sa charge) en utilisant uniquement les efforts moteurs.

Tableau 3-3 : Stäubli *TX40*, paramètres essentiels identifiés avec la méthode *DIDIM*

Paramètre	$\hat{\chi}^0$	$\hat{\chi}^1$	$\hat{\chi}^2$	$\hat{\chi}^3$	% $\hat{\sigma}_{\hat{\chi}}$	% $e_{\hat{\chi}}$
ZZ _{1R}	1	1.27	1.27	1.27	0.681	0.00
FV ₁	0	8.22	8.22	8.21	0.891	0.49
Fc ₁	0	7.29	7.3	7.28	2.91	2.28
XX _{2R}	0	-0.489	-0.489	-0.489	2.94	2.20
XZ _{2R}	0	-0.161	-0.161	-0.161	5.85	3.21
ZZ _{2R}	0	1.07	1.07	1.07	0.928	0.00
MX _{2R}	0	2.16	2.16	2.16	0.777	0,00
FV ₂	0	5.71	5.7	5.7	1.57	3.64
Fc ₂	0	8.53	8.50	8.48	2.44	2.64
XX _{3R}	0	0.102	0.101	0.101	13.1	8.18
ZZ _{3R}	1	0.12	0.12	0.12	6.93	4.00
MY _{3R}	0	-0.598	-0.599	-0.599	2.4	2.04
Ia ₃	0	0.0894	0.0893	0.0892	7.05	4.21
FV ₃	0	1.96	1.96	1.96	2.67	0.00
Fc ₃	0	6.84	6.84	6.84	2.67	0.44
Ia ₄	1	0.0329	0.0329	0.0329	5.24	0.30
FV ₄	0	1.12	1.12	1.12	1.59	0.90
Fc ₄	0	2.57	2.56	2.57	2.61	4.81
MY _{5R}	0	-0.0291	-0.0290	-0.0289	17.1	0.70
Ia ₅	2	0.0408	0.0408	0.0409	8.37	2.76
FV ₅	0	1.87	1.87	1.87	1.95	0.00
Fc ₅	0	3.31	3.29	3.29	2.98	0.92
Ia ₆	1	0.0118	0.0118	0.0118	10.2	3.51
Iam ₆	1	0.0104	0.0104	0.0103	13.3	11.21
FV ₆	0	0.672	0.672	0.672	1.74	1.90
Fs ₆	0	0.461	0.466	0.47	19.6	5.86
Fvm ₆	0	0.647	0.65	0.647	1.7	1.07
Fcm ₆	0	1.81	1.8	1.81	4.11	1.12
XX _L	0	0.0287	0.0288	0.0288	7.77	2.70
YY _L	0	0.0294	0.0294	0.0294	7.69	2.33
ZZ _L	0	0.00582	0.00583	0.00584	19.9	5.99
MZ _L	0	0.266	0.268	0.269	3.09	1.10
M _L	0	2.33	2.33	2.33	3.17	0.87

Comme pour la méthode *IDIM-LS*, le test de normalité de Kolmogorov-Smirnov (voir figure 3-17) vérifie bien que l'erreur suit une loi de distribution gaussienne pour $D_{\text{obs}}=0.020$ inférieur à $D_{\alpha}=0.0281$ (avec $\alpha=0.5$, voir annexe E). Les vecteurs des mesures \mathbf{Y}_j sont proches des vecteurs des mesures estimés par les équations $\mathbf{W}_j\hat{\chi}$ pour j de 1 à 6 (voir figure 3-18).

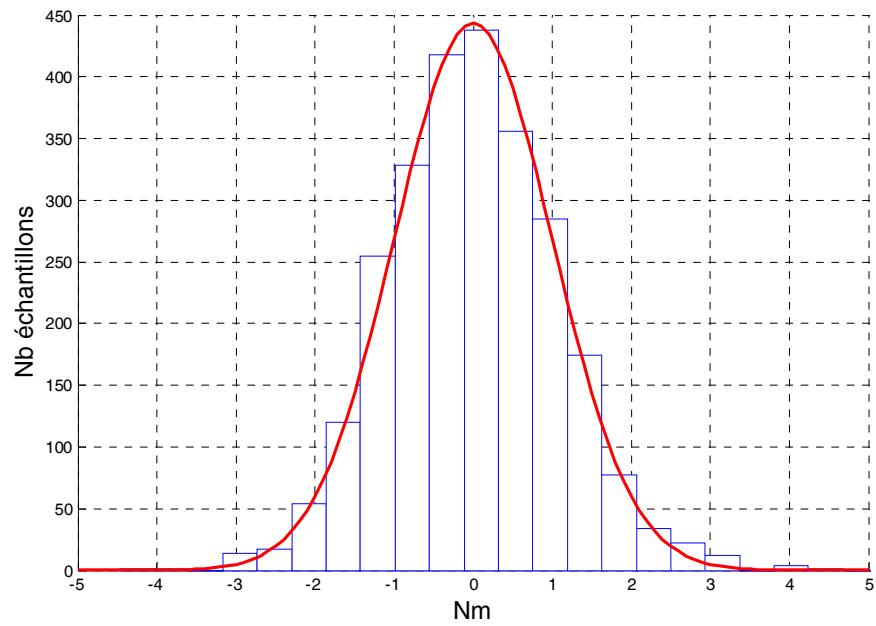
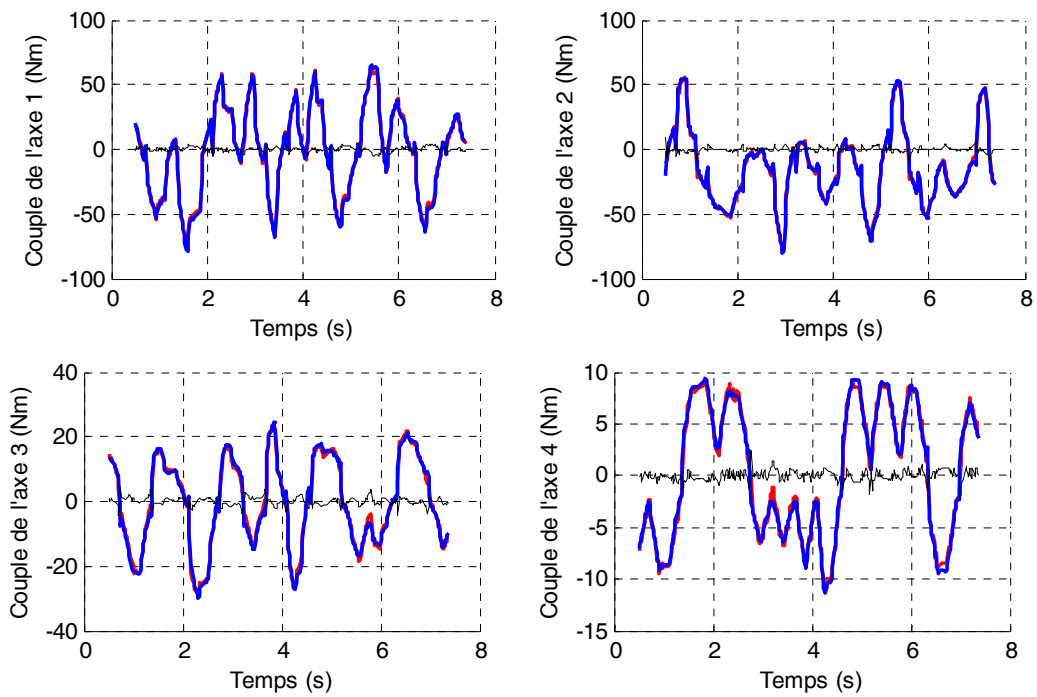


Figure 3-17 : Stäubli *TX40*, Histogramme des erreurs (en bleu) et Gaussienne (en rouge) avec la méthode *DIDIM* à l'itération 3



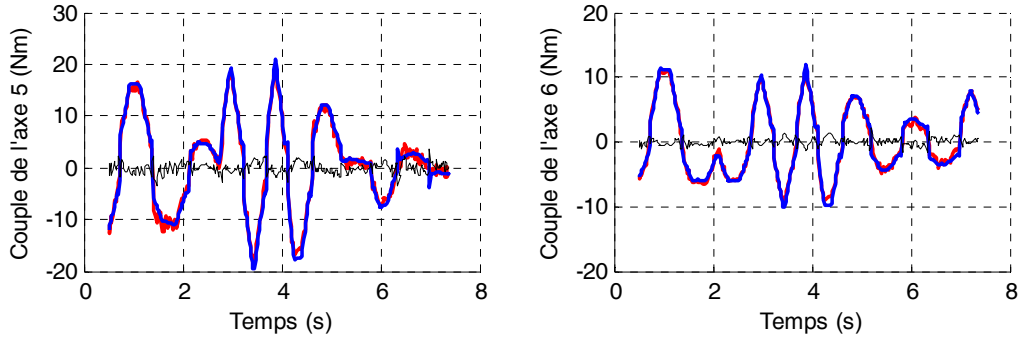


Figure 3-18 : Stäubli TX40, Validation directe avec la méthode DIDIM (sans charge), couples mesurés (rouge), couple reconstruits (bleu) et erreur (noir)

Le tableau 3-4 donne les normes relatives du résidu pour la méthode *IDIM-LS* et pour chacune des itérations de la méthode *DIDIM*. Dès la première itération, la norme relative du résidu obtenue avec la méthode *DIDIM* est inférieure à celle obtenue avec la méthode *IDIM-LS* et elle va en décroissant au fur et à mesure des itérations.

Tableau 3-4 : Norme relative du résidu *IDIM-LS* et *DIDIM*

Itération	<i>IDIM-LS</i>	<i>DIDIM</i>		
	-	1	2	3
$\ W\chi - Y\ / \ Y\ $	8,54%	8,05%	7,82%	7,78%

3.6 Effet de la pondération et du filtre Decimate

Les identifications précédentes du robot Stäubli *TX40* utilisent la pondération et le filtre Decimate avec une fréquence de coupure appropriée.

On rappelle que l'utilisation des moindres carrés pondérés (voir partie 2.4) permet d'améliorer l'estimation des paramètres et des écart-types en attribuant à chaque équation de (2.9) un facteur de confiance. Les moindres carrés pondérés permet de normaliser la variance de l'erreur ρ dans (2.9) après pondération.

Le filtre Decimate (voir partie 2.5) applique un filtre passe-bas de fréquence de coupure ω_p de type Chebyshev de type 1 et d'ordre 8 et un sous-échantillonnage sur chaque colonne de W et le vecteur Y . Il permet de diminuer la taille du système surdéterminé (2.21), d'améliorer l'indépendance statistique entre les échantillons et d'améliorer l'estimation des écart-types.

Afin d'observer les effets de l'utilisation ou non de la méthode des moindres carrés pondérés et du filtre Decimate, 4 cas de figure sont étudiés sur le *TX40* avec la méthode *IDIM-LS*:

- un cas avec pondération et utilisation du filtre Decimate (voir partie 3.5, identification avec la méthode *IDIM-LS*),
- un cas avec pondération sans utilisation du filtre Decimate,
- un cas avec utilisation du filtre Decimate sans pondération,
- un cas sans utilisation du filtre Decimate et sans pondération.

Les histogrammes des erreurs pour chaque cas sont donnés à la figure 3-19, figure 3-20, figure 3-21 et figure 3-22.

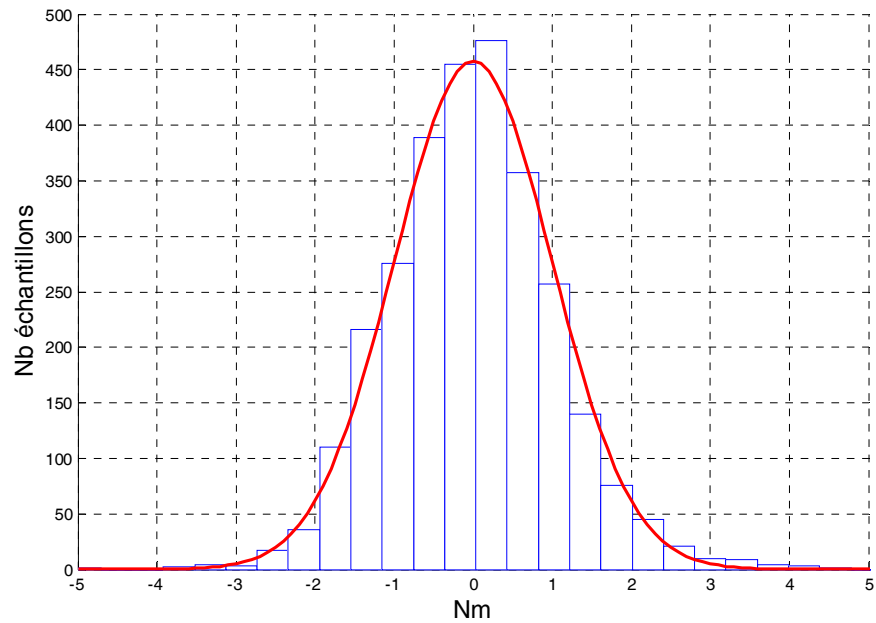


Figure 3-19 : Stäubli TX40, Histogramme des erreurs (en bleu) et Gaussienne (en rouge) avec la méthode *IDIM-LS*, cas avec pondération et utilisation du filtre Decimate

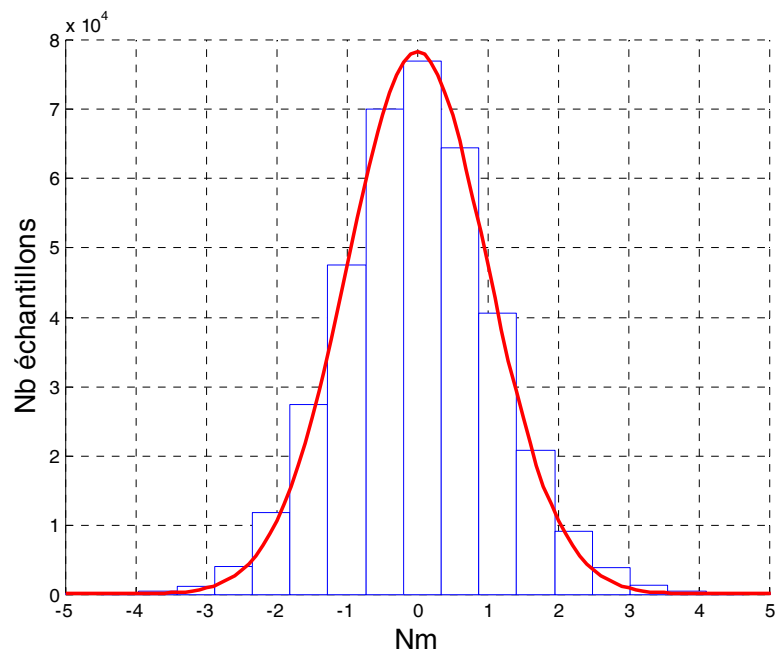


Figure 3-20 : Stäubli TX40, Histogramme des erreurs (en bleu) et Gaussienne (en rouge) avec la méthode *IDIM-LS*, cas avec pondération sans utilisation du filtre Decimate

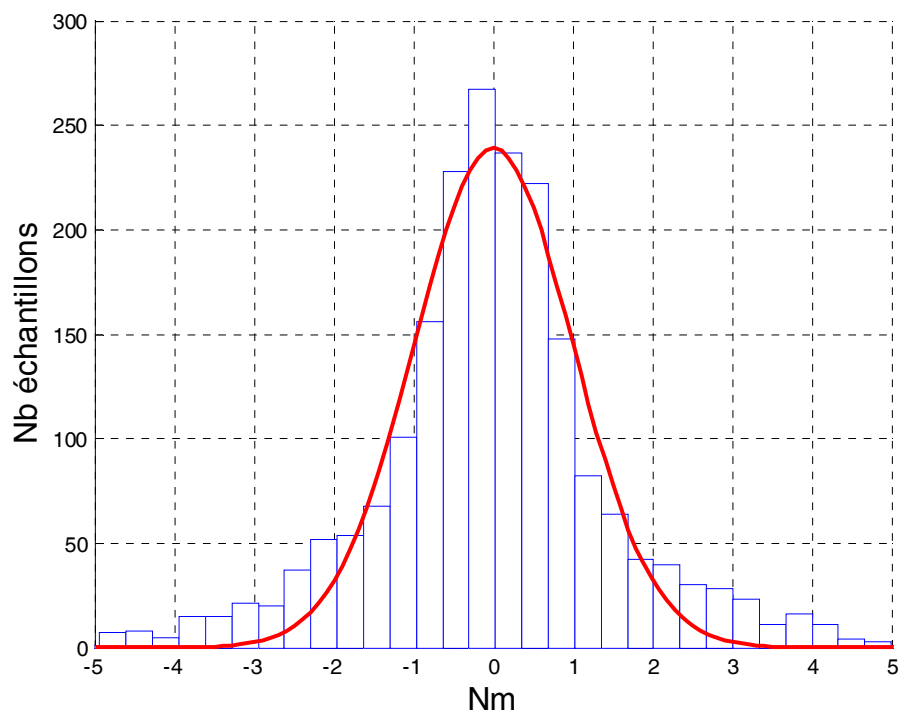


Figure 3-21 : Stäubli *TX40*, Histogramme des erreurs (en bleu) et Gaussienne (en rouge) avec la méthode *IDIM-LS*, cas avec utilisation du filtre Decimate sans pondération

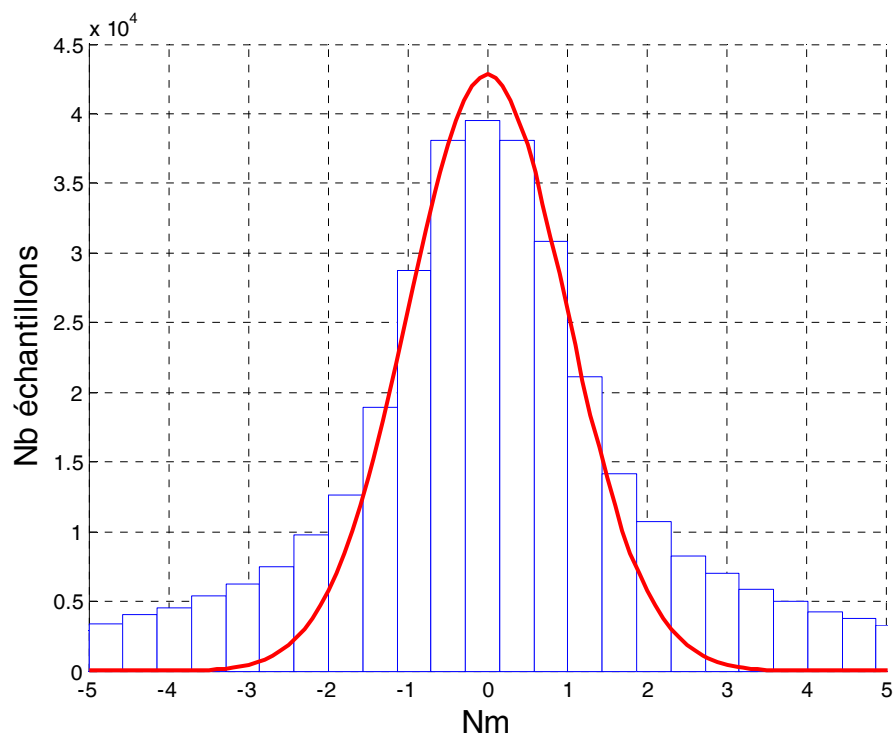


Figure 3-22 : Stäubli *TX40*, Histogramme des erreurs (en bleu) et Gaussienne (en rouge) avec la méthode *IDIM-LS*, cas sans utilisation du filtre Decimate et sans pondération

La non-utilisation du filtre Decimate a pour effet logique d'augmenter le nombre d'échantillons car aucun sous échantillonnage n'est appliqué (voir figure 3-20 et figure 3-22). La norme relative du résidu augmente par rapport au cas où le filtre Decimate est utilisé et les écarts-types sont sous-évalués car l'indépendance statistique entre les échantillons n'est plus de mise. Les paramètres (non représentés ici) sont faiblement biaisés.

La non-utilisation de la méthode des moindres carrés pondérés empêche de normaliser la variance de l'erreur \mathbf{p} dans (2.9) (voir figure 3-21 et figure 3-22). La norme relative du résidu augmente par rapport au cas où la méthode des moindres carrés pondérés est utilisée. Les paramètres sont aussi faiblement biaisés. De plus la méthode de suppression des paramètres mal identifiés et de réduction de modèle décrite à la partie 2.8 ne peut plus être utilisée car elle nécessite que la variance de l'erreur \mathbf{p} soit normale.

Cette étude montre simplement et clairement que l'utilisation du filtre Decimate et de la méthode des moindres carrés pondérés est nécessaire pour identifier un robot de manière pertinente.

3.7 Simplification du coefficient d'adaptation ${}^k\alpha_j$

La théorie de cette partie est issue des travaux de (Robet et al. 2012) (Robet et al. 2013) (Robet et Gautier. 2013a-c)

On rappelle que la condition d'identité des fonctions de transfert simulées et réelles du système articulaire double intégrateur (figure 3-5 et figure 3-6) est obtenue avec le coefficient correcteur idéal suivant:

$${}^k\alpha_j = \frac{{}^a\mathbf{g}_{\tau_j} \quad {}^k\hat{\mathbf{M}}_{I_j}}{{}^{ap}\mathbf{g}_{\tau_j} \quad {}^a\mathbf{M}_{I_j}} \quad (3.47)$$

En pratique on choisit :

$${}^k\alpha_j = \frac{{}^k\hat{\mathbf{M}}_{I_j}}{{}^{ap}\mathbf{M}_{I_j}} \quad (3.48)$$

Avec une initialisation :

$$\hat{\chi}^0 = 0 \text{ sauf pour } I a_j^0 = 1 \text{ pour } j = 1, 2, \dots, n \quad (3.49)$$

Supposons que ${}^{ap}\mathbf{g}_{\tau_j} = {}^a\mathbf{g}_{\tau_j}$ et que l'algorithme converge idéalement sans biais : ${}^k\hat{\mathbf{M}}_{I_j}$ tend vers ${}^a\mathbf{M}_{I_j}$ pour l'itération finale k_f et ${}^k\alpha_j$ tend vers ${}^{k_f}\alpha_j = {}^a\mathbf{M}_{I_j} / {}^{ap}\mathbf{M}_{I_j}$ pour l'itération finale k_f . Dans (3.47), avec ${}^{ap}\mathbf{g}_{\tau_j} = {}^a\mathbf{g}_{\tau_j}$ et ${}^{ap}\mathbf{M}_{I_j} = {}^a\mathbf{M}_{I_j}$, on obtient le coefficient idéal $\alpha_j = 1$.

Le rapport ${}^{k_f}\alpha_j / \alpha_j$ représente l'erreur entre les transferts de boucle ouverte simulées et réelles soit ${}^{k_f}\alpha_j / \alpha_j = {}^a\mathbf{M}_{I_j} / {}^{ap}\mathbf{M}_{I_j} \neq 1$. Il y a donc un biais.

L'identification en simulation du robot permet de montrer ce biais. L'identification des paramètres est effectuée avec les efforts réels simulés en utilisant les paramètres identifiés précédemment avec la méthode *DIDIM*. L'identification est réalisée avec la même trajectoire excitante, à la même fréquence de coupure du filtre decimate et en prenant ${}^a g_{\tau_j} = {}^{ap} g_{\tau_j}$.

Différents facteurs d'erreur sont testés sur $Ir_j N_{rj}^2$ par rapport à ${}^{ap} M_{I_j}$ calculé avec (3.34) pour évaluer l'importance du biais et de la variation de la norme relative du résidu (voir tableau 3-5). L'intervalle de test est fixé de $0.5 Ir_j N_{rj}^2$ à $10 Ir_j N_{rj}^2$ car en dessous de $0.5 Ir_j N_{rj}^2$ l'algorithme ne converge pas et au-delà de $10 Ir_j N_{rj}^2$ la norme relative du résidu et l'erreur relative sur les paramètres continuent simplement d'augmenter.

Tableau 3-5 : Norme relative du résidu *DIDIM*

${}^{ap} M_{I_j}$	$0.5 Ir_j N_{rj}^2$	$2 Ir_j N_{rj}^2$	$5 Ir_j N_{rj}^2$	$10 Ir_j N_{rj}^2$
$\ W\chi - Y\ / \ Y\ $	2.32%	1.1%	1.79%	5.30%
$\% e $	1.60%	0.71%	1.64%	4.51%

On remarque que l'erreur sur les paramètres varie entre 1.60% et 4.51% et que la norme relative du résidu varie entre 1.1% et 5.30% pour un facteur d'erreur compris entre 0.5 et 10. Le biais sur les paramètres est bien présent.

La solution proposée pour palier ce problème est de simplement utiliser la valeur suivante de α_j :

$$\alpha_j = 1 \text{ pour toute itération } k \quad (3.50)$$

L'utilisation de $\alpha_j=1$ évite le biais dû au rapport ${}^a M_{I_j} / {}^{ap} M_{I_j}$. Il ne subsiste que le biais dû au rapport ${}^a g_{\tau_j} / {}^{ap} g_{\tau_j}$. Le nouveau schéma de la simulation, représenté figure 3-23, est identique au schéma (figure 3-5) du robot réel.

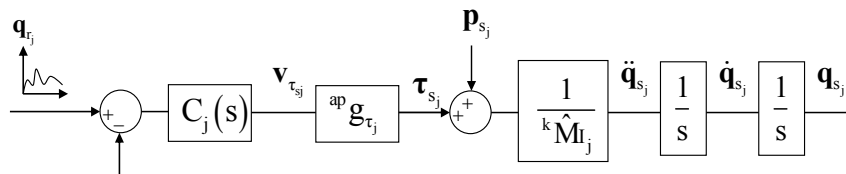


Figure 3-23 : Commande en position articulaire avec $\alpha_j=1$

On choisit les conditions initiales suivantes :

$$\hat{\chi}^0 = 0 \text{ sauf pour } Ia_j^0 = {}^{ap} M_{I_j} = Ir_j N_{rj}^2 \text{ pour } j=1,2,\dots,n \quad (3.51)$$

En choisissant l'initialisation (3.51), on obtient ${}^0I_{a_j} = {}^{ap}M_{I_j}$. La fonction de transfert en boucle ouverte dans la figure 3-23 est :

$${}^0T_{bo}(s) = C_j(s) \frac{{}^{ap}g_{\tau_j}}{{}^{ap}M_{I_j}} \frac{1}{s^2} \quad (3.52)$$

(3.52) est exactement la même fonction de transfert que celle obtenue dans le premier réglage de ${}^k\alpha_j$ avec ${}^k\alpha_j = {}^k\hat{M}_{I_j} / {}^{ap}M_{I_j}$ et avec l'initialisation (3.41) $\hat{\chi}^0 = 0$ sauf pour $I_{a_j}^0 = 1$ pour $j=1,2,\dots,n$.

A la première itération, les deux méthodes de réglages de ${}^k\alpha_j$ sont identiques. Les itérations suivantes convergent sans biais pour ${}^k\alpha_j=1$.

La méthode *DIDIM* avec $\alpha_j=1$ est appliquée sur le robot Stäubli *TX40* avec la même charge et les mêmes trajectoires que précédemment. L'initialisation proposée à l'équation (3.51) est utilisée. Les résultats sont donnés dans le tableau 3-6.

L'algorithme converge en 2 itérations et comme précédemment on remarque que les valeurs identifiées à la première itération sont très proches des valeurs finales. Les écart-types relatifs des valeurs identifiées avec la méthode *DIDIM* avec $\alpha_j=1$ sont proches des écart-types calculés avec la méthode *IDIM-LS* et *DIDIM* classique.

Par la suite, seule la méthode *DIDIM* avec $\alpha_j=1$ est utilisée.

Tableau 3-6 : Stäubli *TX40*, paramètres essentiels identifiés avec la méthode *DIDIM* avec $\alpha=1$

Paramètre	$\hat{\chi}^0$	$\hat{\chi}^1$	$\hat{\chi}^2$	% $\hat{\sigma}_{\hat{\chi}}$	%e
ZZ_{1R}	0.3625	1.27	1.27	0.682	0.00
Fv_1	0	8.22	8.23	0.888	0.73
Fc_1	0	7.29	7.3	2.9	2.01
XX_{2R}	0	-0.489	-0.489	2.93	2.20
XZ_{2R}	0	-0.161	-0.161	5.79	3.21
ZZ_{2R}	0.3625	1.07	1.07	0.916	0.00
MX_{2R}	0	2.16	2.16	0.773	0.00
Fv_2	0	5.71	5.71	1.54	3.82
Fc_2	0	8.53	8.53	2.4	2.07
XX_{3R}	0	0.102	0.103	12.9	6.36
ZZ_{3R}	0	0.12	0.119	6.96	4.80
MY_{3R}	0	-0.598	-0.599	2.42	2.04
Ia_3	0.0808	0.0894	0.0892	7.15	4.21
Fv_3	0	1.96	1.96	2.73	0.00
Fc_3	0	6.84	6.82	2.74	0.73
Ia_4	0.0139	0.0329	0.0329	5.32	0.30
Fv_4	0	1.12	1.12	1.62	0,90
Fc_4	0	2.57	2.57	2.65	4,81
MY_{5R}	0	-0.0291	-0.0294	17.1	2,44
Ia_5	0.0316	0.0408	0.0409	8.4	2.76
Fv_5	0	1.87	1.88	1.94	0.53
Fc_5	0	3.31	3.28	2.99	0.61
Ia_6	0.0062	0.0118	0.0121	10.2	6.14
Iam_6	0.0062	0.0104	0.0105	13.2	9.48
Fv_6	0	0.672	0.675	1.76	1.46
Fs_6	0	0.461	0.469	19.8	5.63
Fvm_6	0	0.647	0.649	1.72	0.76
Fcm_6	0	1.81	1.8	4.16	0.56
XX_L	0	0.0287	0.0287	7.79	3.04
YY_L	0	0.0294	0.0294	7.69	2.33
ZZ_L	0	0.00582	0.0057	20.5	3.45
MZ_L	0	0.266	0.266	3.12	2.21
M_L	0	2.33	2.33	3.11	0.87

Comme précédemment, le test de normalité de Kolmogorov-Smirnov vérifie bien que l'erreur suit une loi de distribution gaussienne pour $D_{obs}=0.019$ inférieur à $D_{\alpha}=0.0281$ (avec $\alpha=0.5$, voir annexe E).

3.8 Comparaison entre la méthode *CLOE* à erreur de position et la méthode *DIDIM*

3.8.1 Comparaison analytique

Par la suite, nous allons expliquer de manière analytique pourquoi la méthode *DIDIM* possède de meilleures performances que la méthode *CLOE* à erreur de position. Cette partie s'inspire des travaux publiés dans (Janot et al. 2014b).

La première explication se focalise sur le calcul de la position et de l'effort moteur de l'articulation j en fonction du terme de perturbation et du terme ${}^k\hat{M}_{I_j}$.

On rappelle le schéma du simulateur à la figure 3-24 :

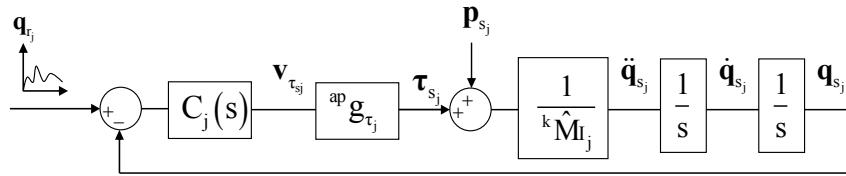


Figure 3-24 : Commande en position articulaire

La fonction de transfert en boucle ouverte $Tbo(s)$ est la suivante (avec $p_{s_j}=0$):

$$Tbo(s) = \frac{q_{s_j}}{e_{q_{s_j}}} = \frac{{}^{ap}g_{\tau_j} C_j(s)}{{}^k\hat{M}_{I_j} s^2} \text{ avec } e_{q_{s_j}} = q_{r_j} - q_{s_j} \quad (3.53)$$

La fonction de transfert en boucle fermée s'écrit de la manière suivante

$$Tbf(s) = \frac{q_{s_j}}{q_{r_j}} = \frac{Tbo(s)}{1 + Tbo(s)} \quad (3.54)$$

En prenant en compte la perturbation, il vient pour q_{s_j} :

$$q_{s_j} = Tbf(s) \left(q_{r_j} + \frac{p_{s_j}}{{}^{ap}g_{\tau_j} C_j(s)} \right) \quad (3.55)$$

L'objectif en poursuite de q_r de la commande, via le correcteur $C_j(s)$, est d'avoir $Tbf(s) \approx 1$ dans la bande passante de $Tbf(s)$ ce qui induit $q_{s_j} \approx q_{r_j}$ et $1 / ({}^{ap}g_{\tau_j} C_j(s)) \approx 0$. La consigne de position est très proche de la position simulée et la perturbation est rejetée dans la bande passante

de $Tbf(s)$. q_{s_j} est donc peu sensible aux variations des paramètres du système dans ${}^k\hat{M}_{I_j}$ et dans la perturbation $p_{s_j}(\hat{\chi})$ rejetée par $1/({}^{ap}g_{\tau_j}C_j(s))$.

Si on compare avec τ_{s_j} , qui s'exprime selon :

$$\begin{aligned}\tau_{s_j} &= {}^k\hat{M}_{I_j}s^2q_{s_j} + p_{s_j} \\ &= {}^k\hat{M}_{I_j}s^2Tbf(s)q_{r_j} + p_{s_j}\end{aligned}\quad (3.56)$$

τ_{s_j} dépend de ${}^k\hat{M}_{I_j}$, avec une amplification de q_{s_j} dans les hautes fréquences (avec le terme double dérivateur s^2), et dépend directement du terme de perturbation $p_{s_j}(\hat{\chi})$. Les efforts moteurs sont donc beaucoup plus sensibles aux variations des paramètres que les positions.

La deuxième explication se focalise sur le calcul des efforts en fonction des positions articulaires. Les équations de l'effort en fonction du correcteur, de la consigne de position et de la position pour une articulation j sont rappelées pour le robot réel et le robot simulé :

$$\begin{aligned}\tau_j &= {}^a g_{\tau_j} C_j(q_{r_j} - q_j) \\ \tau_{s_j} &= {}^{ap} g_{\tau_j} C_j(q_{r_j} - q_{s_j})\end{aligned}\quad (3.57)$$

Pour simplifier la suite des expressions, nous émettons l'hypothèse que ${}^a g_{\tau_j} = {}^{ap} g_{\tau_j}$. Il vient :

$$\tau_j - \tau_{s_j} = -{}^{ap} g_{\tau_j} C_j(q_j - q_{s_j})\quad (3.58)$$

Soit :

$$e_{\tau_j} = -{}^{ap} g_{\tau_j} C_j e_{q_j}\quad (3.59)$$

Où :

- $e_{\tau_j} = \tau_j - \tau_{s_j}$,
- $e_{q_j} = q_j - q_{s_j}$.

Soit :

$$\frac{e_{\tau_j}(s)}{e_{q_j}(s)} = -{}^{\text{ap}}g_{\tau_j} C_j(s) \quad (3.60)$$

Le correcteur est calculé de manière à avoir un bon suivi de trajectoire, soit $Tbf(s) \sim 1$. Le gain ${}^{\text{ap}}g_{\tau_j} |C_j(j\omega)|$ doit donc être grand dans la bande passante de $Tbf(s)$ donc l'erreur entre l'effort moteur réel et l'effort moteur simulé est beaucoup plus grande que l'erreur entre la position réelle et la position simulée.

Les équations précédentes illustrent l'intérêt de la méthode *DIDIM* comparée aux méthodes à erreur de sortie classiques. La méthode *DIDIM* est une méthode *CLOE* pondérée par le contrôleur du robot, le critère utilisé est beaucoup plus sensible aux variations de paramètres que le critère utilisé par les méthodes *CLOE* classiques.

3.8.2 Comparaison expérimentale

Cette partie traite de la comparaison expérimentale entre les méthodes *CLOE* à erreur de position et la méthode *DIDIM*. Trois méthodes de programmation non linéaire sont testées: la méthode de Gauss-Newton, la méthode *SQP* (Sequential Quadratic Programming) et la méthode du Simplex de Nelder-Mead, avec différentes initialisations.

Nous rappelons que la méthode de Gauss-Newton nécessite le calcul des fonctions de sensibilité de manière numérique par différence centrée alors que ces dernières ne sont pas nécessaires pour la méthode du Simplex. Les temps de convergence et les valeurs finales identifiées sont analysés.

Cette étude est effectuée sur le robot Stäubli *TX40* mais ici aucune charge n'est utilisée pour éviter de complexifier le critère quadratique utilisé dans les méthodes de programmation non linéaire.

Méthode de Gauss-Newton

Nous rappelons que la solution optimale $\hat{\chi}$ minimise le critère quadratique suivant (Janot et al. 2014a-b):

$$J(\chi) = \| \mathbf{Y} - \mathbf{Y}_s \|^2 = (\mathbf{Y} - \mathbf{Y}_s)^T (\mathbf{Y} - \mathbf{Y}_s) \quad (3.61)$$

- \mathbf{Y} est le vecteur (rx1) obtenu après échantillonnage et filtrage du vecteur d'échantillon \mathbf{Y}_{fm} qui dépend des mesures des positions,
- \mathbf{Y}_s est le vecteur (rx1) obtenu après échantillonnage et filtrage du vecteur d'échantillon \mathbf{Y}_{sim} qui dépend des positions simulées.

Deux initialisations sont testées:

- la même initialisation que pour la méthode *DIDIM* (voir équation (3.51)),

- une initialisation avec des valeurs assez proches des valeurs optimales, calculées avec les paramètres identifiés avec *DIDIM* donnés dans le tableau 3-6, avec une erreur relative de 20%.

En utilisant la méthode de Gauss-Newton, la matrice de variance-covariance des estimées a pour expression :

$$\mathbf{C}_{\chi\chi_{gn}} = \hat{\sigma}_{\rho_{gn}}^2 \left(\mathbf{W}_{\delta}^T \mathbf{W}_{\delta} \right)^{-1} \quad (3.62)$$

Où \mathbf{W}_{δ} est obtenu après échantillonnage et filtrage des fonctions de sensibilité $\delta_{y_s/x}$ (voir équations (3.11) et (3.12)). En pratique, les fonctions de sensibilité sont calculées de manière numérique par le logiciel Matlab.

L'écart-type $\sigma_{\rho_{gn}}$ peut être calculé par l'estimateur non biaisé suivant:

$$\hat{\sigma}_{\rho_{gn}}^2 = \frac{\| \mathbf{Y} - \mathbf{Y}_s \|^2}{(r - b)} \quad (3.63)$$

Similairement à (2.15), les écart-types des paramètres de base sont les racines carrées de la diagonale de $\mathbf{C}_{\chi\chi_{gn}}$:

$$\hat{\sigma}_{\hat{\chi}_i} = \sqrt{\mathbf{C}_{\chi\chi_{gn}}(i, i)} \quad (3.64)$$

Où $\mathbf{C}_{\chi\chi_{gn}}(i, i)$ est le $i^{\text{ème}}$ élément diagonal de $\mathbf{C}_{\chi\chi}$.

Similairement à (2.16), l'écart type relatif $\% \hat{\sigma}_{\hat{\chi}_{ri}}$ du paramètre de base d'indice i est donné par :

$$\% \hat{\sigma}_{\hat{\chi}_{ri}} = 100 \frac{\hat{\sigma}_{\hat{\chi}_i}}{|\hat{\chi}_i|} \text{ pour } |\hat{\chi}_i| \neq 0 \quad (3.65)$$

La méthode Gauss-Newton permet le calcul des écart-types relatifs, nous avons donc un indicateur sur l'incertitude de l'estimation des paramètres avec cette méthode.

Ces écart-types relatifs ainsi que les équations (3.62), (3.63) et (3.64) sont calculés seulement à la dernière itération de l'algorithme car il est inutile d'avoir un indicateur sur l'incertitude de l'estimation des paramètres à chaque itération de l'algorithme.

Les méthodes *IDIM-LS*, *DIDIM* (voir tableau 3-7) et la méthode *CLOE* à erreur de position avec l'algorithme de Gauss-Newton sont appliquées sur le Stäubli *TX40* (Janot et al. 2014b).

La première initialisation est testée. L'algorithme converge vers un optimum local erroné au bout de 30 itérations et 1h40 de temps de calcul sur un processeur dual-cœur de 2 (GHz). L'algorithme de Gauss-Newton ne permet donc pas l'utilisation de la même initialisation que la méthode *DIDIM*.

La deuxième initialisation est testée (les valeurs initiales sont les paramètres identifiés avec *DIDIM* avec une erreur relative de 20%). L'algorithme converge vers des valeurs proches de celles obtenues avec *DIDIM*. Il converge en 15 itérations et 50min, la méthode *DIDIM* converge, elle, en 2 itérations soit 8s seulement. Les résultats sont donnés dans le tableau 3-7.

La simplification de modèle pour calculer les paramètres essentiels est différente de la méthode proposé dans la partie (2.8) car elle n'est pas applicable en l'état pour la méthode *CLOE* à erreur de position. A la place, les paramètres de base possédant un écart-type relatif supérieur à 20% sont supprimés.

Tableau 3-7 : Stäubli TX40, paramètres essentiels identifiés avec les méthodes, *IDIM-LS*, *DIDIM* et la méthode *CLOE* à erreur de position

Par.	<i>IDIM-LS</i>		<i>DIDIM</i>		<i>CLOE</i> (position)	
	$\hat{\chi}$	$\% \hat{\sigma}_{\hat{\chi}_{ri}}$	$\hat{\chi}^2$	$\% \hat{\sigma}_{\hat{\chi}_{ri}}$	$\hat{\chi}^{15}$	$\% \hat{\sigma}_{\hat{\chi}_{ri}}$
ZZ _{1R}	1.26	0.90	1.27	0.76	1.25	1.0
FV ₁	7.92	1.2	8.00	1.0	8.2	0.7
FC ₁	7.25	3.7	7.07	3.4	6.53	2.3
XX _{2R}	-0.489	3.6	-0.483	3.4	-0.48	2.1
XZ _{2R}	-0.156	6.4	-0.158	6.6	-0.15	4.0
ZZ _{2R}	1.06	0.94	1.07	1.0	1.08	1.2
MX _{2R}	2.16	0.83	2.15	0.80	2.2	2.7
FV ₂	5.32	1.8	5.54	1.7	5.73	0.8
FC ₂	8.34	2.7	8.10	2.7	7.70	1.7
XX _{3R}	0.105	16	0.0957	15	0.13	9.3
ZZ _{3R}	0.131	6.9	0.124	6.8	0.115	7.9
MY _{3R}	-0.582	2.7	-0.597	2.4	-0.0606	2.1
Ia ₃	0.0812	8.3	0.0847	7.3	0.0990	9.0
FV ₃	1.98	2.9	1.98	2.5	2.00	1.4
FC ₃	6.34	3.0	6.32	2.8	6.02	1.3
Ia ₄	0.0319	6.7	0.032	5.8	0.0303	9.1
FV ₄	1.11	2.1	1.13	1.8	1.14	1.1
FC ₄	2.49	3.5	2.41	3.2	2.24	3.1
MY _{5R}	-0.0315	16	-0.0316	14	-0.00400	15.0
Ia ₅	0.0434	9.7	0.0421	9.1	0.00402	9.2
FV ₅	1.82	2.5	1.81	2.3	1.85	1.4
FC ₅	2.98	4.0	3.04	3.7	2.90	2.9
Ia ₆	0.0120	12	0.0121	11	0.0115	9.5
Iam ₆	0.0112	16	0.0103	15	0.0129	17
FV ₆	0.687	2.0	0.647	2.1	0.650	1.3
FC ₆	0.480	5.9	0.471	6.1	2.22	2.7
Fvm ₆	0.608	2.2	0.623	2.0	0.600	1.3
Fcm ₆	1.97	3.3	1.74	4.9	1.93	3.2

Les positions mesurées, les positions simulées et l'erreur entre ces dernières sont représentées figure 3-25. Les erreurs relatives sont données dans le tableau 3-8, ces dernières sont toutes inférieures à 0.1%. Elles ont été calculées aussi pour la dernière itération de *DIDIM* et sont très proches de ces dernières.

Les normes relatives du résidu pour les méthodes *IDIM-LS* et *DIDIM* sont respectivement de 7.6% et 6.4%.

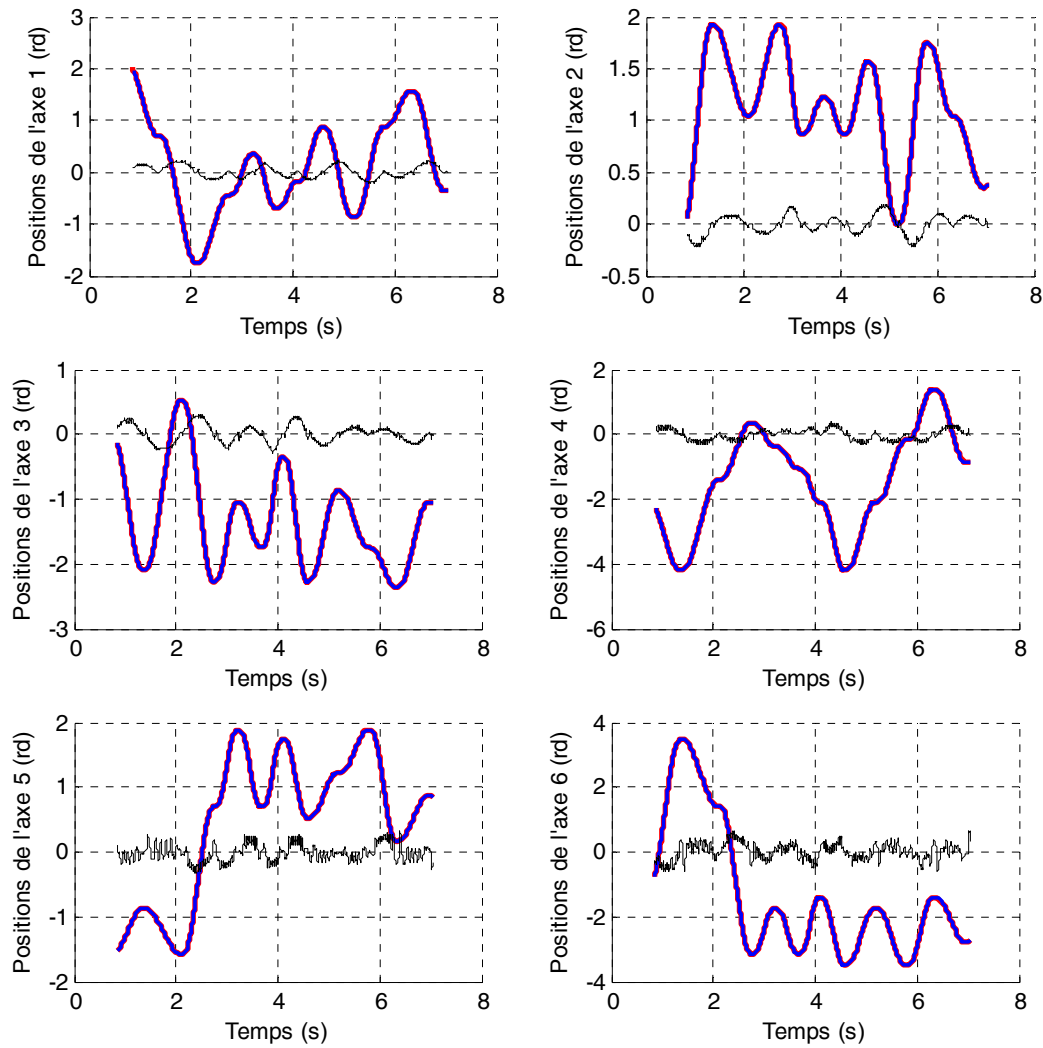


Figure 3-25 : Staubli TX40, Positions mesurées (rouge), positions simulées (bleu) et erreur x 200 (noir), algorithme de Gauss-Newton

Tableau 3-8 : Staubli TX40, Gauss-Newton, erreurs relatives entre les positions mesurées et simulées et entre les couples mesurés et simulés

Articulation	$\ \mathbf{Y}(\mathbf{q}) - \mathbf{Y}_s(\mathbf{q}_s) \ / \ \mathbf{Y}(\mathbf{q}) \ $	$\ \mathbf{Y}(\boldsymbol{\tau}) - \mathbf{Y}_s(\boldsymbol{\tau}_s) \ / \ \mathbf{Y}(\boldsymbol{\tau}) \ $
1	0.065 %	10.37 %
2	0.037 %	5.91 %
3	0.045 %	7.19 %
4	0.036 %	5.75 %
5	0.054 %	8.63 %
6	0.058 %	9.26 %

On remarque que les valeurs identifiées avec la méthode *CLOE* à erreur de position avec l'algorithme de Gauss-Newton et la méthode *DIDIM* sont proches.

Au vu des erreurs relatives entre les positions mesurées et simulées, l'approximation (3.19) $(\mathbf{q}_s, \dot{\mathbf{q}}_s, \ddot{\mathbf{q}}_s) \simeq (\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$ est respectée et permet la convergence de l'algorithme de *DIDIM* en 2 itérations seulement.

Les conditionnements des matrices d'observation pour les méthodes *IDIM-LS*, *DIDIM* et à erreur de sortie sont respectivement de $\text{cond}(\mathbf{W})=260$, $\text{cond}(\mathbf{W})=244$ et $\text{cond}(\mathbf{W}_g)>10^5$ (voir équation (3.12)). Un conditionnement proche de 1 indique un bon conditionnement du problème et un conditionnement élevé indique un mauvais conditionnement. Un mauvais conditionnement implique des difficultés et/ou un temps de convergence allongé pour calculer la solution numérique au problème car le critère est très peu sensible aux variations par rapport aux paramètres.

Les matrices d'observation des méthodes *IDIM-LS* et *DIDIM* sont bien conditionnées, par contre la matrice \mathbf{W}_g est très mal conditionnée. Cela est dû au fait que les erreurs de poursuite de trajectoire sont peu sensibles aux variations des paramètres. En conséquence, la méthode *CLOE* classique converge lentement et est sensible aux conditions initiales sur les paramètres.

Méthode avec l'algorithme SQP (Sequential Quadratic Programming)

Une méthode *CLOE* à erreur de position utilisant l'algorithme SQP (Sequential Quadratic Programming) est testée dans un deuxième temps. Cet algorithme est adapté pour résoudre les moindres carrés non linéaires. Les mêmes initialisations que pour la méthode Gauss-Newton sont testées.

La même initialisation que pour la méthode *DIDIM* ne permet pas la convergence de l'algorithme qui est stoppé au bout de 3h.

L'initialisation avec les paramètres identifiés avec *DIDIM* avec une erreur relative de 20% permet la convergence de l'algorithme en 23 itérations et 1h30 de temps de calcul. La méthode *SQP* converge donc plus lentement que la méthode de Gauss-Newton.

Les paramètres identifiés sont très proches des paramètres identifiés avec la méthode de Gauss-Newton (représentés tableau 3-7). Ils ne sont pas donnés ici. Les erreurs relatives sont données dans le tableau 3-9, ces dernières sont toutes inférieures à 0.1%. Les erreurs relatives sont très faibles car le robot Stäubli *TX40* est équipé de codeurs précis.

Tableau 3-9 : Stäubli *TX40*, *SQP*, erreurs relatives entre les positions mesurées et simulées et entre les couples mesurés et simulés

Articulation	$\ \mathbf{Y}(\mathbf{q}) - \mathbf{Y}_s(\mathbf{q}_s) \ / \ \mathbf{Y}(\mathbf{q}) \ $	$\ \mathbf{Y}(\boldsymbol{\tau}) - \mathbf{Y}_s(\boldsymbol{\tau}_s) \ / \ \mathbf{Y}(\boldsymbol{\tau}) \ $
1	0.066 %	10.45 %
2	0.036 %	5.70 %
3	0.047 %	7.44 %
4	0.036 %	5.70 %
5	0.054 %	8.55 %
6	0.059 %	9.34 %

Méthode du Simplex de Nelder-Mead

Une méthode *CLOE* à erreur de position utilisant l'algorithme de Nelder-Mead est testée dans un troisième temps (Lagarias et al. 1998) (Jubien et al. 2013). Il ne nécessite pas le calcul des fonctions de sensibilité. En conséquence, il n'est pas possible de calculer directement les écart-types relatifs des paramètres.

Seule l'initialisation (3.51) de la méthode *DIDIM* est testée. La solution optimale $\hat{\chi}$ minimise le même critère quadratique que précédemment (3.61). Le modèle n'est pas simplifié et seuls les paramètres de base sont identifiés. Les résultats sont donnés dans le tableau 3-10

. Les paramètres possédant des écart-types relatifs inférieurs à 25% sont de couleur rouge.

Contrairement à l'algorithme de Gauss-Newton et de *SQP*, l'algorithme du Simplex permet la convergence des paramètres avec la même initialisation que de la méthode *DIDIM*. Cet algorithme est beaucoup moins sensible aux conditions initiales, seules les moments d'inertie de la chaîne d'actionnement doivent être proche des valeurs réelles, les autres paramètres peuvent être initialisés à des valeurs nulles.

Cependant 4305 itérations et 9590 simulations du modèle dynamique direct sont nécessaires (plus de 150 par paramètres). Le temps de convergence est approximativement de 16 heures sur un processeur double-cœurs de 2 (GHz). Ceci est à comparer aux 2 itérations/simulations du modèle dynamique direct nécessaires pour la méthode *DIDIM* (<10 secondes).

Il n'est pas possible de calculer directement les écart-types relatifs avec cette méthode, ou toute autre valeur qui permettrait de connaître la confiance dans les paramètres identifiés. Il est possible de contourner le problème en utilisant les expressions de l'identification *IDIM-LS* sur les positions, vitesses et accélérations utilisées à la dernière itération de l'algorithme. Mais ceci nécessite la mesure des efforts moteurs. Il est aussi possible de calculer la matrice \mathbf{W}_δ par différences finies à la fin de l'algorithme et de calculer les écart-types relatifs.

Les paramètres identifiés avec la méthode *DIDIM* et qui possèdent des écart-types inférieur à 25% sont proches des paramètres identifiés avec la méthode *CLOE* usuelle avec l'algorithme du Simplex (paramètres repérés en rouge).

Les positions mesurées, les positions simulées et l'erreur entre ces dernières sont représentées figure 3-26. Les erreurs relatives sont toutes inférieures à 0.1% et très proches des valeurs données dans le tableau 3-8.

Tableau 3-10 : Stäubli *TX40*, paramètres de base identifiés avec la méthode *DIDIM* et la méthode *CLOE*

Méthode	<i>DIDIM</i>		<i>CLOE</i>	Méthode	<i>DIDIM</i>		<i>CLOE</i>
Par.	$\hat{\chi}^2$	$\% \hat{\sigma}_{\hat{\chi}_n}$	$\hat{\chi}^{4305}$	Par.	$\hat{\chi}^2$	$\% \hat{\sigma}_{\hat{\chi}_n}$	$\hat{\chi}^{4305}$
ZZ _{1R}	1.24	1.84	1.19	MY _{4R}	-0.00796	106	0.000111
FV ₁	8.04	1.1	10.1	Ia ₄	0.0302	11.2	0.0219
Fc ₁	6.95	3.64	8.16	Fv ₄	1.13	1.94	1.20
Off ₁	0.187	72.9	0.405	Fc ₄	2.38	3.37	2.33
XX _{2R}	-0.47	4.4	-0.418	Off ₄	-0.0932	45.1	-0.984
XY ₂	0.00411	243	0.0084	XX _{5R}	0.00394	78.2	0.0222
XZ _{2R}	-0.158	7.78	-0.178	XY ₅	0.000948	162	-0.0013
YZ ₂	-0.00222	371	0.161	XZ ₅	0.000755	215	-0.000181
ZZ _{2R}	1.06	1.73	1.01	YZ ₅	0.00095	173	0.0014
MX _{2R}	2.17	4.58	2.12	ZZ _{5R}	-0.000889	371	5.17.10-6
MY _{2R}	0.0579	124	-0.0469	MX _{5R}	-0.00769	96.3	0.000503
Fv ₂	5.65	1.77	5.77	MY _{5R}	-0.0303	18.8	-0.00173
Fc ₂	7.98	2.86	7.49	Ia ₅	0.0421	12.2	0.00483
Off ₂	0.737	144	-10.7	Fv ₅	1.82	2.39	1.90
XX _{3R}	0.119	14.9	0.112	Fc ₅	3	4.02	3.36
XY ₃	-0.00114	906	-0.0011	Off ₅	0.0159	483	-0.234
XZ ₃	-0.00393	273	0.0235	XX _{6R}	-0.00111	116	0.0028
YZ ₃	0.00286	232	0.0020	XY ₆	0.000133	501	0.000265
ZZ _{3R}	0.109	13.2	0.0102	XZ ₆	0.00184	52.3	0.0018
MX _{3R}	0.0669	62.3	0.0538	YZ ₆	0.000353	257	0.0018
MY _{3R}	-0.605	3.18	-0.0606	ZZ _{6R}	0.00212	54.3	0.000259
Ia ₃	0.0976	12.1	0.0078	MX _{6R}	0.00102	407	0.0101
Fv ₃	1.97	2.56	2.03	MY _{6R}	-0.00134	296	0.0165
Fc ₃	6.28	2.8	5.96	Ia ₆	0.0103	16.1	0.0095
Off ₃	0.217	152	-0.152	Iam ₆	0.0097	17.5	0.0082
XX _{4R}	0.00265	268	0.0031	Fv ₆	0.646	2.1	0.760
XY ₄	-0.00589	50.9	0.0029	Fc ₆	0.445	24.5	0.460
XZ ₄	-0.000676	529	-0.0019	Fvm ₆	0.618	2.08	0.662
YZ ₄	-0.00572	60.6	0.0031	Fcm ₆	1.76	4.99	1.98
ZZ _{4R}	-2.57e-05	142	1.60.10-7	Off ₆	0.141	29.6	0.0285
MX _{4R}	-0.0178	43.8	2.38.10-5				

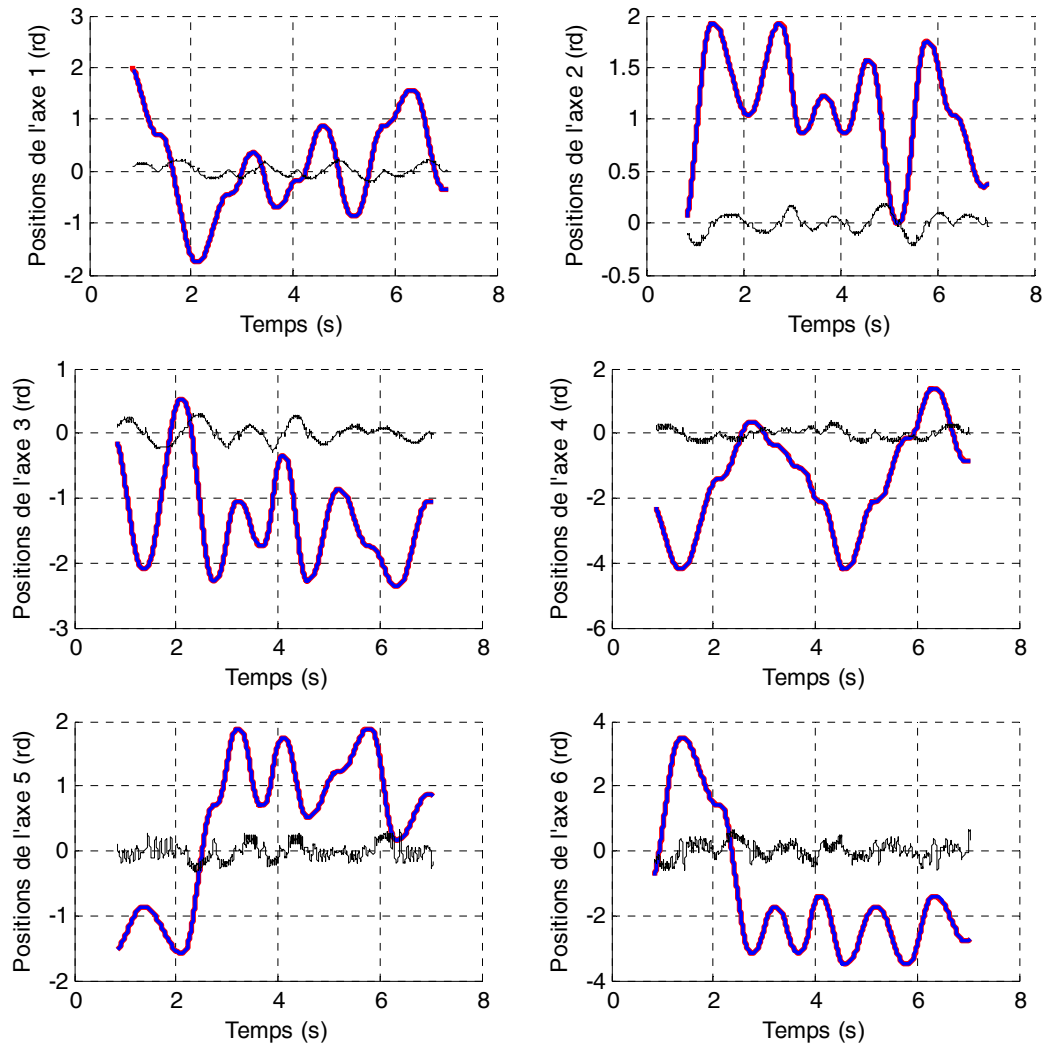


Figure 3-26 : Stäubli TX40, Positions mesurées (rouge), positions simulées (bleu) et erreur x 200 (noir), algorithme du Simplex

3.8.3 Discussion

La méthode *CLOE* classique à erreur de position a été testée avec l'algorithme de Gauss-Newton, avec l'algorithme *SQP* et avec l'algorithme du Simplex de Nelder Mead. La solution optimale minimise l'erreur quadratique entre les positions simulées et les positions mesurées pour donner les 61 paramètres de base du robot.

L'algorithme de Gauss-Newton et la méthode *SQP* étant sensibles à l'initialisation, les conditions initiales sur les paramètres doivent être proches des bonnes valeurs.

L'algorithme du Simplex permet l'utilisation de la même initialisation que *DIDIM*.

Les trois méthodes d'optimisation non linéaire ont un temps de convergence très long (de quasiment 1h à 16h) comparé à la méthode *DIDIM* qui ne nécessite qu'une dizaine de seconde.

La méthode *DIDIM* approche les fonctions de sensibilité grâce au modèle dynamique inverse, et utilise les outils numériques de la méthode *IDIM-LS* pour le calcul des écart-types relatifs. La solution optimale minimise l'erreur quadratique sur les efforts à la place des positions

car les efforts sont beaucoup plus sensibles aux variations des paramètres que les positions car le robot est commandé en boucle fermée (voir partie 3.8.1).

3.9 Comparaison expérimentale entre la méthode *CLOE* à erreur d'effort et la méthode *DIDIM*

Afin de compléter la comparaison entre les méthodes *CLOE* classiques et la méthode *DIDIM*, une comparaison est effectuée entre une méthode *CLOE* à erreur d'effort et la méthode *DIDIM*.

On rappelle que la méthode *DIDIM* est une méthode *CLOE* à erreur d'effort où le modèle dynamique inverse est utilisé pour approcher les fonctions de sensibilité de la matrice Jacobienne par l'équation (3.22).

Ici la comparaison est effectuée avec une méthode *CLOE* à erreur d'effort utilisant l'algorithme de Nelder-Mead qui évite le calcul des fonctions de sensibilité (Lagarias et al. 1998) (Bogdan et Abba, 2009) (Bogdan et Abba, 2012).

La solution optimale $\hat{\chi}$ minimise le critère quadratique suivant :

$$J(\chi) = \| \mathbf{Y} - \mathbf{Y}_s \|^2 = (\mathbf{Y} - \mathbf{Y}_s)^T (\mathbf{Y} - \mathbf{Y}_s) \quad (3.66)$$

Avec :

- \mathbf{Y} est le vecteur (rx1) obtenu après échantillonnage et filtrage des mesures de $\mathbf{y}=\boldsymbol{\tau}$ à la fréquence de mesure f_m sur une fenêtre d'observation T_{obs} ,
- \mathbf{Y}_s est le vecteur (rx1) obtenu après échantillonnage et filtrage des mesures de $\mathbf{y}_s=\boldsymbol{\tau}_s$ à la fréquence de mesure f_m sur une fenêtre d'observation T_{obs} .

L'initialisation (3.51) est utilisée. 1232 itérations et 2798 simulations du modèle dynamique direct sont nécessaires. Le temps de convergence est approximativement de 5 heures sur un processeur double-cœurs de 2 (GHz). Comme pour la méthode *CLOE* usuelle à erreur de position avec l'algorithme du Simplex, les paramètres identifiés avec la méthode *DIDIM* et qui possèdent des écart-types inférieur à 25% sont proches des paramètres identifiés avec la méthode *CLOE* à erreur d'effort avec l'algorithme du Simplex (non représentés ici).

La méthode *DIDIM* est donc 2000 fois plus rapide que la méthode *CLOE* à erreur d'effort utilisant l'algorithme du Simplex. La simplification utilisée par la méthode *DIDIM* pour approcher les fonctions de sensibilité permet une diminution importante du temps de calcul.

3.10 Conclusion

Ce chapitre a présenté la méthode *DIDIM* appliquée sur les robots rigides. Cette méthode itérative ne nécessite que la mesure des efforts moteurs et est économe en temps de calcul. Une variante a été proposée et permet de supprimer le léger biais qui pouvait apparaître dans les paramètres estimés avec la méthode originale.

Une comparaison analytique et expérimentale avec les méthodes *CLOE* classiques à erreur de position sur un robot à 6 degrés de liberté a été effectuée, elle montre clairement l'intérêt de la

méthode *DIDIM* face aux méthodes *CLOE* classiques qui se basent sur l'erreur entre les positions simulées et mesurées.

On remarque que pour la méthode du Simplex de Nelder-Mead, pour la même initialisation que la méthode *DIDIM*, le temps de calcul est 7000 fois plus important. Pour les méthodes de Gauss-Newton et *SQP* qui nécessitent de bonnes conditions initiales (différences de 20% par rapport aux paramètres identifiés avec la méthode *DIDIM*), le temps de calcul est entre 350 et 700 fois plus important.

Si les efforts moteurs peuvent s'exprimer linéairement par rapport aux paramètres et que la commande du robot est connue, la méthode *DIDIM* permet d'estimer les paramètres du robot en très peu d'itérations avec leurs écart-types. Aucune méthode *CLOE* classique ne peut permettre l'identification d'un robot avec de 61 paramètres en seulement 2 à 3 simulations du modèle dynamique direct.

Contrairement à la méthode *IDIM-LS*, la méthode *DIDIM* évite le calcul des vitesses et accélérations par filtrage passe-bande des positions articulaires grâce à l'utilisation du modèle dynamique direct.

Cette méthode présente un intérêt certain pour les robots flexibles pour lesquels les degrés de liberté élastiques ne sont pas mesurés. L'extension de cette méthode aux robots à flexibilités articulaires fait l'objet du chapitre suivant.

Chapitre 4 Méthode *DIDIM* appliquée aux robots à flexibilités articulaires

Sommaire:

4.1 Introduction	101
4.2 Etude préliminaire sur le bras <i>EMPS</i>	101
4.2.1 Identification statique et dynamique avec des essais à sortie bloquée	101
4.2.2 Identification du modèle dynamique rigide du bras <i>EMPS</i> avec <i>IDIM-LS</i>	105
4.2.3 Identification du modèle dynamique flexible du bras <i>EMPS</i> avec <i>IDIM-LS</i>	108
4.2.4 Etude de la convergence de la méthode <i>DIDIM</i> en fonction des conditions initiales	110
4.3 Extension de la méthode <i>DIDIM</i> aux robots à flexibilités articulaires	113
4.3.1 Introduction	113
4.3.2 Etape 1 : identification du modèle dynamique rigide du robot	113
4.3.3 Etape 2 : identification des raideurs et des rapports d'inerties	113
4.3.4 Etape 3 : identification globale du modèle dynamique flexible	117
4.3.5 Structure de l'algorithme pour les robots à flexibilités articulaires	118
4.4 Application au bras <i>EMPS</i>	118
4.4.1 Identification du modèle dynamique flexible avec la méthode <i>DIDIM</i>	118
4.4.2 Comparaison avec la méthode <i>CLOE</i> usuelle	123
4.5 Application au robot Kuka <i>LWR</i>	125
4.5.1 Introduction	125
4.5.2 Identification expérimentale du modèle dynamique rigide avec la méthode <i>IDIM-LS</i>	126
4.5.3 Identification expérimentale des raideurs articulaires avec les capteurs de couple	131
4.5.4 Identification du modèle dynamique flexible avec la méthode <i>DIDIM</i>	137
4.5.4.1 Introduction	137
4.5.4.2 Paramètres utilisés pour la simulation	141
4.5.4.3 Méthode <i>DIDIM</i> sans bruit	143
4.5.4.4 Méthode <i>DIDIM</i> avec bruit	151
4.6 Conclusion	156

Résumé: Ce chapitre présente l'extension de la méthode *DIDIM* aux robots à flexibilités articulaires. Après une étude expérimentale sur les propriétés de convergence de la méthode *DIDIM* sur un robot flexible un axe (le bras *EMPS*), une méthode en trois étapes est proposée pour identifier tous les paramètres dynamiques d'un robot à flexibilités articulaires avec seulement les efforts moteurs. Cette méthode est appliquée expérimentalement sur le bras *EMPS*. Elle est ensuite appliquée sur des mesures produites en simulation (sans et avec bruit) sur un robot 7 axes (le Kuka

LWR) après une identification des trois premières raideurs grâce aux mesures des couples côté corps.

4.1 Introduction

Dans le chapitre précédent, il a été vu que la méthode *DIDIM* présente un intérêt certain pour les robots à flexibilités articulaires car contrairement à la méthode *IDIM-LS* et aux méthodes *CLOE* classiques à erreur de position, la mesure des positions n'est pas nécessaire, en particulier celle des degrés de liberté élastiques.

Pour les robots à flexibilités articulaires, les degrés de liberté élastiques ne sont ni observés ni actionnés.

Rappelons que la méthode *DIDIM* converge seulement si l'approximation suivante est respectée :

$$\left(\mathbf{q}_s(\hat{\chi}^k), \dot{\mathbf{q}}_s(\hat{\chi}^k), \ddot{\mathbf{q}}_s(\hat{\chi}^k) \right) = (\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) \text{ pour tout } \hat{\chi}^k \quad (4.1)$$

Pour les robots rigides, le respect de cette convergence s'effectue en simulant le robot avec la même commande que celle du robot réel (voir figure 3-5 et figure 3-23) et en initialisant la matrice d'inertie avec les inerties des rotors des moteurs (voir équations (3.51) et (3.34)). Ces conditions initiales doivent être adaptées pour les robots à flexibilités articulaires.

Une étude préliminaire est effectuée sur un robot un axe avec une flexibilité articulaire, le bras *EMPS*. Après une identification de la raideur avec des essais à sortie bloquée et une identification des modèles dynamiques rigide et flexible avec la méthode *IDIM-LS*, une étude de la convergence de la méthode *DIDIM* est effectuée pour évaluer la robustesse de l'algorithme vis à vis des conditions initiales.

4.2 Etude préliminaire sur le bras EMPS

4.2.1 Identification statique et dynamique avec des essais à sortie bloquée

L'identification de la raideur statique est d'abord effectuée avec un essai à sortie bloquée avec une consigne de position moteur constante. Ensuite, l'identification des paramètres de la chaîne d'actionnement et de la raideur dynamique est effectuée avec un essai à sortie bloquée. Ces deux identifications, simples à mettre en œuvre sur le bras *EMPS*, permettent d'avoir une première estimation des paramètres considérés (Janot, 2007).

Pour chacun des essais suivants, l'axe de la vis à billes est bloqué avec un serre-joint afin de fixer la position q_{L1} à une valeur constante nommée q_{L0} , en conséquence $q_{m1} = -q_{e1} + q_{L0}$.

La fréquence des mesures pour le bras *EMPS* est fixée à $f_m = 1$ (kHz).

Essai statique avec une consigne de position moteur constante

La consigne q_r est fixée à 0.13 mm (un seul palier). L'effort moteur est représenté figure 4-1 et la position du moteur q_{m1} est représentée figure 4-2. Avec un correcteur *PD* (voir partie 1.3), sans action intégrale, la position moteur s'établit à 0.08 (mm).

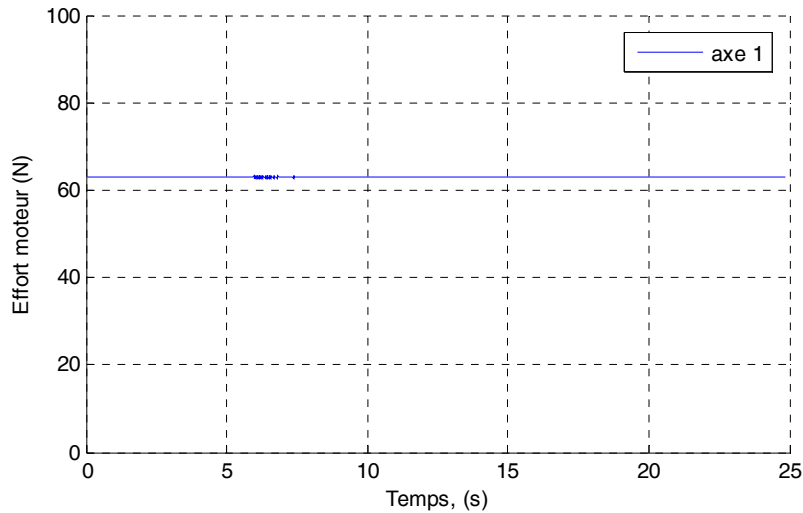


Figure 4-1 : Bras *EMPS*, effort moteur, essai statique

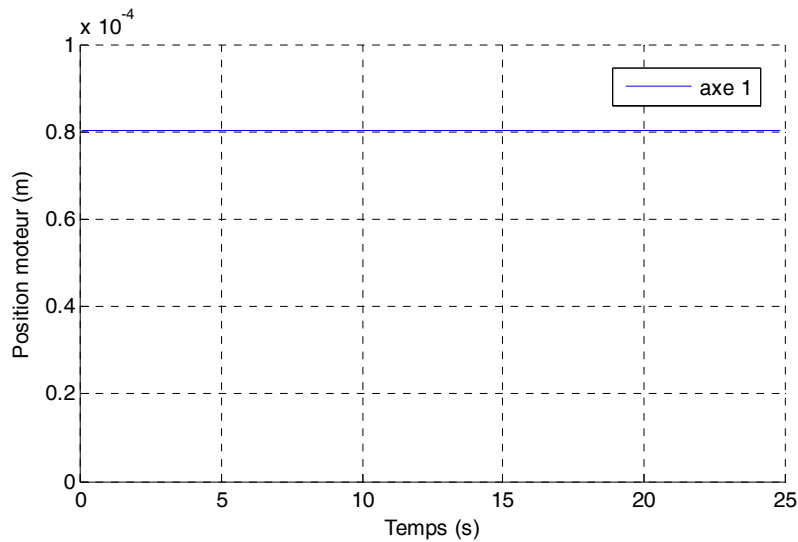


Figure 4-2 : Bras *EMPS*, position moteur, essai statique

L'effort moteur exprimé linéairement par rapport aux paramètres pour cet essai est le suivant :

$$\tau_{m_{di}} = \mathbf{MDI} \boldsymbol{\chi} = \begin{bmatrix} 1 & q_{m_1} \end{bmatrix} \begin{bmatrix} \text{Off}k_1 & k_1 \end{bmatrix}^T \quad (4.2)$$

Où $offk_1$ est un terme d'offset qui prend en compte l'offset de l'amplificateur de courant, le frottement sec d'adhérence et le terme $-k_1q_{1_0}$.

La valeur numérique de la raideur statique vaut $7.82 \cdot 10^5$ (N/m) avec un écart-type de 1.2% pour une fréquence de coupure des filtres Decimate et Butterworth réglée à 1 (Hz) (l'offset identifié vaut 164 (N))

Le vecteur de mesure Y se superpose au vecteur du modèle $W\chi$, représenté figure 4-3.

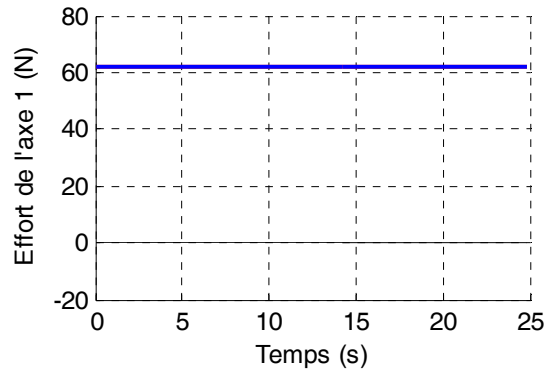


Figure 4-3 : Bras *EMPS*, essai statique à sortie bloquée, validation directe, Y (rouge), $W\chi$ (bleu) et erreur (noir)

Essai dynamique

L'effort moteur exprimé linéairement par rapport aux paramètres pour cet essai en dynamique à sortie bloquée est le suivant :

$$\tau_{mdi} = \mathbf{MDI} \chi \quad (4.3)$$

Avec :

- $\mathbf{MDI} = \begin{bmatrix} \ddot{q}_{m_1} & \dot{q}_{m_1} & \text{sign}(\dot{q}_{m_1}) & 1 & q_{m_1} \end{bmatrix}$,
- $\chi = [Iam_1 \quad Fvm_1 \quad Fcm_1 \quad Offk_1 \quad k_1]^T$.

Où $offk_1$ est un terme d'offset qui prend en compte l'offset de l'amplificateur de courant et le terme $-k_1q_{1_0}$.

La consigne de position est calculée avec un générateur de mouvement point à point avec un profil de trapèze en vitesse. Comme précédemment l'absence de correction intégrale et la force de rappel de la raideur produit un écart entre q_{r1} et la position moteur q_{m1} . La fréquence de coupure du filtre de Butterworth est réglée à 200 (Hz) et la fréquence de coupure du filtre Decimate est

réglée à 60 (Hz). L'effort moteur est représenté figure 4-4 et la position et la vitesse du moteur q_{m1} sont représentées figure 4-5. Les paramètres identifiés sont donnés dans le tableau 4-1.

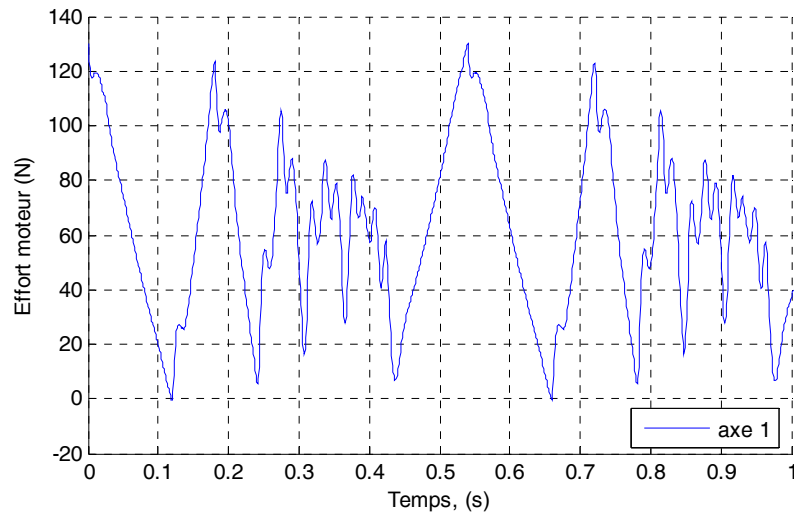


Figure 4-4 : Bras *EMPS*, effort moteur, essai dynamique

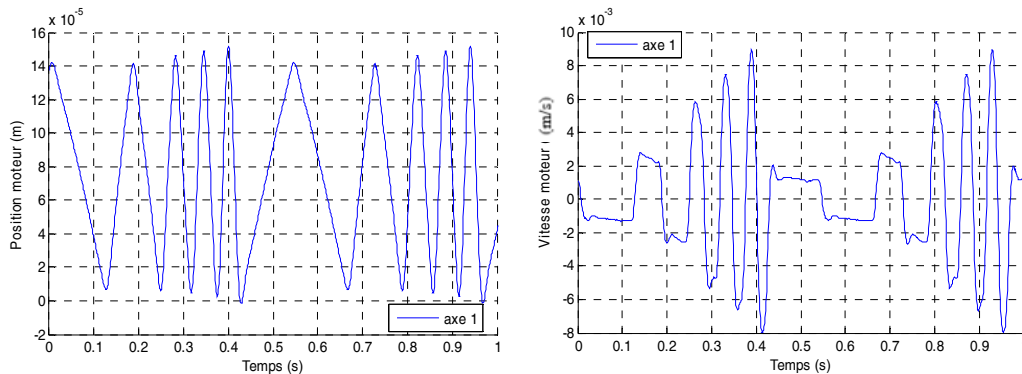


Figure 4-5 : Bras *EMPS*, position et vitesse moteur, essai dynamique

Tableau 4-1 : Bras *EMPS*, essai à sortie bloquée, paramètres identifiés

Paramètre	$\hat{\chi}$	% $\hat{\sigma}_{\chi}$
Iam ₁	64.8 (kg)	0.42
Fcm ₁	9.13 (N/m/s)	2.3
Offm ₁	-9.47 (N)	2.0
k ₁	8.65 10⁵ (N/m)	0.24

Le frottement visqueux n'est pas identifiable avec cet essai, cependant les autres paramètres du modèle sont bien identifiés avec une norme relative du résidu de 6.26%. Le vecteur de mesure \mathbf{Y} est proche du modèle $\mathbf{W}\hat{\chi}$, (figure 4-6). La valeur identifiée de la raideur dynamique diffère de 10.6% de la raideur statique par rapport à celle obtenue avec l'essai statique. Cette différence peut

être due à l'effet du frottement sec non pris en compte dans l'essai statique et à l'utilisation d'un seul palier de position.

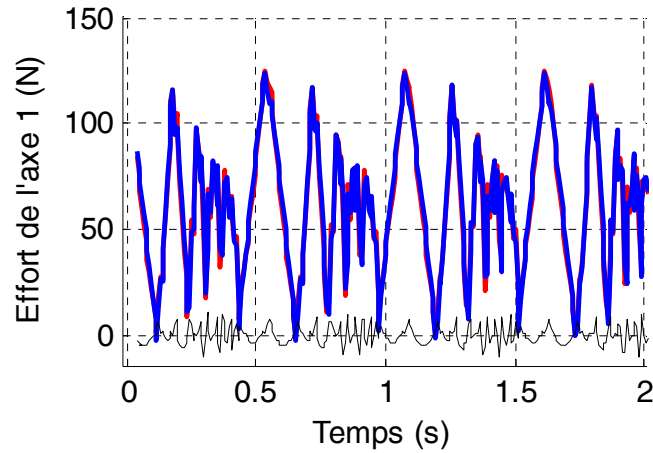


Figure 4-6 : Bras *EMPS*, essai à sortie bloquée en dynamique, validation directe, Y (rouge), W_χ (bleu) et erreur (noir)

4.2.2 Identification du modèle dynamique rigide du bras *EMPS* avec *IDIM-LS*

Ici, l'axe de la vis à billes n'est plus bloqué, le degré de liberté flexible du robot est laissé libre. Dans un premier temps, le modèle dynamique rigide du bras *EMPS* est identifié avec la méthode *IDIM-LS*. La fréquence de coupure du filtre de Butterworth est fixée à 20 (Hz) et la fréquence de coupure du filtre Decimate est fixée à 5 (Hz).

La consigne de position est calculée par le générateur de mouvement point à point avec un profil de vitesse en trapèze et des allers-retours du chariot d'amplitude 25 cm. La consigne de position et la consigne de vitesse sont représentées figure 4-7 et figure 4-8 respectivement. L'effort moteur mesuré est représenté figure 4-9.

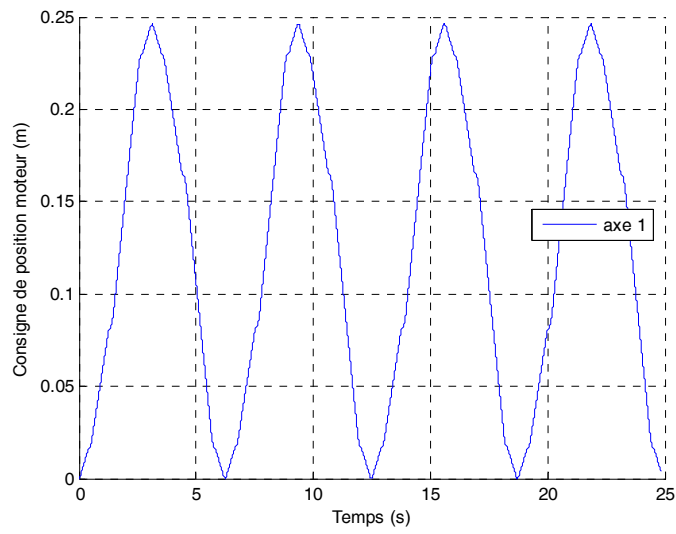


Figure 4-7 : Bras *EMPS*, consigne de position moteur

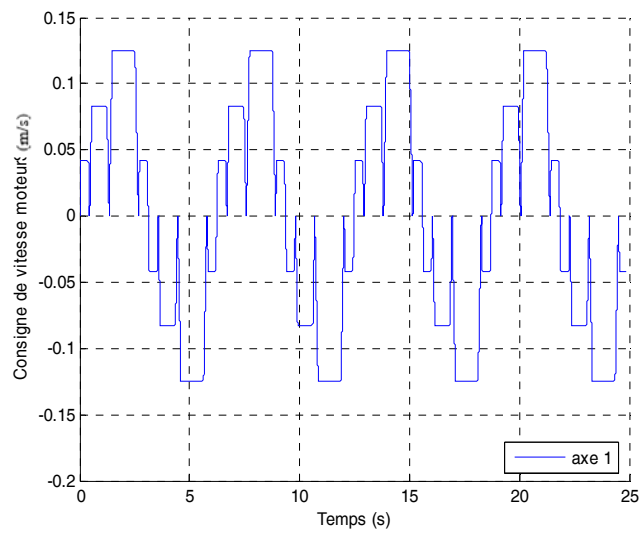


Figure 4-8 : Bras *EMPS*, consigne de vitesse moteur

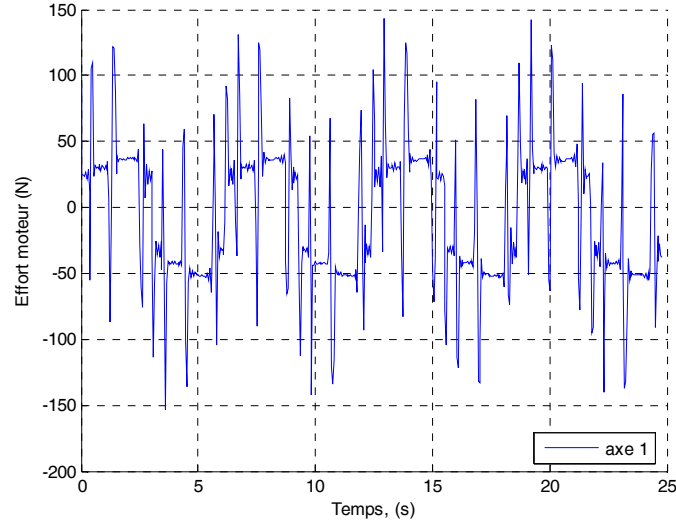


Figure 4-9 : Bras *EMPS*, effort moteur

L'identification expérimentale est effectuée avec une charge de 9.916 (kg) (± 5 (g)) avec la méthode décrite dans la partie (2.10). Seule la masse M_L de la charge est identifiable parmi ses 10 paramètres inertiels car le chariot du robot ne bouge qu'en translation.

Le modèle dynamique inverse du bras *EMPS* rigide sans charge est rappelé (voir paragraphe 1.3.1):

$$\tau_{\text{mdi}} = \tau_{\text{mdi}_{m_1}} = M_{1R}\ddot{q}_{m_1} + Fv_1\dot{q}_{m_1} + Fc_1\text{sign}(\dot{q}_{m_1}) + \text{Off}_1 \quad (4.4)$$

L'effort moteur s'exprime linéairement par rapport aux paramètres dynamiques de la manière suivante (avec identification de la charge, voir paragraphe 2.10):

$$\begin{bmatrix} \tau_{\text{mdi}_a} \\ \tau_{\text{mdi}_b} \end{bmatrix} = \begin{bmatrix} \tau_{\text{mdi}_{m_1a}} \\ \tau_{\text{mdi}_{m_1b}} \end{bmatrix} = \mathbf{MDI} \boldsymbol{\chi} \quad (4.5)$$

avec : $\mathbf{MDI} = \begin{bmatrix} \ddot{q}_{m_1a} & \dot{q}_{m_1a} & \text{sign}(\dot{q}_{m_1a}) & 1 & 0 \\ \ddot{q}_{m_1b} & \dot{q}_{m_1b} & \text{sign}(\dot{q}_{m_1b}) & 1 & \ddot{q}_{m_1b} \end{bmatrix}$, $\boldsymbol{\chi} = [M_{1R} \quad Fv_1 \quad Fc_1 \quad \text{Off}_1 \quad M_L]^T$

L'indice (a) correspond à l'essai à vide et l'indice (b) correspond à l'essai en charge.

Les paramètres identifiés du modèle dynamique rigide sont donnés dans le tableau 4-2. Le vecteur des mesures \mathbf{Y} avec le vecteur des mesures estimé par le modèle $\mathbf{W}\hat{\boldsymbol{\chi}}$ sont représentés figure 4-10 pour l'essai à vide.

Tableau 4-2 : Bras *EMPS*, paramètres identifiés du modèle dynamique rigide avec la méthode *IDIM-LS*

Paramètre	$\hat{\chi}$	% $\hat{\sigma}_{\hat{\chi}}$
M_{1R}	105 (kg)	0.37
F_{V1}	217 (N/m/s)	1.16
F_{c1}	19.8 (N)	1.19
Off_1	-5.24 (N)	1.83
M_L	10.9 (kg)	0.98

Tous les paramètres sont bien identifiés. En particulier la masse $M_L=10.9$ (kg) recoupe la valeur pesée (9.916 (kg)). La norme relative du résidu est de 4.44%. Ces valeurs serviront de référence par la suite.

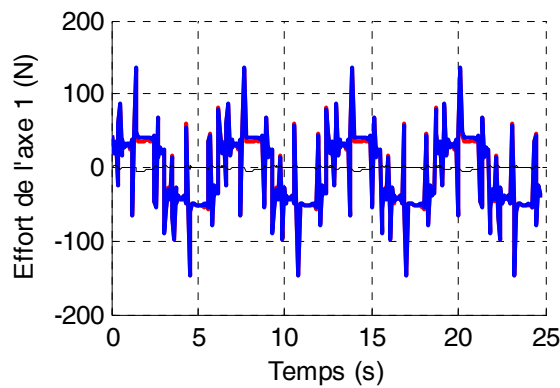


Figure 4-10 : Bras *EMPS*, modèle rigide, validation directe, Y (rouge), $W\chi$ (bleu) et erreur (noir)

4.2.3 Identification du modèle dynamique flexible du bras *EMPS* avec *IDIM-LS*

La méthode *IDIM-LS* est maintenant utilisée pour identifier le modèle dynamique flexible du bras *EMPS*. Elle est applicable sur l'*EMPS* car le bras est équipé d'un codeur haute-résolution pour mesurer la position moteur q_{m1} et d'un codeur haute-résolution pour mesurer la position de la vis après flexibilité et calculer avec précision la position q_{L1} du chariot.

Ces deux codeurs haute-résolution utilisés en quadrature (12500 traits par tours soit 50000 impulsions par tour) permettent de calculer avec précision l'amplitude du degré de liberté élastique $q_{e1} = q_{L1} - q_{m1}$.

Le modèle dynamique inverse du bras *EMPS* flexible sans charge est rappelé et permet de calculer l'effort moteur en fonction des positions et de leurs dérivées de la manière suivante :

$$\begin{aligned} \tau_{mdi_m1} &= Iam_1 \ddot{q}_{m1} + Fvm_1 \dot{q}_{m1} + Fcm_1 \text{sign}(\dot{q}_{m1}) - k_1 q_{e1} + Offm_1 \\ 0 &= M_{1R} \ddot{q}_{L1} + FvL_1 \dot{q}_{L1} + FcL_1 \text{sign}(\dot{q}_{L1}) + k_1 q_{e1} \end{aligned} \quad (4.6)$$

L'effort moteur peut s'exprimer linéairement par rapport aux paramètres dynamiques comme suit :

$$\begin{bmatrix} \tau_{\text{mdi}_a} \\ \tau_{\text{mdi}_b} \end{bmatrix} = \begin{bmatrix} \tau_{\text{mdi}_{m_{1a}}} \\ 0 \\ \tau_{\text{mdi}_{m_{1b}}} \\ 0 \end{bmatrix} = \mathbf{MDI} \boldsymbol{\chi} \quad (4.7)$$

Avec :

$$\bullet \mathbf{MDI} = \begin{bmatrix} \ddot{q}_{m_{1a}} & \dot{q}_{m_{1a}} & \text{sign}(\dot{q}_{m_{1a}}) & 1 & -q_{e_{1a}} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & q_{e_{1a}} & \ddot{q}_{L_{1a}} & \dot{q}_{L_{1a}} & \text{sign}(\dot{q}_{L_{1a}}) & 0 \\ \ddot{q}_{m_{1b}} & \dot{q}_{m_{1b}} & \text{sign}(\dot{q}_{m_{1b}}) & 1 & -q_{e_{1b}} & 0 & 0 & 0 & \ddot{q}_{L_{1b}} \\ 0 & 0 & 0 & 0 & q_{e_{1b}} & \ddot{q}_{L_{1b}} & \dot{q}_{L_{1b}} & \text{sign}(\dot{q}_{L_{1b}}) & \ddot{q}_{L_{1b}} \end{bmatrix},$$

$$\bullet \boldsymbol{\chi} = [\text{Iam}_1 \quad \text{Fvm}_1 \quad \text{Fcm}_1 \quad \text{Offm}_1 \quad k_1 \quad M_{1R} \quad \text{FvL}_1 \quad \text{FcL}_1 \quad M_L]^T.$$

Par rapport à la consigne en position utilisée pour le modèle rigide, on choisit ici d'ajouter un sinus wobulé à faible amplitude (0.13 (mm)) qui varie de 0 à 50 (Hz) avec une période de 6.2 (s) afin d'exciter la flexibilité (voir partie (2.7)).

Les résultats d'identification sont donnés dans le tableau 4-3.

Tableau 4-3 : Bras *EMPS*, paramètres identifiés du modèle dynamique flexible avec la méthode *IDIM-LS*

Paramètre	$\hat{\boldsymbol{\chi}}$	% $\hat{\sigma}_{\boldsymbol{\chi}}$
Iam_1	69 (kg)	0.20
Fvm_1	105 (N/m/s)	1.14
Fcm_1	9.56 (N)	0.94
Off_1	-6.48 (N)	0.58
k_1	8.03 10⁵ (N/m)	0.48
M_{1R}	34 (kg)	0.87
FvL_1	97 (N/m/s)	0.94
FcL_1	9.75 (N)	0.31
M_L	9.92 (N)	1.01

Tous les paramètres sont bien identifiés. La raideur identifiée est proche des valeurs statique et dynamique identifiées avec les essais à sortie bloquée. Le vecteur de mesure \mathbf{Y} est très proche du modèle $\mathbf{W}\boldsymbol{\chi}$ (voir figure 4-12), les oscillations (figure de droite) sont bien reconstruites. La norme relative du résidu est de 7.34%. La masse de la charge est bien identifiée. Les valeurs identifiées avec la méthode *IDIM-LS* vont servir de référence pour la suite. L'histogramme de l'erreur est représenté figure 4-11, le test de normalité de Kolmogorov-Smirnov vérifie bien que l'erreur suit une loi de distribution gaussienne.

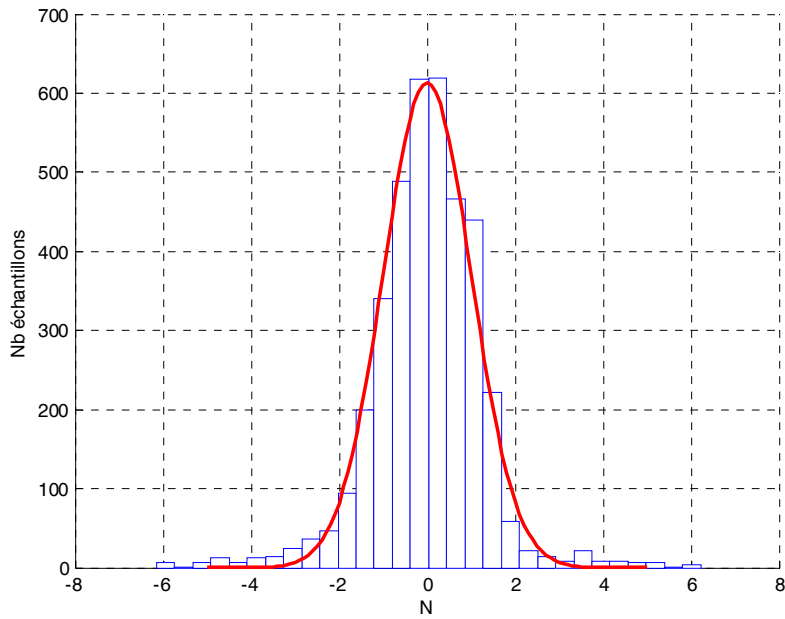


Figure 4-11 : *EMPS*, Histogramme des erreurs (en bleu) et Gaussienne (en rouge) avec la méthode *IDIM-LS*

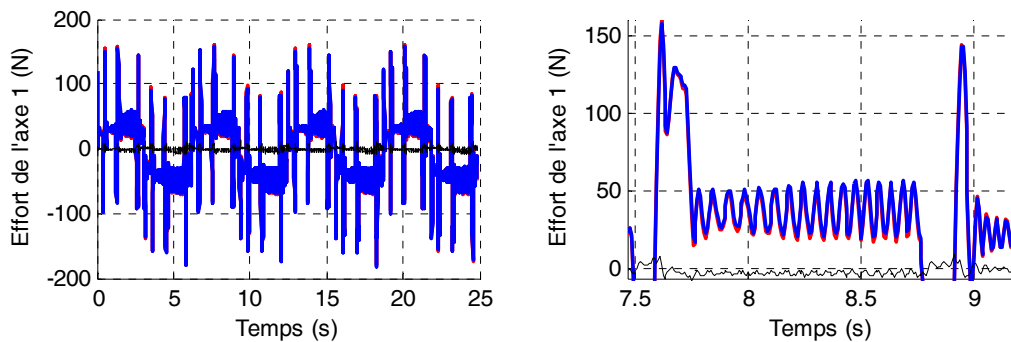


Figure 4-12: Bras *EMPS*, modèle flexible, validation directe, Y (rouge), $W\chi$ (bleu) et erreur (noir), sans charge

4.2.4 Etude de la convergence de la méthode *DIDIM* en fonction des conditions initiales

Il est nécessaire de trouver les bonnes conditions initiales sur les raideurs et les inerties pour que la méthode *DIDIM* converge.

Une étude est effectuée sur le bras *EMPS* à vide en initialisant les valeurs de I_{am1} , M_{1R} et k_1 du modèle dynamique flexible à différentes valeurs, les autres paramètres sont mis à 0.

Le simulateur utilisé est programmé avec le logiciel Matlab/Simulink et est représenté figure 4-13. Le contrôleur et le modèle dynamique direct du robot sont représentés figure 4-14 et figure 4-15 respectivement.

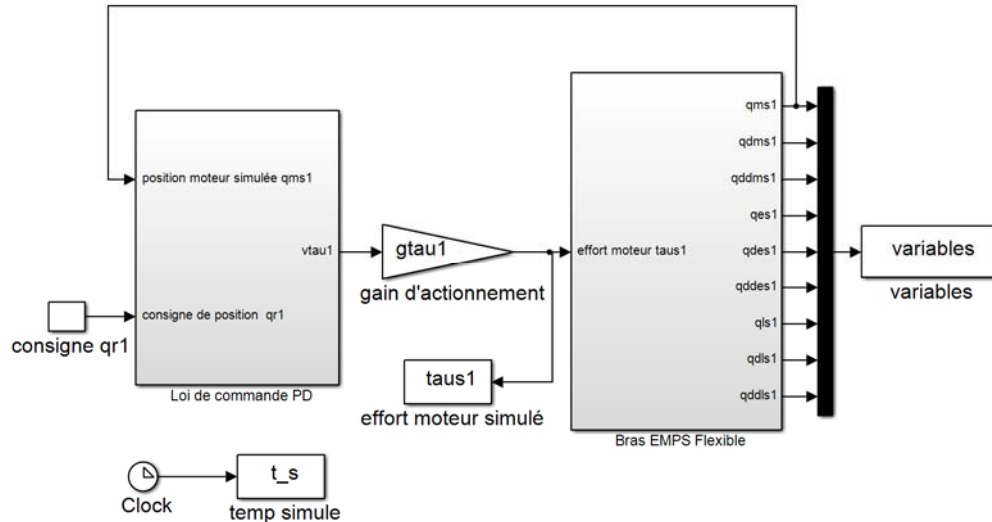


Figure 4-13: Simulateur du bras *EMPS* flexible sous Matlab/Simulink

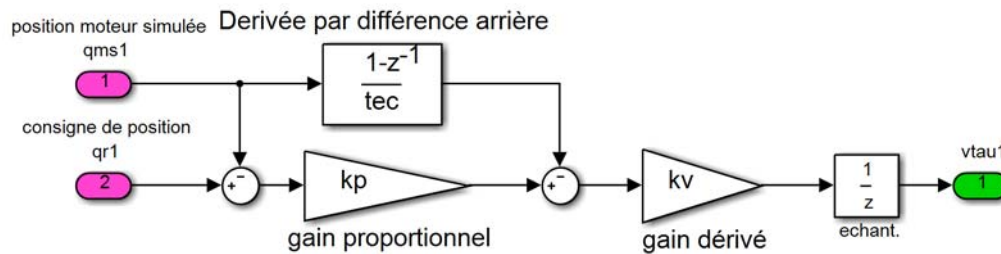


Figure 4-14 : Contrôleur du bras *EMPS* flexible

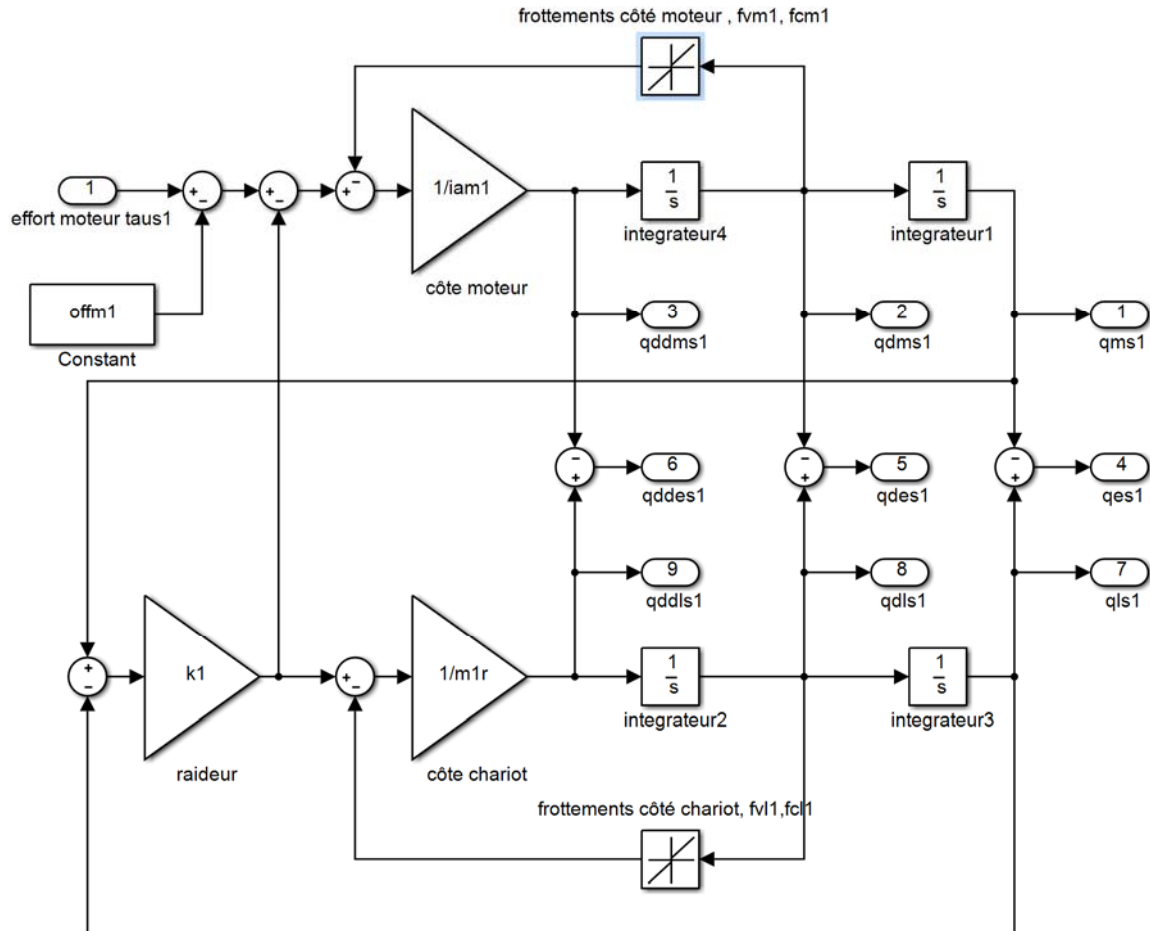


Figure 4-15 : Simulateur du bras *EMPS* flexible

Les fréquences de coupure des filtres de Butterworth et Decimate sont fixées aux mêmes fréquences que pour l'identification du modèle dynamique flexible avec la méthode *IDIM-LS*.

Les conditions initiales sur les paramètres sont des valeurs pondérées des paramètres identifiés avec la méthode *IDIM-LS*.

La méthode converge uniquement si l'erreur sur les conditions initiales reste inférieure à 15% de la valeur des paramètres identifiés avec la méthode *IDIM-LS*.

Cette étude montre clairement que la méthode *DIDIM* est beaucoup plus sensible aux conditions initiales sur les robots à flexibilités articulaires que sur les robots rigides.

Il est nécessaire d'estimer à des valeurs proches (minimum 85%) les inerties et les raideurs afin d'initialiser correctement la méthode *DIDIM* pour identifier le modèle dynamique flexible du robot. La méthode proposée est décrite dans la partie suivante.

4.3 Extension de la méthode *DIDIM* aux robots à flexibilités articulaires

4.3.1 Introduction

Dans la plupart des cas, il est toujours possible d'avoir une idée des premiers modes d'un robot à flexibilités articulaires. Soit par le fabricant du robot (rare) ou soit par le fabricant des réducteurs, ou soit par la réponse vibratoire au sonnage du marteau. Par contre il est plus difficile d'avoir accès aux inerties de part et d'autre des flexibilités.

Il est possible d'identifier le modèle dynamique rigide complet du robot avec la méthode *IDIM-LS* ou la méthode *DIDIM* car dans ce cas la mesure des déformations élastiques n'est pas nécessaire. Ce qui permet de connaître les frottements totaux et la somme des inerties avant et après flexibilités pour chaque articulation du robot.

En conséquence, une procédure en 3 étapes est proposée pour identifier les robots à flexibilités articulaires.

La première étape consiste à identifier le modèle dynamique rigide du robot avec la méthode *DIDIM*. Cette étape permet d'identifier la somme des frottements et des inerties avant et après flexibilités en plus des paramètres inertiels des corps.

La deuxième étape permet d'estimer les raideurs et les rapports d'inerties de part et d'autre des flexibilités en utilisant les paramètres identifiés à la première étape. Ils sont estimés par moindres carrés non-linéaires. L'algorithme d'optimisation du Simplex de Nelder-Mead est utilisé car il est plus robuste aux conditions initiales que celui de Gauss-Newton ou de *SQP*. Cette étape permet de connaître de manière approchée les rapports d'inerties et les raideurs.

J'ai choisi d'estimer les rapports d'inerties plutôt que les inerties car ce sont des paramètres sans dimension compris entre 0 et 1, facile à interpréter pour chaque articulation j et pour tout robot.

La troisième étape est l'application de la méthode *DIDIM* sur le modèle dynamique flexible complet du robot, en utilisant les valeurs identifiées dans les étapes 1 et 2. La méthode *DIDIM* permet d'estimer avec plus de précision tous les paramètres du modèle dynamique flexible simultanément (inerties, raideurs et frottements).

Cette procédure en trois étapes est détaillée par la suite.

4.3.2 Etape 1 : identification du modèle dynamique rigide du robot

La méthode *DIDIM* est appliquée pour identifier les paramètres du modèle dynamique rigide du robot dans les basses fréquences. Pour cela, la fréquence de coupure du filtre Decimate doit être fixée à une valeur inférieure au premier mode flexible du robot. L'algorithme est initialisé avec (voir équation (3.34)):

$${}^{ap}M_{I_j} = I_{a_j} N_j^2 \quad (4.8)$$

4.3.3 Etape 2 : identification des raideurs et des rapports d'inerties

Théorie

On définit le rapport d'inertie suivant pour l'articulation i_{art} du robot réel :

$$\begin{aligned}
 {}^a\lambda_{i_{art}} &= \frac{{}^aI_{am_{i_{art}}}}{{}^aI_{am_{i_{art}}} + {}^aM_{i_{art}} + {}^aI_{al_{i_{art}}}} \text{ pour une articulation prismatique (p) } i_{art} \\
 {}^a\lambda_{i_{art}} &= \frac{{}^aI_{am_{i_{art}}}}{{}^aI_{am_{i_{art}}} + {}^aZZ_{i_{art}} + {}^aI_{al_{i_{art}}}} \text{ pour une articulation rotoïde (r) } i_{art}
 \end{aligned} \tag{4.9}$$

Où :

- ${}^aM_{i_{art}}$ (kg) est la masse du corps $C_{i_{art}}$,
- ${}^aZZ_{i_{art}}$ (kg.m²) est le moment d'inertie du corps i_{art} autour de Z_j ,
- ${}^aI_{al_{i_{art}}}$ (kg (p.) ou kg.m² (r.)) est le moment d'inertie des éléments de la chaîne d'actionnement après flexibilités, côté corps, de l'articulation i_{art} ,
- ${}^aI_{am_{i_{art}}}$ (kg (p.) ou kg.m² (r.)) est le moment d'inertie des éléments de la chaîne d'actionnement avant flexibilité, côté moteur, de l'articulation i_{art} .

Les vecteurs ($n_{art} \times 1$) suivants sont définis :

$${}^a\mathbf{k} = \begin{bmatrix} {}^ak_1 \\ {}^ak_2 \\ \vdots \\ {}^ak_{n_{art}} \end{bmatrix} \tag{4.10}$$

$${}^a\boldsymbol{\lambda} = \begin{bmatrix} {}^a\lambda_1 \\ {}^a\lambda_2 \\ \vdots \\ {}^a\lambda_{n_{art}} \end{bmatrix} \tag{4.11}$$

Où ${}^ak_{i_{art}}$ (N/m (p.) ou Nm/rd (r.)) est la raideur articulaire de l'articulation i_{art} ,

Variables de l'algorithme d'optimisation non linéaire

Cette étape consiste à estimer les valeurs $\hat{\mathbf{k}}$ et $\hat{\boldsymbol{\lambda}}$ de ${}^a\mathbf{k}$ et ${}^a\boldsymbol{\lambda}$ pour permettre l'initialisation de l'algorithme de *DIDIM* décrite à la prochaine étape.

Les vecteurs optimaux minimisent le critère quadratique suivant :

$$(\hat{\mathbf{k}}, \hat{\boldsymbol{\lambda}}) = \underset{\mathbf{k}, \boldsymbol{\lambda}}{\operatorname{argmin}} \|\mathbf{Y} - \mathbf{Y}_s\|^2 \tag{4.12}$$

Où \mathbf{Y} et \mathbf{Y}_s sont respectivement les vecteurs obtenus après échantillonnage et filtrage des efforts réel et simulé.

Dans cette étape, seuls les vecteurs \mathbf{k} et $\boldsymbol{\lambda}$ sont estimés avec l'algorithme du Simplex de Nelder Mead (Lagarias et al, 1998).

Paramètres du simulateur du robot

A chaque itération de l'optimisation non linéaire, les raideurs varient dans la simulation du robot selon ${}^k \hat{\mathbf{k}}$. Les inerties/masses varient selon ${}^k \hat{\boldsymbol{\lambda}}$:

$$\begin{aligned} {}^k \mathbf{Iam}_{i_{art}} &= {}^k \hat{\boldsymbol{\lambda}}_{i_{art}}^{ap} \left(\mathbf{Iam}_{i_{art}} + \mathbf{M}_{i_{art}} + \mathbf{IaL}_{i_{art}} \right) \\ {}^k \mathbf{M}_{i_{art}} &= \left(1 - {}^k \hat{\boldsymbol{\lambda}}_{i_{art}} \right)^{ap} \left(\mathbf{Iam}_{i_{art}} + \mathbf{M}_{i_{art}} + \mathbf{IaL}_{i_{art}} \right) \text{ pour une articulation prismatique } i_{art} \quad (4.13) \\ {}^k \mathbf{IaL}_{i_{art}} &= 0 \end{aligned}$$

$$\begin{aligned} {}^k \mathbf{Iam}_{i_{art}} &= {}^k \hat{\boldsymbol{\lambda}}_{i_{art}}^{ap} \left(\mathbf{Iam}_{i_{art}} + \mathbf{ZZ}_{i_{art}} + \mathbf{IaL}_{i_{art}} \right) \\ {}^k \mathbf{ZZ}_{i_{art}} &= \left(1 - {}^k \hat{\boldsymbol{\lambda}}_{i_{art}} \right)^{ap} \left(\mathbf{Iam}_{i_{art}} + \mathbf{ZZ}_{i_{art}} + \mathbf{IaL}_{i_{art}} \right) \text{ pour une articulation rotoïde } i_{art} \quad (4.14) \\ {}^k \mathbf{IaL}_{i_{art}} &= 0 \end{aligned}$$

Où :

- ${}^k \mathbf{ZZ}_{i_{art}}$ est le terme (3,3) du tenseur d'inertie du corps i_{art} à l'itération k ,
- ${}^k \mathbf{M}_{i_{art}}$ est la masse du corps i_{art} à l'itération k ,
- ${}^k \mathbf{IaL}_{i_{art}}$ est le moment d'inertie des éléments de la chaîne d'actionnement après flexibilités de l'articulation i_{art} à l'itération k ,
- ${}^k \mathbf{Iam}_{i_{art}}$ est le moment d'inertie des éléments de la chaîne d'actionnement avant flexibilité de l'articulation i_{art} à l'itération k ,
- ${}^k \hat{\boldsymbol{\lambda}}_{i_{art}}$ est l'estimation du rapport d'inertie de l'articulation i_{art} à l'itération k .

Les valeurs des moments d'inertie des éléments de la chaîne d'actionnement après flexibilités $\mathbf{IaL}_{i_{art}}$ sont fixées à 0 quelle que soit l'itération k car ces derniers sont souvent regroupés avec d'autres paramètres ou sont très faibles devant les autres masses/inerties (voir exemple de l'*EMPS*, partie 1.3.2).

Le calcul des paramètres ${}^a \mathbf{ZZ}_{i_{art}}$, ${}^a \mathbf{M}_{i_{art}}$ et ${}^a \mathbf{Iam}_{i_{art}}$ pour i_{art} de 1 à n_{art} nécessite de connaître les valeurs *a priori* ${}^{ap} \left(\mathbf{Iam}_{i_{art}} + \mathbf{ZZ}_{i_{art}} + \mathbf{IaL}_{i_{art}} \right)$ et ${}^{ap} \left(\mathbf{Iam}_{i_{art}} + \mathbf{M}_{i_{art}} + \mathbf{IaL}_{i_{art}} \right)$ du modèle dynamique flexible. Ces valeurs sont estimées à partir des paramètres identifiés du modèle dynamique rigide:

$$\begin{aligned}
 {}^{ap}(\mathbf{I}am_{i_{art}} + M_{i_{art}} + \mathbf{I}a_{L_{i_{art}}}) &= (\hat{\mathbf{I}}a_j + \hat{M}_j)_{\text{rigide}} \text{ pour une articulation prismatique } i_{art} \\
 {}^{ap}(\mathbf{I}am_{i_{art}} + \mathbf{Z}\mathbf{Z}_{i_{art}} + \mathbf{I}a_{L_{i_{art}}}) &= (\hat{\mathbf{I}}a_j + \hat{\mathbf{Z}}\mathbf{Z}_j)_{\text{rigide}} \text{ pour une articulation rotoïde } i_{art}
 \end{aligned} \tag{4.15}$$

Où :

- $j=i_{art}$,
- \hat{M}_j est la masse du corps j ,
- $\hat{\mathbf{Z}}\mathbf{Z}_j$ est le moment d'inertie autour de z_j du corps C_j ,
- $\hat{\mathbf{I}}a_j$ le moment d'inertie de la chaîne d'actionnement de l'articulation j ,

\hat{M}_j , $\hat{\mathbf{Z}}\mathbf{Z}_j$ et $\hat{\mathbf{I}}a_j$ sont les valeurs du modèle rigide identifiées à la première étape de l'algorithme.

Tous les autres paramètres du modèle dynamique flexible sont les valeurs constantes identifiées à l'étape 1 (exemple: ${}^kMX_{i_{art}} = (\widehat{MX}_j)_{\text{rigide}}$, ${}^kMY_{i_{art}} = (\widehat{MY}_j)_{\text{rigide}}$, etc...).

Les frottements côté corps du robot sont fixés à 0 car les frottements côté moteur sont souvent prépondérants sur les robots industriels (Jubien et al. 2014a) :

$$\begin{aligned}
 {}^kFvm_{i_{art}} &= \widehat{Fv}_j \\
 {}^kFcm_{i_{art}} &= \widehat{Fc}_j \\
 {}^kFvL_{i_{art}} &= 0 \\
 {}^kFcL_{i_{art}} &= 0
 \end{aligned} \tag{4.16}$$

Seul l'essai sans charge est utilisé dans cette étape, les paramètres de la charge sont donc mis à 0.

Initialisation de l'algorithme de programmation non linéaire

S'il n'est pas possible d'avoir des valeurs *a priori* des raideurs, la raideur *a priori* de l'articulation i_{art} peut se calculer à partir de la fréquence naturelle à entrée bloquée *a priori* ${}^{ap}f_{n_{i_{art}}}$ (moteur bloqué, $q_{m_{i_{art}}} = \text{cte}$) de la manière suivante :

$$\begin{aligned}
 {}^{ap}k_{i_{art}} &= 4\pi^2 {}^{ap}f_{n_{i_{art}}}^2 M_{eq_{i_{art}}} \text{ pour une articulation prismatique } i_{art} \\
 {}^{ap}k_{i_{art}} &= 4\pi^2 {}^{ap}f_{n_{i_{art}}}^2 \mathbf{Z}\mathbf{Z}_{eq_{i_{art}}} \text{ pour une articulation rotoïde } i_{art}
 \end{aligned} \tag{4.17}$$

Où :

- $M_{eq_{iart}}$ est la masse équivalente côté corps pour une articulation prismatique,
- $ZZ_{eq_{iart}}$ est le moment d'inertie équivalent côté corps. Elle est fonction de la configuration de blocage du robot.

Il faut avoir une idée des fréquences naturelles à entrée bloquée (données constructeurs, essais vibratoire par sonnage, essai à sortie bloquée...) par contre les rapports d'inerties peuvent être initialisés à la valeur 0.5, tout en respectant la définie positivité de la matrice d'inertie du modèle dynamique flexible du robot (voir équation (1.18)).

Ce qui donne :

$$\begin{aligned} {}^0\hat{k}_{iart} &= {}^{ap}k_{iart} \\ {}^0\hat{\lambda}_{iart} &= 0.5 \text{ et } \mathbf{M}(\mathbf{q}_L) \text{ définie positive} \end{aligned} \quad (4.18)$$

Où :

- ${}^0\hat{k}_{iart}$ est la valeur initiale de la raideur de l'articulation i_{art} ,
- ${}^{ap}k_{iart}$ est la raideur *a priori* de l'articulation i_{art} ,
- ${}^0\hat{\lambda}_{iart}$ est la valeur initiale du rapport d'inertie de l'articulation i_{art} .

4.3.4 Etape 3 : identification globale du modèle dynamique flexible

La méthode *DIDIM* est appliquée pour estimer tous les paramètres dynamiques du modèle dynamique flexible du robot.

Les paramètres de raideurs et d'inerties/masses sont initialisés aux valeurs suivantes :

$${}^0\hat{k}_{iart} \text{ (étape 3)} = \hat{k}_{iart} \text{ (étape 2)} \text{ pour toute articulation } i_{art} \quad (4.19)$$

$$\begin{aligned} {}^0\widehat{\mathbf{I}}_{am_{iart}} &= \hat{\lambda}_{iart} \left(\widehat{\mathbf{I}}_{a_j} + \widehat{\mathbf{M}}_j \right)_{rigide} \\ {}^0\widehat{\mathbf{M}}_{iart} &= (1 - \hat{\lambda}_{iart}) \left(\widehat{\mathbf{I}}_{a_j} + \widehat{\mathbf{M}}_j \right)_{rigide} \end{aligned} \quad \text{pour une articulation prismatique (p) } i_{art} \quad (4.20)$$

$$\begin{aligned} {}^0\widehat{\mathbf{I}}_{am_{iart}} &= \hat{\lambda}_{iart} \left(\widehat{\mathbf{I}}_{a_j} + \widehat{\mathbf{Z}}\mathbf{Z}_j \right)_{rigide} \\ {}^0\widehat{\mathbf{Z}}\mathbf{Z}_{iart} &= (1 - \hat{\lambda}_{iart}) \left(\widehat{\mathbf{I}}_{a_j} + \widehat{\mathbf{Z}}\mathbf{Z}_j \right)_{rigide} \end{aligned} \quad \text{pour une articulation rotoïde (r) } i_{art} \quad (4.21)$$

Tous les autres paramètres sont initialisés à des valeurs nulles sauf les paramètres de la charge si elle est présente pour l'essai en charge (voir partie 3.4).

Cette initialisation permet d'utiliser des conditions initiales proches des paramètres optimaux afin de permettre la convergence de l'algorithme sur un robot à flexibilités articulaires.

4.3.5 Structure de l'algorithme pour les robots à flexibilités articulaires

L'algorithme *DIDIM* appliqué aux robots à flexibilités articulaires peut être résumé comme suit :

Etape 1 : identification de modèle dynamique rigide avec la méthode *DIDIM*

Etape 2 : identification de \mathbf{m} et \mathbf{k} avec un algorithme de programmation non linéaire en utilisant l'initialisation décrite dans (4.18).

Etape 3 : identification du modèle dynamique flexible avec la méthode *DIDIM* en utilisant les paramètres estimés aux étapes 1 et 2 pour calculer les conditions initiales.

4.4 Application au bras EMPS

Dans cette partie l'algorithme en trois étapes est appliqué expérimentalement sur le bras *EMPS*. Une comparaison avec une méthode *CLOE* classique est effectuée par la suite. Les résultats de cette partie sont inspirés de (Gautier et al. 2013).

4.4.1 Identification du modèle dynamique flexible avec la méthode *DIDIM*

Etape 1

Le modèle dynamique (4.5) est utilisé pour l'identification du modèle dynamique rigide avec la méthode *DIDIM*. La fréquence de coupure du filtre Decimate est fixée à 5 (Hz). La consigne et les mesures décrites dans la partie (4.2.2) sont utilisées (voir figure 4-7, figure 4-8 et figure 4-9) avec une trajectoire à vide et une trajectoire avec la charge de 9.916 (kg) (± 5 (g)) pour l'essai avec charge. L'identification de la charge utilise le modèle dynamique inverse (4.5) et la méthode décrite à la partie 3.4.

L'initialisation (3.51) est utilisée :

$${}^0M_{1R} = {}^{ap}Ia_1 \text{ avec } {}^{ap}Ia_1 = 78 \text{ (kg)} \quad (4.22)$$

On rappelle que le paramètre Ia_1 est regroupé avec le paramètre M_1 pour le cas du bras *EMPS* (voir tableau 1-4).

Les paramètres identifiés du modèle dynamique rigide sont donnés dans le tableau 4-4.

Tableau 4-4 : Bras *EMPS*, paramètres identifiés du modèle dynamique rigide avec la méthode *DIDIM*

Paramètre	$\hat{\chi}^0$	$\hat{\chi}^2$	$\% \hat{\sigma}_{\chi^2}$	$\% e_{\hat{\chi}}$
M_{IR}	78	104 (kg)	0.36	0.58
F_{V1}	0	215 (N/m/s)	1.14	0.92
F_{c1}	0	19.6 (N)	1.16	1.01
Off_1	0	-5.22 (N)	1.81	0.38
M_L	0	10.8 (kg)	0.89	0.92

Tous les paramètres sont bien identifiés et sont proches de ceux identifiés avec la méthode *IDIM-LS* (voir tableau 4-2 et dernière colonne $\%e_{\hat{\chi}}$ du tableau 4-4 -équation (3.46)-), la norme relative du résidu est de 4.42%. Le vecteur de mesure Y est proche du modèle $W\chi$, représenté figure 4-16.

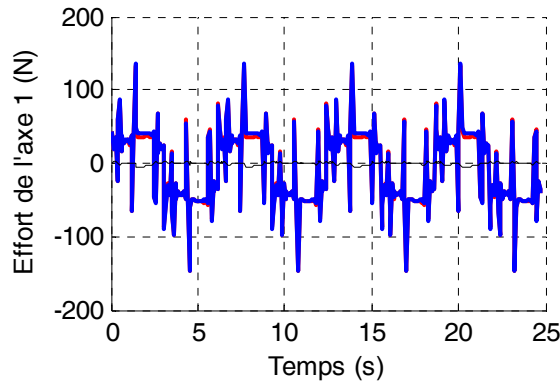


Figure 4-16 : Bras *EMPS*, modèle rigide, validation directe avec la méthode *DIDIM*, validation directe, Y (rouge), $W\chi$ (bleu) et erreur (noir)

Les paramètres identifiés vont être utilisés dans les deux étapes suivantes. L'algorithme de *DIDIM* est très robuste sur le modèle dynamique rigide du bras *EMPS* car il converge pour une valeur de ${}^0M_{IR}$ comprise entre 25 (kg) et 1000 (kg).

Etape 2

Les valeurs de la raideur et du rapport d'inertie sont identifiées dans cette étape avec l'algorithme de programmation non linéaire du Simplex de Nelder-Mead.

La raideur à l'itération k est une variable de l'algorithme de programmation non linéaire. Les autres paramètres du robot sont calculés à chaque itération k de l'algorithme de programmation non linéaire avec :

$$\begin{aligned}
 {}^k \mathbf{I}am_1 &= {}^k \lambda_1 \left(\hat{\mathbf{M}}_{1R} \right)_{\text{rigide}} \\
 {}^k \mathbf{F}vm_1 &= \hat{\mathbf{F}}v_1 \\
 {}^k \mathbf{F}cm_1 &= \hat{\mathbf{F}}c_1 \\
 {}^k \mathbf{O}ffm_1 &= \hat{\mathbf{O}}ff_1 \\
 {}^k \mathbf{M}_{1R} &= (1 - {}^k \lambda_1) \left(\hat{\mathbf{M}}_{1R} \right)_{\text{rigide}} \\
 {}^k \mathbf{F}vL_1 &= 0 \\
 {}^k \mathbf{F}cL_1 &= 0
 \end{aligned} \tag{4.23}$$

L'initialisation proposée dans (4.18) est utilisée :

$$\begin{aligned}
 {}^0 k_1 &= 4\pi^2 (1 - {}^0 \lambda_1) {}^{ap} f_{n_1}^2 \left(\hat{\mathbf{M}}_{1R} \right)_{\text{rigide}} \text{ avec } {}^{ap} f_{n_1} = 40 \text{ (Hz)} \\
 &= 3.28 \cdot 10^6 \text{ (N/m)}
 \end{aligned} \tag{4.24}$$

On a choisi une valeur initiale éloignée de la valeur identifiée à la partie 4.2.1 pour tester la robustesse de l'algorithme aux conditions initiales.

La fréquence de coupure du filtre de Butterworth aller-retour est fixée à 60 (Hz). Les résultats sont donnés dans le tableau 4-5. Les valeurs λ_1 et k_1 sont données en fonction de l'itération k .

Tableau 4-5 : Valeurs identifiées de λ_1 et k_1

Itération k	0	16
λ_1	0.5	0.62
k_1	$3.28 \cdot 10^6$	$8.87 \cdot 10^5$

La norme relative du résidu sur les efforts moteurs est de 8.80% pour l'itération finale. L'algorithme converge en moins de 20 itérations. Pour le cas du bras *EMPS*, il est robuste par rapport aux valeurs initiales : ${}^0 \lambda_1$ peut être choisi entre 0,25 et 0,95 et ${}^0 k_1$ peut être choisi entre $2.3 \cdot 10^5$ et $9.8 \cdot 10^6$ (N/m) pour les mêmes résultats finaux.

Etape 3

Les paramètres du simulateur du robot flexible sont initialisés avec (4.19) et (4.20) :

$$\begin{aligned}
 {}^0\widehat{\text{Iam}}_1 &= \hat{\lambda}_1 \left(\widehat{\text{M}}_{1\text{R}} \right)_{\text{rigide}} \\
 {}^0\widehat{\text{Fvm}}_1 &= 0 \\
 {}^0\widehat{\text{Fcm}}_1 &= 0 \\
 {}^0\widehat{\text{Offm}}_1 &= 0 \\
 {}^0\hat{k}_1 &= \hat{k}_1 \text{ (étape 2)} \\
 {}^0\widehat{\text{M}}_{1\text{R}} &= (1 - \hat{\lambda}_1) \left(\widehat{\text{M}}_{1\text{R}} \right)_{\text{rigide}} \\
 {}^0\widehat{\text{FVL}}_1 &= 0 \\
 {}^0\widehat{\text{FCL}}_1 &= 0
 \end{aligned} \tag{4.25}$$

La fréquence de coupure du filtre Decimate est fixée à 60 (Hz). Les résultats sont donnés dans le tableau 4-6.

Tableau 4-6 : Bras *EMPS*, paramètres identifiés du modèle dynamique flexible avec la méthode *DIDIM*

Paramètre	$\hat{\chi}^0$	$\hat{\chi}^3$	$\% \hat{\sigma}_{\hat{\chi}^3}$	$\% e_{\hat{\chi}}$
Iam ₁	64.5	66.3 (kg)	0.46	0.90
Fvm ₁	0	215 (N/m/s)	0.72	-
Fcm ₁	0	17.5 (N)	0.85	-
Off ₁	0	-6.50 (N)	0.59	0.31
k ₁	8.87 10 ⁵	8.45 10⁵ (N/m)	0.45	0.84
M _{1R}	39.5	37.4 (kg)	0.92	1.35
FVL ₁	0	-	-	-
FCL ₁	0	-	-	-
M _L	10.8	9.90 (kg)	0.95	0.20

L'algorithme converge en seulement 3 itérations. Le vecteur de mesure \mathbf{Y} est très proche du modèle $\mathbf{W}\chi$, les oscillations (figure de droite) sont bien reconstruites (voir figure 4-12). La norme relative du résidu est de 7.31%.

Tous les paramètres sont bien identifiés et sont proches des valeurs identifiées avec la méthode *IDIM-LS* (voir dernière colonne du tableau 4-6) sauf pour les frottements car seuls les frottements de l'entrée de la chaîne d'actionnement côté moteur du robot sont identifiables.

Tous les frottements se regroupent côté moteur pour l'initialisation proposée et leurs sommes sont proches des frottements du modèle dynamique rigide. Différents facteurs de répartition entre les frottements de la sortie de la chaîne d'actionnement côté corps et les frottements de l'entrée de la chaîne d'actionnement côté moteur du robot pour leurs initialisations ont été testés. Mais hormis pour un facteur proche du facteur réel, les frottements ne convergent pas vers les valeurs identifiées par la méthode *IDIM-LS*.

Contrairement à la méthode *IDIM-LS* qui utilise trois mesures (effort moteur, position moteur et position du degré de liberté flexible), la méthode *DIDIM* n'utilise que la mesure des efforts

moteurs (sur un essai où la flexibilité est excitée). Cette seule information n'est pas suffisante pour séparer les frottements côté moteur et côté corps pour le cas du bras *EMPS*.

Cependant, la sensibilité de l'effort moteur par rapport à la répartition des frottements côté moteur et côté corps est très faible. Cela a donc peu d'incidence sur la précision du modèle dynamique flexible identifié.

L'histogramme de l'erreur est représenté figure 4-17, le test de normalité de Kolmogorov-Smirnov vérifie bien que l'erreur suit une loi de distribution gaussienne.

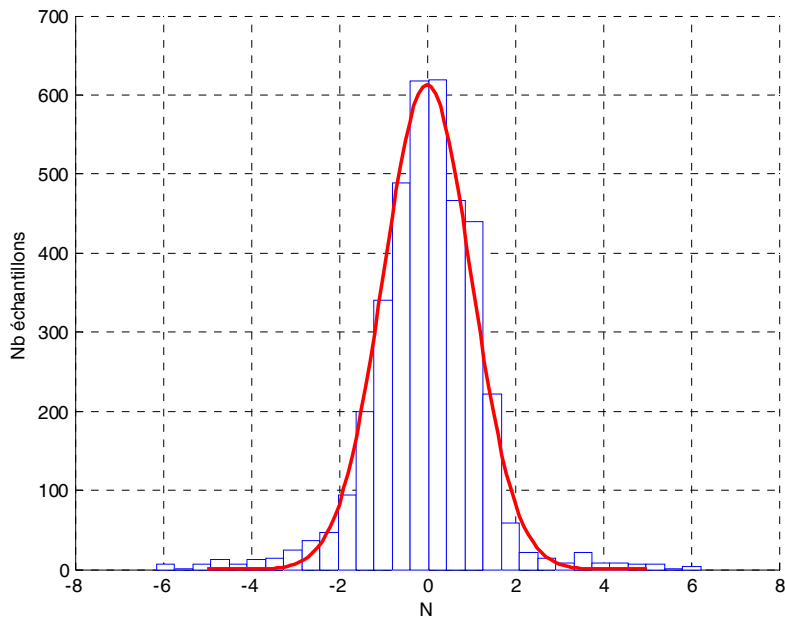


Figure 4-17 : *EMPS*, Histogramme des erreurs (en bleu) et Gaussienne (en rouge) avec la méthode *DIDIM*

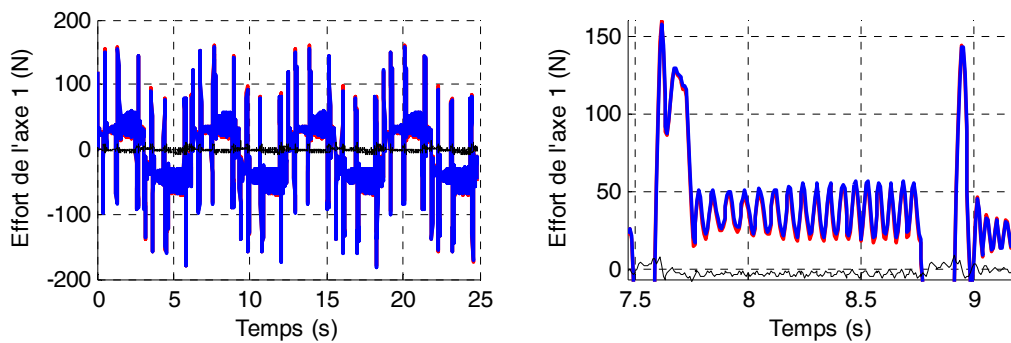


Figure 4-18 : Bras *EMPS*, modèle flexible, validation directe avec la méthode *DIDIM* (sans charge)

L'algorithme en trois étapes proposé permet d'identifier un axe de robot à flexibilités articulaires avec seulement la mesure des efforts moteurs avec une convergence rapide et une bonne

précision. Les trois étapes s'exécutent en seulement 64 secondes sur un processeur double-cœur de 2 (GHz).

4.4.2 Comparaison avec la méthode *CLOE* usuelle

Dans cette partie, la méthode *DIDIM* appliquée aux robots à flexibilités articulaires est comparée à une méthode de sortie usuelle. Les paramètres du modèle dynamique flexible du bras *EMPS* sans charge sont identifiés avec l'algorithme du Simplex de Nelder-Mead.

Dans un premier temps le critère quadratique suivant est utilisé :

$$J(\chi)_{\mathbf{q}_{m_1}} = \|\mathbf{Y}(\mathbf{q}_{m_1}) - \mathbf{Y}_s(\mathbf{q}_{ms_1})\|^2 \quad (4.26)$$

Avec :

- \mathbf{Y} est le vecteur (rx1) obtenu après échantillonnage et filtrage de la position moteur mesurée \mathbf{q}_{m_1} ,
- \mathbf{Y}_s est le vecteur (rx1) obtenu après échantillonnage et filtrage de la position moteur simulée \mathbf{q}_{ms_1} .

Dans un deuxième temps le critère quadratique suivant est utilisé :

$$J(\chi)_{\boldsymbol{\tau}_{m_1}} = \|\mathbf{Y}(\boldsymbol{\tau}_{m_1}) - \mathbf{Y}_s(\boldsymbol{\tau}_{ms_1})\|^2 \quad (4.27)$$

Avec :

- \mathbf{Y} est le vecteur (rx1) obtenu après échantillonnage et filtrage de l'effort moteur mesuré $\boldsymbol{\tau}_{m_1}$,
- \mathbf{Y}_s est le vecteur (rx1) obtenu après échantillonnage et filtrage de l'effort moteur simulé $\boldsymbol{\tau}_{ms_1}$.

L'algorithme du Simplex de Nelder-Mead n'utilise pas les dérivées du critère. En conséquence, les fonctions de sensibilité ne sont pas nécessaires.

Les conditions initiales sur les paramètres nécessitent la connaissance des paramètres identifiés du modèle dynamique rigide :

$$\begin{aligned}
 {}^0\widehat{I}_{am_1} &= {}^{ap}\lambda_1 \left(\widehat{M}_{1R} \right)_{\text{rigide}} \\
 {}^0\widehat{F}_{vm_1} &= 0.5 F_{V_1} \\
 {}^0\widehat{F}_{cm_1} &= 0.5 F_{C_1} \\
 {}^0\widehat{O}_{ffm_1} &= Off_1 \\
 {}^0\widehat{k}_1 &= 4\pi^2 {}^{ap}f_n^2 (1 - {}^{ap}\lambda_1) \left(\widehat{M}_{1R} \right)_{\text{rigide}} \\
 {}^0\widehat{M}_{1R} &= (1 - {}^{ap}\lambda_1) \left(\widehat{M}_{1R} \right)_{\text{rigide}} \\
 {}^0\widehat{F}_{VL_1} &= 0.5 F_{V_1} \\
 {}^0\widehat{F}_{CL_1} &= 0.5 F_{C_1}
 \end{aligned} \tag{4.28}$$

Ici les valeurs *a priori* de la fréquence naturelle à entrée bloquée ${}^{ap}f_n$ et du rapport d'inertie ${}^{ap}\lambda_1$ sont fixées respectivement à 21 (Hz) et 0.5. Contrairement à *DIDIM*, la valeur de la fréquence à entrée bloquée est choisie plus proche de la réalité pour limiter le nombre d'itérations de l'algorithme du Simplex de Nelder-Mead.

Les résultats sont donnés dans le tableau 4-7 et le tableau 4-8. Les différences relatives par rapport aux valeurs identifiées avec *DIDIM* sont affichées en dernière colonne.

Tableau 4-7 : Valeurs identifiées avec la méthode *CLOE* à erreur de sortie sur la position moteur

Paramètre	$\hat{\chi}^{179}$	%e $_{\hat{\chi}}$
I $_{am_1}$	64.1 (kg)	4.1
F $_{vm_1}$	126 (N/m/s)	20
F $_{cm_1}$	13.2 (N)	38
O $_{ff_1}$	-6.45 (N)	0.46
k $_1$	8.7 10⁵ (N/m)	3.8
M $_{1R}$	40.1 (kg)	8.7
F $_{VL_1}$	84.6 (N/m/s)	13
F $_{CL_1}$	5.2 (N)	47

Tableau 4-8 : Valeurs identifiées avec la méthode *CLOE* à erreur de sortie sur l'effort moteur

Paramètre	$\hat{\chi}^{172}$	%e _{χ}
Iam ₁	67.1 (kg)	0.30
Fvm ₁	76.5 (N/m/s)	27
Fcm ₁	12.1 (N)	27
Off ₁	-6.21 (N)	4.2
k ₁	7.68 10⁵ (N/m)	8.3
M _{IR}	37.8 (kg)	2.4
FvL ₁	135 (N/m/s)	39
FcL ₁	6.40 (N)	34

L'algorithme du Simplex converge en un minimum de 172 itérations (276 simulations du modèle dynamique direct) en 9 minutes soit 9 fois plus lentement que les trois étapes de l'algorithme proposé à la partie 4.3. Les valeurs identifiées (hors frottements) ont des écarts assez significatifs par rapport aux valeurs identifiées par la méthode *IDIM-LS* (entre 0.3 et 8%). Les sommes des frottements visqueux et secs sont correctes mais comme pour la méthode *DIDIM*, ils ne sont pas séparables.

Comme pour les robots rigides, la méthode *DIDIM* appliquée aux robots à flexibilités articulaires est une importante amélioration des méthodes à erreur de sortie classiques.

4.5 Application au robot Kuka *LWR*

4.5.1 Introduction

Le robot Kuka LightWeight Robot possède 7 articulations rotoïdes. Le réducteur mécanique de chaque articulation est un engrenage à onde de déformation (harmonic drive) caractérisé par une flexibilité articulaire importante.

Les mesures des couples moteurs, des positions moteurs \mathbf{q}_m (mesurées par des codeurs côté moteur à l'entrée des réducteurs) et des couples côté corps $\boldsymbol{\tau}_e$ (mesurés par les capteurs de couple en sortie de réducteur) sont disponibles dans le contrôleur.

Cependant aucune information précise sur la commande implantée dans le robot réel n'est fournie par Kuka. En conséquence la méthode *DIDIM* appliquée aux robots rigides et à flexibilités articulaires ne peut pas être validée expérimentalement.

Dans cette partie, le modèle dynamique rigide du robot est identifié expérimentalement. Ensuite les raideurs sont identifiées expérimentalement grâce aux capteurs de couple côté corps.

Enfin la méthode *DIDIM* appliquée aux robots à flexibilités articulaires est utilisée pour identifier le modèle dynamique flexible du robot mais sur des données simulées avec une commande de même structure que la commande implantée dans le robot réel.

4.5.2 Identification expérimentale du modèle dynamique rigide avec la méthode *IDIM-LS*

Le modèle dynamique rigide du robot est identifié avec la méthode *IDIM-LS* en utilisant la mesure des efforts moteurs et des positions moteurs.

La fréquence de coupure du filtre Butterworth est réglée à 10 (Hz) et la fréquence de coupure du filtre Decimate est réglée à 0.8 (Hz), pour une valeur du test de Durbin-Watson de 1.7 (voir annexe F). Une masse de 4.6136 (kg) (+0.1 (g)) est fixée sur l'effecteur du robot. Afin d'identifier la charge, le robot effectue des trajectoires sans charge et des trajectoires avec la charge fixée sur l'effecteur.

Comme pour le robot Stäubli *TX40*, les consignes de positions articulaires sont calculées par le générateur de mouvement du contrôleur. C'est un mouvement point à point avec un profil d'accélération en trapèze.

La fréquence d'échantillonnage des mesures est de 1 (kHz). Les consignes articulaires sont représentées figure 4-19, figure 4-20 et figure 4-21 avec 21 points d'arrêt et les couples moteurs sans charge (ramené côté articulaire) sont représentés figure 4-23. Un zoom sur la consigne d'accélération de l'axe 1 est représenté figure 4-22.

Les résultats de l'identification du modèle dynamique rigide sont donnés dans le tableau 4-9.

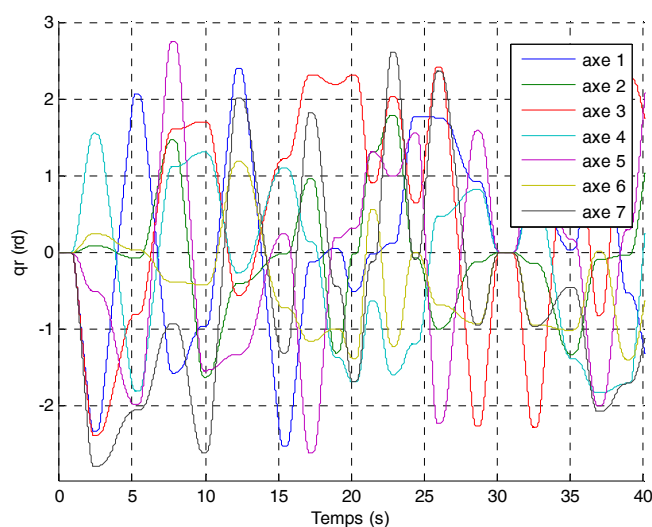


Figure 4-19 : Kuka *LWR*, Consignes de positions articulaires

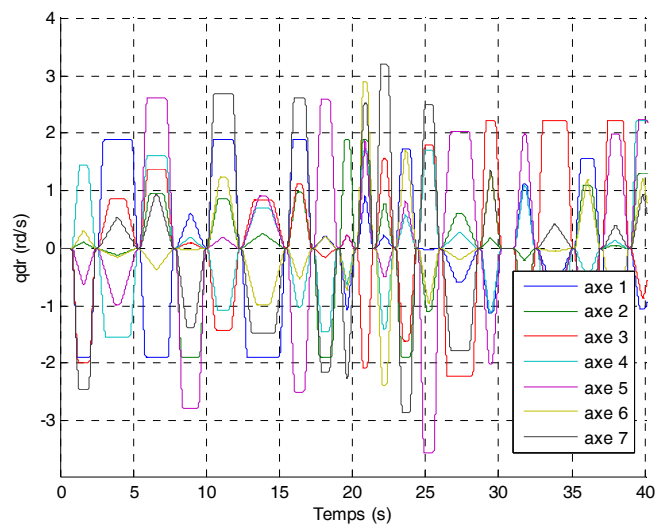


Figure 4-20 : Kuka *LWR*, Consignes de vitesses articulaires

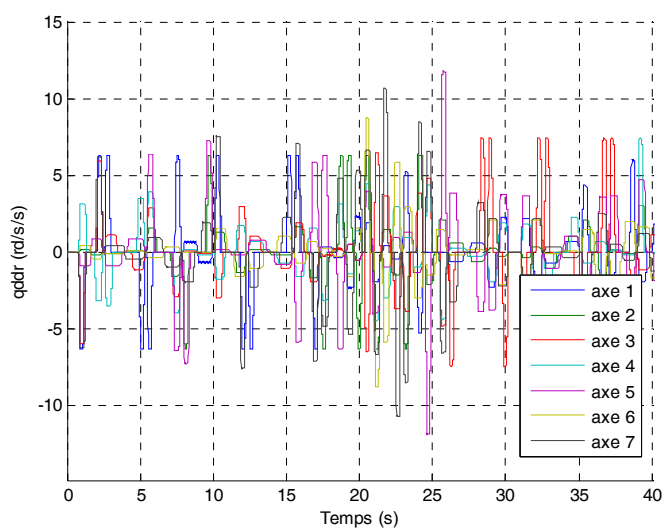


Figure 4-21 : Kuka *LWR*, Consignes d'accélérations articulaires

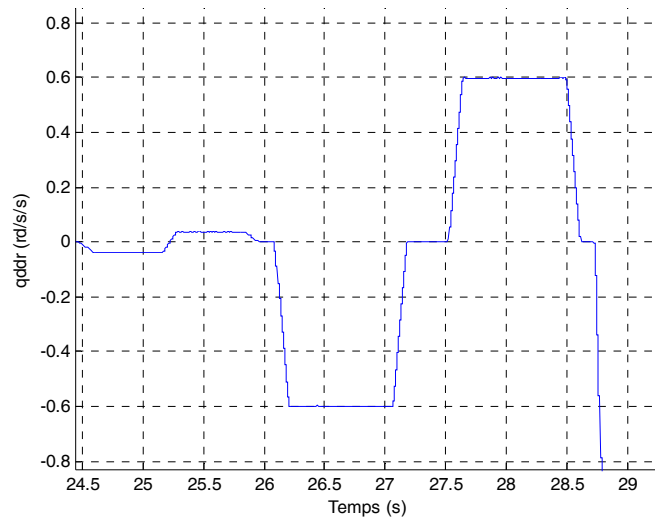


Figure 4-22 : Kuka LWR, Zoom sur la consigne d'accélération articulaire de l'axe 1

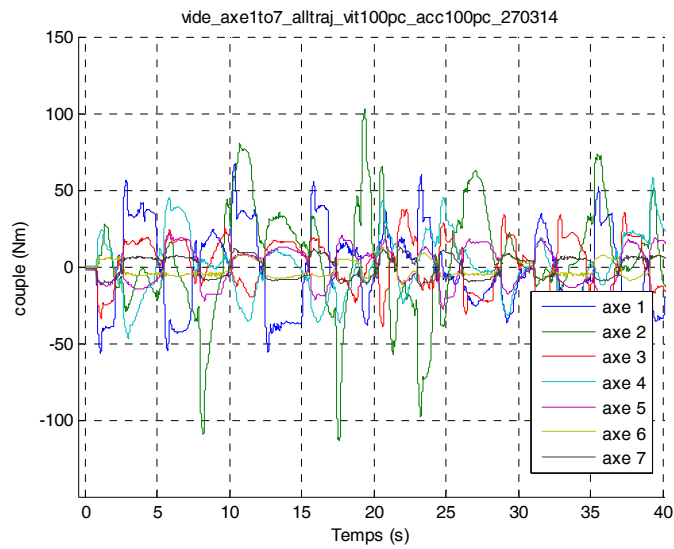


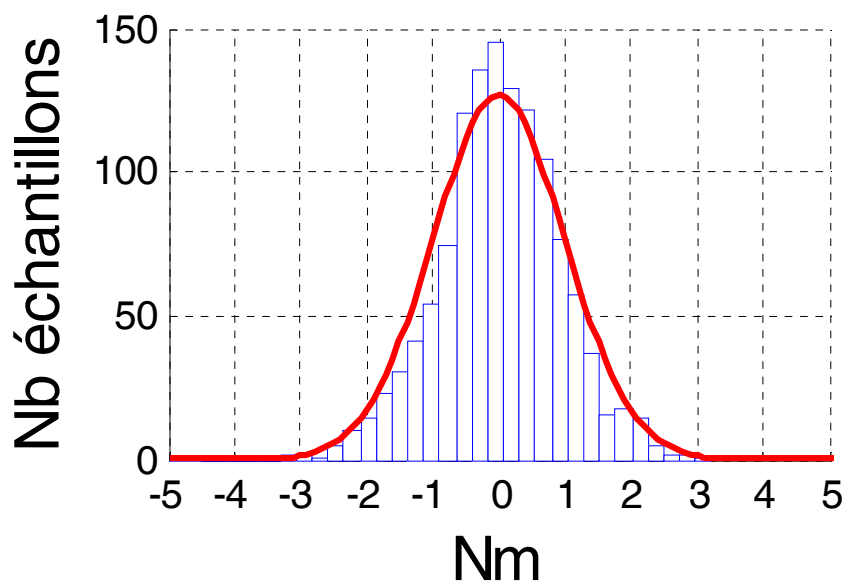
Figure 4-23 : Kuka LWR, Efforts moteurs mesurés

Tableau 4-9 : Kuka *LWR*, paramètres identifiés du modèle dynamique rigide avec la méthode *IDIM-LS*

Par.	$\hat{\chi}$	$\% \sigma_{\hat{\chi}_i}$	Par.	$\hat{\chi}$	$\% \sigma_{\hat{\chi}_i}$
ZZ _{1R}	3.18	1.97	MY _{5R}	0.0422	19.5
FV ₁	13.2	1.34	Ia ₅	0.71	3.55
Fc ₁	13.7	1.66	FV ₅	3.94	2.01
XX _{2R}	1.33	8.47	Fc ₅	9.31	1.32
ZZ _{2R}	3.33	1.40	MY _{6R}	0.0352	23.0
MY _{2R}	4.51	3.62	Ia ₆	0.41	9.46
FV ₂	13.3	2.92	FV ₆	1.77	6.68
Fc ₂	14.0	2.85	Fc ₆	5.42	1.76
Ia ₃	1.93	2.46	Ia ₇	0.394	4.19
FV ₃	5.78	2.41	FV ₇	1.41	3.91
Fc ₃	10.3	1.81	Fc ₇	6.33	1.42
XX _{4R}	0.422	6.94	XX _L	0.101	21.3
ZZ _{4R}	0.539	7.55	XZ _L	-0.034	22.1
MY _{4R}	-1.38	1.08	MX _L	0.33	2.29
Ia ₄	1.75	4.41	MY _L	-0.314	2.44
FV ₄	9.89	2.01	MZ _L	0.603	1.58
Fc ₄	9.75	1.99	M _L	4.64	1.27

Seuls 32 paramètres essentiels sont conservés: 26 paramètres du robot (sur les 69 paramètres de base) et 6 paramètres de la charge (sur ses 10 paramètres inertiels).

Les paramètres sont bien identifiés car leurs écart-types relatifs sont faibles. La masse de la charge est estimée à 4.64 (kg) pour une valeur pesée de 4.61 (kg) soit une erreur de 0.03 (kg) (inférieure à $2 \hat{\sigma}_{M_L}$). L'histogramme des erreurs est représenté figure 3-12, le test de normalité de Kolmogorov-Smirnov vérifie bien que l'erreur suit une loi de distribution gaussienne pour $D_{\text{obs}}=0.0172$ inférieur à $D_{\alpha}=0.0181$ (avec $\alpha=0.5$, voir annexe E). Les courbes de la validation directe pour l'essai sans charge sont représentées Figure 4-25. Les vecteurs des mesures \mathbf{Y}_j sont proches des vecteurs des mesures estimés par les équations $\mathbf{W}_j \hat{\chi}$ pour j de 1 à 7.


Figure 4-24 : Kuka *LWR*, Histogramme des erreurs (en bleu) et Gaussienne (en rouge) avec la méthode *IDIM-LS*

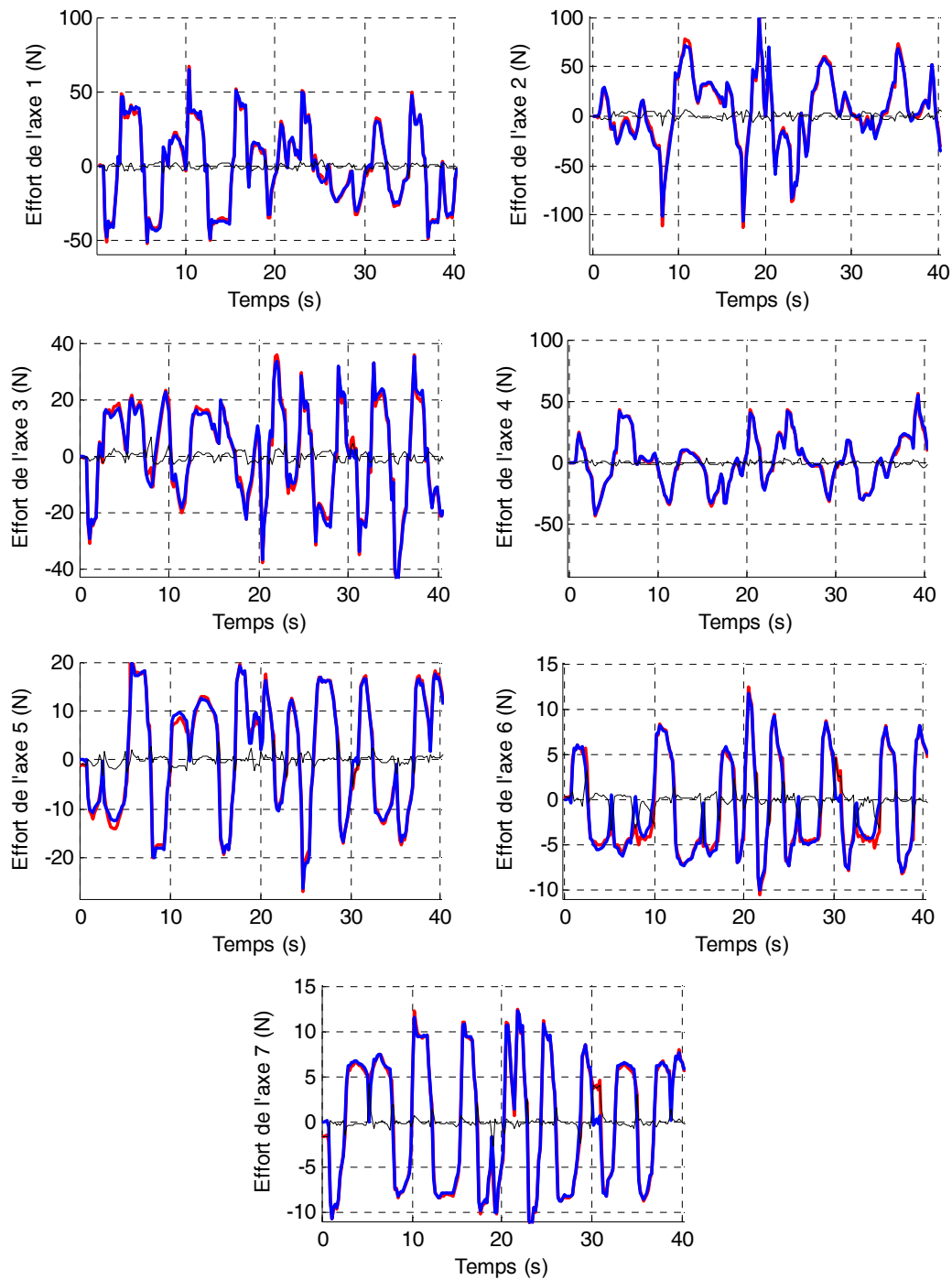


Figure 4-25 : Kuka *LWR*, Validation directe avec la méthode *IDIM-LS*, Y (rouge), W_χ (bleu) et erreur (noir)

4.5.3 Identification expérimentale des raideurs articulaires avec les capteurs de couple

Avant d'appliquer la méthode *DIDIM* sur le Kuka *LWR* en simulation, je propose une méthode pour identifier une partie des raideurs du robot basée sur l'utilisation des capteurs de couples côté corps.

Théorie

On rappelle que le Kuka *LWR* possède des capteurs de couple en sortie de réducteur qui mesurent les couples côté corps τ_e et que les mesures des positions articulaires, côté corps, de chaque articulation peuvent être estimées à partir des positions moteurs \mathbf{q}_m , des couples τ_e et d'une estimation des raideurs articulaires \mathbf{k}^{ap} , une matrice colonne d'offset \mathbf{Off}_L est rajoutée pour tenir compte des offsets sur les mesures des capteurs :

$$\tau_e = -{}^{ap}\mathbf{k}(\mathbf{q}_L - \mathbf{q}_m) + \mathbf{Off}_L \quad (4.29)$$

$$\mathbf{q}_L = -{}^{ap}\mathbf{k}^{-1}(\tau_e - \mathbf{Off}_L) + \mathbf{q}_m \quad (4.30)$$

Où :

- ${}^{ap}\mathbf{k}$ est la matrice (7x7) diagonale des raideurs articulaires estimées par le fabricant :

$${}^{ap}\mathbf{k} = \begin{bmatrix} {}^{ap}k_1 & 0 & \dots & 0 \\ 0 & {}^{ap}k_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & {}^{ap}k_7 \end{bmatrix} \quad (4.31)$$

- \mathbf{Off}_L est la matrice colonne (7x1) des offsets :

$$\mathbf{Off}_L = [\text{Off}_{L_1} \quad \text{Off}_{L_2} \quad \dots \quad \text{Off}_{L_7}]^T \quad (4.32)$$

On rappelle que le modèle dynamique inverse d'un robot à flexibilités articulaires s'exprime de la manière suivante (voir partie 1.2.2 et figure 1-4) en prenant en compte les offsets sur les mesures des capteurs :

$$\begin{aligned} \tau_{\text{mdi}_m} &= \mathbf{Iam} \ddot{\mathbf{q}}_m + \tau_{\text{mdi}_e} + \tau_{\text{fm}} \\ \tau_{\text{mdi}_e} &= \mathbf{Ia}_L \ddot{\mathbf{q}}_L + \mathbf{M}(\mathbf{q}_L) \ddot{\mathbf{q}}_L + \mathbf{H}(\mathbf{q}_L, \dot{\mathbf{q}}_L) + \mathbf{Off}_L \\ \tau_{\text{mdi}_e} &= -\mathbf{k}\mathbf{q}_e + \mathbf{Off}_L \end{aligned} \quad (4.33)$$

En prenant ${}^{\text{ap}}\mathbf{k} = \mathbf{k}$ et en reportant (4.30) et ses dérivées $(\mathbf{q}_L, \dot{\mathbf{q}}_L, \ddot{\mathbf{q}}_L)$ dans (4.33) :

$$\begin{aligned} \boldsymbol{\tau}_{\text{mdi}_e} = & \mathbf{Ia}_L \left[-\mathbf{k}^{-1} \ddot{\boldsymbol{\tau}}_e + \ddot{\mathbf{q}}_m \right] + \mathbf{M} \left(-\mathbf{k}^{-1} (\boldsymbol{\tau}_e - \mathbf{Off}_L) + \mathbf{q}_m \right) \left[-\mathbf{k}^{-1} \ddot{\boldsymbol{\tau}}_e + \ddot{\mathbf{q}}_m \right] \dots \\ & + \mathbf{H} \left(-\mathbf{k}^{-1} (\boldsymbol{\tau}_e - \mathbf{Off}_L) + \mathbf{q}_m, -\mathbf{k}^{-1} \dot{\boldsymbol{\tau}}_e + \dot{\mathbf{q}}_m \right) + \mathbf{Off}_L \end{aligned} \quad (4.34)$$

Après développement de l'équation (4.34) :

$$\begin{aligned} \boldsymbol{\tau}_{\text{mdi}_e} = & \mathbf{Ia}_L \ddot{\mathbf{q}}_m - \mathbf{Ia}_L \mathbf{k}^{-1} \ddot{\boldsymbol{\tau}}_e + \mathbf{M} \left(-\mathbf{k}^{-1} (\boldsymbol{\tau}_e - \mathbf{Off}_L) + \mathbf{q}_m \right) \ddot{\mathbf{q}}_m \dots \\ & - \mathbf{M} \left(-\mathbf{k}^{-1} (\boldsymbol{\tau}_e - \mathbf{Off}_L) + \mathbf{q}_m \right) \mathbf{k}^{-1} \ddot{\boldsymbol{\tau}}_e \dots \\ & + \mathbf{H} \left(-\mathbf{k}^{-1} (\boldsymbol{\tau}_e - \mathbf{Off}_L) + \mathbf{q}_m, -\mathbf{k}^{-1} \dot{\boldsymbol{\tau}}_e + \dot{\mathbf{q}}_m \right) + \mathbf{Off}_L \end{aligned} \quad (4.35)$$

On remarque que le développement de l'équation (4.35) est compliqué et que les couples côté corps ne pourront pas s'exprimer linéairement par rapport aux paramètres ou à des combinaisons de paramètres à cause des fonctions non-linéaires telles que les fonctions trigonométriques (par exemple les termes de gravités qui dépendent en partie des fonctions cosinus et sinus des positions articulaires).

L'équation (4.35) est réécrite seulement pour l'articulation $i_{\text{art}} = i_c$ en émettant l'hypothèse que :

$$\begin{aligned} \mathbf{q}_{m_{i_{\text{art}}}} &= \text{cte pour } i_{\text{art}} \neq i_c \\ \mathbf{q}_{L_{i_{\text{art}}}} &= \text{cte pour } i_{\text{art}} \neq i_c \end{aligned} \quad (4.36)$$

C'est à dire que toutes les articulations sont immobiles sauf l'articulation i_c .

Pour une articulation i_c orientée suivant la direction verticale (vecteur gravité $\bar{\mathbf{g}}$ colinéaire à $\bar{\mathbf{z}}_{i_c}$), il n'y a pas de couple de gravité et il vient (Janot, 2007):

$$\begin{aligned} \tau_{\text{mdi}_{e_c}} = & \mathbf{Ia}_{L_{i_c}} \ddot{\mathbf{q}}_{m_{i_c}} - \frac{\mathbf{Ia}_{L_{i_c}}}{k_{i_c}} \ddot{\boldsymbol{\tau}}_{e_c} + \mathbf{ZZ}_{i_c} \ddot{\mathbf{q}}_{m_{i_c}} - \frac{\mathbf{ZZ}_{i_c}}{k_{i_c}} \ddot{\boldsymbol{\tau}}_{e_c} \dots \\ & + \mathbf{Fv}_{L_{i_c}} \dot{\mathbf{q}}_{m_{i_c}} - \frac{\mathbf{Fv}_{L_{i_c}}}{k_{i_c}} \dot{\boldsymbol{\tau}}_{e_c} + \mathbf{Fcl}_{i_{\text{art}}} \text{sign} \left(\dot{\mathbf{q}}_{m_{i_c}} - \frac{\dot{\boldsymbol{\tau}}_{e_c}}{k_{i_c}} \right) \dots \\ & + \mathbf{Off}_{L_{i_c}} \end{aligned} \quad (4.37)$$

En pratique le comportement de $\text{sign}(\dot{\mathbf{q}}_{m_{i_c}} - \dot{\boldsymbol{\tau}}_{e_c} / k_{i_c})$ est très proche du comportement de $\text{sign}(\dot{\mathbf{q}}_{m_{i_c}})$, il est donc possible d'approcher $\text{sign}(\dot{\mathbf{q}}_{m_{i_c}} - \dot{\boldsymbol{\tau}}_{e_c} / k_{i_c})$ par $\text{sign}(\dot{\mathbf{q}}_{m_{i_c}})$ (erreur relative $< 0.5\%$ pour le cas de l'*EMPS*).

Les termes $\mathbf{Ia}_{L_{i_c}}$ et \mathbf{ZZ}_{i_c} peuvent être regroupés sous le terme $\mathbf{ZZ}_{R_{i_c}} = \mathbf{ZZ}_{i_c} + \mathbf{Ia}_{L_{i_c}}$, alors il vient :

$$\begin{aligned} \tau_{\text{mdi}_{e_{i_c}}} = & ZZ_{R_{i_c}} \ddot{q}_{m_{i_c}} - \frac{ZZ_{R_{i_c}}}{k_{i_c}} \ddot{\tau}_{e_{i_c}} + F_{VL_{i_c}} \dot{q}_{m_{i_c}} - \frac{F_{VL_{i_c}}}{k_{i_c}} \dot{\tau}_{e_{i_c}} \dots \\ & + F_{CL_{iart}} \text{sign}(\dot{q}_{m_{i_c}}) + \text{Offl}_{i_c} \end{aligned} \quad (4.38)$$

Le calcul de $\tau_{\text{mdi}_{e_{i_c}}}$ dépend des termes $ZZ_{R_{i_c}}$, $ZZ_{R_{i_c}} / k_{i_c}$, $F_{VL_{i_c}}$, $F_{VL_{i_c}} / k_{i_c}$, $F_{CL_{iart}}$ et Offl_{i_c} ainsi que du couple côté corps (et de ses dérivées première et seconde) et de la vitesse et l'accélération moteur de l'articulation i_c .

Le couple côté corps peut être exprimé linéairement par rapport aux paramètres dynamiques comme suit pour une articulation orientée suivant la direction verticale :

$$\tau_{\text{mdi}_{e_{i_c}}} = \mathbf{MDI}^{i_c} \boldsymbol{\chi}^{i_c} \quad (4.39)$$

Avec :

- $\mathbf{MDI}^{i_c} = \begin{bmatrix} \dot{q}_{m_{i_c}} & -\ddot{\tau}_{e_{i_c}} & \dot{q}_{m_{i_c}} & -\dot{\tau}_{e_{i_c}} & \text{sign}(\dot{q}_{m_{i_c}}) & 1 \end{bmatrix}$,
- $\boldsymbol{\chi}^{i_c} = \begin{bmatrix} ZZ_{R_{i_c}} & \frac{ZZ_{R_{i_c}}}{k_{i_c}} & F_{VL_{i_c}} & \frac{F_{VL_{i_c}}}{k_{i_c}} & F_{CL_{iart}} & \text{Offl}_{i_c} \end{bmatrix}^T$.

La modèle inverse (4.39) de l'articulation i_c montre qu'il est possible d'estimer la raideur articulaire k_{i_c} à l'aide d'une des deux équations suivantes pour une articulation orientée à la verticale :

$$\hat{k}_{i_c} = \frac{\hat{\chi}^{i_c}(1)}{\hat{\chi}^{i_c}(2)} \quad (4.40)$$

$$\hat{k}_{i_c} = \frac{\hat{\chi}^{i_c}(3)}{\hat{\chi}^{i_c}(4)} \quad (4.41)$$

Il est possible d'étendre le modèle dynamique inverse (4.38) avec l'hypothèse (4.36) à l'articulation i_c si elle est orientée horizontalement (vecteur gravité \vec{g} perpendiculaire à \vec{z}_{i_c}), en introduisant les deux derniers termes du premier moment d'inertie du corps C_{i_c} , MX_{i_c} et MY_{i_c} :

$$\begin{aligned}
 \tau_{\text{mdi}_{e_{i_c}}} = & ZZ_{R_{i_c}} \ddot{q}_{m_{i_c}} - \frac{ZZ_{R_{i_c}}}{k_{i_c}} \ddot{\tau}_{e_{i_c}} \dots \\
 & - MX_{i_c} g \cos\left(q_{m_{i_c}} - \frac{\tau_{e_{i_c}} - \text{Offl}_{i_c}}{k_{i_c}}\right) + MY_{i_c} g \sin\left(q_{m_{i_c}} - \frac{\tau_{e_{i_c}} - \text{Offl}_{i_c}}{k_{i_c}}\right) \dots \\
 & + FvL_{i_c} \dot{q}_{m_{i_c}} - \frac{FvL_{i_c}}{k_{i_c}} \dot{\tau}_{e_{i_c}} + FcL_{i_{\text{art}}} \text{sign}(\dot{q}_{m_{i_c}}) + \text{Offl}_{i_c}
 \end{aligned} \quad (4.42)$$

Où g est l'accélération de la gravité avec $g=9.81$ (m/s²).

En pratique le terme $(\tau_{e_{i_c}} - \text{Offl}_{i_c}) / k_{i_c}$ est très faible devant le terme $q_{m_{i_c}}$, il est donc possible d'utiliser les approximations suivantes :

$$\begin{aligned}
 \cos\left(q_{m_{i_c}} - \frac{\tau_{e_{i_c}} - \text{Offl}_{i_c}}{k_{i_c}}\right) & \simeq \cos(q_{m_{i_c}}) \\
 \sin\left(q_{m_{i_c}} - \frac{\tau_{e_{i_c}} - \text{Offl}_{i_c}}{k_{i_c}}\right) & \simeq \sin(q_{m_{i_c}})
 \end{aligned} \quad (4.43)$$

Il vient à partir de (4.42) et (4.43) :

$$\begin{aligned}
 \tau_{\text{mdi}_{e_{i_c}}} = & ZZ_{R_{i_c}} \ddot{q}_{m_{i_c}} - \frac{ZZ_{R_{i_c}}}{k_{i_c}} \ddot{\tau}_{e_{i_c}} \dots \\
 & - MX_{i_c} g \cos(q_{m_{i_c}}) + MY_{i_c} g \sin(q_{m_{i_c}}) \dots \\
 & + FvL_{i_c} \dot{q}_{m_{i_c}} - \frac{FvL_{i_c}}{k_{i_c}} \dot{\tau}_{e_{i_c}} + FcL_{i_{\text{art}}} \text{sign}(\dot{q}_{m_{i_c}}) + \text{Offl}_{i_c}
 \end{aligned} \quad (4.44)$$

Le couple côté corps peut être exprimé linéairement par rapport aux paramètres dynamiques comme suit pour une articulation orientée suivant la direction horizontale :

$$\tau_{\text{mdi}_{e_{i_c}}} = \mathbf{MDI}^{i_c} \boldsymbol{\chi}^{i_c} \quad (4.45)$$

Avec :

- $\mathbf{MDI}^{i_c} = \begin{bmatrix} \dot{q}_{m_{i_c}} & -\dot{\tau}_{e_{i_c}} & -g \cos(q_{m_{i_c}}) & g \sin(q_{m_{i_c}}) & \dot{q}_{m_{i_c}} & -\dot{\tau}_{e_{i_c}} & \text{sign}(\dot{q}_{m_{i_c}}) & 1 \end{bmatrix}$,
- $\boldsymbol{\chi}^{i_c} = \begin{bmatrix} ZZ_{R_{i_c}} & \frac{ZZ_{R_{i_c}}}{k_{i_c}} & MX_{i_c} & MY_{i_c} & FvL_{i_c} & \frac{FvL_{i_c}}{k_{i_c}} & FcL_{i_{\text{art}}} & \text{Offl}_{i_c} \end{bmatrix}^T$.

La modèle inverse (4.39) de l'articulation i_c montre qu'il est possible d'estimer la raideur articulaire k_{i_c} à l'aide d'une des deux équations suivantes pour une articulation orientée suivant la direction horizontale :

$$\hat{k}_{i_c} = \frac{\hat{\chi}^{i_c}(1)}{\hat{\chi}^{i_c}(2)} \quad (4.46)$$

$$\hat{k}_{i_c} = \frac{\hat{\chi}^{i_c}(5)}{\hat{\chi}^{i_c}(6)} \quad (4.47)$$

L'équation choisie entre (4.40) et (4.41) (ou (4.46) et (4.47)) est celle qui contient les paramètres avec les écart-types les plus faibles afin d'avoir une estimation la plus précise de la raideur.

Pour utiliser les modèles dynamiques inverses (4.38) et (4.44), il est nécessaire de respecter l'hypothèse (4.36). Sur un robot à plusieurs degrés de liberté, afin de respecter cette hypothèse, toutes les articulations $i_{art} \neq i_c$ doivent être perpendiculaires à l'articulation en cours d'identification i_c . Il n'est pas possible d'avoir des articulations $i_{art} \neq i_c$ colinéaires à l'articulation en cours d'identification i_c à cause des raideurs articulaires et des raideurs d'asservissement qui peuvent être excitées.

Identification expérimentale

Il est possible d'identifier les raideurs des 3 premières articulations du Kuka *LWR* en respectant l'hypothèse (4.36) , la première avec le modèle (4.38) et les deux suivantes avec le modèle (4.44).

Les configurations du robot pour l'identification des 3 premières articulations sont représentées figure 4-26, figure 4-27 et figure 4-28.

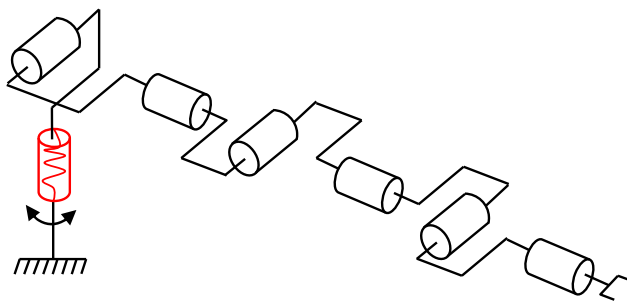


Figure 4-26 : Kuka *LWR*, configuration pour l'identification de l'articulation $i_c=1$

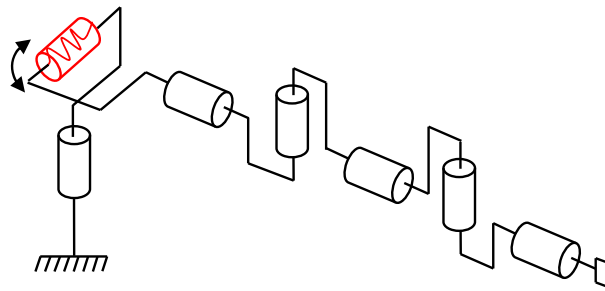


Figure 4-27 : Kuka *LWR*, configuration pour l'identification de l'articulation $i_c=2$

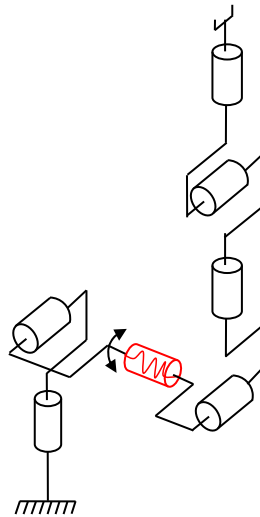


Figure 4-28 : Kuka *LWR*, configuration pour l'identification de l'articulation $i_c=3$

Afin d'augmenter les amplitudes du couple côté corps de l'articulation en cours d'identification, il est décidé de fixer la charge de 4.61 (kg) sur l'effecteur du robot. Seules les données acquises avec cette charge sont utilisées pour l'identification (il n'y a pas d'identification de la charge seule comme décrit à la partie 2.10). En conséquence, les paramètres $ZZ_{R_{iart}}$, MX_{iart} et MY_{iart} prennent en compte l'effet de la charge. Ceci n'a pas d'importance car on cherche seulement à identifier la raideur articulaire.

Seules les raideurs des articulations 1 à 3 sont identifiables car seules les configurations représentées aux figure 4-26, figure 4-27 et figure 4-28 permettent de respecter la condition (4.36).

Les résultats sont donnés dans le tableau 4-10. Les valeurs a priori des raideurs de Kuka (Jubien et al. 2014a) et fournies par le laboratoire Deutsches Zentrum für Luft-und Raumfahrt (DLR, Wessling, Allemagne) sont données dans le tableau 4-11.

Tableau 4-10 : Kuka *LWR*, paramètres et raideurs identifiées avec la méthode *IDIM-LS* (n.i. : non identifiable)

Par.	$\hat{\chi}$	$\% \hat{\sigma}_{\hat{\chi}}$	Par.	$\hat{\chi}$	$\% \hat{\sigma}_{\hat{\chi}}$	Par.	$\hat{\chi}$	$\% \hat{\sigma}_{\hat{\chi}}$
ZZ_{1R}	3.02	0.60	ZZ_{2R}	2.83	0.32	ZZ_{3R}	0.500	0.51
ZZ_{1R}/k_1	$1.99 \cdot 10^{-4}$	0.95	ZZ_{2R}/k_2	$1.85 \cdot 10^{-4}$	0.71	ZZ_{3R}/k_3	$3.35 \cdot 10^{-4}$	0.95
FvL_1	n.i.	-	MX_2	n.i.	-	MX_3	n.i.	-
FcL_1 / k_1	n.i.	-	MY_2	4.86	0.24	MY_3	-0.0077	9.01
FcL_1	n.i.	-	FvL_2	n.i.	-	FvL_3	n.i.	-
$OffL_1$	n.i.	-	FcL_2 / k_2	n.i.	-	FcL_3 / k_3	n.i.	-
-	-	-	FcL_2	n.i.	-	FcL_3	n.i.	-
-	-	-	$OffL_2$	n.i.	-	$OffL_3$	-0.198	8.23
k_1	15176	1.55	k_2	15297	1.03	k_3	14925	1.46

Tableau 4-11 : Kuka *LWR*, raideurs a priori

Par.	Valeur Kuka	Valeur <i>DLR</i>
k_1	10000	15000
k_2	10000	15000
k_3	10000	15000
k_4	10000	15000
k_5	10000	10000
k_6	7500	10000
k_7	7500	10000

Les raideurs identifiées (avec (4.40) pour l'axe 1 et (4.46) pour les axes 2 et 3) sont très proches des valeurs fournies par le laboratoire *DLR*. Les valeurs des raideurs utilisées par la suite seront celles du laboratoire *DLR* pour toutes les articulations. Les termes de frottements ne sont pas identifiables car trop faibles.

Cette méthode montre qu'il est possible d'identifier une partie des raideurs articulaires d'un robot si ce dernier est pourvu de capteurs de couples côté corps en sortie de réducteur.

4.5.4 Identification du modèle dynamique flexible avec la méthode *DIDIM*

4.5.4.1 Introduction

Kuka ne fournit aucune information précise sur le contrôleur implanté dans la baie de commande du *LWR*. Les informations fournies par le laboratoire *DLR* (*DLR*, 2014) impliquent que le contrôleur du robot est de type commande en couple calculé à l'aide du modèle dynamique inverse du robot (en anglais: Computed Torque Control). Hormis le type de contrôleur, aucune autre information n'est fournie (gains, bande passante, paramètres utilisés pour le calcul du modèle dynamique).

Le modèle dynamique inverse utilise les paramètres *a priori* de la matrice d'inertie (1.7) et du vecteur de gravité (A.12) (voir annexe G). Le couple de commande τ_s est calculé de la manière suivante (Khalil et Dombre, 2007):

$$\boldsymbol{\tau}_s = {}^{ap}\mathbf{M}(\mathbf{q}_{m_r})\mathbf{w}_r + {}^{ap}\mathbf{Q}(\mathbf{q}_{m_r}) \quad (4.48)$$

Où :

- ${}^{ap}\mathbf{M}(\mathbf{q}_{m_r})$ est la matrice d'inertie (7x7) *a priori* du robot fournie par le fabricant,
- ${}^{ap}\mathbf{Q}(\mathbf{q}_{m_r})$ est le vecteur (7x1) *a priori* des efforts de gravité fourni par le fabricant,
- \mathbf{w}_r est le vecteur (7x1) d'entrée de commande de type Proportionnelle, Intégrale, Dérivée :

$$\mathbf{w}_r = \mathbf{k}_p(\mathbf{q}_{m_r} - \mathbf{q}_m) + \mathbf{k}_v(\dot{\mathbf{q}}_{m_r} - \hat{\mathbf{q}}_m) + \mathbf{k}_i \int_{t_0}^t (\mathbf{q}_{m_r} - \mathbf{q}_m) dt + \ddot{\mathbf{q}}_m \quad (4.49)$$

Où :

- \mathbf{k}_p est la matrice (7x7) diagonale des 7 gains proportionnels normalisés,
- \mathbf{k}_v est la matrice (7x7) diagonale des 7 gains dérivés normalisés,
- \mathbf{k}_i est la matrice (7x7) diagonale des 7 gains intégraux normalisés,
- $\hat{\mathbf{q}}_m$ est le vecteur (7x1) des estimations des vitesses moteur en ligne (calculée numériquement à partir des positions moteurs par différence arrière).
- $\int_{t_0}^t (\mathbf{q}_{m_r} - \mathbf{q}_m) dt$ est le vecteur (7x1) des estimations de l'intégrale des erreurs de poursuite (calculée numériquement par la méthode d'Euler avant).

Sous hypothèse de modèle parfait et d'un suivi parfait de trajectoire, on a :

$$\begin{aligned} \mathbf{q}_{m_r} &= \mathbf{q}_m \\ {}^{ap}\mathbf{M}(\mathbf{q}_{m_r}) &= {}^a\mathbf{M}(\mathbf{q}_m) \\ {}^{ap}\mathbf{Q}(\mathbf{q}_{m_r}) &= {}^a\mathbf{Q}(\mathbf{q}_m) \end{aligned} \quad (4.50)$$

Où :

- ${}^a\mathbf{M}(\mathbf{q}_m)$ est la matrice d'inertie (7x7) du robot réel,
- ${}^a\mathbf{Q}(\mathbf{q}_m)$ est le vecteur (7x1) des efforts de gravité du robot réel.

L'équation (4.48) s'écrit :

$$\boldsymbol{\tau}_s = {}^{ap}\mathbf{M}(\mathbf{q}_{m_r})\mathbf{w}_r + {}^{ap}\mathbf{Q}(\mathbf{q}_{m_r}) = {}^a\mathbf{M}(\mathbf{q}_m)\ddot{\mathbf{q}}_m + {}^a\mathbf{Q}(\mathbf{q}_m) \quad (4.51)$$

Soit :

$$\mathbf{w}_r = \ddot{\mathbf{q}}_m \quad (4.52)$$

La commande dynamique idéale sans erreur de modèle linéarise et découple le système non linéaire en n_{iart} doubles intégrateurs articulaires (les flexibilités, les efforts centrifuges, les efforts de Coriolis et les frottements sont négligés par Kuka pour la commande du *LWR*).

L'équation (4.49), avec $\mathbf{w}_r = \ddot{\mathbf{q}}_m$, s'écrit sous la forme d'une équation d'erreur libre :

$$\ddot{e}_{m_j} + k_{v_j} \dot{e}_{m_j} + k_{p_j} e_{m_j} + k_{i_j} \int_{t_0}^t e_{m_j} dt = 0 \quad (4.53)$$

Où $e_{m_j} = q_{m_j} - q_{m_j}$ est l'erreur de poursuite de l'articulation j .

Cette équation différentielle correspond à une fonction de transfert $h_j(s)$ pour l'articulation j :

$$h_j(s) = \frac{1}{s^3 + k_{v_j} s^2 + k_{p_j} s + k_{i_j}} \quad (4.54)$$

Les 3 pôles de $h_j(s)$ définissent la bande passante et la marge de stabilité du contrôle de l'erreur de poursuite e_{m_j} . On choisit un pôle triple ${}^d\omega_{n_j}$ correspondant à un pôle réel et un pôle double complexe de système de second ordre avec un amortissement de 1 et une pulsation naturelle désirée ${}^d\omega_{n_j}$ (voir annexe G).

Ce choix correspond à l'équation suivante :

$$\begin{aligned} (s + {}^d\omega_{n_j})^3 &= s^3 + k_{v_j} s^2 + k_{p_j} s + k_{i_j} \\ &= s^3 + 3 {}^d\omega_{n_j} s^2 + 3 {}^d\omega_{n_j}^2 s + {}^d\omega_{n_j}^3 \end{aligned} \quad (4.55)$$

Soit :

$$\begin{aligned} k_{v_j} &= 3 {}^d\omega_{n_j} \\ k_{p_j} &= 3 {}^d\omega_{n_j}^2 \\ k_{i_j} &= {}^d\omega_{n_j}^3 \end{aligned} \quad (4.56)$$

Le simulateur utilisé et programmé avec le logiciel Matlab/Simulink est représenté figure 4-29. La partie *PID* du contrôleur et le modèle dynamique direct du robot sont représentés respectivement figure 4-30 et figure 4-31 (avec $tec=T_e$). Ce dernier est codé en langage C et est compilé sous forme de 'C-Mex function' pour diminuer le temps de simulation. Le modèle dynamique direct du robot est celui du robot flexible pour prendre en compte les flexibilités articulaires. La fréquence de mesure f_m et de commande sont fixées à 1 (kHz).

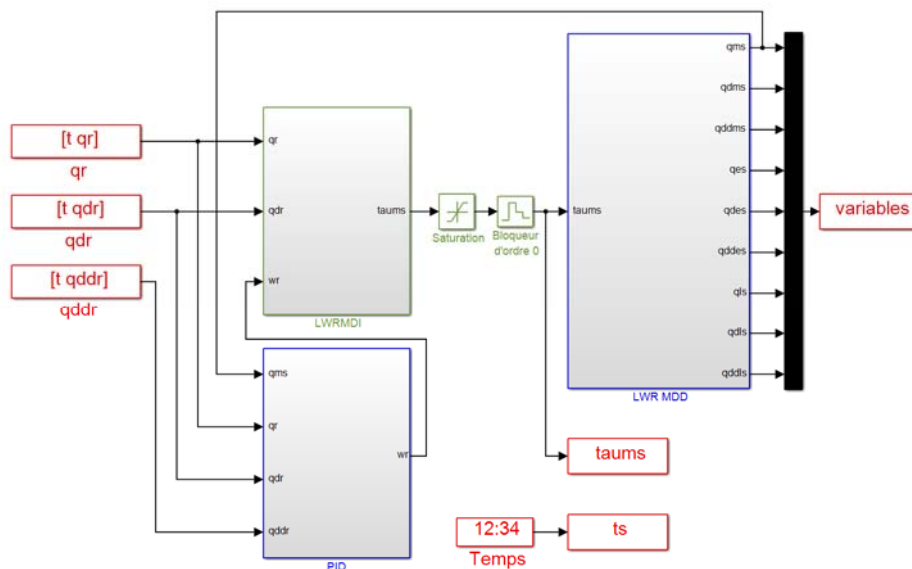


Figure 4-29 : Simulateur du Kuka *LWR* sous Matlab/Simulink

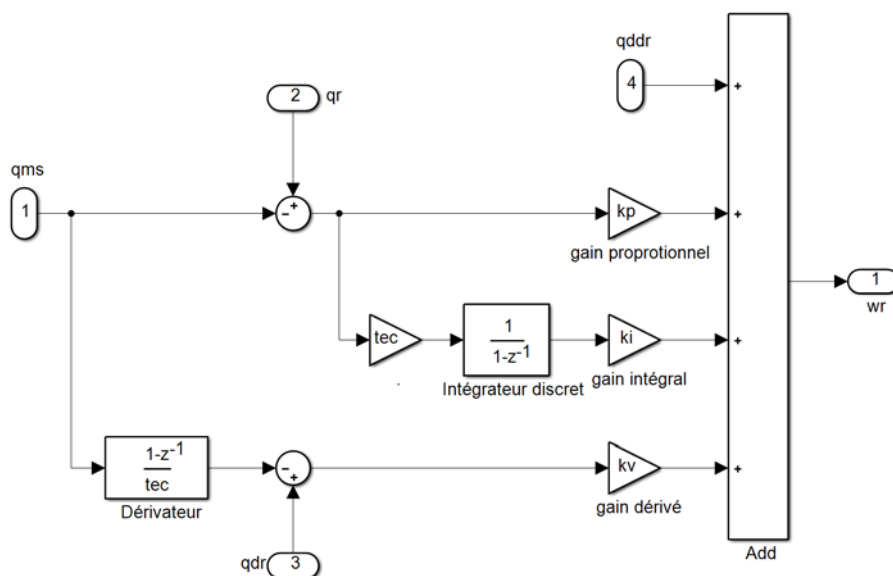
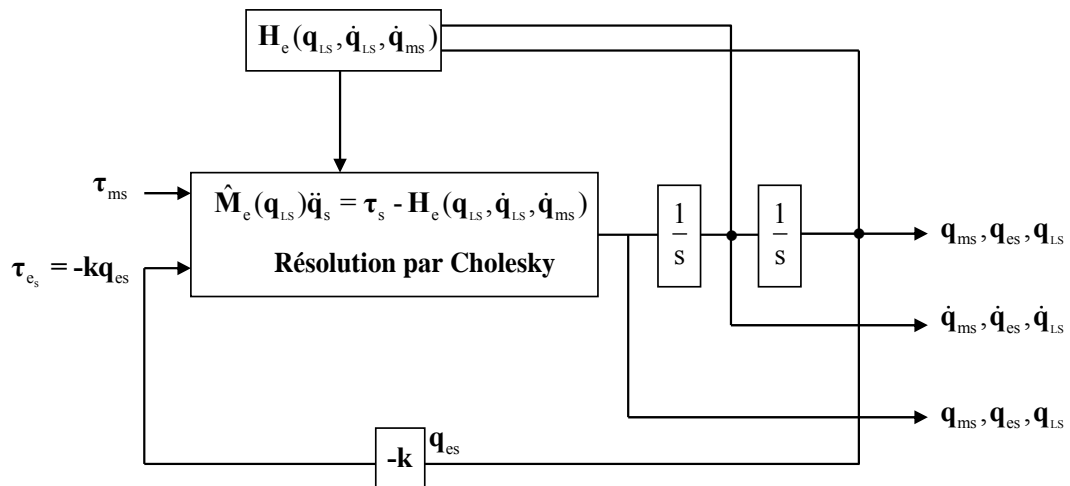


Figure 4-30 : Contrôleur simulé du Kuka *LWR*


 Figure 4-31 : Modèle dynamique du Kuka *LWR*

4.5.4.2 Paramètres utilisés pour la simulation

Les paramètres du modèle dynamique flexible sont calculés à partir de ceux du modèle dynamique rigide identifiés à la partie 4.5.2 et des raideurs a priori du laboratoire *DLR* données dans le tableau 4-11. Ils sont utilisés par le simulateur du Kuka *LWR* flexible pour calculer les 7 couples moteurs qui servent de couples moteurs "réels" par la suite.

La répartition des inerties a lieu dans la chaîne d'actionnement et on suppose que la flexibilité est localisée dans l'harmonic drive de chaque articulation, proche de la sortie du réducteur (voir figure 1-12). En conséquence l'inertie de la chaîne d'actionnement est prépondérante côté moteur et les rapports d'inertie utilisés sont fixés à 0.8 pour chaque articulation.

Les frottements visqueux en sortie de la chaîne d'actionnement côté corps du robot sont nuls et ceux en entrée de la chaîne d'actionnement côté moteur prennent les valeurs des frottements visqueux du modèle dynamique rigide (Jubien et al. 2014b). La répartition des frottements de Coulomb est de 0.05 pour les frottements de Coulomb en sortie de la chaîne d'actionnement côté corps du robot (0.95 pour les frottements de Coulomb en entrée de la chaîne d'actionnement côté moteur). Les offsets du modèle dynamique flexible sont ceux du modèle dynamique rigide.

Les paramètres de base (différents de 0) utilisés par la simulation sont donnés dans le tableau 4-12.

Tableau 4-12 : Kuka *LWR*, paramètres de base (différents de 0) du modèle dynamique flexible utilisés pour la simulation

Par.	^{ap} χ	Par.	^{ap} χ
Iam ₁	2.51	Fcm ₅	8.87
Fvm ₁	13.2	k ₅	10000
Fcm ₁	13.1	Offm ₅	0.0158
k ₁	15000	XX _{5R}	-0.00397
Offm ₁	0.121	XY ₅	0.0162
ZZ _{1R} (flexible)	0.627	XZ ₅	0.0138
Fcl ₁	0.654	YZ ₅	-0.0208
Iam ₂	3.66	ZZ _{5R}	0.14
Fvm ₂	13.3	MX ₅	0.0093
Fcm ₂	13.4	MY _{5R}	0.0419
k ₂	15000	Fcl ₅	0.444
Offm ₂	0.549	Iam ₆	0.3
XX _{2R}	1.39	Fvm ₆	1.78
XY ₂	0.07	Fcm ₆	5.13
XZ ₂	0.0479	k ₆	10000
YZ ₂	-0.0213	Offm ₆	0.0288
ZZ _{2R} (flexible)	0.916	XX _{6R}	-0.00526
MX ₂	-0.0391	XY ₆	0.0117
MY _{2R}	3.32	XZ ₆	0.00545
Fcl ₂	0.668	YZ ₆	-0.017
Iam ₃	1.5	ZZ _{6R}	0.0751
Fvm ₃	5.76	MX ₆	-0.0165
Fcm ₃	9.78	MY _{6R}	0.0341
k ₃	15000	Fcl ₆	0.256
Offm ₃	0.0854	Iam ₇	0.301
XX _{3R}	-0.162	Fvm ₇	1.43
XY ₃	-0.081	Fcm ₇	5.98
XZ ₃	-0.0259	k ₇	10000
YZ ₃	-0.0207	Offm ₇	0.0578
ZZ _{3R}	0.375	XX _{7R}	0.0198
MX ₃	-0.0496	XY ₇	-0.00427
MY _{3R}	-0.0198	XZ ₇	-0.00312
Fcl ₃	0.489	YZ ₇	-0.00279
Iam ₄	1.78	ZZ ₇	0.0753
Fvm ₄	9.89	MX ₇	0.00877
Fcm ₄	9.25	MY ₇	0.00371
k ₄	15000	Fcl ₇	0.299
Offm ₄	0.0333	XX _L	0.0983
XX _{4R}	0.413	XY _L	0.0212
XY ₄	0.0374	XZ _L	-0.0333
XZ ₄	-0.00665	YY _L	0.117
YZ ₄	-0.00712	YZ _L	0.0307
ZZ _{4R}	0.445	ZZ _L	0.0407
MX ₄	-0.0176	MX _L	0.323
MY _{4R}	-1.36	MY _L	-0.309
Fcl ₄	0.463	MZ _L	0.551
Iam ₅	0.561	M _L	4.70
Fvm ₅	3.93		

On remarque que les fréquences naturelles à entrée bloquée *a priori* sont comprises entre 12 (Hz) (axe 1) et 60 (Hz) (axe 7) pour le robot en position verticale ($q_{m1...7}=0$ rd). La fréquence de coupure du filtre decimate est de 120 (Hz) pour permettre l'identification du modèle dynamique flexible.

Les consignes de positions utilisées pour la simulation sont celles utilisés pour l'identification du modèle dynamique rigide. On choisit pour ce robot d'y superposer un train d'impulsion de $1 \cdot 10^{-3}$ (rd) d'amplitude et de durée de 200 (ms) avec une fréquence de 0.25 (Hz) afin d'exciter les flexibilités.

4.5.4.3 Méthode *DIDIM* sans bruit

Pour valider la convergence de la procédure d'optimisation non linéaire, l'algorithme proposé est testé avec les efforts moteurs simulés sans aucun bruit sur les mesures (représentés figure 4-32 et figure 4-33).

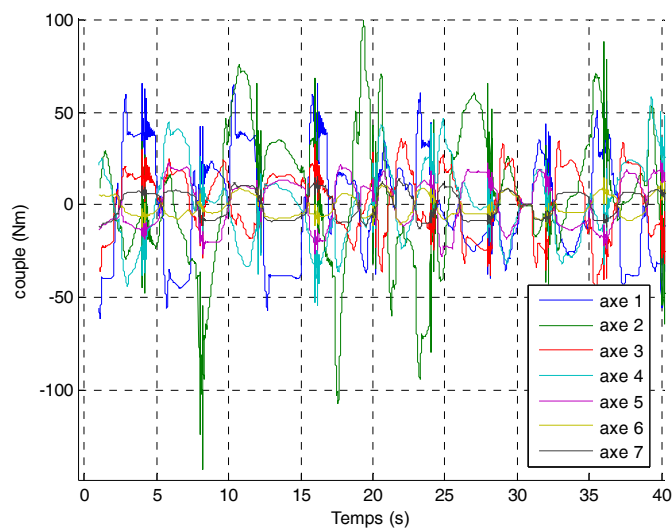


Figure 4-32 : Efforts moteurs simulés pour le Kuka *LWR* flexible

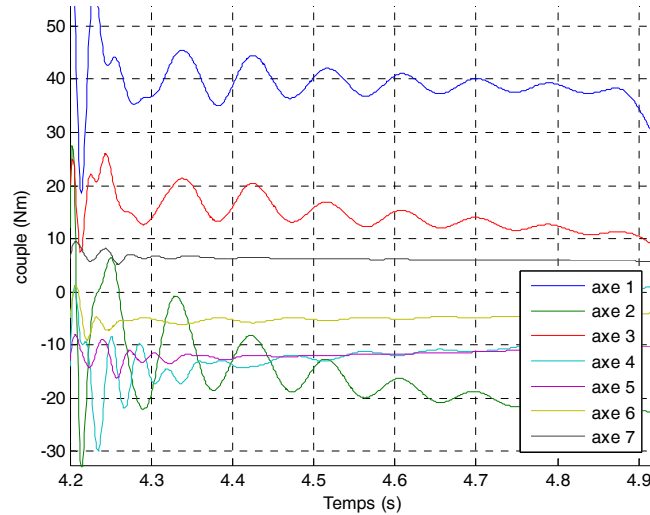


Figure 4-33 : Zoom sur les efforts moteurs simulés pour le Kuka *LWR* flexible

Etape 1

Le modèle dynamique rigide décrit à la partie 1.5.1 est identifié avec la méthode *DIDIM* durant la première étape. La fréquence de coupure du filtre Decimate est fixée à 0.8 (Hz).

Les consignes décrites à la partie précédente sont utilisées. L'identification est effectuée avec la charge de 4.61 (kg).

L'initialisation donnée tableau 4-13 est utilisée.

Tableau 4-13 : Kuka *LWR*, conditions initiales sur les moments d'inerties

j	${}^0 I_a = {}^{ap} I_a$
1	2
2	2
3	2
4	2
5	1
6	1
7	1

Les moments d'inerties des chaînes d'actionnement sont choisis assez éloignés des valeurs données dans le tableau 4-12 pour tester la robustesse de la méthode *DIDIM*.

Les paramètres de base identifiés du modèle dynamique rigide du Kuka *LWR* à partir des couples du Kuka *LWR* flexible sont donnés dans le tableau 4-12.

. La 3ème colonne et la 6ème colonne donnent les erreurs relatives par rapport aux paramètres du modèle dynamique flexible du tableau 4-12. Pour les cas particuliers des frottements secs et des paramètres ZZ_{1R} et ZZ_{2R} , ils sont comparés respectivement aux sommes des frottements sec côté moteur et côté corps, à la somme de l'inertie de la chaîne d'actionnement coté moteur de

l'articulation 1 et du terme ZZ_{1R} (flexible) et à la somme de l'inertie de la chaîne d'actionnement coté moteur de l'articulation 2 et du terme ZZ_{2R} (flexible).

Tableau 4-14 : Kuka *LWR*, paramètres de base identifiés pour l'étape 1

Par.	$\hat{\chi}^2$	%e $_{\hat{\chi}}$	Par.	$\hat{\chi}^2$	%e $_{\hat{\chi}}$
ZZ_{1R} (rigide)	3.15	0.41	ZZ_{5R}	0.138	1.43
FV ₁	13.2	0	MX ₅	0.00763	18
Fc ₁	13.7	0.39	MY _{5R}	0.0429	2.39
Off ₁	0.119	1.65	Ia ₅	0.562	0.18
XX _{2R}	1.41	1.44	Fv ₅	3.97	1.01
XY ₂	0.0718	2.57	Fc ₅	9.25	0.69
XZ ₂	0.0505	5.42	Off ₅	0.00282	82
YZ ₂	-0.0284	33	XX _{6R}	-0.00731	39
MX ₂	-0.0403	3.07	XY ₆	0.0124	5.98
MY _{2R}	3.32	0	XZ ₆	0.00784	44
ZZ_{2R} (rigide)	4.6	0.52	YZ ₆	-0.0179	5.29
FV ₂	13.3	0	ZZ _{6R}	0.0714	4.92
Fc ₂	14	0.49	MX ₆	-0.0152	7.88
Off ₂	0.562	2.37	MY _{6R}	0.0365	7.04
XX _{3R}	-0.162	0	Ia ₆	0.297	1.0
XY ₃	-0.0736	9.14	Fv ₆	1.8	1.12
XZ ₃	-0.0282	8.88	Fc ₆	5.37	0.30
YZ ₃	-0.0231	11.6	Off ₆	0.0287	0.35
ZZ _{3R}	0.364	2.93	XX _{7R}	0.0179	9.60
MX ₃	-0.0499	0.60	XY ₇	-0.00421	1.40
MY _{3R}	-0.021	6.07	XZ ₇	-0.00306	1.92
Ia ₃	1.52	1.33	YZ ₇	-0.00226	19
FV ₃	5.77	0.18	ZZ ₇	0.0755	0.27
Fc ₃	10.3	0.30	MX ₇	0.00811	7.52
Off ₃	0.0721	16	MY ₇	0.00327	11
XX _{4R}	0.428	3.64	Ia ₇	0.293	2.66
XY ₄	0.0366	2.14	Fv ₇	1.44	0.70
XZ ₄	-0.00481	28	Fc ₇	6.27	0.15
YZ ₄	-0.00427	40	Off ₇	0.0495	15
ZZ _{4R}	0.451	1.35	XX _L	0.106	7.84
MX ₄	-0.0183	3.98	XY _L	0.0203	4.25
MY _{4R}	-1.36	0	XZ _L	-0.0348	4.50
Ia ₄	1.77	0.57	YY _L	0.125	6.84
Fv ₄	9.9	0.10	YZ _L	0.0311	1.30
Fc ₄	9.7	0.13	ZZ _L	0.0406	0.25
Off ₄	0.0279	17	MX _L	0.324	0.31
XX _{5R}	-0.00676	77	MY _L	-0.309	0
XY ₅	0.0179	10	MZ _L	0.55	0.18
XZ ₅	0.0135	2.28	M _L	4.72	0.43
YZ ₅	-0.0218	4.80			

Hormis certains paramètres très faibles et certains offsets, les paramètres sont proches des paramètres du tableau 4-12 (inférieurs à 10% en moyenne). L'algorithme converge en 2 itérations (76 secondes) avec une norme relative du résidu finale de 0.56%. Cette faible norme montre qu'avec l'application du filtrage parallèle à une fréquence inférieure au premier mode flexible, il est possible d'identifier un modèle dynamique rigide d'un robot à flexibilités articulaires qui représente le

comportement basse fréquence du robot. Les vecteurs de mesure \mathbf{Y}_j sont proches des modèles $\mathbf{W}_j\boldsymbol{\chi}$ représentés figure 4-34 et figure 4-35. Un test a été effectué avec une fréquence de coupure du filtre decimate réglée à 80 (Hz), dans ce cas la norme relative du résidu finale est de 13.5% avec des variations sur les paramètres de l'ordre de 20% en moyenne comparés aux paramètres du tableau 4-12, il est donc nécessaire de régler la fréquence de coupure du filtre decimate à une fréquence compatible avec le comportement basse fréquence du modèle rigide.

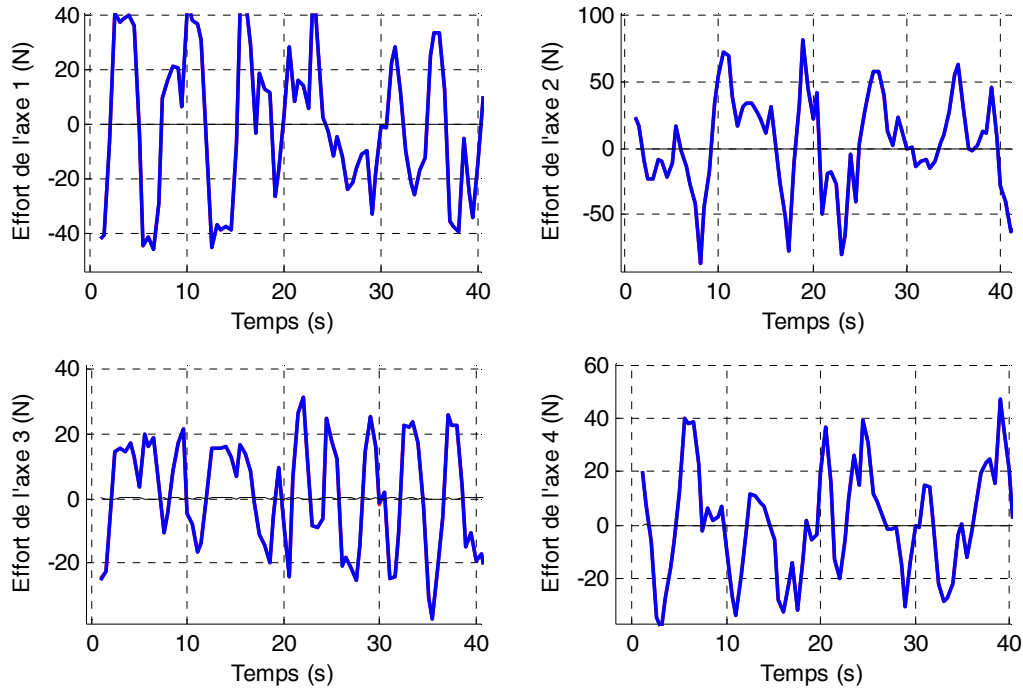


Figure 4-34 : Kuka *LWR*, Validation directe avec la méthode *DIDIM*, étape 1, \mathbf{Y} (rouge), $\mathbf{W}\boldsymbol{\chi}$ (bleu) et erreur (noir), axes 1 à 4

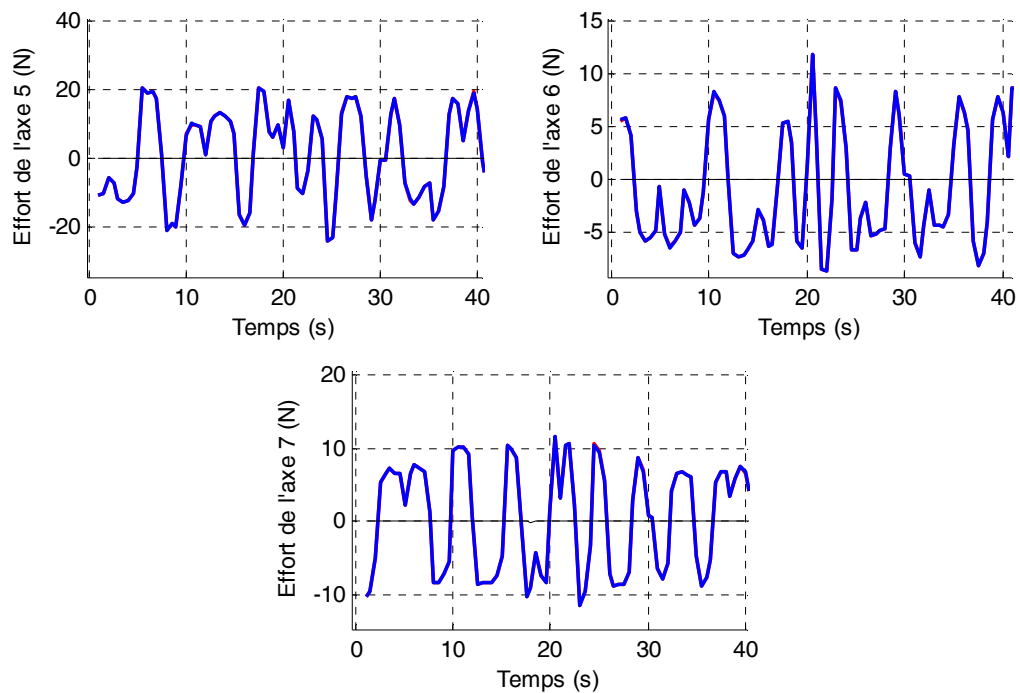


Figure 4-35 : Kuka LWR, Validation directe avec la méthode *DIDIM*, étape 1, Y (rouge), $W\chi$ (bleu) et erreur (noir), axes 5 à 7

Les paramètres identifiés durant cette première étape vont être utilisés aux deux étapes suivantes.

Etape 2

Cette étape permet d'identifier les 7 raideurs et les 7 rapports d'inertie du Kuka LWR flexible.

Les paramètres du robot sont calculés à chaque itération k de l'algorithme de programmation non linéaire avec l'équation (4.14).

L'algorithme d'optimisation non linéaire est celui du Simplex de Nelder Mead. Il y a en tout 14 variables à optimiser.

Les conditions initiales sont données dans le tableau 4-15.

L'algorithme converge en 102 itérations et 1h42min. Le temps de convergence élevé s'explique car il faut simuler le robot flexible plusieurs fois à chaque itération et il y a 14 paramètres à identifier. Chaque simulation du robot flexible prend environ 20 secondes.

Les valeurs identifiées sont proches des valeurs données dans la partie 4.5.4.2. Les différences relatives entre les valeurs données dans la partie 4.5.4.2 et les valeurs identifiées dans le tableau 4-15 sont dues au fait que les paramètres des corps du modèle dynamique rigide identifiés à l'étape 1 sont légèrement différents des paramètres des corps du modèle dynamique flexible "réel".

Tableau 4-15 : Kuka *LWR*, raideurs, k et λ en fonction de k

Itération	k=0		k=102	
	0k	${}^0\lambda$	${}^{102}k$	${}^{102}\lambda$
1	5000	0.5	15180	0.81
2	5000	0.5	15201	0.80
3	5000	0.5	14908	0.79
4	5000	0.5	15121	0.79
5	5000	0.5	9920	0.81
6	5000	0.5	10181	0.79
7	5000	0.5	10065	0.80

Différentes initialisations ont été testées sur les raideurs (entre 3000 (Nm/rad) et 40000 (Nm/rad)) et les rapports d'inerties (entre 0.4 et 0.9), elles permettent toutes la convergence de l'optimisation non linéaire en un temps limité (inférieur à 3h) vers des valeurs identiques ou très proches de celle du tableau 4-15.

Etape 3

Les valeurs identifiées aux 2 étapes précédentes vont permettre d'identifier le modèle dynamique flexible du Kuka *LWR*.

Les paramètres du simulateur du Kuka *LWR* flexible sont initialisés avec (4.21).

La fréquence de coupure du filtre Decimate est fixée à 160 (Hz).

L'algorithme converge en 4 itérations (160 secondes). Les vecteurs de mesure \mathbf{Y}_j se superposent aux modèles $\mathbf{W}_j\boldsymbol{\chi}$, (voir figure 4-36 - plus zoom sur les oscillations - et figure 4-37). La norme relative du résidu est de 10^{-11} pour l'itération finale due seulement aux erreurs numériques durant le processus d'intégration du modèle dynamique direct par le logiciel Matlab/Simulink.

Tous les paramètres sont bien identifiés et sont égaux aux valeurs données dans le tableau 4-12 (erreurs relatives comprises entre 10^{-9} et 10^{-13}).

Comme pour l'*EMPS*, l'algorithme en trois étapes proposé est validé en simulation sur un robot à flexibilités articulaires avec seulement la mesure des efforts moteurs. Les trois étapes s'exécutent en 2h, essentiellement dû à la deuxième étape sur un processeur double-cœur de 2 (GHz).

Tableau 4-16 : Kuka *LWR*, paramètres de base différents de 0 identifiés pour l'étape 3

Par.	$\hat{\chi}^4$	Par.	$\hat{\chi}^4$
Iam ₁	2.51	Fcm ₅	8.87
Fvm ₁	13.2	k ₅	10000
Fcm ₁	13.1	Offm ₅	0.0158
k ₁	15000	XX _{5R}	-0.00397
Offm ₁	0.121	XY ₅	0.0162
ZZ _{1R} (flexible)	0.627	XZ ₅	0.0138
FCL ₁	0.654	YZ ₅	-0.0208
Iam ₂	3.66	XX _{5R}	0.14
Fvm ₂	13.3	MX ₅	0.0093
Fcm ₂	13.4	MY _{5R}	0.0419
k ₂	15000	FCL ₅	0.444
Offm ₂	0.549	Iam ₆	0.3
XX _{2R}	1.39	Fvm ₆	1.78
XY ₂	0.07	Fcm ₆	5.13
XZ ₂	0.0479	k ₆	10000
YZ ₂	-0.0213	Offm ₆	0.0288
ZZ _{2R} (flexible)	0.916	XX _{6R}	-0.00526
MX ₂	-0.0391	XY ₆	0.0117
MY _{2R}	3.32	XZ ₆	0.00545
FCL ₂	0.668	YZ ₆	-0.017
Iam ₃	1.5	ZZ _{6R}	0.0751
Fvm ₃	5.76	MX ₆	-0.0165
Fcm ₃	9.78	MY _{6R}	0.0341
k ₃	15000	FCL ₆	0.256
Offm ₃	0.0854	Iam ₇	0.301
XX _{3R}	-0.162	Fvm ₇	1.43
XY ₃	-0.081	Fcm ₇	5.98
XZ ₃	-0.0259	k ₇	10000
YZ ₃	-0.0207	Offm ₇	0.0578
ZZ _{3R}	0.375	XX _{7R}	0.0198
MX ₃	-0.0496	XY ₇	-0.00427
MY _{3R}	-0.0198	XZ ₇	-0.00312
FCL ₃	0.489	YZ ₇	-0.00279
Iam ₄	1.78	ZZ ₇	0.0753
Fvm ₄	9.89	MX ₇	0.00877
Fcm ₄	9.25	MY ₇	0.00371
k ₄	15000	FCL ₇	0.299
Offm ₄	0.0333	XX _L	0.0983
XX _{4R}	0.413	XY _L	0.0212
XY ₄	0.0374	XZ _L	-0.0333
XZ ₄	-0.00665	YY _L	0.117
YZ ₄	-0.00712	YZ _L	0.0307
ZZ _{4R}	0.445	ZZ _L	0.0407
MX ₄	-0.0176	MX _L	0.323
MY _{4R}	-1.36	MY _L	-0.309
FCL ₄	0.463	MZ _L	0.551
Iam ₅	0.561	M _L	4.70
Fvm ₅	3.93		

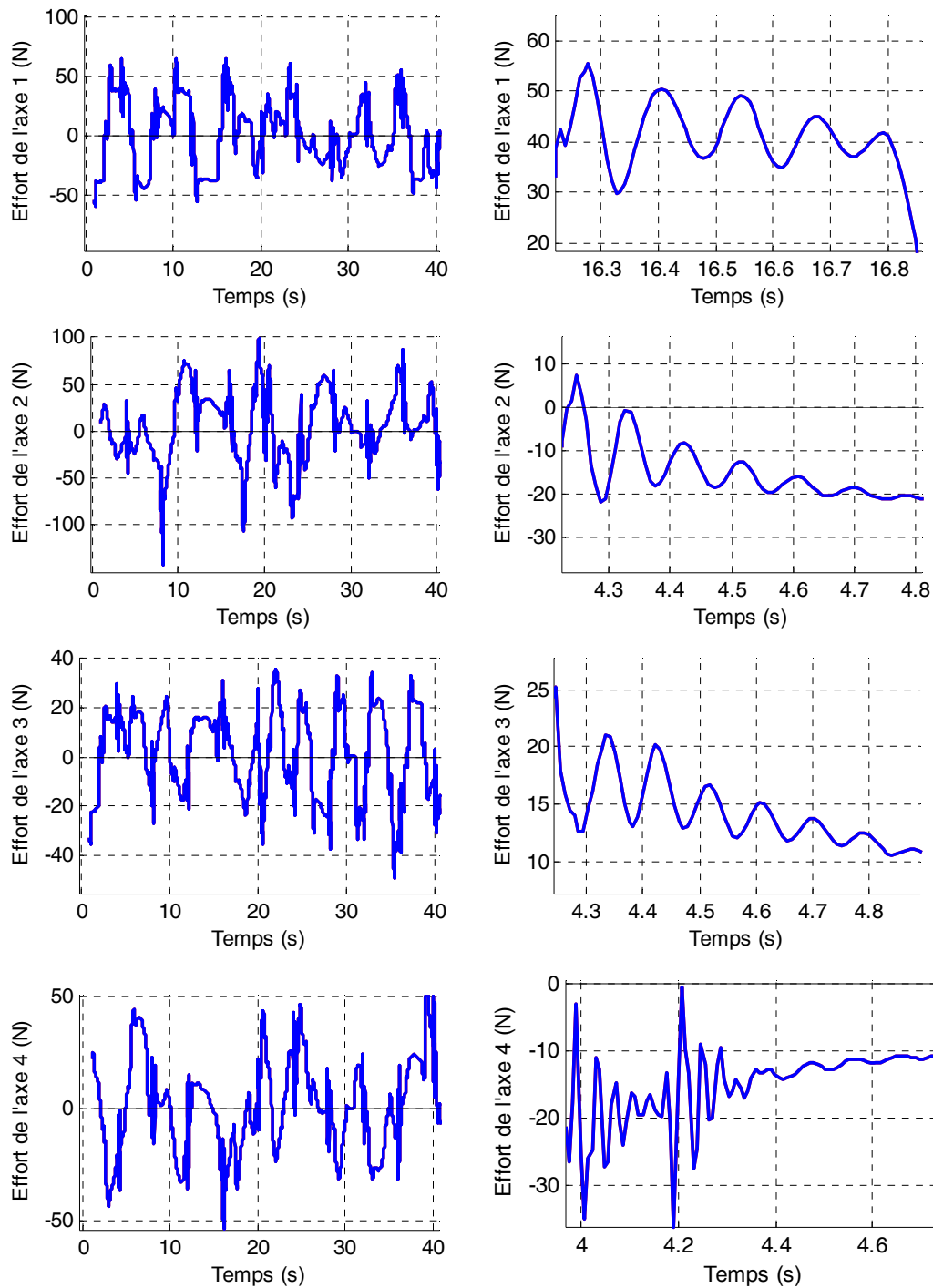


Figure 4-36 : Kuka LWR, Validation directe avec la méthode *DIDIM*, étape 3, Y (rouge), W_χ (bleu) et erreur (noir), axes 1 à 4, avec zoom sur les oscillations

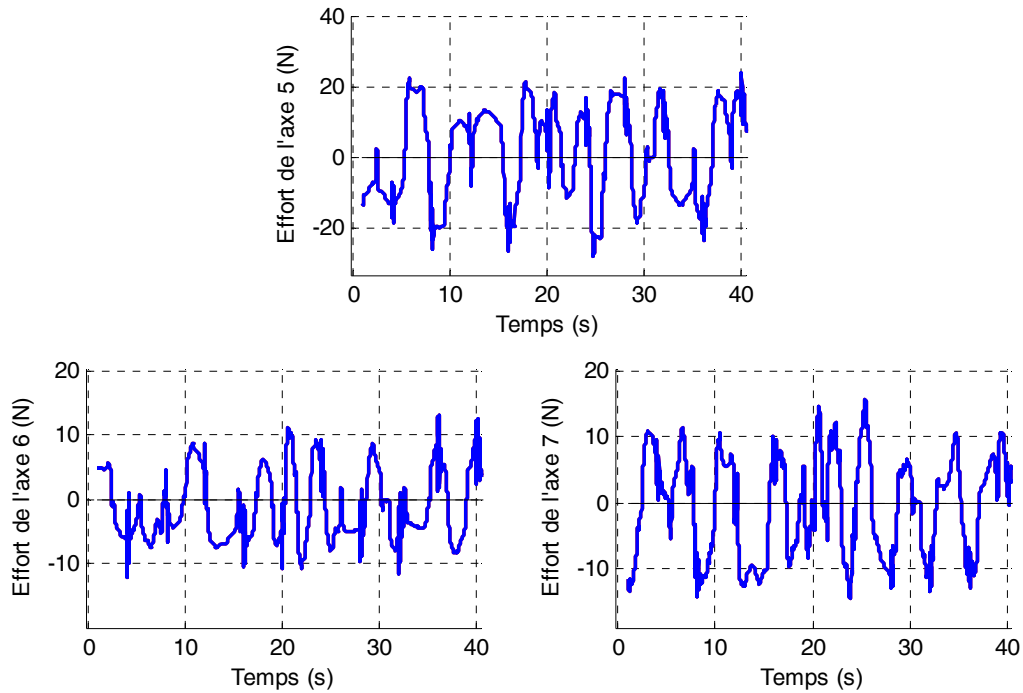


Figure 4-37 : Kuka LWR, Validation directe avec la méthode *DIDIM*, étape 3, Y (rouge), $W\chi$ (bleu) et erreur (noir), axes 5 à 7

4.5.4.4 Méthode *DIDIM* avec bruit

Dans un deuxième temps, pour tester la robustesse de l'algorithme d'optimisation non linéaire proposé vis à vis des bruits de mesure et pour s'approcher de la réalité, un bruit gaussien centré d'écart-type 0.5 (Nm) est rajouté sur les 7 efforts moteurs. Ce niveau de bruit est proche de celui observé sur les efforts moteurs mesurés du robot (de l'ordre de 1 à 4% du couple nominal). Les nouveaux efforts moteurs sont représentés figure 4-38.

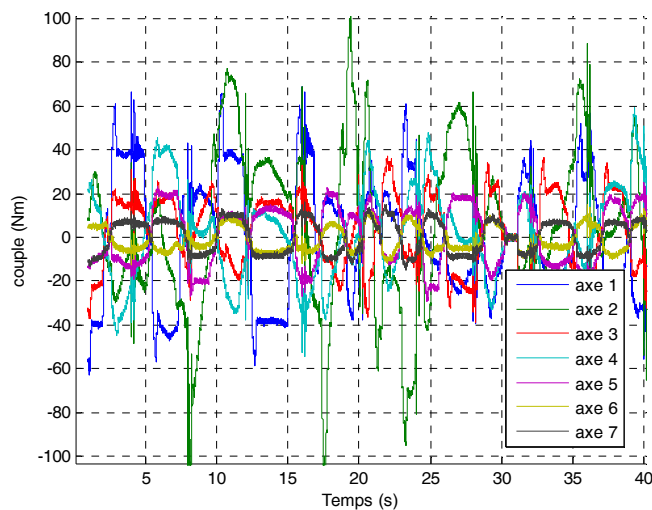


Figure 4-38 : Kuka LWR, Efforts moteurs simulés avec bruit

Etape 1

Avec les mêmes conditions (modèle, filtrage, conditions initiales) qu'à l'étape 1 de la partie 4.5.4.3, les paramètres du modèle dynamique rigide du Kuka *LWR* sont identifiés à partir des couples du Kuka *LWR* flexible avec bruit. Les résultats sont donnés dans le tableau 4-17.

Tableau 4-17 : Kuka *LWR*, paramètres identifiés pour l'étape 1

Par.	$\hat{\chi}^2$	$\% \hat{\sigma}_{\hat{\chi}^2}$	Par.	$\hat{\chi}^2$	$\% \hat{\sigma}_{\hat{\chi}^2}$
ZZ _{1R} (rigide)	3.15	0.176	XZ ₅	0.0136	11.6
FV ₁	13.2	0.106	YZ ₅	-0.0221	9.76
FS ₁	13.7	0.13	ZZ _{5R}	0.138	1.87
Off ₁	0.119	6.78	MY _{5R}	0.0428	2.21
XX _{2R}	1.41	0.648	Ia ₅	0.562	0.616
XY ₂	0.0716	5.97	FV ₅	3.97	0.244
XZ ₂	0.051	9.41	Fc ₅	9.25	0.164
ZZ _{2R} (rigide)	4.6	0.202	XY ₆	0.0123	9.76
MX ₂	-0.0401	7.07	XZ ₆	0.00804	19.5
MY _{2R}	3.32	0.0652	YZ ₆	-0.0181	9.71
FV ₂	13.3	0.117	ZZ _{6R}	0.0715	4.57
Fc ₂	14	0.116	MX ₆	-0.0151	7.8
Off ₂	0.56	3.81	MY _{6R}	0.0363	4.04
XX _{3R}	-0.162	4.88	Ia ₆	0.297	2.03
XY ₃	-0.0739	4.67	FV ₆	1.8	0.97
XZ ₃	-0.0279	10.9	Fc ₆	5.37	0.264
ZZ _{3R}	0.365	1.06	XX _{7R}	0.0183	13.9
MX ₃	-0.0497	3.24	ZZ ₇	0.0754	2.25
MY _{3R}	-0.021	7.26	Ia ₇	0.294	1.18
Ia ₃	1.52	0.325	FV ₇	1.44	0.698
FV ₃	5.77	0.211	Fc ₇	6.27	0.263
Fc ₃	10.3	0.16	XX _L	0.106	3.12
XX _{4R}	0.427	0.885	XY _L	0.0199	8.43
XY ₄	0.0367	4.69	XZ _L	-0.0348	3.8
ZZ _{4R}	0.451	0.934	YY _L	0.125	2.87
MX ₄	-0.0183	7.12	YZ _L	0.0312	4.11
MY _{4R}	-1.36	0.118	ZZ _L	0.0402	4.35
Ia ₄	1.77	0.368	MX _L	0.323	0.383
FV ₄	9.9	0.166	MY _L	-0.309	0.393
Fc ₄	9.7	0.166	MZ _L	0.55	0.334
XY ₅	0.0175	10.4	M _L	4.72	0.0826

On rappelle qu'un bruit gaussien centré a été ajouté aux efforts moteurs simulés avec le modèle dynamique flexible, en conséquence certains paramètres ne sont plus identifiables. Mais la majorité des paramètres sont bien identifiés et possèdent de faibles écart-types. L'algorithme converge en 2 itérations (76 secondes) avec une norme relative du résidu finale de 0.61%. Les vecteurs de mesure Y_j sont proches des modèles $W_j \chi$ représentés figure 4-39.

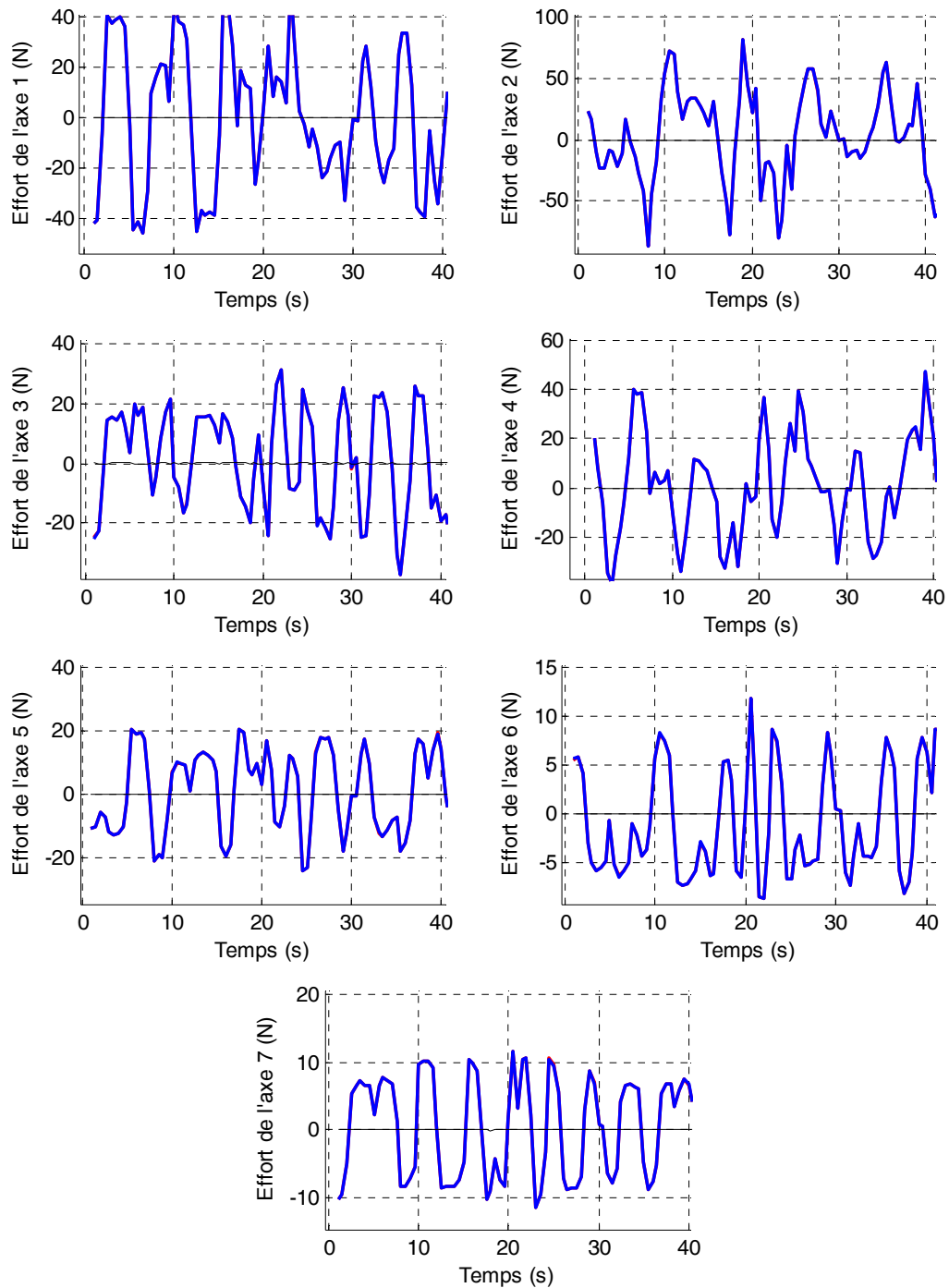


Figure 4-39 : Kuka *LWR*, Validation directe avec la méthode *DIDIM*, étape 1, Y (rouge), $W\chi$ (bleu) et erreur (noir)

Etape 2

Cette étape permet d'identifier les 7 raideurs et les 7 rapports d'inertie du Kuka *LWR* flexible avec les mêmes conditions qu'à l'étape 2 de la partie 4.5.4.3 mais sur les efforts bruités.

L'algorithme converge en 111 itérations et 2h12min. L'utilisation d'efforts moteurs bruités filtré par un filtre de type Butterworth aller-retour (fréquence de coupure fixée à 120 (Hz)) rallonge

légèrement le temps de convergence. Les valeurs identifiées sont un peu moins proches des valeurs donnés dans la partie 4.5.4.2 que pour le cas sans bruit.

Tableau 4-18 : Kuka *LWR*, raideurs, k et λ en fonction de k

Itération	k=0		K=111	
	0k	${}^0\lambda$	${}^{111}k$	${}^{111}\lambda$
1	5000	0.5	14935	0.81
2	5000	0.5	14850	0.82
3	5000	0.5	15099	0.81
4	5000	0.5	15252	0.78
5	5000	0.5	9910	0.83
6	5000	0.5	10277	0.79
7	5000	0.5	10156	0.80

Etape 3

Les valeurs identifiées aux 2 étapes précédentes vont permettre d'identifier le modèle dynamique flexible du Kuka *LWR* avec des efforts moteurs bruités. L'algorithme converge en 4 itérations. Les vecteurs de mesure \mathbf{Y}_j sont proches des modèles $\mathbf{W}_j\boldsymbol{\chi}$, (voir figure 4-40). La norme relative du résidu est de 0.7% pour l'itération finale.

Tous les paramètres sont bien identifiés et sont proches des valeurs données dans la partie 4.5.4.2. Contrairement au bras *EMPS*, les frottements sont bien répartis et bien identifiés avec le bruit simulé. Ce résultat devra être confirmé avec des mesures réelles et bruitées.

Cette expérimentation prouve que la convergence de la procédure d'optimisation non linéaire est robuste à un bruit simulé sur les couples.

Tableau 4-19 : Kuka *LWR*, paramètres identifiés pour l'étape 3

Par.	$\hat{\chi}^4$	$\% \hat{\sigma}_{\hat{\chi}_i^4}$	Par.	$\hat{\chi}^4$	$\% \hat{\sigma}_{\hat{\chi}_i^4}$
Iam ₁	2.51	0.0162	Fcl ₄	0.463	0.182
Fvm ₁	13.2	0.00923	Iam ₅	0.561	0.0452
Fcm ₁	13.1	0.0133	Fvm ₅	3.93	0.0217
k ₁	15005	0.0111	Fcm ₅	8.87	0.0175
Offm ₁	0.121	0.655	k ₅	10001	0.0347
ZZ _{1R} (flexible)	0.628	0.0535	XY ₅	0.0164	0.527
Fcl ₁	0.654	0.128	XZ ₅	0.0138	0.604
Iam ₂	3.66	0.0123	YZ ₅	-0.0207	0.506
Fvm ₂	13.3	0.0102	ZZ _{5R}	0.14	0.0909
Fcm ₂	13.4	0.0122	MY _{5R}	0.042	0.145
k ₂	14989	0.0027	Fcl ₅	0.444	0.19
Offm ₂	0.551	0.19	Iam ₆	0.3	0.137
XX _{2R}	1.39	0.0356	Fvm ₆	1.78	0.0903
XY ₂	0.0701	0.352	Fcm ₆	5.13	0.0304
XZ ₂	0.0479	0.531	k ₆	10000	0.0264
ZZ _{2R} (flexible)	0.916	0.0454	XY ₆	0.0116	0.534
MX ₂	-0.0391	0.26	XZ ₆	0.00561	1.38
MY _{2R}	3.32	0.00455	YZ ₆	-0.017	0.467
Fcl ₂	0.668	0.133	ZZ _{6R}	0.0753	0.2
Iam ₃	1.5	0.0206	MX ₆	-0.0165	0.441
Fvm ₃	5.76	0.0186	MY _{6R}	0.0341	0.269
Fcm ₃	9.78	0.0168	Fcl ₆	0.256	0.342
k ₃	14992	0.00638	Iam ₇	0.301	0.084
XX _{3R}	-0.162	0.241	Fvm ₇	1.44	0.061
XY ₃	-0.0815	0.219	Fcm ₇	5.98	0.0274
XZ ₃	-0.0261	0.581	k ₇	10000	0.036
ZZ _{3R}	0.375	0.048	XX _{7R}	0.0196	0.574
MX ₃	-0.0497	0.199	ZZ ₇	0.0752	0.114
MY _{3R}	-0.0197	0.5	Fcl ₇	0.299	0.277
Fcl ₃	0.489	0.171	XX _L	0.0981	0.162
Iam ₄	1.78	0.0231	XY _L	0.0213	0.357
Fvm ₄	9.88	0.0146	XZ _L	-0.0333	0.217
Fcm ₄	9.26	0.0176	YY _L	0.117	0.145
k ₄	15001	0.00482	YZ _L	0.0306	0.232
Offm ₄	0.0335	2.49	ZZ _L	0.0406	0.235
XX _{4R}	0.413	0.0501	MX _L	0.323	0.0246
XY ₄	0.0375	0.252	MY _L	-0.309	0.0256
ZZ _{4R}	0.444	0.0426	MZ _L	0.551	0.0227
MX ₄	-0.0176	0.445	M _L	4.7	0.00579
MY _{4R}	-1.36	0.00793			

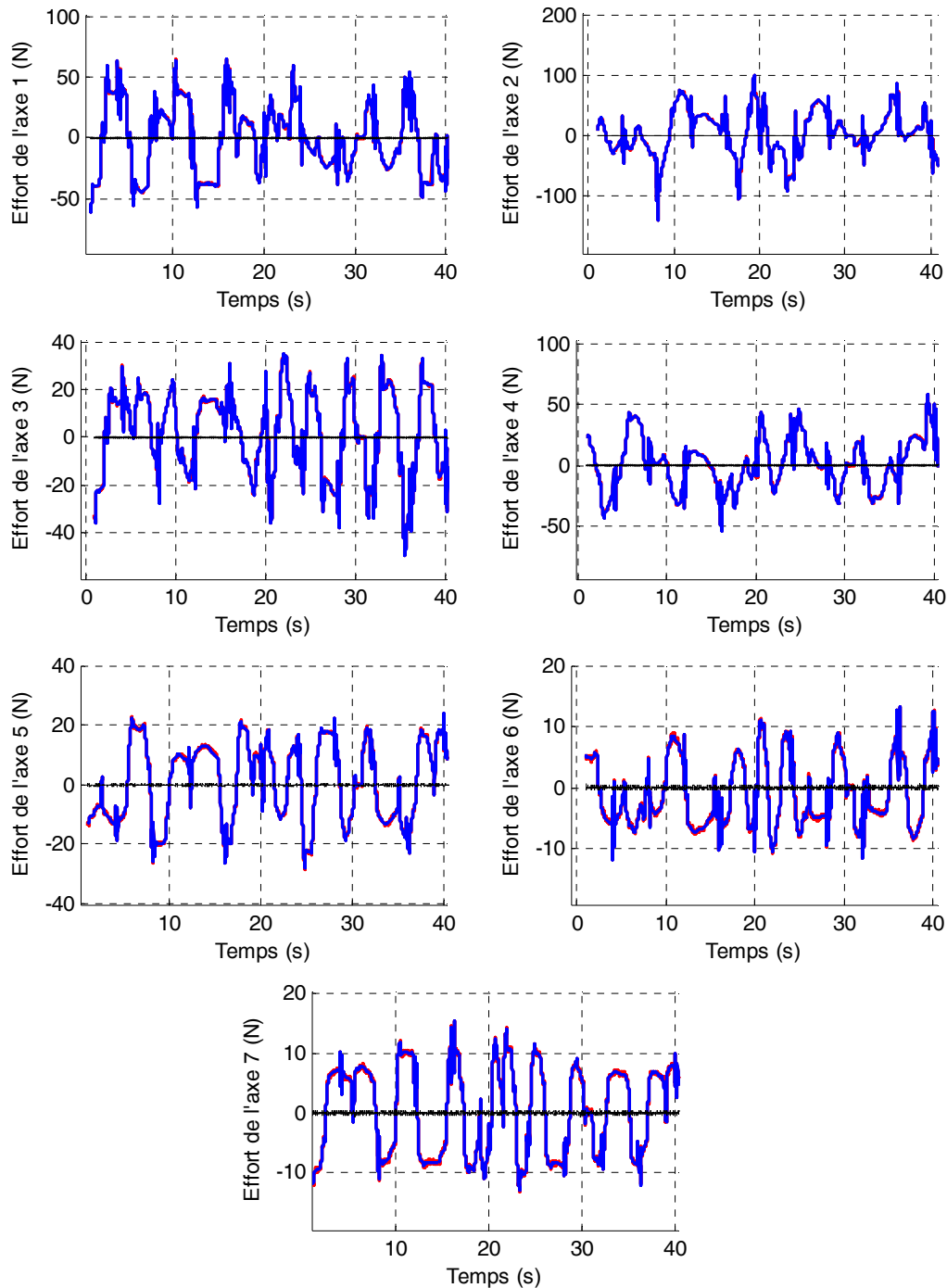


Figure 4-40 : Kuka LWR, Validation directe avec la méthode *DIDIM*, étape 3, Y (rouge), $W\chi$ (bleu) et erreur (noir)

4.6 Conclusion

Dans ce chapitre, j'ai présenté l'extension de la méthode *DIDIM* aux robots à flexibilités articulaires. Une méthode en trois étapes a été proposée pour identifier tous les paramètres dynamiques d'un robot à flexibilités articulaires en utilisant seulement les efforts moteurs. Cette procédure d'optimisation non linéaire a été testée expérimentalement sur le bras *EMPS* avec une articulation prismatique et une flexibilité articulaire et en simulation sur le Kuka *LWR* à 7

articulations rotoïdes (avec efforts moteurs sans bruit et avec bruits). Les résultats excellents valident l'algorithme proposé en terme de convergence et de robustesse vis à vis de bruit de mesure sur les couples.

La méthode *DIDIM* converge pratiquement 10 fois plus vite sur un robot un axe à une flexibilité articulaire, le bras *EMPS*, qu'une méthode *CLOE* classique à erreur de position.

Conclusion générale et perspectives

L'objectif de ce travail de thèse qui consistait à étendre la méthode *DIDIM* (Direct and Inverse Dynamic Identification Models) aux robots à flexibilités articulaires a été atteint. *DIDIM* est une nouvelle méthode d'identification dynamique qui n'utilise que les mesures des efforts moteurs.

Cette méthode est donc particulièrement intéressante à utiliser pour identifier les robots industriels à flexibilités articulaires pour lesquels les mesures des déformations élastiques ne sont pas disponibles.

Les étapes et les contributions qui ont conduit à ce succès sont résumées ci-après. Une extension de la description et de la modélisation des robots rigides au cas des robots à flexibilités articulaires avec la description *DHM* a été proposée et est applicable à tout robot, sans limitation du nombre d'articulations.

Elle permet un calcul automatique des modèles géométriques, cinématiques, dynamiques, et d'identification en utilisant le logiciel de calcul symbolique Symoro+ développé par l'équipe robotique de l'*IRCCyN*. Les robots étudiés dans cette thèse ont été modélisés : le bras *EMPS* avec une articulation prismatique; le robot rigide Stäubli *TX40* à 6 articulations rotoïdes et le robot Kuka LightWeight Robot avec 7 articulations rotoïdes flexibles.

La méthode d'identification *IDIM-LS* (Inverse Dynamic Identification Model with Least Squares) largement utilisée et validée en robotique a été présentée, comme une référence pour les autres approches utilisées par la suite. Elle utilise le modèle dynamique inverse et les moindres carrés linéaires et nécessite à la fois la mesure des efforts moteurs et des positions articulaires et l'estimation des vitesses et accélérations articulaires par filtrage passe-bande des positions.

La méthode *DIDIM* appliquée sur les robots rigides a été présentée avec une première contribution concernant une étude comparative de réglage du coefficient α dans le simulateur. Cette étude montre que cette contribution supprime le biais de la méthode avec le gain α adapté à chaque itération. Bien que ce biais soit faible, la méthode avec le coefficient $\alpha=1$ est préférable et est utilisée par la suite.

Ensuite, pour bien mettre en évidence l'intérêt théorique et applicatif de la méthode *DIDIM* comme nouvelle approche de la méthode *CLOE*, on a réalisé une comparaison de la méthode *CLOE* classique en erreur de position sur le robot rigide Stäubli *TX40* à 6 articulations rotoïdes et 61 paramètres dynamiques à identifier. Les résultats montrent nettement la supériorité de la méthode *DIDIM* sur la méthode *CLOE* classique avec une convergence plusieurs centaines de fois plus rapide et une robustesse aux erreurs sur les conditions initiales largement supérieure. On rappelle les points clés à l'origine des performances de *DIDIM* :

- minimisation d'une norme d'erreur en efforts moteurs à la place d'une norme d'erreur en positions
- utilisation d'une expression analytique des fonctions de sensibilité approchées par le modèle dynamique inverse au lieu de l'utilisation d'une intégration d'un système compliqué d'équation différentielles ou un calcul numérique par différences dans les méthodes d'optimisation non linéaires utilisées comme pour les méthodes *CLOE* usuelles.
- le troisième point clé est l'identification en boucle fermée, nécessaire aux deux points clés précédents.

La commande en boucle fermée amplifie la sensibilité de l'erreur en efforts moteurs par rapport à la variation des paramètres et conditionne la précision de l'approximation des fonctions de sensibilité par le modèle inverse.

Ensuite la principale contribution de cette thèse est la généralisation de la méthode *DIDIM* aux robots à flexibilités articulaires sous la forme d'un algorithme en trois étapes. Dans la première étape, le modèle dynamique rigide du robot est identifié dans les basses fréquences avec la méthode *DIDIM*. La deuxième étape utilise les résultats de la première étape pour identifier les valeurs approchées des raideurs ainsi que les inerties de part et d'autre des flexibilités. Cette étape utilise un algorithme de programmation non linéaire classique avec un nombre réduit de paramètres pour limiter le nombre d'itérations et accélérer la convergence. Dans la troisième étape, le modèle dynamique flexible complet est identifié avec la méthode *DIDIM* en utilisant les valeurs approchées identifiées dans les deux premières étapes pour définir des conditions initiales sur les paramètres proches de la solution optimale.

L'algorithme a été validé expérimentalement sur le bras *EMPS* avec une articulation prismatique et une flexibilité. Une comparaison de la méthode *DIDIM* avec la méthode *CLOE* à erreur de sortie en position et la méthode *CLOE* à erreur de sortie en effort moteur montre une convergence pratiquement 10 fois plus rapide et une meilleure robustesse aux conditions initiales sur les inerties et les raideurs. Cependant les frottements identifiés se regroupent du côté moteur sans pouvoir identifier leur répartition par rapport à la flexibilité.

Pour finir, l'algorithme est testé en simulation sur le robot Kuka *LWR* à 7 articulations rotoïdes flexibles, seulement en simulation, car la commande n'est pas fournie par Kuka. Afin de vérifier les valeurs des raideurs *a priori* fournies par Kuka et le laboratoire allemand *DLR*, on a proposé et validé une méthode d'identification des raideurs avec des essais axe par axe. Cette méthode utilise les valeurs des couples côté corps, mesurés par des capteurs de couples placés en sortie des réducteurs du Kuka *LWR*. Les raideurs identifiées sur les 3 premiers axes sont très proches (à 10% près) des valeurs *a priori* du laboratoire *DLR* (à 10% près).

Les paramètres utilisés pour la simulation proviennent de l'identification du modèle dynamique rigide avec la méthode *IDIM-LS*, et des raideurs *a priori* du laboratoire *DLR*. Les résultats en simulation sans bruit montrent sur ce cas d'étude la convergence des différents algorithmes de programmation non linéaire avec une très grande précision.

Les résultats en simulation avec un bruit réaliste sur les efforts moteurs valident l'algorithme proposé avec des données bruitées. En utilisant uniquement des valeurs *a priori* approximatives des raideurs et la mesure des efforts moteurs, il est possible d'identifier complètement le modèle dynamique d'un robot à flexibilités articulaires.

La loi de commande implantée dans le contrôleur du robot est connue sur les prototypes, de même que dans le cas des robots industriels dans le cadre d'une collaboration avec le fabricant pour améliorer les performances du robot. Le problème d'une loi de commande inconnue peut se poser dans le cas d'un objectif restreint à la simulation du robot.

Bilan :

On peut dire que l'utilisation du modèle inverse pour le calcul des fonctions de sensibilité a permis de lever les difficultés de l'optimisation non linéaire qui limitait l'utilisation des méthodes *CLOE* usuelles à des systèmes simples avec peu de paramètres.

Le cas des modèles non linéaires avec une centaine de paramètres à identifier illustre bien le succès de la méthode *DIDIM*.

L'utilisation de cette méthode n'est pas restreinte aux robots, elle a été appliquée avec succès à l'identification des paramètres électriques et mécaniques de la chaîne de transmission actionnée par un moteur à courant continu, une machine synchrone et une machine asynchrone (Robet et al. 2012)(Robet et Gautier, 2013a)(Robet et Gautier, 2013b).

Perspectives :

Tout d'abord l'algorithme en trois étapes de la méthode *DIDIM*, validé en simulation, doit être validé expérimentalement sur le robot Kuka *LWR* dès que sa commande sera disponible.

La somme des frottements identifiés expérimentalement pour chaque axe est correcte, cependant ils se regroupent du côté moteur. Il est nécessaire d'améliorer la méthode *DIDIM* avec de nouvelles trajectoires excitantes qui permettront d'identifier la répartition des frottements de part et d'autre de la flexibilité.

La méthode proposée pour identifier les raideurs du Kuka *LWR* grâce aux mesures des capteurs de couple côté corps peut être étendue au cas multi-axes en mouvements quelconques avec le développement d'un nouveau modèle dynamique.

Enfin, la méthode *DIDIM* appliquée aux robots à flexibilités articulaires peut être étendue aux robots modélisables par des flexibilités localisées dans les corps du robot.

Communications en revue internationale

Janot, A., Gautier, M., **Jubien, A.**, Vandanjon, P.O., 2014. Comparison between the CLOE Method and the DIDIM Method for Robots Identification. *IEEE Transaction on Control Systems Technology (TCST)*. vol 22(5). pp. 1935-1941. DOI: 10.1109/TCST.2014.2299544.

Communications avec actes en conférences internationales

Jubien, A., Gautier, M., Janot, A., 2014. Dynamic identification of the Kuka LWR robot using motor torques and joint torque sensors data. *World Congress of the International Federation of Automatic Control (IFAC), Cape Town (Afrique du sud). 24-29 août 2014*. pp. 8391-8396.

Briot, S., Gautier, M, **Jubien, A.**, 2014. In situ calibration of joint torque sensors of the KUKA LightWeight robot using only internal controller data. *IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM), Besançon (France). 8-11 juillet 2014*. pp. 470-475. DOI: 10.1109/AIM.2014.6878122.

Jubien, A., Gautier, M., Janot, A., 2014. Dynamic identification of the Kuka LightWeight robot: Comparison between actual and confidential Kuka's parameters. *IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM), Besançon (France). 8-11 juillet 2014*. pp. 483-488. DOI: 10.1109/AIM.2014.6878124.

Jubien, A., Gautier, M., 2013. Global identification of spring balancer, dynamic parameters and drive gains of heavy industrial robots. *IEEE/RJS International Conference on Intelligent Robots and Systems (IROS), Tokyo (Japon), 3-7 novembre 2013*. pp. 1355-1360. DOI : 10.1109/IROS.2013.6696525.

Gautier, M., **Jubien, A.**, Janot, A., 2013. Iterative Learning Identification and Computed Torque Control of Robots. *IEEE/RJS International Conference on Intelligent Robots and Systems (IROS), Tokyo (Japon), 3-7 novembre 2013*. pp. 3419-3424. DOI: 10.1109/IROS.2013.6696843.

Robet, P.P., Gautier, M., **Jubien, A.**, Janot, A., 2013. Global identification of mechanical and electrical parameters of DC motor driven joint with a fast CLOE method. *IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM), Wollongong (Australie). 9-12 juillet 2013*. pp. 1205-1210. DOI: 10.1109/AIM.2013.6584258.

Jubien, A., Gautier, M., Janot, A., 2013. Effectiveness of the DIDIM method with respect to the usual CLOE method. Application to the dynamic parameters identification of an industrial robot. *9^{ème} Asian Control Conference (ASCC), Istanbul (Turquie). 23-26 juin 2013*. pp 1-6. DOI : 10.1109/ASCC.2013.6606308.

Gautier, M., **Jubien, A.**, Janot, A., 2013. A new iterative online dynamic identification method of robots from only force/torque data. *9^{ème} Asian Control Conference (ASCC), Istanbul (Turquie). 23-26 juin 2013*. pp.1-6. DOI : 10.1109/ASCC.2013.6606307.

- Gautier, M., **Jubien, A.**, Janot, A., Robet, P.P., 2013. Dynamic Identification of flexible joint manipulators with an efficient closed loop output error method based on motor torque output data. *IEEE International Conference on Robotics and Automation (ICRA), Karlsruhe (Allemagne), 6-10 mai 2013.* pp. 2949-2955. DOI : 10.1109/ICRA.2013.6630986.
- Robet, P.P., Gautier, M., **Jubien, A.**, Janot, A., 2012. A new output error method for a decoupled identification of electrical and mechanical dynamic parameters of DC motor-driven robots. *10 ème IFAC Symposium on Robot Control (SYROCO), Dubrovnik (Croatie), 5-7 septembre 2012.* vol 10(1). pp.25-30. DOI: 10.3182/20120905-3-HR-2030.00147.
- Jubien, A.**, Gautier, M., Janot, A., Vandanjon, P.O., 2012. Méthode à erreur de sortie pour l'identification en boucle fermée des paramètres dynamiques d'un robot à flexibilité localisée, sans mesure de flexibilité. *7ème Conférence Internationale Francophone d'Automatique (CIFA), Grenoble (France), 4-6 juillet 2012.* pp. 683-688.
- Gautier, M., Janot, A., **Jubien, A.**, Vandanjon, P.O., 2011. Joint stiffness identification from only motor force/torque data. *50ème IEEE Conference on Decision and Control and European Control Conference (CDC), Orlando (Floride, Etats-Unis), 12-15 décembre 2011.* pp. 5088–5093. DOI : 10.1109/CDC.2011.6160589.
- Janot, A., Gautier, M., **Jubien, A.**, Vandanjon, P.O., 2011. Experimental joint stiffness identification depending on measurements availability. *50ème IEEE Conference on Decision and Control and European Control Conference (CDC), Orlando (Floride, Etats-Unis), 12-15 décembre 2011.* pp. 5112–5117. DOI: 10.1109/CDC.2011.6160402.

Toutes les fiches détaillées de ces publications sont disponibles dans la base HAL des archives ouvertes (<http://hal.archives-ouvertes.fr>), en recherchant les notices bibliographiques sous le nom de Jubien Anthony. Les publications sont aussi référencées et téléchargeables sous IEEEXplore (<http://ieeexplore.ieee.org>) ou sous IFAC-PapersOnLine (<http://www.ifac-papersonline.net/>).

Bibliographie

- Albu-Schäffer, A., Haddadin, S., Ott, C., Stemmer, A., Wimböck, T., Hirzinger, G., 2007. The DLR lightweight robot: design and control concepts for robots in human environments. *Industrial Robot: An International Journal*, vol. 34(5), pp. 376-385.
- Albu-Schäffer, A., Hirzinger, G., 2001. Parameter identification and passivity based joint control for a 7 DOF torque controlled light weight robot. *IEEE International Conference on Robotics and Automation*, Séoul, Corée du Sud, vol 3, pp. 2852-2858.
- Alici, G., Shirinzadeh, B., 2005. Enhanced Stiffness Modeling, Identification and Characterization for Robot Manipulators. *IEEE Transactions on Robotics*, vol 21(4), pp. 554-564.
- Bischoff, R., Kurth, J., Schreiber, G., Koeppe, R., Albu-Schäffer, A., Beyer, A., Eiberger, O., Haddadin, S., Stemmer, A., Grunwald, G., Hirzinger, G., 2010. The KUKA-DLR Lightweight Robot arm - a new reference platform for robotics research and manufacturing. *International Symposium on robotics and German Conference on Robotics*, Munich, Allemagne, pp. 1-8.
- Bogdan, I.-C., Abba, G., 2009. Identification of the servomechanism used for micro-displacement. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Saint-Louis, MO, USA, pp. 1986-1991.
- Bogdan, I.-C., Abba, G., 2012. Identification of mechanical parameters at low velocities for a micropositioning stage using a velocity hysteresis model. *IEEE International Conference on Robotics and Automation*, Saint-Paul, MN, USA, pp. 430-435.
- Carrillo, F., Baysse, A., Habbadi, A., 2009. Output Error Identification Algorithms for Continuous-Time Systems Operating in Closed-Loop. *IFAC Symposium on System Identification*, Saint-Malo, France, vol 15(1), pp. 408-413.
- Davidson, R., MacKinnon, J.G., 1993. Estimation and Inference in Econometrics. *Oxford University Press, New York*.
- DLR, 2014. Laboratoire Deutsches Zentrum für Luft- und Raumfahrt. <http://www.dlr.de/dlr/en/>.
- dSPACE, 2014. ControlDesk. www.dspace.com.
- Dumas, C., Caro, S., Garnier, S., Furet, B., 2011. Joint Stiffness Identification of Six-revolute Industrial Serial Robots. *Journal of Robotics and Computer Integrated Manufacturing*, vol 27(4), pp 881-888.
- Durbin, J., Watson, G.S., 1950. Testing for Serial Correlation in Least Squares Regression. *Biometrika*, vol 37, pp. 409-428.
- Farebrother, R.W., 1980. Pan's Procedure for the Tail Probabilities of the Durbin-Watson Statistic. *Applied Statistics*, vol 29, pp. 224-227.

- Flacco, F., De Luca, A., 2011. Residual-based stiffness estimation in robot with flexible transmissions. *IEEE International Conference on Robotics and Automation*, Shanghai, Chine, pp. 5541-5547.
- Garnier, H., Gilson, M., Bastogne, T., Mensler, M., 2008. The CONTSID Toolbox: A Software Support for Data-based Continuous-time Modelling, in: Identification of Continuous-time Models from Sampled Data. *Springer London*, pp. 249-290.
- Gautier, M., 1986. Identification of robots dynamics. *IFAC/IFIP/IMACS Symposium on Theory of Robots*, pp. 125-130.
- Gautier, M., 1990. Contribution à la modélisation et à l'identification des robots (Thèse de Doctorat). *Université de Nantes*.
- Gautier, M., 1991. Numerical calculation of the base inertial parameters of robots. *Journal of Robotic Systems*, vol 8, pp. 485-506.
- Gautier, M., 1997. Dynamic identification of robots with power model. *IEEE International Conference on Robotics and Automation*, Albuquerque, NM, USA, vol 3, pp. 1922-1927.
- Gautier, M., Briot, S., 2012. Global identification of drive gains parameters of robots using a known payload. *IEEE International Conference on Robotics and Automation*, Saint-Paul, MN, USA, pp. 2812-2817.
- Gautier, M., Briot, S., 2014. Global Identification of Joint Drive Gains and Dynamic Parameters of Robots. *ASME Journal of Dynamic Systems, Measurement and Control*, vol 136(5).
- Gautier, M., Janin, C., Presse, C., 1993. Dynamic Identification of Robots using Least Squares and Extended Kalman Filtering Methods. *European Control Conference*, Groningen, Pays-Bas, vol 4, pp. 2291-2297.
- Gautier, M., Janot, A., Vandanjon, P.O., 2008. DIDIM: A new method for the dynamic identification of robots from only torque data. *IEEE International Conference on Robotics and Automation*, Pasadena, CA, USA, pp. 2122-2127.
- Gautier, M., Janot, A., Vandanjon, P.O., 2012. A New Closed-Loop Output Error Method for Parameter Identification of Robot Dynamics. *IEEE Transactions on Control Systems Technology*, vol 21, pp. 428-444.
- Gautier, M., Jubien, A., Janot, A., Robet, P.P., 2013. Dynamic Identification of flexible joint manipulators with an efficient closed loop output error method based on motor torque output data. *IEEE International Conference on Robotics and Automation*, Karlsruhe, Allemagne, pp. 2949-2955.
- Gautier, M., Khalil, W., 1990. Direct calculation of minimum set of inertial parameters of serial robots. *IEEE Transactions on Robotics and Automation*, pp. 368-373.

- Gautier, M., Khalil, W., 1991. Exciting trajectories for the identification of base inertial parameters of robots. *IEEE Conference on Decision and Control*, Brighton, Angleterre, vol 1, pp. 494-499.
- Gautier, M., Khalil, W., 2001. Identification des paramètres des modèles. *Traité IC2, Analyse et modélisation des robots manipulateurs*, chapitre 4. *Hermes Science Publications*.
- Gautier, M., Khalil, W., Restrepo, P.P., 1995. Identification of the dynamic parameters of closed loop robots. *IEEE International Conference on Robotics and Automation*, Nagoya, Aichi, Japon, pp. 3045-3050.
- Gautier, M., Poignet, P., 2001. Extended Kalman filtering and weighted least squares dynamic identification of robot. *Control Engineering Practice*, vol 9(12), pp. 1361-1372.
- Gautier, M., Poignet, P., 2002. Identification en boucle fermée par modèle inverse des paramètres physiques de systèmes mécatroniques. *Journal Européen des Systèmes Automatisés*, vol 36(3), pp. 465-480.
- Gautier, M., Vandanjon, P.-O., Janot, A., 2011. Dynamic identification of a 6 dof robot without joint position data. *IEEE International Conference on Robotics and Automation*, Shanghai, Chine, pp. 234-239.
- Guglielmi, M., Jonker, E., Piasco, J.M., 1987. Modelling and identification of a two degrees of freedom SCARA robot using extended Kalman filtering. *International Conference on Advanced Robotics*, Versailles, France, pp. 137-148.
- Hamon, P., 2011. Modélisation et identification dynamiques des robots avec un modèle de frottement sec fonction de la charge et de la vitesse (Thèse de Doctorat). *Université de Nantes*.
- Hollerbach, J., Khalil, W., Gautier, M., 2008. Model Identification. *Springer Handbook of Robotics*, Springer Berlin, pp. 321-344.
- Janot, A., 2007. Contribution à la modélisation et à l'identification des interfaces haptiques. (Thèse de Doctorat). *Université de Nantes*.
- Janot, A., Gautier, M., Jubien, A., Vandanjon, P.O., 2011. Experimental joint stiffness identification depending on measurements availability. *IEEE Conference on Decision and Control and European Control Conference*, Orlando, FL, USA, pp. 5112-5117.
- Janot, A., Gautier, M., Jubien, A., Vandanjon, P.O., 2014b. Comparison between the CLOE Method and the DIDIM Method for Robots Identification. *IEEE Transaction on Control Systems Technology*, vol 22(5). pp. 1935-1941
- Janot, A., Vandanjon, P.O., Gautier, M., 2014a. A Generic Instrumental Variable Approach for Industrial Robot Identification. *IEEE Transactions on Control Systems Technology*, vol 22(1), pp. 132-145.

- Jubien, A., Gautier, M., Janot, A., 2013. Effectiveness of the DIDIM method with respect to the usual CLOE method: Application to the dynamic parameters identification of an industrial robot. *Asian Control Conference*, Istanbul, Turquie, pp 1-6.
- Jubien, A., Gautier, M., Janot, A., 2014a. Dynamic identification of the Kuka LightWeight robot: Comparison between actual and confidential Kuka's parameters. *IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, Besançon, France, pp. 483-488.
- Jubien, A., Gautier, M., Janot, A., 2014b. Dynamic identification of the Kuka LWR robot using motor torques and joint torque sensors data. *World Congress of the International Federation of Automatic Control*, Cape Town, Afrique du Sud, pp. 8391-8396.
- Khalil, W., Creusot, D., 1997. SYMORO+ : A system for the symbolic modelling of robots. *Robotica*, vol 15(2), pp. 153-161.
- Khalil, W., Dombre, E., 1999. Modélisation, identification et commande des robots. *Hermes Science Publications*.
- Khalil, W., Dombre, E., 2002. Modeling, Identification and Control of Robots, 3rd edition. *Taylor and Francis Group*, New York.
- Khalil, W., Gautier, M., 2000. Modeling of mechanical systems with lumped elasticity. *IEEE International Conference on Robotics and Automation*, San Francisco, CA, USA, vol 4, pp. 3964-3969.
- Khalil, W., Gautier, M., Lemoine, P., 2007. Identification of the payload inertial parameters of industrial manipulators. *IEEE International Conference on Robotics and Automation*, Rome, Italie, pp. 4943-4948.
- Khalil, W., Kleininger, J., 1986. A new geometric notation for open and closed-loop robots. *IEEE International Conference on Robotics and Automation*, San Francisco, CA, USA, pp. 1174-1179.
- Landau, I.D., Anderson, B.D.O., De Bruyne, F., 1999. Closed-loop output error identification algorithms for nonlinear plants. *IEEE Conference on Decision and Control*, Phoenix, AZ, USA, pp. 606-611.
- Lagarias, J.C., Reeds, J.A., Wright, M.H., Wright, P.E., 1998. Convergence Properties of the Nelder – Mead Simplex Method in Low Dimensions. *Society for Industrial and Applied Mathematics*, vol 9, pp. 112-147.
- Lawson, C.L., Hanson, R.J., 1974. Solving Least Squares Problems. *Society for Industrial and Applied Mathematics*.
- Lightcap, C., Banks, S., 2007. Dynamic identification of a mitsubishi pa10-6ce robot using motion capture. *IEEE. IEEE/RSJ International Conference on Intelligent Robots and Systems*, San Diego, CA, USA, pp. 3860-3865.

- Ljung, L., 2000. System identification toolbox—user’s guide. *The MathWorks Inc.*
- Makarov, M., Grossard, M., Rodriguez-Ayerbe, P., Dumur, D., 2012. A Frequency-Domain Approach for Flexible-Joint Robot Modeling and Identification. *IFAC Symposium on System Identification*, vol 16(1), Bruxelles, Belgique, pp. 583-588.
- Marsaglia, G., Tsang, W., Wang, J., 2003. Evaluating Kolmogorov’s Distribution. *Journal of Statistical Software*, vol 8(18), pp. 1-4.
- Massey, F.G., 1951. The Kolmogorov-Smirnov Test for Goodness of Fit. *Journal of the American Statistical Association*, vol 46, pp. 68-78.
- Mathworks, 2014. Matlab. *www.mathworks.fr.*
- Mayeda, H., Yoshida, K., Osuka, K., 1990. Base parameters of manipulator dynamic models. *IEEE Transactions on Robotics and Automation*, vol 6(3), pp. 312-321.
- Miller, L.H., 1956. Table of Percentage Points of Kolmogorov Statistics. *Journal of the American Statistical Association*, vol 51(273), pp. 111-121.
- Minoux, M., 1983. Programmation Mathématique. *Théorie et algorithmes, Dunod.*
- Olsen, M.M., Petersen, H.G., 2001. A new method for estimating parameters of a dynamic robot model. *IEEE Transactions on Robotics and Automation*. vol 17, pp. 95-100.
- Olsen, M.M., Swevers, J., Verdonck, W., 2002. Maximum Likelihood Identification of a Dynamic Robot Model: Implementation Issues. *The International Journal of Robotics Research*, vol 21(2), pp. 89-96.
- Östring, M., 2003. Closed-loop identification of an industrial robot containing flexibilities. *Control Engineering Practice*, vol 11(3), pp. 291-300.
- Pfeiffer, F., Holzl, J., 1995. Parameter identification for industrial robots. *IEEE International Conference on Robotics and Automation*, Nagoya, Aichi, Japon, vol 2, pp. 1468-1476.
- Pham, M.T., 2002. Contribution à la modélisation, l’identification et la commande de systèmes mécaniques à flexibilités localisées. Application à des axes de machines-outils rapides. (Thèse de Doctorat). *Université de Nantes.*
- Pham, M.T., Gautier, M., Poignet, P., 2001. Identification of joint stiffness with bandpass filtering. *IEEE International Conference on Robotics and Automation*, Séoul, Corée du sud, vol 3, pp. 2867-2872.
- Pham, M.T., Gautier, M., Poignet, P., 2002. Accelerometer based identification of mechanical systems. *IEEE International Conference on Robotics and Automation*, Washington, DC, USA, vol 4, pp. 4293-4298.

- Presse, C., Gautier, M., 1993. New criteria of exciting trajectories for robot identification. *IEEE International Conference on Robotics and Automation*, Atlanta, GE, USA, vol 3, pp. 907-912.
- Rackl, W., Lampariello, R., Hirzinger, G., 2012. Robot excitation trajectories for dynamic parameter estimation using optimized B-splines. *IEEE International Conference on Robotics and Automation*, Saint-Paul, MN, USA, pp. 2042-2047.
- Restrepo, P.P., Gautier, M., 1995. Calibration of drive chain of robot joints. *IEEE Conference on Control Applications*, Albany, NY, USA, pp. 526-531.
- Richalet, J., Rault, A., Pouliquen, R., 1971. Identification des processus par la méthode du modèle. *Gordon and Breach*.
- Robet, P.Ph, Gautier, M., 2013a. Decoupled identification of electrical and mechanical parameters of synchronous motor-driven chain with an efficient CLOE method. *IEEE Conference on Industrial Electronics and Applications*, Melbourne, VIC, Australie, pp. 1780-1785.
- Robet, P.Ph, Gautier, M., 2013b. Global identification of mechanical and electrical parameters of synchronous motor driven joint with a fast CLOE method. *European Control Conference*, Zürich, Suisse, pp. 4604-4609.
- Robet, P.Ph, Gautier, M., 2013c. Global identification of mechanical and electrical parameters of asynchronous motor driven joint with a fast CLOE method. *European Conference on Power Electronics and Applications*, Lille, France, pp. 1-10.
- Robet, P.Ph, Gautier, M., Jubien, A., Janot, A., 2012. A new output error method for a decoupled identification of electrical and mechanical dynamic parameters of DC motor-driven robots. *IFAC Symposium on Robot Control*, Dubrovnik, Croatie, pp. 25-30.
- Robet, P.Ph, Gautier, M., Jubien, A., Janot, A., 2013. Global identification of mechanical and electrical parameters of DC motor driven joint with a fast CLOE method. *IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, Wollongong, NSW, Australie, pp. 1205-1210.
- Spong, M.W., 1987. Modeling and control of elastic joint robots. *Journal of Dynamic Systems Measurement and Control*, vol 109(4), pp. 310-318.
- Swevers, J., Gansema, C., Tukul, D.B., De Schutter, J., Van Brussel, H., 1997. Optimal Robot Excitation and Identification. *IEEE Transactions on Robotics and Automation*, vol 13(5), pp. 730-740.
- Universal Robot, 2014. Universal Robots UR5 et UR10. <http://www.universal-robots.fr/FR/Produits.aspx>.
- Walter, E., Pronzato, L., 1997. Identification of parametric models from experimental data. *Springer London*.

A. Energie linéaire par rapport aux paramètres inertiels

On représente le corps C_j (représenté figure A-1) avec :

- L_j le vecteur (3x1) des coordonnées de $O_{j-1}O_j$ dans R_j ,
- S_j le vecteur (3x1) des coordonnées de O_jG_j dans R_j ,
- G_j le centre de masse du corps C_j dans R_j .

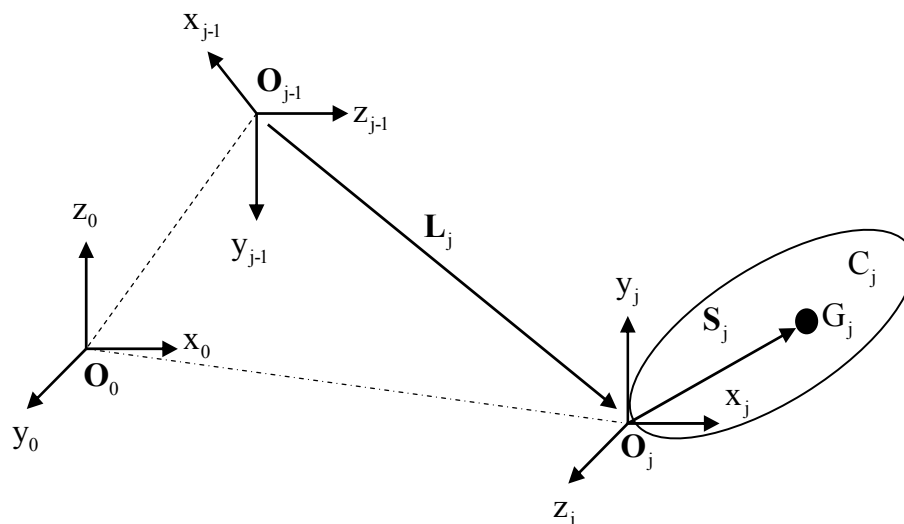


Figure A-1 : Représentation d'un corps C_j et composition des vitesses

L'énergie cinétique totale E du système est donnée par la relation :

$$E = \sum_{j=1}^n E^j \quad (\text{A.1})$$

Avec E^j l'énergie cinétique du corps C_j qui s'exprime par :

$$E^j = \frac{1}{2} [\boldsymbol{\omega}_j^T \mathbf{I}_{G_j} \boldsymbol{\omega}_j + M_j \mathbf{V}_{G_j}^T \mathbf{V}_{G_j}] \quad (\text{A.2})$$

où :

- \mathbf{I}_{G_j} est le tenseur d'inertie du corps C_j par rapport à un repère de même orientation que R_j et d'origine G_j ,
- $\boldsymbol{\omega}_j$ est la vitesse angulaire du corps C_j ,
- M_j est la masse du corps C_j ,
- \mathbf{V}_{G_j} est la vitesse du centre de gravité G_j .

En appliquant la loi de composition des vitesses, on obtient :

$$\mathbf{V}_{Gj} = \mathbf{V}_j + \boldsymbol{\omega}_j \wedge \mathbf{S}_j \quad (\text{A.3})$$

où :

- \mathbf{V}_j est la vitesse du point O_j ,
- \wedge est l'opérateur du produit vectoriel.

En appliquant le théorème de Huygens, on obtient :

$$\mathbf{J}_j = \mathbf{I}_{Gj} - M_j \hat{\mathbf{S}}_j \hat{\mathbf{S}}_j \quad (\text{A.4})$$

où :

- $\hat{\mathbf{S}}_j$ est la matrice antisymétrique du préproduit vectoriel,
- \mathbf{J}_j est le tenseur d'inertie du corps C_j par rapport au repère R_j :

$$\mathbf{J}_j = \begin{bmatrix} \text{XX}_j & \text{XY}_j & \text{XZ}_j \\ \text{XY}_j & \text{YY}_j & \text{YZ}_j \\ \text{XZ}_j & \text{XY} & \text{ZZ}_j \end{bmatrix} = \begin{bmatrix} \int (y^2 + z^2) dm & -\int xy dm & -\int xz dm \\ -\int xy dm & \int (x^2 + z^2) dm & -\int yz dm \\ -\int xz dm & -\int yz dm & \int (x^2 + y^2) dm \end{bmatrix} \quad (\text{A.5})$$

On pose \mathbf{MS}_j le premier moment d'inertie du corps C_j autour de l'origine O_j du repère R_j et égal à $M_j \mathbf{S}_j$ tel que :

$$\mathbf{MS}_j = [\text{MX}_j \quad \text{MY}_j \quad \text{MZ}_j]^T \quad (\text{A.6})$$

On obtient l'expression suivante pour l'énergie cinétique :

$$E^j = \frac{1}{2} \left[\boldsymbol{\omega}_j^T \mathbf{J}_j \boldsymbol{\omega}_j + M_j \mathbf{V}_j^T \mathbf{V}_j + 2 \mathbf{MS}_j^T (\mathbf{V}_j \wedge \boldsymbol{\omega}_j) \right] \quad (\text{A.7})$$

Le calcul des variables eulériennes $\boldsymbol{\omega}_j$ et \mathbf{V}_j en fonction des positions, vitesses et accélération généralisées est rappelé en annexe A.

A partir de l'énergie cinétique totale E du système, on en déduit la matrice d'inertie $\mathbf{M}(\mathbf{q})$ et la matrice $(n \times n)$ $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ du vecteur des efforts centrifuges et de Coriolis $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{q}}$.

Le terme d'indice ij avec $i \neq j$ de la matrice d'inertie $\mathbf{M}(\mathbf{q})$ est égal au coefficient de $\dot{q}_i \dot{q}_j$ de l'énergie cinétique totale et le terme d'indice jj est égal au coefficient $\dot{q}_j^2 / 2$ de l'énergie cinétique totale.

Le coefficient d'indice ij C_{ij} de la matrice $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ s'exprime suivant les coefficients de la matrice d'inertie :

$$C_{ij} = \sum_{k=1}^n c(i, j)_k \dot{q}_k \text{ avec} \quad (A.8)$$

$$c(i, j)_k = \frac{1}{2} \left(\frac{\partial M_{ij}(\mathbf{q})}{\partial q_k} + \frac{\partial M_{ik}(\mathbf{q})}{\partial q_j} - \frac{\partial M_{jk}(\mathbf{q})}{\partial q_i} \right)$$

L'énergie potentielle totale U du système s'écrit :

$$U = \sum_{j=1}^n U^j \quad (A.9)$$

Avec U^j l'énergie potentielle du corps C_j qui s'exprime pour un robot à articulations rigides par :

$$U^j = -M_j \mathbf{g}^T ({}^0\mathbf{L}_j + \mathbf{S}_j) \quad (A.10)$$

où :

- \mathbf{g}^T le vecteur de l'accélération de la pesanteur exprimé dans le repère R_0 ,
- ${}^0\mathbf{L}_j$ le vecteur (3x1) des coordonnées de $\mathbf{O}_0\mathbf{O}_j$ dans R_0 .

En projetant les vecteurs ${}^0\mathbf{L}_j$ et \mathbf{S}_j dans le repère R_0 , on obtient :

$$U^j = -\mathbf{g}^T (M_j \mathbf{P}_j + {}^0\mathbf{A}_j \mathbf{M} \mathbf{S}_j) \quad (A.11)$$

où :

- ${}^0\mathbf{A}_j$ est la matrice orthogonale (3x3) de rotation (1.3) définissant l'orientation du repère R_j dans le repère R_0 ,
- \mathbf{P}_j est la matrice colonne (3x1) des coordonnées du vecteur $\overline{\mathbf{O}_0\mathbf{O}_j}$ exprimé dans le repère R_0 .

Cette relation est linéaire par rapport aux paramètres M_j et $\mathbf{M} \mathbf{S}_j$.

${}^0\mathbf{P}_j$ et ${}^0\mathbf{A}_j$ dépendent uniquement des positions \mathbf{q} donc U est fonction de \mathbf{q} et on en déduit le vecteur des efforts de gravité $\mathbf{Q}(\mathbf{q})$:

$$\mathbf{Q}(\mathbf{q}) = [\mathbf{Q}_1(\mathbf{q}) \quad \mathbf{Q}_2(\mathbf{q}) \quad \cdots \quad \mathbf{Q}_n(\mathbf{q})]^T$$

avec :

$$\mathbf{Q}_j(\mathbf{q}) = \frac{\partial U}{\partial \mathbf{q}_j} \quad (\text{A.12})$$

On définit le vecteur (10x1) des paramètres inertiels standards du corps C_j :

$$\boldsymbol{\chi}_i^j = \begin{bmatrix} \text{XX}_j & \text{XY}_j & \text{XZ}_j & \text{YY}_j & \text{YZ}_j \\ \text{ZZ}_j & \text{MX}_j & \text{MY}_j & \text{MZ}_j & \text{M}_j \end{bmatrix}^T \quad (\text{A.13})$$

Où :

- $\text{XX}_j, \text{XY}_j, \text{XZ}_j, \text{YY}_j, \text{YZ}_j$ et ZZ_j sont les 6 composantes du tenseur d'inertie du corps C_j par rapport au repère R_j noté \mathbf{J}_j ,
- MX_j, MY_j et MZ_j sont les 3 composantes du premier moment d'inertie produit de la masse par les coordonnées du centre de gravité du corps C_j dans le repère R_j ,
- M_j est la masse du corps C_j .

Les relations (A.7) et (A.11) montrent que l'énergie totale H^j du corps C_j est linéaire par rapport aux paramètres inertiels (Gautier, 1990)(Gautier et Khalil, 1990):

$$H^j = E^j + U^j = \mathbf{h}^j \boldsymbol{\chi}_i^j \quad (\text{A.14})$$

où \mathbf{h}^j est le vecteur (10x1) définit par l'équation suivante :

$$\mathbf{h}^j = \frac{\partial H^j}{\partial \boldsymbol{\chi}_i^j} = \begin{bmatrix} h_{\text{XX}_j} & h_{\text{XY}_j} & h_{\text{XZ}_j} & h_{\text{YY}_j} & h_{\text{YZ}_j} \\ h_{\text{ZZ}_j} & h_{\text{MX}_j} & h_{\text{MY}_j} & h_{\text{MZ}_j} & h_{\text{M}_j} \end{bmatrix} \quad (\text{A.15})$$

Chaque fonction d'énergie \mathbf{h}^j est calculée à partir de (A.7) et (A.11) :

- $h_{\text{XX}_j} = \partial H^j / \partial \text{XX}_j = \frac{1}{2} \omega_{1,j}^2$,
- $h_{\text{XY}_j} = \partial H^j / \partial \text{XY}_j = \omega_{1,j} \omega_{2,j}$,
- $h_{\text{XZ}_j} = \partial H^j / \partial \text{XZ}_j = \omega_{1,j} \omega_{3,j}$,
- $h_{\text{YY}_j} = \partial H^j / \partial \text{YY}_j = \frac{1}{2} \omega_{2,j}^2$,

- $h_{YZj} = \partial H^j / \partial YZ_j = \omega_{2,j} \omega_{3,j}$,
- $h_{ZZj} = \partial H^j / \partial ZZ_j = \frac{1}{2} \omega_{3,j}^2$,
- $h_{MXj} = \partial H^j / \partial MX_j = \omega_{3,j} V_{2,j} - \omega_{2,j} V_{3,j} - {}^0 \mathbf{g}^T {}^0 \mathbf{s}_j$,
- $h_{MYj} = \partial H^j / \partial MY_j = \omega_{1,j} V_{3,j} - \omega_{3,j} V_{1,j} - {}^0 \mathbf{g}^T {}^0 \mathbf{n}_j$,
- $h_{MZj} = \partial H^j / \partial MZ_j = \omega_{2,j} V_{1,j} - \omega_{1,j} V_{2,j} - {}^0 \mathbf{g}^T {}^0 \mathbf{a}_j$,
- $h_{Mj} = \partial H^j / \partial M_j = \frac{1}{2} \mathbf{V}_j^T \mathbf{V}_j - {}^0 \mathbf{g}^T {}^0 \mathbf{P}_j$.

où :

- $\omega_{i,j}$ est la composante i du vecteur $\boldsymbol{\omega}_j$,
- $V_{i,j}$ est la composante i du vecteur \mathbf{V}_j .

Le robot comporte (10n) paramètres inertiels standards au total :

$$\boldsymbol{\chi}_l = [\boldsymbol{\chi}_l^{1T} \quad \boldsymbol{\chi}_l^{2T} \quad \cdots \quad \boldsymbol{\chi}_l^{nT}]^T \quad (\text{A.16})$$

L'énergie totale H du robot s'exprime de la manière suivante :

$$H = E + U = \sum_{j=1}^n H^j = \sum_{i=1}^{10n} h_i \chi_{li} \quad \text{avec } h_i = \frac{\partial H}{\partial \chi_{li}} \quad (\text{A.17})$$

où χ_{li} est un coefficient de $\boldsymbol{\chi}_l$ et h_i sa fonction d'énergie correspondante.

Pour les articulations motorisées, avec des rapports de réduction importants (>50), l'énergie cinétique de la chaîne d'actionnement est principalement due à la vitesse relative \dot{q}_j :

$$Ea_j = \frac{1}{2} I_{a_j} \dot{q}_j \quad (\text{A.18})$$

Où I_{a_j} est le moment d'inertie de la chaîne d'actionnement j exprimée en unité S.I côté corps.

L'effort de frottement est approché par l'équation (1.12), linéaire par rapport aux coefficients de frottements F_{vj} et F_{cj} et par rapport à l'offset Off_j .

L'équation de Lagrange (1.7) du modèle dynamique inverse s'exprime linéairement par rapport au vecteur de paramètres standards (1.14) selon (1.15).

B. Formalisme de Newton-Euler pour le calcul du modèle dynamique

Une modification de l'algorithme de Newton-Euler, proposée par (Khalil et Dombre, 2002), permet d'exprimer le modèle inverse linéairement par rapport aux paramètres inertiels M_j , MS_j et J_j .

Les équations de Newton-Euler modifiées s'écrivent :

$$\begin{aligned} \mathbf{F}_j &= M_j \dot{\mathbf{V}}_j + \dot{\boldsymbol{\omega}}_j \otimes \mathbf{MS}_j + \boldsymbol{\omega}_j \otimes (\boldsymbol{\omega}_j \otimes \mathbf{MS}_j) \\ \mathbf{M}_j &= \mathbf{J}_j \dot{\boldsymbol{\omega}}_j + \mathbf{MS}_j \otimes \dot{\mathbf{V}}_j + \boldsymbol{\omega}_j \otimes (\mathbf{J}_j \boldsymbol{\omega}_j) \end{aligned} \quad (\text{B.1})$$

Où :

- \mathbf{F}_j est la résultante des forces extérieures appliquées sur le corps C_j ,
- \mathbf{M}_j est le moment résultant des forces extérieures exercées sur le corps C_j autour de \mathbf{O}_j ,
- M_j est la masse du corps C_j ,
- \mathbf{MS}_j est le premier moment d'inertie du corps C_j autour de l'origine \mathbf{O}_j du repère R_j et égal à $M_j \mathbf{S}_j$,
- $\boldsymbol{\omega}_j$ est la vitesse angulaire du corps C_j exprimée dans R_j ,
- \mathbf{V}_j est le vecteur (3x1) vitesse du point \mathbf{O}_j dans R_j ,
- \mathbf{J}_j est le tenseur d'inertie du corps C_j par rapport au repère R_j ,
- \otimes est l'opérateur du produit vectoriel.

Les équations de Newton-Euler sont obtenues par double récurrence :

- la récurrence avant part de la base du robot vers son organe terminal. Elle calcule les vitesses et accélérations des corps,
- la récurrence arrière part de l'organe terminal vers la base. Elle calcule les torseurs des efforts sur chaque articulation.

Récurrence avant : calcul des vitesses et accélérations de la base du robot vers l'organe terminal

Elle permet de calculer $\boldsymbol{\omega}_j$, $\dot{\boldsymbol{\omega}}_j$ et $\dot{\mathbf{V}}_j$. Les formules de compositions des vitesses donnent

:

$$\begin{aligned} \boldsymbol{\omega}_j &= \boldsymbol{\omega}_{j-1} + \bar{\sigma}_j \dot{q}_j \mathbf{a}_j \\ \mathbf{V}_j &= \mathbf{V}_{j-1} + \boldsymbol{\omega}_{j-1} \otimes \mathbf{L}_j + \sigma_j \dot{q}_j \mathbf{a}_j \end{aligned} \quad (\text{B.2})$$

En dérivant les équations précédentes, on obtient :

$$\begin{aligned}
\dot{\boldsymbol{\omega}}_j &= \dot{\boldsymbol{\omega}}_{j-1} + \bar{\sigma}_j (\ddot{q}_j \mathbf{a}_j + \boldsymbol{\omega}_{j-1} \otimes \dot{q}_j \mathbf{a}_j) \\
\dot{\mathbf{V}}_j &= \dot{\mathbf{V}}_{j-1} + \dot{\boldsymbol{\omega}}_{j-1} \otimes \mathbf{L}_j + \boldsymbol{\omega}_{j-1} \otimes (\boldsymbol{\omega}_{j-1} \otimes \mathbf{L}_j) + \sigma_j (\ddot{q}_j \mathbf{a}_j + 2\boldsymbol{\omega}_{j-1} \otimes \dot{q}_j \mathbf{a}_j)
\end{aligned} \tag{B.3}$$

Si la base est fixe, la récurrence est initialisée par :

$$\begin{aligned}
\boldsymbol{\omega}_0 &= \mathbf{0} \\
\dot{\boldsymbol{\omega}}_0 &= \mathbf{0} \\
\mathbf{V}_0 &= \mathbf{0}
\end{aligned} \tag{B.4}$$

L'équation (B.1) permet de calculer \mathbf{F}_j et \mathbf{M}_j .

Récurrence arrière : calcul de l'organe terminal vers la base du robot

Les équations composant la récurrence arrière sont obtenues à partir du bilan des efforts sur chaque corps, écrit à l'origine \mathbf{O}_j :

$$\begin{aligned}
\mathbf{F}_j &= \mathbf{f}_j - \mathbf{f}_{j+1} + \mathbf{M}_j \mathbf{g} - \mathbf{f}_{ej} \\
\mathbf{M}_j &= \mathbf{m}_j - \mathbf{m}_{j+1} - \mathbf{L}_{j+1} \otimes \mathbf{f}_{j+1} + \mathbf{S}_j \otimes \mathbf{M}_j \mathbf{g} - \mathbf{m}_{ej}
\end{aligned} \tag{B.5}$$

Où :

- \mathbf{f}_j est la force résultante du torseur dynamique des efforts exercés sur le corps C_j par son antécédent et par l'actionneur j ,
- \mathbf{f}_{ej} est la résultante du torseur dynamique des efforts exercés par le corps C_j sur l'environnement,
- \mathbf{m}_j est le moment du torseur dynamique des efforts exercés sur le corps C_j par son antécédent et par l'actionneur autour de \mathbf{O}_j ,
- \mathbf{m}_{ej} est le moment du torseur dynamique des efforts exercés par le corps C_j sur l'environnement autour de \mathbf{O}_j .

Les équations (B.1) et (B.5) montrent que l'on peut faire intervenir l'effet de la gravité sans avoir à la prendre en compte explicitement dans le bilan des efforts des corps en prenant :

$$\dot{\mathbf{V}}_0 = -\mathbf{g} \tag{B.6}$$

Ce qui permet d'écrire :

$$\begin{aligned}
\mathbf{f}_j &= \mathbf{F}_j + \mathbf{f}_{j+1} + \mathbf{f}_{ej} \\
\mathbf{m}_j &= \mathbf{M}_j + \mathbf{m}_{j+1} + \mathbf{L}_{j+1} \otimes \mathbf{f}_{j+1} + \mathbf{m}_{ej}
\end{aligned} \tag{B.7}$$

La récurrence est initialisée par :

$$\begin{aligned}\mathbf{f}_{n+1} &= 0 \\ \mathbf{m}_{n+1} &= 0\end{aligned}\tag{B.8}$$

\mathbf{F}_j et \mathbf{M}_j sont calculés par (B.1) à (B.5).

On obtient alors les efforts moteurs τ_{mdij} en projetant (B.7) sur l'axe j du mouvement articulaire. On ajoute enfin les termes représentant la dynamique de la chaîne d'actionnement, ce qui donne :

$$\tau_{\text{mdi } j} = \left(\sigma_j \mathbf{f}_j + \bar{\sigma}_j \mathbf{m}_j \right)^T \mathbf{a}_j + \tau_{\text{fj}} + \text{Ia}_j \ddot{q}_j\tag{B.9}$$

On déduit de (B.7) que \mathbf{f}_j et \mathbf{m}_j ne dépendent que des paramètres inertiels du corps j et de ceux des corps situés entre j et n qui sont introduits par les termes \mathbf{f}_{j+1} et \mathbf{m}_{j+1} .

C. Méthodes formelle et numérique pour le calcul des paramètres de base

Deux méthodes qui permettent de calculer un jeu minimal de paramètres inertiels, appelés paramètres de base, sont présentées. L'utilisation de ces paramètres dans le calcul du modèle dynamique diminue sa complexité.

En outre, cette étape est un préliminaire indispensable pour l'identification des paramètres inertiels, les paramètres de base constituant les seuls paramètres identifiables avec le modèle dynamique inverse. Ils sont obtenus à partir des paramètres standards en éliminant ceux qui n'ont pas d'effet sur le modèle dynamique et en regroupant certains autres entre eux. On présente une méthode formelle qui utilise le modèle d'énergie du robot et une méthode numérique qui utilise la factorisation QR.

Méthode formelle

L'énergie totale H_j du corps C_j est rappelée (voir équation (A.14)). Elle est donnée par la relation linéaire suivante en fonction des paramètres inertiels (Gautier, 1990)(Gautier et Khalil, 1990)(Mayeda et al., 1990) :

$$H^j = E^j + U^j = \mathbf{h}^j \boldsymbol{\chi}_1^j \quad (\text{C.1})$$

A partir des équations (A.7) et (A.11), la relation générale de récurrence entre les fonctions d'énergie des corps j et $j-1$ s'écrit :

$$\mathbf{h}^j = \mathbf{h}^{j-1} {}^{j-1}\boldsymbol{\lambda}_j + \dot{\mathbf{q}}_j \boldsymbol{\eta}_j \quad (\text{C.2})$$

Où :

- ${}^{j-1}\boldsymbol{\lambda}_j$ est une matrice (10x10) fonction des paramètres géométriques du repère j (voir tableau C-1),
- $\boldsymbol{\eta}_j$ est un vecteur ligne :

$$\begin{aligned} \boldsymbol{\eta}_j = & \bar{\sigma}_j \left[\begin{array}{cccccccccc} 0 & 0 & \omega_{1,j} & 0 & \omega_{2,j} & \omega_{3,j} - \frac{1}{2}\dot{\mathbf{q}}_j & V_{2,j} & -V_{1,j} & 0 & 0 \end{array} \right] + \dots \\ & + \sigma_j \left[\begin{array}{cccccccccc} 0 & 0 & 0 & 0 & 0 & 0 & -\omega_{2,j} & \omega_{1,j} & 0 & V_{3,j} - \frac{1}{2}\dot{\mathbf{q}}_j \end{array} \right] \end{aligned} \quad (\text{C.3})$$

La matrice ${}^{j-1}\boldsymbol{\lambda}_j$ permet d'exprimer les paramètres inertiels du corps j dans le repère $j-1$ selon la relation :

$${}^{j-1}\boldsymbol{\chi}_1^j = {}^{j-1}\boldsymbol{\lambda}_j {}^j\boldsymbol{\chi}_1^j \quad (\text{C.4})$$

Pour prendre en compte les inerties des actionneurs, on introduit le paramètre inertiel I_a , comme onzième paramètre standard du corps C_j , ce qui porte à 11 n leur nombre total pour le robot. La matrice \mathbf{h}_j est alors étendue à 11 composantes avec :

$$h_{11,j} = \frac{1}{2} \dot{q}_j^2 \quad (C.5)$$

Les paramètres de base s'obtiennent à partir des équations de l'énergie :

- Un paramètre inertiel χ_{ij} n'a pas d'effet sur le modèle dynamique si sa fonction d'énergie est constante : $h_j = \text{constante}$. Dans ce cas, la fonction h_j et le paramètre χ_{ij} sont éliminés du modèle (C.1).
- Un paramètre inertiel χ_{ij} peut être regroupé avec r autres paramètres $\chi_{i1}, \dots, \chi_{ir}$ si sa fonction d'énergie est linéaire par rapport aux fonctions de ces paramètres :

$$h_j = \sum_{p=1}^r t_{jp} h_{jp} + \text{cte} \quad (C.6)$$

où t_{jp} est un coefficient constant.

On obtient alors la relation suivante pour chaque paramètre χ_{ijp} :

$$\chi_{ijpR} = \chi_{ijp} + t_{jp} \chi_{ij} \quad (C.7)$$

L'indice R signifie que le paramètre est regroupé avec d'autres paramètres.

L'utilisation des expressions de \mathbf{h}^j , permet d'établir des règles générales pour déterminer les paramètres qui n'affectent pas le modèle dynamique sans avoir à calculer l'énergie.

A partir de la relation de récurrence (C.2) entre les fonctions d'énergie \mathbf{h}^j et \mathbf{h}^{j-1} , on en déduit les relations de regroupement entre paramètres (différentes selon le type d'articulation : rotoïde ou prismatique).

Finalement, différentes règles sont obtenues afin d'éliminer ou de regrouper des paramètres, pour ne conserver que les paramètres de base. La démarche est donnée ci-après.

Tableau C-1 : Expression des éléments de la matrice ${}^i\lambda_j$

$i \backslash j$	1	2	3	4	5	6	7	8	9	10
1	CC θ	-2 CS θ	0	SS θ	0	0	0	0	2r	r ²
2	CS θ C α	(CC θ - SS θ) C α	-C θ S α	-CS θ C α	S θ S α	0	-dS θ C α +rC θ S α	-dC θ C α -rS θ S α	dS α	drS
3	CS θ S α	(CC θ - SS θ) S α	C θ C α	-CS θ S α	-S θ C α	0	-dS θ S α +rC θ C α	-dC θ S α +rS θ C α	-dC α	-drC α
4	SS θ C α	2CS θ C α	-2S θ C α	CC θ C α	-2C θ C α	SS α	2(dC θ +rS θ C α)	2(-dS θ +rC θ C α)	2rC α	d α +r ² C α
5	SS θ S α	2CS θ S α	S θ (C α - SS α)	CC θ S α	C θ (C α - SS α)	-C α S α	rS θ (SS α - C α)	rC θ (SS α - C α)	2rC α S α	r ² C α
6	SS θ SS α	2CS θ SS α	2S θ C α S α	CC θ SS α	2C θ C α S α	CC α	2(dC θ -rS θ C α)	-2(dS θ +rC θ C α)	2rSS α	d α +r ² SS α
7	0	0	0	0	0	0	C θ	-S θ	0	d
8	0	0	0	0	0	0	S θ C α	C θ C α	-S α	-rS α
9	0	0	0	0	0	0	S θ S α	C θ S α	C α	rC α
10	0	0	0	0	0	0	0	0	0	1

Notations: CC α =cos²(α), SS α =sin²(α), CS α =cos(α)sin(α), d=d, r=r

Soit r_1 la première articulation rotoïde en partant de la base et r_2 la première articulation rotoïde succédant à l'articulation r_1 et d'axe z_{r_2} non parallèle à z_{r_1} . On applique pour $j = n, \dots, 1$ les règles suivantes :

1) Si l'articulation j est une rotoïde ($\sigma_j=0$), les paramètres YY_j , MZ_j et M_j peuvent être regroupés avec les autres paramètres des corps j et $j-1$. Les formules de regroupements sont les suivantes :

- $XX_{jR} = XX_j - YY_j$,
- $XX_{j-1R} = XX_{j-1} + YY_j + 2r_jMZ_j + r_j^2M_j$,
- $XY_{j-1R} = XY_{j-1} + d_jS\alpha_jMZ_j + d_jr_jS\alpha_jM_j$,
- $XZ_{j-1R} = XZ_{j-1} - d_jC\alpha_jMZ_j - d_jr_jC\alpha_jM_j$,
- $YY_{j-1R} = YY_{j-1} + CC\alpha_jYY_j + 2r_jCC\alpha_jMZ_j + (d_j^2 + r_j^2CC\alpha_j) M_j$,
- $YZ_{j-1R} = YZ_{j-1} + CS\alpha_jYY_j + 2r_jCS\alpha_jMZ_j + r_j^2CS\alpha_j M_j$,
- $ZZ_{j-1R} = ZZ_{j-1} + SS\alpha_jYY_j + 2r_jSS\alpha_jMZ_j + (d_j^2 + r_j^2SS\alpha_j) M_j$,
- $MX_{j-1R} = MX_{j-1} + d_jM_j$,
- $MY_{j-1R} = MY_{j-1} - S\alpha_jMZ_j - r_jS\alpha_jM_j$,
- $MZ_{j-1R} = MZ_{j-1} + C\alpha_jMZ_j + r_jC\alpha_jM_j$,
- $M_{j-1R} = M_{j-1} + M_j$.

2) Si l'articulation j est une prismatique ($\sigma_j=1$), les paramètres du tenseur d'inertie J_j peuvent être regroupés avec les éléments du tenseur d'inertie J_{j-1} . Les formules de regroupements sont les suivantes :

- $XX_{j-1R} = XX_{j-1} + CC\theta_jXX_j - 2CS\theta_jXY_j + SS\theta_jYY_j$,
- $XY_{j-1R} = XY_{j-1} + CS\theta_jC\alpha_jXX_j + (CC\theta_j - SS\theta_j) C\alpha_jXY_j - C\theta_jS\alpha_jXZ_j \dots$
 $\dots - CS\theta_jC\alpha_jYY_j + S\theta_jS\alpha_jYZ_j$,
- $XZ_{j-1R} = XZ_{j-1} + CS\theta_jS\alpha_jXX_j + (CC\theta_j - SS\theta_j) S\alpha_jXY_j + C\theta_jC\alpha_jXZ_j \dots$
 $\dots - CS\theta_jS\alpha_jYY_j - S\theta_jC\alpha_jYZ_j$,
- $YY_{j-1R} = YY_{j-1} + SS\theta_jCC\alpha_jXX_j + 2CS\theta_jCC\alpha_jXY_j - 2S\theta_jCS\alpha_jXZ_j \dots$
 $\dots + CC\theta_jCC\alpha_jYY_j - 2C\theta_jCS\alpha_jYZ_j + SS\alpha_jZZ_j$,
- $YZ_{j-1R} = YZ_{j-1} + SS\theta_jCS\alpha_jXX_j + 2CS\theta_jCS\alpha_jXY_j - S\theta_j(CC\alpha_j - SS\alpha_j)XZ_j \dots$
 $\dots + CC\theta_jCS\alpha_jYY_j + C\theta_j(CC\alpha_j - SS\alpha_j)YZ_j - CS\alpha_jZZ_j$,
- $ZZ_{j-1R} = ZZ_{j-1} + SS\theta_jSS\alpha_jXX_j + 2CS\theta_jSS\alpha_jXY_j - 2S\theta_jCS\alpha_jXZ_j \dots$
 $\dots + CC\theta_jSS\alpha_jYY_j + 2C\theta_jCS\alpha_jYZ_j + CC\alpha_jZZ_j$.

3) Si j est prismatique ($\sigma_j=1$), pour $r_1 < j < r_2$, on déduit :

- si \mathbf{a}_j est parallèle à \mathbf{a}_{r_1} , éliminer MZ_j et regrouper MX_j et MY_j en utilisant les relations suivantes :

- $MX_{j-1R} = MX_{j-1} + C\theta_jMX_j - S\theta_jMY_j$,
- $MY_{j-1R} = MY_{j-1} + S\theta_jC\alpha_jMX_j + C\theta_jC\alpha_jMY_j$,
- $MZ_{j-1R} = MZ_{j-1} + S\theta_jS\alpha_jMX_j + C\theta_jS\alpha_jMY_j$,

- $ZZ_{jR} = ZZ_j + 2d_j C\theta_j MX_j - 2d_j S\theta_j MY_j$.
- si \mathbf{a}_j n'est pas perpendiculaire ni parallèle à \mathbf{a}_{r1} , regrouper MZ_j :
- $MX_{jR} = MX_j - ({}^j a_{xr1}/{}^j a_{zr1})MZ_j$,
- $MY_{jR} = MY_j - ({}^j a_{yr1}/{}^j a_{zr1})MZ_j$.
- si \mathbf{a}_j est perpendiculaire à \mathbf{a}_{r1} (donc ${}^j a_{zr1}=0$), éliminer MX_j si la composante ${}^j a_{yr1}$ est nulle, éliminer MY_j si ${}^j a_{xr1}$ est nulle, sinon (pour ${}^j a_{xr1}$ et ${}^j a_{yr1}$ non nulles) regrouper MY_j avec la relation :
- $MX_{jR} = MX_j - ({}^j a_{xr1}/{}^j a_{yr1})MY_j$.

4) Si l'articulation j est rotoïde ($\sigma_j=0$) et $r_1 \leq j < r_2$, éliminer XX_j , XY_j , XZ_j , et YZ_j qui n'ont pas d'effet.

5) Si l'articulation j est rotoïde ($\sigma_j=0$) et $r_1 \leq j < r_2$, \mathbf{a}_j est confondu avec \mathbf{a}_{r1} , et \mathbf{a}_{r1} est parallèle à \mathbf{a}_i et \mathbf{g} pour $i < j$, alors éliminer MX_j et MY_j qui n'ont pas d'effet.

6) Le paramètre Ia_j peut être regroupé avec ZZ_j ou M_j dans les cas suivants :

- si $j=r_1$
- si $j=r_1$ et son axe est perpendiculaire à r_1
- si $j \leq r_2$ et que l'axe de la première articulation prismatique, notée p_1 , est perpendiculaire à la gravité et son axe est aligné avec les axes rotoïdes en amont ou $p_1 = 1$.

Méthode numérique (Gautier, 1990)(Gautier, 1991)

La méthode numérique utilise la décomposition QR d'une matrice \mathbf{W}_{st} obtenue par échantillonnage de la matrice $\mathbf{MDI}_{st}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$ permettant de détecter les dépendances linéaires entre ses colonnes et de sélectionner un ensemble de colonnes indépendantes et un jeu minimal de paramètres.

$(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$ peuvent être des réalisations de variables aléatoires (bruit uniforme par exemple) bornées respectivement aux positions, vitesses et accélérations maximales ou directement les signaux expérimentaux. Dans ce dernier cas, les paramètres identifiables peuvent être différents des paramètres de base suivant le type d'essai (essais avec des axes bloqués par exemple).

La décomposition QR de la matrice \mathbf{W}_{st} permet d'écrire :

$$\mathbf{Q}^T \mathbf{W}_{st} = \begin{bmatrix} \mathbf{R} \\ 0_{(r-c) \times c} \end{bmatrix} \quad (\text{C.8})$$

où :

- \mathbf{Q} est une matrice orthogonale de dimension (rxr) ,
- \mathbf{R} une matrice triangulaire supérieure de dimension (cxc) .

Le rang de la matrice \mathbf{W}_{st} est égal au nombre des b paramètres minimaux :

$$\text{rang}(\mathbf{W}_{\text{st}}) = b \quad (\text{C.9})$$

Ce rang se détermine par le nombre d'éléments sur la diagonale R_{ii} non nuls de la matrice \mathbf{R} . Numériquement, on définit un seuil de tolérance et on considère les différents R_{ii} nuls s'ils vérifient la relation $|R_{ii}| \leq \text{tol}$.

Les b éléments R_{ii} de la diagonale de \mathbf{R} tels que $|R_{ii}| > \text{tol}$ fournissent les indices des paramètres inertiels de base χ_1 , les colonnes indépendantes correspondantes de \mathbf{W}_{st} étant notées \mathbf{W}_1 . Les autres fournissent les paramètres à regrouper χ_2 et les colonnes dépendantes \mathbf{W}_2 . On écrit ainsi que :

$$\mathbf{W}_{\text{st}}\chi_{\text{st}} = [\mathbf{W}_1 \quad \mathbf{W}_2] \begin{bmatrix} \chi_1 \\ \chi_2 \end{bmatrix} \quad (\text{C.10})$$

La matrice β exprime les relations linéaires entre les colonnes de \mathbf{W}_{st} sous la forme explicite :

$$\mathbf{W}_2 = \mathbf{W}_1\beta \quad (\text{C.11})$$

En conséquence, on peut écrire que :

$$\mathbf{W}_{\text{st}}\chi_{\text{st}} = [\mathbf{W}_1 \quad \mathbf{W}_1\beta] \begin{bmatrix} \chi_1 \\ \chi_2 \end{bmatrix} = \mathbf{W}_1 [\chi_1 + \beta\chi_2] \quad (\text{C.12})$$

soit :

$$\mathbf{W}_{\text{st}}\chi_{\text{st}} = [\mathbf{W}_1 \quad \mathbf{W}_2] \begin{bmatrix} \chi \\ 0 \end{bmatrix} = \mathbf{W}_1\chi \quad (\text{C.13})$$

Avec la relation de regroupement :

$$\chi = \chi_1 + \beta\chi_2 \quad (\text{C.14})$$

La décomposition QR de la matrice $[\mathbf{W}_1 \quad \mathbf{W}_2]$ s'écrit :

$$\begin{aligned}
[\mathbf{W}_1 \quad \mathbf{W}_2] &= [\mathbf{Q}_1 \quad \mathbf{Q}_2] = \begin{bmatrix} \mathbf{R}_1 & \mathbf{R}_2 \\ \mathbf{0}_{(r-b) \times b} & \mathbf{0}_{(r-b) \times (c-b)} \end{bmatrix} \\
&= [\mathbf{Q}_1 \mathbf{R}_1 \quad \mathbf{Q}_1 \mathbf{R}_2]
\end{aligned} \tag{C.15}$$

Où :

- \mathbf{R}_1 est une matrice triangulaire supérieure régulière de dimension $(b \times b)$,
- \mathbf{R}_2 est une matrice de dimension $b \times (c-b)$.

On obtient finalement :

$$\begin{aligned}
\mathbf{W}_1 &= \mathbf{Q}_1 \mathbf{R}_1 \\
\mathbf{Q}_1 &= \mathbf{W}_1 \mathbf{R}_1^{-1} \\
\mathbf{W}_2 &= \mathbf{Q}_1 \mathbf{R}_2 = \mathbf{W}_1 \mathbf{R}_1^{-1} \mathbf{R}_2
\end{aligned} \tag{C.16}$$

Ce résultat fournit les $(c-b)$ relations linéaires entre les colonnes de \mathbf{W}_2 et \mathbf{W}_1 . On obtient ainsi la relation de regroupement des paramètres χ_2 sur les paramètres χ_1 à partir de laquelle on établit le vecteur des paramètres de base χ :

$$\chi = \chi_1 + \mathbf{R}_1^{-1} \mathbf{R}_2 \chi_2 \tag{C.17}$$

La méthode numérique est particulièrement intéressante pour le calcul du modèle minimal de tout modèle linéaire par rapport aux paramètres.

D. Table de Fisher-Snedecor

Si F est une variable aléatoire suivant la loi de Fisher-Snedecor à (ν_1, ν_2) degrés de liberté, la table donne la valeur $F_{(1-\alpha)}$ telle que (voir tableau D-1):

$$P\{\hat{F} \geq F_{(1-\alpha)}\} = \alpha \quad (D.1)$$

$F_{(1-\alpha)}$ est le quantile d'ordre $1-\alpha$ de la loi de Fisher-Snedecor. On définit $\nu_1=b-b_c$ iet $\nu_2=n_e-b$ et on prend α égal à 0.05.

Tableau D-1 : Table de Fisher-Snedecor ($\alpha=0.05$)

$\nu_2 \backslash \nu_1$	1	2	3	4	5	6	8	12	24	∞
1	161	199	216	225	230	234	239	244	249	254
2	18,5	19,0	19,2	19,2	19,3	19,33	19,3	19,4	19,4	19,5
3	10,1	9,55	9,28	9,12	9,01	8,94	8,85	8,74	8,64	8,53
4	7,71	6,94	6,59	6,39	6,26	6,16	6,04	5,91	5,77	5,63
5	6,61	5,79	5,41	5,19	5,05	4,95	4,82	4,68	4,53	4,37
6	5,99	5,14	4,76	4,53	4,39	4,28	4,15	4,00	3,84	3,67
7	5,59	4,74	4,35	4,12	3,97	3,87	3,73	3,57	3,41	3,23
8	5,32	4,46	4,07	3,84	3,69	3,58	3,44	3,28	3,12	2,93
9	5,12	4,26	3,86	3,63	3,48	3,37	3,23	3,07	2,90	2,71
10	4,96	4,10	3,71	3,48	3,33	3,22	3,07	2,91	2,74	2,54
11	4,84	3,98	3,59	3,36	3,20	3,09	2,95	2,79	2,61	2,41
12	4,75	3,89	3,49	3,26	3,11	3,00	2,85	2,69	2,51	2,30
13	4,67	3,81	3,41	3,18	3,03	2,92	2,77	2,60	2,42	2,21
14	4,60	3,74	3,34	3,11	2,96	2,85	2,70	2,53	2,35	2,14
15	4,54	3,68	3,29	3,06	2,90	2,79	2,64	2,48	2,29	2,07
16	4,49	3,63	3,24	3,01	2,85	2,74	2,59	2,42	2,24	2,02
17	4,45	3,59	3,20	2,96	2,81	2,70	2,55	2,38	2,19	1,97
18	4,41	3,55	3,16	2,93	2,77	2,66	2,51	2,34	2,15	1,92
19	4,38	3,52	3,13	2,90	2,74	2,63	2,48	2,31	2,11	1,88
20	4,35	3,49	3,10	2,87	2,71	2,60	2,45	2,28	2,08	1,85
21	4,32	3,47	3,07	2,84	2,68	2,57	2,42	2,25	2,05	1,82
22	4,30	3,44	3,05	2,82	2,66	2,55	2,40	2,23	2,03	1,79
23	4,28	3,42	3,03	2,80	2,64	2,53	2,37	2,20	2,01	1,76
24	4,26	3,40	3,01	2,78	2,62	2,51	2,36	2,18	1,98	1,74
25	4,24	3,39	2,99	2,76	2,60	2,49	2,34	2,16	1,96	1,72
26	4,23	3,37	2,98	2,74	2,59	2,47	2,32	2,15	1,95	1,70
27	4,21	3,35	2,96	2,73	2,57	2,46	2,31	2,13	1,93	1,68
28	4,20	3,34	2,95	2,71	2,56	2,45	2,29	2,12	1,91	1,66
29	4,18	3,33	2,93	2,70	2,55	2,43	2,28	2,10	1,90	1,65
30	4,17	3,32	2,92	2,69	2,53	2,42	2,27	2,09	1,89	1,63
40	4,08	3,23	2,84	2,61	2,45	2,34	2,18	2,00	1,79	1,52
60	4,00	3,15	2,76	2,53	2,37	2,25	2,10	1,92	1,70	1,40
125	3,92	3,07	2,68	2,44	2,29	2,17	2,01	1,83	1,60	1,26
∞	3,85	3,00	2,61	2,38	2,22	2,10	1,94	1,76	1,52	1,09

E. Test de normalité de Kolmogorov-Smirnov

Le test de normalité de Kolmogorov-Smirnov permet de vérifier qu'un échantillon de n_e variables aléatoires indépendantes (x_1, \dots, x_{n_e}) suit une loi de distribution normale (Massey, 1951)(Miller, 1956)(Marsaglia et al., 2003). La densité de probabilité f_n de cette loi normale est la suivante:

$$f_n(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2} \quad (\text{E.1})$$

Où:

- μ est l'espérance mathématique,
- σ est l'écart-type.

On se limite par la suite au cas d'une loi de distribution normale centrée réduite où μ est nul et $\sigma=1$. Les valeurs de probabilité utilisées sont issues des tables de Massey (Massey, 1951). L'hypothèse H_0 est que la densité de probabilité de l'ensemble des valeurs de x suit une loi normale centrée réduite ($x \sim N(0,1)$):

$$F(x_i) = F_n(x_i) \quad \forall x_i \quad (\text{E.2})$$

Où:

- F est la fonction de répartition de l'ensemble de valeurs à tester,
- F_n est la fonction de répartition d'une loi de distribution normale centrée réduite.

Le test de normalité de Kolmogorov-Smirnov teste que la fonction de répartition de l'ensemble des échantillons est égale à la fonction de répartition de la loi normale centrée réduite:

$$D_{\text{obs}} = \max_{x_i} \left(\left| F(x_i) - F_n(x_i) \right| \right) \quad \forall x_i \quad (\text{E.3})$$

Où:

- F est la fonction de répartition de l'ensemble des échantillons,
- F_n est la fonction de répartition de la loi normale centrée réduite.

L'hypothèse H_0 est fautive si $D_{\text{obs}} > D_\alpha$ avec $\alpha=0.05$:

$$D_{\alpha=0.05} = \frac{0.895}{S} \quad \text{avec} \quad S = \sqrt{n_e} - 0.01 + \frac{0.85}{\sqrt{n_e}} \quad (\text{E.4})$$

Où n_e est la taille de l'échantillon.

Le test de Kolmogorov-Smirnov est implanté dans le logiciel Matlab dans la fonction 'kstest'. Il ne peut être utilisé que si l'indépendance statistique entre les échantillons est vérifiée.

F. Test de Durbin-Watson

Le test de Durbin-Watson permet de savoir s'il existe une indépendance des résidus d'ordre 1 (Durbin et Watson, 1950). Ce dernier repose sur l'estimation d'un modèle autorégressif de premier ordre pour l'estimation des résidus.

La statistique de Durbin-Watson est donnée par l'équation suivante :

$$dw = \frac{\sum_{i=2}^r (\rho_p(i) - \rho_p(i-1))^2}{\sum_{i=1}^r (\rho_p(i))^2} \quad (F.1)$$

La statistique de Durbin-Watson est comprise entre 0 et 4. Les valeurs proches de 2 montrent une absence d'indépendance des résidus, les valeurs proches de 0 montrent une indépendance positive et les valeurs proches de 4 montrent une autocorrélation négative.

La difficulté rencontrée avec l'utilisation de ce test réside dans l'estimation de la valeur-p. La valeur-p est la probabilité d'obtenir la même valeur du test du l'hypothèse nulle est vraie. L'algorithme de Pan peut être utilisé pour calculer la valeur-p du test de Durbin-Watson (Farebrother, 1980). Les seuils suivants sont utilisés:

- valeur-p < 0.05 : présomption contre l'hypothèse nulle
- 0.05 < valeur-p: faible présomption ou pas de présomption contre l'hypothèse nulle

Le test avec le calcul de la valeur-p par l'algorithme de Pan est implémenté dans Matlab dans la fonction 'dwtest'.

Le test de de Durbin-Watson ne peut être appliqué que si variance de l'erreur ρ est homoscédastique.

G. Paramètres du contrôleur simulé du Kuka *LWR*

Dans (Jubien et al. 2014a), l'identification des paramètres des corps utilisés dans la baie de commande du robot Kuka *LWR* est effectuée à l'aide de la matrice d'inertie et du vecteur gravité fournis en temps réel. Ces paramètres sont donnés dans le tableau G-1.

Tableau G-1 : Paramètres des corps Kuka différents de 0

Par.	χ^{kuka}	Par.	χ^{kuka}
ZZ _{1R}	1.15 10⁻²	MY _{4R}	-1.33
XX _{2R}	1.25	XX _{5R}	3.95 10⁻³
YZ ₂	-5.43 10⁻⁴	YZ ₅	4.22 10⁻⁴
ZZ _{2R}	1.25	ZZ _{5R}	6.33 10⁻³
MX ₂	1.35 10⁻³	MX ₅	-6.56 10⁻⁴
MY _{2R}	3.46	MY _{5R}	4.07 10⁻²
XX _{3R}	6.36 10⁻³	XX _{6R}	1.17 10⁻³
YZ ₃	7.26 10⁻⁴	YZ ₆	1.02 10⁻⁵
ZZ _{3R}	1.08 10⁻²	ZZ _{6R}	3.77 10⁻³
MX ₃	9.45 10⁻⁴	MX ₆	8.35 10⁻⁴
MY _{3R}	-4.72 10⁻⁴	MY _{6R}	2.86 10⁻²
XX _{4R}	0.413	ZZ ₇	1.20 10⁻⁴
YZ ₄	5.32 10⁻⁴	MX ₇	-2.98 10⁻⁴
ZZ _{4R}	0.418	MY ₇	9.54 10⁻⁴
MX ₄	-3.50 10⁻³		

Ces paramètres sont utilisés pour le calcul de la matrice d'inertie et du vecteur de gravité de l'équation (4.48).

Cependant, l'effet des moments d'inertie Ia_j des chaînes d'actionnement n'est pas présent dans le modèle de Kuka. Les valeurs ZZ_{1R} et ZZ_{2R} sont remplacées par les valeurs identifiées données par le tableau 4-9, et les moments d'inertie Ia_j des chaînes d'actionnement des articulations 3 à 7 sont pris en compte en prenant les valeurs identifiées données par le tableau 4-9.

Les gains sont calculés pour les pulsations naturelles désirées suivantes :

$$\begin{aligned}
 {}^d\omega_1 &= 125.6 \text{ rd / s} \\
 {}^d\omega_2 &= 125.6 \text{ rd / s} \\
 {}^d\omega_3 &= 125.6 \text{ rd / s} \\
 {}^d\omega_4 &= 188.5 \text{ rd / s} \\
 {}^d\omega_5 &= 188.5 \text{ rd / s} \\
 {}^d\omega_6 &= 188.5 \text{ rd / s} \\
 {}^d\omega_7 &= 251.3 \text{ rd / s}
 \end{aligned} \tag{G.1}$$

RESUMÉ et MOTS CLÉS

IDENTIFICATION DYNAMIQUE DES ROBOTS A FLEXIBILITÉS ARTICULAIRES

Ce travail résulte d'une collaboration entre l'IRCCyN et l'ONERA sur l'identification dynamique des robots à flexibilités articulaires, utilisés par exemple dans les applications de la robotique collaborative. La technique d'identification usuelle en robotique nécessite la mesure des positions moteurs et la mesure des déformations élastiques, non disponibles sur les robots industriels. Récemment, une nouvelle technique nommée *DIDIM* (Direct and Inverse Dynamic Identification Models), qui utilise uniquement la mesure des efforts moteurs, a été proposée et validée sur les robots rigides. Cette thèse propose une extension de *DIDIM*, qui n'utilise aucune mesure de position, aux cas des robots à flexibilités articulaires. On réalise d'abord une étude comparative sur un robot rigide 6 axes avec 61 paramètres, qui démontre la supériorité de *DIDIM* sur une méthode usuelle en boucle fermée à erreur de sortie en position (*CLOE*) : *DIDIM* converge 100 fois plus vite et est largement plus robuste vis à vis des erreurs sur les conditions initiales. Ensuite *DIDIM* est étendue aux robots à flexibilités articulaires avec une procédure en trois étapes : une identification du modèle rigide en basses fréquences, une identification du mode flexible et des inerties de part et d'autre de la flexibilité et enfin une identification globale précise du modèle dynamique flexible complet. Une validation expérimentale est réalisée sur un banc d'essai de robot un axe avec une flexibilité. Ensuite, une validation en simulation sur le robot 7 axes Kuka Light Weight Robot montre l'efficacité de la méthode *DIDIM* appliquée aux robots à flexibilités articulaires industriels, dans le cas où la commande est connue.

Mots-clés : robotique, modélisation, identification, paramètres dynamiques, flexibilité articulaire

DYNAMIC IDENTIFICATION OF FLEXIBLE JOINT ROBOTS

This work is the result of collaboration between *IRCCyN* and *ONERA* on dynamic identification of robots with joint flexibilities, used for example in new applications for collaborative robotics. The usual identification technique in robotics requires the actual data of motor positions and the actual data of elastic deformations, which are usually not available in industrial robots. Recently, a new technique called *DIDIM* (Direct and Inverse Dynamic Identification Models), which uses only the data of motor torques, has been proposed and validated on rigid robots. This thesis proposes an extension of *DIDIM*, which uses no actual position data at all, to the case of robots with joint flexibilities. First, a comparative study on a rigid 6-axis robot with 61 parameters, shows the superiority of *DIDIM* over a conventional method *CLOE* (Closed-Loop Output Error) in position: *DIDIM* converges 100 times faster and is strongly more robust with respect to errors in the initial conditions. Second, *DIDIM* is extended to robots with joint flexibilities in a three steps procedure: a rigid model identification at low frequencies, an approximated identification of the flexible mode and of the inertia of each side of the flexibility, and finally the overall accurate identification of the full flexible dynamic model. A first experimental validation is performed on a test bench robot with one joint and one flexibility. A second validation in simulation on the 7 axes Kuka Light Weight Robot shows the effectiveness of *DIDIM* applied to industrial robots with joint flexibilities, in the case where the actual control law is known.

Keywords: robotics, modeling, identification, dynamic parameters, flexible joint