



HAL
open science

Modélisation d'un robot manipulateur en vue de la commande robuste en force utilisé en soudage FSW

Ke Wang

► **To cite this version:**

Ke Wang. Modélisation d'un robot manipulateur en vue de la commande robuste en force utilisé en soudage FSW. Automatique / Robotique. Ecole nationale supérieure d'arts et métiers - ENSAM, 2016. Français. NNT : 2016ENAM0003 . tel-01305951

HAL Id: tel-01305951

<https://pastel.hal.science/tel-01305951>

Submitted on 22 Apr 2016

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.

École doctorale n° 432 : Science des Métiers de l'Ingénieur

Doctorat ParisTech

THÈSE

pour obtenir le grade de docteur délivré par

l'École Nationale Supérieure d'Arts et Métiers

Spécialité " Automatique "

présentée et soutenue publiquement par

Ke WANG

le 28 janvier 2016

**Robot manipulator modeling for robust force control
used in Friction Stir Welding (FSW)**

~~~

**Modélisation d'un robot manipulateur en vue de  
la commande robuste en force utilisé en soudage FSW**

Directeur de thèse : **Gabriel ABBA**

Co-encadrement de la thèse : **François LEONARD**

### Jury

**M. Guillaume MOREL**, Professeur, Université de Pierre et Marie Curie, Paris VI

**M. Wolfgang SEEMANN**, Professeur, Karlsruhe Institute of Technology

**M. Stéphane CARO**, Chargé de recherche, IRCCyN, CNRS

**M. Gabriel ABBA**, Professeur, LCFC, Université de Lorraine

**M. François LEONARD**, Maître de conférences, LCFC, Université de Lorraine

**M. Aurélien ROBINEAU**, Ingénieur, Institut de Soudure

Président  
Rapporteur  
Rapporteur  
Examineur  
Examineur  
Invité

**T  
H  
È  
S  
E**



# Contents

|                                                                        |           |
|------------------------------------------------------------------------|-----------|
| <b>Acknowledgements</b>                                                | <b>4</b>  |
| <b>List of Figures</b>                                                 | <b>7</b>  |
| <b>List of Tables</b>                                                  | <b>10</b> |
| <b>List of Abbreviations</b>                                           | <b>12</b> |
| <br>                                                                   |           |
| <b>I English Version</b>                                               | <b>15</b> |
| <br>                                                                   |           |
| <b>1 General Introduction and Literature Review</b>                    | <b>17</b> |
| 1.1 Research Background and Motivations . . . . .                      | 17        |
| 1.2 Literature Review on Industrial Robot Manipulators . . . . .       | 18        |
| 1.2.1 History of Industrial Robot Manipulators . . . . .               | 18        |
| 1.2.2 Current Developments of Industrial Robots . . . . .              | 21        |
| 1.3 Literature Review on Modeling of Flexible Joint Robot Manipulators | 23        |
| 1.4 Literature Review on Control of Flexible Joint Robots . . . . .    | 24        |
| 1.4.1 Singular Perturbation and Integral Manifold . . . . .            | 24        |
| 1.4.2 Feedback Linearization . . . . .                                 | 25        |
| 1.4.3 Cascaded System and Integral Backstepping . . . . .              | 26        |
| 1.4.4 PD Control . . . . .                                             | 28        |
| 1.4.5 Other Control Methods . . . . .                                  | 28        |
| 1.5 Dissertation Outline . . . . .                                     | 29        |
| <br>                                                                   |           |
| <b>2 Modeling of Flexible Joint Robot Manipulators</b>                 | <b>31</b> |
| 2.1 Introduction . . . . .                                             | 31        |
| 2.2 Spatial Descriptions and Coordinate Transformations . . . . .      | 31        |
| 2.2.1 Descriptions of Positions, Orientations and Frames . . . . .     | 31        |
| 2.2.2 Homogeneous Transformations . . . . .                            | 33        |
| 2.2.3 X-Y-Z fixed angles . . . . .                                     | 34        |
| 2.3 Robot Kinematics . . . . .                                         | 35        |
| 2.3.1 Modified Denavit-Hartenberg Convention . . . . .                 | 35        |
| 2.3.2 Forward Kinematics: Direct Geometric Model . . . . .             | 36        |
| 2.3.3 Inverse Kinematics: Inverse Geometric Model . . . . .            | 39        |
| 2.3.4 Forward Instantaneous Kinematics: Direct Kinematic Model .       | 43        |
| 2.3.5 Inverse Instantaneous Kinematics: Inverse Kinematic Model .      | 44        |

|          |                                                                           |           |
|----------|---------------------------------------------------------------------------|-----------|
| 2.4      | Robot Dynamics . . . . .                                                  | 44        |
| 2.4.1    | Dynamic Modeling of the Robot . . . . .                                   | 44        |
| 2.4.2    | Flexibility Model . . . . .                                               | 46        |
| 2.4.3    | Friction Model . . . . .                                                  | 47        |
| 2.4.4    | Introduction of SYMORO+ . . . . .                                         | 48        |
| 2.5      | Conclusion . . . . .                                                      | 48        |
| <b>3</b> | <b>Simplification of Robot Dynamic Model Using Interval Method</b>        | <b>49</b> |
| 3.1      | Introduction . . . . .                                                    | 49        |
| 3.2      | Interval Analysis . . . . .                                               | 51        |
| 3.2.1    | Basic Definitions and Operations of Intervals . . . . .                   | 51        |
| 3.2.2    | Inclusion Functions and Overestimation . . . . .                          | 52        |
| 3.3      | Symbolic Robot Dynamic Model . . . . .                                    | 54        |
| 3.4      | Simplification Using Interval Method . . . . .                            | 55        |
| 3.4.1    | Simplification Algorithm . . . . .                                        | 56        |
| 3.4.2    | Example of Simplification of Component $M_{44}$ . . . . .                 | 58        |
| 3.4.3    | Results of Simplified Model for Whole Workspace . . . . .                 | 62        |
| 3.4.4    | Error Analysis of Simplified Model for Whole Workspace . . . . .          | 63        |
| 3.4.5    | Further Simplification and Results . . . . .                              | 64        |
| 3.5      | Case Study: Simplification on Three Different Test Trajectories . . . . . | 65        |
| 3.5.1    | Evaluation Indexes of Simplification . . . . .                            | 66        |
| 3.5.2    | Case I: Simplification on an Identification Trajectory . . . . .          | 67        |
| 3.5.3    | Case II: Simplification on a Linear FSW Trajectory . . . . .              | 72        |
| 3.5.4    | Case III: Simplification on a Circular FSW Trajectory . . . . .           | 77        |
| 3.5.5    | Case IV: Torques Analysis in Robot Dynamic Equation . . . . .             | 81        |
| 3.6      | Discussion on Usage of the Simplified Model . . . . .                     | 83        |
| 3.7      | Conclusion . . . . .                                                      | 84        |
| <b>4</b> | <b>Dynamic Modeling and Identification of Robotic FSW Process</b>         | <b>85</b> |
| 4.1      | Introduction . . . . .                                                    | 85        |
| 4.2      | Description of the FSW Process . . . . .                                  | 87        |
| 4.3      | Static Modeling of Process Forces in FSW . . . . .                        | 89        |
| 4.4      | Experimental Setup . . . . .                                              | 91        |
| 4.5      | Calculation of Plunge Depth and Data Filtering . . . . .                  | 95        |
| 4.5.1    | Definition of Plunge Depth . . . . .                                      | 95        |
| 4.5.2    | Three Methods for Calculating Plunge Depth . . . . .                      | 96        |
| 4.5.3    | Derivative of Plunge Depth and Data Filtering . . . . .                   | 99        |
| 4.6      | Linear Dynamic Modeling and Identification of Axial Force . . . . .       | 102       |
| 4.6.1    | Linear Dynamic Model of Axial Force . . . . .                             | 102       |
| 4.6.2    | Least Squares Method and Error Analysis Method . . . . .                  | 103       |
| 4.6.3    | Results and Error Analysis of Identified Linear Model . . . . .           | 105       |
| 4.7      | Nonlinear Dynamic Modeling and Identification of Axial Force . . . . .    | 109       |
| 4.7.1    | Nonlinear Dynamic Model and Identification Method . . . . .               | 109       |
| 4.7.2    | Results and Error Analysis of Identified Nonlinear Model . . . . .        | 111       |
| 4.8      | Conclusion . . . . .                                                      | 112       |

|           |                                                                                                                               |            |
|-----------|-------------------------------------------------------------------------------------------------------------------------------|------------|
| <b>5</b>  | <b>Design of Robust Force Controller in Robotic FSW Process</b>                                                               | <b>115</b> |
| 5.1       | Introduction . . . . .                                                                                                        | 115        |
| 5.2       | Force Control of Robot Manipulator . . . . .                                                                                  | 116        |
| 5.2.1     | Basic Force Control Approaches . . . . .                                                                                      | 116        |
| 5.2.2     | Advanced Force Control Approaches . . . . .                                                                                   | 118        |
| 5.3       | Force Control Strategy and System Modeling . . . . .                                                                          | 118        |
| 5.3.1     | Force Control Strategy . . . . .                                                                                              | 118        |
| 5.3.2     | System Modeling in Cartesian Space . . . . .                                                                                  | 119        |
| 5.4       | Parameter Identification of Displacement Model of Rigid Robot . . . . .                                                       | 123        |
| 5.4.1     | Identification using Transfer Function Estimation . . . . .                                                                   | 124        |
| 5.4.2     | Identification using Process Model Estimation . . . . .                                                                       | 126        |
| 5.5       | Design of Robust Force Controller . . . . .                                                                                   | 128        |
| 5.5.1     | Description of Entire Control System . . . . .                                                                                | 128        |
| 5.5.2     | Specifying Desired Performance of Controller . . . . .                                                                        | 129        |
| 5.5.3     | Designing Structure of Controller . . . . .                                                                                   | 130        |
| 5.5.4     | Calculating Parameters of the Controller . . . . .                                                                            | 132        |
| 5.5.5     | Results of Force Controller Design . . . . .                                                                                  | 134        |
| 5.6       | Simulation of Robotic FSW Process . . . . .                                                                                   | 135        |
| 5.6.1     | KUKA Robot Controller and Trajectory Generator . . . . .                                                                      | 135        |
| 5.6.2     | Dynamic Control of KUKA Robot . . . . .                                                                                       | 136        |
| 5.6.3     | Joint Motion Controller of the Simulator . . . . .                                                                            | 138        |
| 5.6.4     | Establishment of Simulator for Robotic FSW Process . . . . .                                                                  | 141        |
| 5.6.5     | Simulations and Results Analysis . . . . .                                                                                    | 144        |
| 5.7       | Vibration Analysis of Axial Force . . . . .                                                                                   | 147        |
| 5.7.1     | Identification of Experimental Axial Force Model . . . . .                                                                    | 147        |
| 5.7.2     | Disturbance Model for Force Vibration and Simulation . . . . .                                                                | 149        |
| 5.8       | Conclusion . . . . .                                                                                                          | 152        |
| <b>6</b>  | <b>General Conclusions and Perspectives</b>                                                                                   | <b>155</b> |
| 6.1       | General Conclusions . . . . .                                                                                                 | 155        |
| 6.2       | Perspectives . . . . .                                                                                                        | 158        |
| <b>II</b> | <b>Résumé Étendu en Français</b>                                                                                              | <b>159</b> |
| <b>7</b>  | <b>Résumé en Français: Modélisation d'un robot manipulateur en vue de la commande robuste en force utilisé en soudage FSW</b> | <b>161</b> |
| 7.1       | Introduction générale et revue de littérature . . . . .                                                                       | 161        |
| 7.2       | Modélisation des robots manipulateurs à articulations flexibles . . . . .                                                     | 176        |
| 7.3       | Simplification des modèles dynamiques des robots en utilisant la méthode d'intervalle . . . . .                               | 182        |
| 7.4       | Modélisation dynamique et identification du procédé de soudage FSW robotisé . . . . .                                         | 194        |
| 7.5       | Conception d'un contrôleur robuste en force pendant le procédé de soudage FSW robotisé . . . . .                              | 203        |
| 7.6       | Conclusions générales et perspectives . . . . .                                                                               | 212        |

---

|                                                                                                              |            |
|--------------------------------------------------------------------------------------------------------------|------------|
| <b>Bibliography</b>                                                                                          | <b>217</b> |
| <b>Appendix</b>                                                                                              | <b>229</b> |
| <b>A Supplementary Materials for Modeling of Robot</b>                                                       | <b>231</b> |
| A.1 Homogeneous Transformation Matrix ${}^0T_t$ . . . . .                                                    | 231        |
| A.2 Jacobian Matrix . . . . .                                                                                | 232        |
| <b>B Supplementary Materials for Model Simplification of Robot</b>                                           | <b>235</b> |
| B.1 Expression of Original Component $M_{21}$ and Simplified component $M_{s21}$ of Inertia Matrix . . . . . | 235        |
| B.2 Tables for choosing appropriate $k_t$ and $k_p$ for linear and circular FSW trajectories . . . . .       | 236        |
| <b>C Supplementary Materials for Design of Force Controller</b>                                              | <b>239</b> |
| C.1 Detailed Computation Procedures for Single-Stage Controller . . . . .                                    | 239        |
| <b>D Technical Documents of the Robot</b>                                                                    | <b>243</b> |
| D.1 Technical Documents of the Robot . . . . .                                                               | 243        |

# Acknowledgements

I would like to acknowledge everyone who has helped me with my doctoral studies in any kind of way throughout the past three years.

First and foremost, I would like to express my sincere gratitude to my thesis supervisor Dr. Gabriel Abba for providing me with the excellent opportunity to achieve this work. I am really grateful for his patient and continuous guidance, tremendous support and encouragement with my scientific research and professional development. His wisdom, enthusiasm and friendly attitude have inspired me to try my best to overcome any difficulty of my work. With the help of his extensive knowledge and countless suggestions, I have learned a great deal of things both for the research and life. I am also grateful for his invaluable help in proofreading the manuscript and assisting me in preparing my doctoral dissertation defense.

I am also extremely grateful to my co-supervisor Dr. François Léonard for affording me technical guidance, generous support, flexible and valuable supervising time as well as his admirable patience and strong confidence on me. He has always been supportive and has provided me scientific knowledge and advice, insightful discussions and helpful suggestions about my research. He has always encouraged me to think more independently and to get over the various obstacles. I have also appreciated the timely help that he has given to me, even in his vacation time, and the immense assistance in proofreading my manuscript and preparing my thesis defense.

I would also like to thank all the members of my PhD thesis committee: Dr. Guillaume Morel, Dr. Wolfgang Seemann, Dr. Stéphane Caro and Mr. Aurélien Robineau for their support, encouragement, insightful questions and constructive suggestions. It is my great honor to have such a high level jury for my PhD defense. I want to give my special thanks to Dr. Wolfgang Seemann and Dr. Stéphane Caro for spending their precious time on reviewing my manuscript carefully and giving back their insightful comments and constructive advice to improve the quality of my thesis. Dr. Guillaume Morel has chaired the doctoral thesis defense and has given me some helpful comments on my research which inspired me a lot. I am also honored to be able to invite the FSW expert Mr. Aurélien Robineau to participate in my thesis defense and he has also given me a lot of valuable suggestions. I am really grateful for the inestimable contributions of all these jury members.

I would like to acknowledge and extend my appreciation to the colleagues and friends in the LCFC: Patrick Martin, Régis Bigot, Laurant Langlois, Sandra Chevret, Jean-yves Dantan, Ali Siadat, Nafissa Lakbakbi Elyaaqoubi, Alain Etienne, Eric Becker, Stéphanie Schiappa, Véronique Ernest, Zakaria Allam, Simon Jung, Florian Baratto, Renaud Gimenez, Mathieu Hobon, Olivier Gyss, Damien Chevalier, Bruno Kaici, Thanh Hung Nguyen, Catalina Gutierrez, François Cristofari, Phillipe



Mayer, Fawzia Dardouri, Maryam Zouhri, Jinna Qin, Guochao Gu, Fan Li, Qing Xia, Peng Wang, Zhicheng Huang, Haitao Tian, Jianjie Zhang, Zhongkai Chen and all others who have also contributed to my advancement throughout my doctoral studies. I want to thank everyone in the laboratory for the inspiring environment and friendly atmosphere they are creating and for all the fun we have had together in the past several years. I am particularly grateful to Maryam Zouhri for her precious friendship, warm care and heartfelt support.

My cordial and sincere thanks go to all my Chinese friends with whom I have shared the wonderful and amazing time in Metz. Special thanks to Jinna Qin, Guochao Gu and Zhongkai Chen for assisting me to adapt to the life in France, and Jinna Qin has helped me a lot both for the research work and daily life. I also want to thank Fan Li, Qing Xia, Peng Wang, Zhicheng Huang, Haitao Tian, Jianjie Zhang, Xianqiong Zhao and Da Li for taking photos during my oral presentation and for preparing the delicious buffet after my doctoral defense. I love all my dear Chinese friends in Metz and I appreciate the pure and precious friendship between us which is the most valuable asset of my life and will not be forgotten forever.

I also owe a special debt of gratitude to all of the staff at ENSAM Metz and the doctoral school in Paris for giving me the immense help and generous support throughout my studies. Special thanks should go to the administrative crew of my laboratory: Stéphanie Schiappa and Véronique Ernest for their admirable patience and countless help in my research work. In addition, I am also indebted to the administrative personnel of the doctoral school: Florence Dumard, Claude Roy and the director Anne Bouteville for their timely assistance and warm encouragement.

The French National Agency of Research (ANR) and the China Scholarship Council (CSC) are gratefully acknowledged for the research project funding (under the project ANR-2010-SEGI-003-01-COROUSO) and the PhD scholarship for my daily life (reference number: 201206020028), without which I would not finish my doctoral study and would not obtain the PhD degree in France.

Last but not least, I would like to give the most heartfelt thanks to my family members: my parents Ruyi Wang and Zhuohui Ren; my wife Fang Liu; my grandparents Daohai Wang, Zhongcai Li, Wenqi Ren, Xiuqi Zhao; my uncle Anyi Wang, my aunt Ping Li, my brother Fei Wang, my little sister Zeyu Wang and other relatives in my beloved motherland China. My parents have sacrificed their lives for me and provided me with unconditioned deep love, admirable patience, selfless support and endless encouragement, without which I would not get to this day to finish my PhD study. Apart from my parents, I am also indebted to my wife Fang Liu for her immense patience, support, encouragement and love that she is constantly giving me without a word of complaint. I am also really grateful to all other relatives and friends for providing me with invaluable help, support and encouragement unconditionally throughout my life.

# List of Figures

|      |                                                                                                                                                                                                                             |    |
|------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| 1.1  | Industrial robots from 1959 to 1974 [9]                                                                                                                                                                                     | 19 |
| 1.2  | Industrial robots from 1974 to 1984 [9]                                                                                                                                                                                     | 20 |
| 1.3  | Industrial robots from 1985 to present [9]                                                                                                                                                                                  | 21 |
| 1.4  | Worldwide annual installations of industrial robots [12]                                                                                                                                                                    | 22 |
| 1.5  | Industrial robots launched by four different companies                                                                                                                                                                      | 22 |
| 1.6  | The outline and flow of information in this dissertation                                                                                                                                                                    | 30 |
|      |                                                                                                                                                                                                                             |    |
| 2.1  | Position description of a point in space                                                                                                                                                                                    | 32 |
| 2.2  | Position and orientation description of an object                                                                                                                                                                           | 33 |
| 2.3  | X-Y-Z fixed angles                                                                                                                                                                                                          | 34 |
| 2.4  | Description of modified Denavit-Hartenberg convention                                                                                                                                                                       | 35 |
| 2.5  | Robot KUKA KR500-2MT                                                                                                                                                                                                        | 36 |
| 2.6  | Geometric description of the robot                                                                                                                                                                                          | 37 |
| 2.7  | Schematic of the flexible joint model                                                                                                                                                                                       | 47 |
| 2.8  | Interface of the software SYMORO+                                                                                                                                                                                           | 47 |
|      |                                                                                                                                                                                                                             |    |
| 3.1  | The function $\mathbf{f}$ and its inclusion function $[\mathbf{f}]$ as well as minimal inclusion function $[\mathbf{f}]^*$ from $\mathbb{X}$ to $\mathbb{Y}$ ( $\mathbb{X}$ and $\mathbb{Y}$ correspond to $\mathbb{R}^2$ ) | 53 |
| 3.2  | The schematic of generation of the symbolic robot dynamic model                                                                                                                                                             | 55 |
| 3.3  | Diagram of the simplification algorithm process                                                                                                                                                                             | 57 |
| 3.4  | Flow chart of the simplification program                                                                                                                                                                                    | 59 |
| 3.5  | Histogram statistics of term norm ratio for $M_{44}$                                                                                                                                                                        | 61 |
| 3.6  | The tool position, joint angular position, velocity and acceleration along the identification trajectory (test IS0811A)                                                                                                     | 67 |
| 3.7  | Comparison between original component of $M(q)$ and simplified component of $M_s(q)$ along the identification trajectory (I)                                                                                                | 69 |
| 3.8  | Comparison between original component of $M(q)$ and simplified component of $M_s(q)$ along the identification trajectory (II)                                                                                               | 70 |
| 3.9  | Comparison between original component of $H(q, \dot{q})$ and simplified component of $H_s(q, \dot{q})$ along the identification trajectory                                                                                  | 71 |
| 3.10 | Comparison between original torque $T$ and simplified torque $T_s$ along the identification trajectory                                                                                                                      | 71 |
| 3.11 | The tool position, joint angular position, velocity and acceleration along the linear FSW trajectory (test IS1709J)                                                                                                         | 72 |
| 3.12 | Pareto diagram: choose the most appropriate proportional factors for simplifying the inertia matrix $M(q)$ in the linear FSW case based on the criteria of accuracy and simplicity                                          | 73 |

|      |                                                                                                                                                                                                                       |     |
|------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|
| 3.13 | Comparison between original component of $M(q)$ and simplified component of $Ms(q)$ along the linear FSW trajectory (I) . . . . .                                                                                     | 75  |
| 3.14 | Comparison between original component of $M(q)$ and simplified component of $Ms(q)$ along the linear FSW trajectory (II) . . . . .                                                                                    | 75  |
| 3.15 | Comparison between original component of $H(q, \dot{q})$ and simplified component of $Hs(q, \dot{q})$ along the linear FSW trajectory . . . . .                                                                       | 76  |
| 3.16 | Comparison between original torque $T$ and simplified torque $Ts$ along the linear FSW trajectory . . . . .                                                                                                           | 76  |
| 3.17 | The tool position, joint angular position, velocity and acceleration along the circular FSW trajectory (test IS1006P) . . . . .                                                                                       | 77  |
| 3.18 | Comparison between original component of $M(q)$ and simplified component of $Ms(q)$ along the circular FSW trajectory (I) . . . . .                                                                                   | 79  |
| 3.19 | Comparison between original component of $M(q)$ and simplified component of $Ms(q)$ along the circular FSW trajectory (II) . . . . .                                                                                  | 79  |
| 3.20 | Comparison between original component of $H(q, \dot{q})$ and simplified component of $Hs(q, \dot{q})$ along the circular FSW trajectory . . . . .                                                                     | 80  |
| 3.21 | Comparison between original torque $T$ and simplified torque $Ts$ along the circular FSW trajectory . . . . .                                                                                                         | 80  |
| 3.22 | Measured currents of all the motors during the linear FSW process (test 1709J) . . . . .                                                                                                                              | 82  |
| 3.23 | External deformation compensation of KUKA robot [73] . . . . .                                                                                                                                                        | 83  |
| 4.1  | Schematic drawing of friction stir welding . . . . .                                                                                                                                                                  | 87  |
| 4.2  | Friction stir welding experiment and its three main stages [2] . . . . .                                                                                                                                              | 88  |
| 4.3  | Measurements of the forces and torque applied on the tool during the FSW process implemented by an instrumented MTS-ISTIR-10 Friction Stir Welder on 6000 aluminum alloy with a welding depth of 6 mm [111] . . . . . | 89  |
| 4.4  | Experimental setup: FSW system based on robot KUKA KR500-2MT (by courtesy of Institut de Soudure, Goin, France) . . . . .                                                                                             | 91  |
| 4.5  | The desired welding trajectory in 3D space, measured (or calculated) joint angular position, velocity and acceleration of motor axes after the gear reduction along the welding trajectory (test IS1505I) . . . . .   | 92  |
| 4.6  | Measured currents of all the motors during the FSW test (test IS1505I) . . . . .                                                                                                                                      | 93  |
| 4.7  | Signal flow schematic of the experimental testbed . . . . .                                                                                                                                                           | 93  |
| 4.8  | Diagram of the discrete observer for robot state estimation [73] . . . . .                                                                                                                                            | 94  |
| 4.9  | Diagram of data acquisition and processing for model identification in the external observer-based test of KUKA robot . . . . .                                                                                       | 95  |
| 4.10 | Schematic of definition of the plunge depth $d_p$ . . . . .                                                                                                                                                           | 96  |
| 4.11 | Schematic of the flexibility model in Cartesian space . . . . .                                                                                                                                                       | 98  |
| 4.12 | Comparison between filtered and original measured axial force $F_z$ during the FSW process (test IS1505I) . . . . .                                                                                                   | 100 |
| 4.13 | Comparison between filtered and calculated plunge depth $d_p$ and its derivative $\frac{d}{dt}d_p$ (Method 1) . . . . .                                                                                               | 101 |
| 4.14 | Comparison between filtered and calculated plunge depth $d_p$ and its derivative $\frac{d}{dt}d_p$ (Method 2) . . . . .                                                                                               | 102 |

|      |                                                                                                                                                       |     |
|------|-------------------------------------------------------------------------------------------------------------------------------------------------------|-----|
| 4.15 | Comparison between filtered and calculated plunge depth $d_p$ and its derivative $\frac{d}{dt}d_p$ (Method 3) . . . . .                               | 103 |
| 4.16 | Free-body diagram of simple lumped parameter model in $z$ direction .                                                                                 | 103 |
| 4.17 | Comparison between filtered and fitted axial force $F_z$ and generated error for Linear Model 1 . . . . .                                             | 107 |
| 4.18 | Comparison between filtered and fitted axial force $F_z$ and generated error for Linear Model 2 . . . . .                                             | 108 |
| 4.19 | Comparison between filtered and fitted axial force $F_z$ and generated error for Linear Model 3 . . . . .                                             | 108 |
| 4.20 | Histogram of error and estimated Gaussian for Linear Model 1 with data filtering . . . . .                                                            | 109 |
| 4.21 | Histogram of error and estimated Gaussian for Linear Model 2 with data filtering . . . . .                                                            | 109 |
| 4.22 | Histogram of error and estimated Gaussian for Linear Model 3 with data filtering . . . . .                                                            | 110 |
| 4.23 | Comparison between filtered and fitted axial force $F_z$ and generated error for the nonlinear model . . . . .                                        | 113 |
| 4.24 | Histogram of error and estimated Gaussian for the nonlinear model with data filtering . . . . .                                                       | 113 |
| 5.1  | Inner position/outer force control scheme for industrial robot KUKA                                                                                   | 119 |
| 5.2  | System modeling in Cartesian space . . . . .                                                                                                          | 120 |
| 5.3  | Time history of model parameters during circular FSW process (test IS1006P) . . . . .                                                                 | 123 |
| 5.4  | Measured and fitted displacements in $x$ direction . . . . .                                                                                          | 125 |
| 5.5  | Measured and fitted displacements in $y$ direction . . . . .                                                                                          | 125 |
| 5.6  | Measured and fitted displacements in $z$ direction . . . . .                                                                                          | 126 |
| 5.7  | Bode diagram of three third-order transfer function models . . . . .                                                                                  | 126 |
| 5.8  | Measured and fitted displacements in $x$ direction . . . . .                                                                                          | 127 |
| 5.9  | Measured and fitted displacements in $y$ direction . . . . .                                                                                          | 127 |
| 5.10 | Measured and fitted displacements in $z$ direction . . . . .                                                                                          | 127 |
| 5.11 | Bode diagram of three kinds of models in $z$ direction . . . . .                                                                                      | 127 |
| 5.12 | Block diagram of the axial force control system . . . . .                                                                                             | 128 |
| 5.13 | Nichols chart of the the open-loop transfer function excluding the rest part of force controller to be designed $G_{FC}^*(s)$ (Phase-lead-lag case) . | 132 |
| 5.14 | Bode diagram of three force controllers . . . . .                                                                                                     | 135 |
| 5.15 | Bode diagram of the open loop systems with three force controllers .                                                                                  | 135 |
| 5.16 | Schematic diagram of the position control of KUKA robot for FSW process (without the force control) . . . . .                                         | 136 |
| 5.17 | Diagram of the continuous-time joint motion controller of the simulator                                                                               | 139 |
| 5.18 | Diagram of the discrete-time joint motion controller of the simulator .                                                                               | 140 |
| 5.19 | Diagram of the main models in the simulator . . . . .                                                                                                 | 142 |
| 5.20 | Three-dimensional model of robot KUKA in the FSW process . . . .                                                                                      | 142 |
| 5.21 | Schematic diagram of the simulator in Matlab/Simulink environment                                                                                     | 143 |
| 5.22 | Comparison of the axial forces during the simulations with three different force controllers . . . . .                                                | 145 |

---

|      |                                                                                                                                                                                                                                     |     |
|------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|
| 5.23 | Desired tool velocity in Cartesian space used in the simulation . . . .                                                                                                                                                             | 145 |
| 5.24 | Comparison of the desired tool position and the actual tool position<br>in 3D space during the simulation with only force control in $z$ direction                                                                                  | 146 |
| 5.25 | Comparison of the desired tool position and the actual tool position<br>in X-Y plane during the simulation with only force control in $z$ direction                                                                                 | 146 |
| 5.26 | Comparison of the desired tool position and the actual tool position<br>in X-Z plane during the simulation with only force control in $z$ direction                                                                                 | 147 |
| 5.27 | Measured axial force during friction stir welding process (test IS1709J:<br>$\Omega = 1100 \text{ tr}/\text{min}$ , $v = 400 \text{ mm}/\text{min}$ , welding depth $6 \text{ mm}$ , material<br><i>Al6082</i> ) . . . . .          | 148 |
| 5.28 | Spectrum of the axial force in FSW process (test IS1709J) . . . . .                                                                                                                                                                 | 148 |
| 5.29 | Welded workpiece after the FSW process (test IS1709J) . . . . .                                                                                                                                                                     | 149 |
| 5.30 | Measured and fitted $z$ coordinates of the weld surface and correspond-<br>ing $x$ coordinates (test 1709J, $z$ coordinates denotes the height of weld<br>surface and $x$ coordinated denotes the position along welding direction) | 150 |
| 5.31 | Height variations of the surface of welded workpiece $\Delta h$ and corre-<br>sponding $x$ coordinates (test 1709J) . . . . .                                                                                                       | 150 |
| 5.32 | Spectrum of height variation of the weld surface in FSW process (test<br>1709J) . . . . .                                                                                                                                           | 151 |
| 5.33 | Simulation of robotic FSW process with the disturbance model (The<br>case of using fast type force controller) . . . . .                                                                                                            | 152 |
| D.1  | Technical data of robot KR500-2MT . . . . .                                                                                                                                                                                         | 243 |
| D.2  | Principle dimensions and working envelope (software values) for robot<br>KR500-2MT . . . . .                                                                                                                                        | 244 |

# List of Tables

|      |                                                                                                                                                                            |    |
|------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| 2.1  | Geometric parameters of robot model . . . . .                                                                                                                              | 37 |
| 3.1  | Values of the robot geometric parameters . . . . .                                                                                                                         | 55 |
| 3.2  | Range of robot motion, velocity and acceleration . . . . .                                                                                                                 | 56 |
| 3.3  | Values of the inertia parameters used in the simplification . . . . .                                                                                                      | 59 |
| 3.4  | The ratio of $ T_i $ to $ T_i _{max}$ . . . . .                                                                                                                            | 60 |
| 3.5  | Norm of term $T_{10}$ and error $e_p(\%)$ when $P = 0$ . . . . .                                                                                                           | 61 |
| 3.6  | Norms of all generated components during simplification process . . . . .                                                                                                  | 61 |
| 3.7  | Number of terms in $M(q)$ and $Ms(q)$ when $k_t = 5\%$ and $k_p = 1\%$ . . . . .                                                                                           | 62 |
| 3.8  | Number of terms in $H(q, \dot{q})$ and $Hs(q, \dot{q})$ when $k_t = 5\%$ and $k_p = 1\%$ . . . . .                                                                         | 62 |
| 3.9  | Negligible Inertia Parameters for $M(q)$ and $H(q, \dot{q})$ . . . . .                                                                                                     | 63 |
| 3.10 | Number of terms in components of $M(q)$ and $Ms(q)$ as well as RMS errors ( $e_{RMS}$ ) when $k_t = 5\%$ and $k_p = 1\%$ . . . . .                                         | 64 |
| 3.11 | The ratio of $ M_{1i}\ddot{q}_i $ to $ M_{1i}\ddot{q}_i _{max}$ in $W_1$ . . . . .                                                                                         | 65 |
| 3.12 | Results of further simplification for $M(q)\ddot{q}$ . . . . .                                                                                                             | 65 |
| 3.13 | Actual intervals of robot states for the identification trajectory . . . . .                                                                                               | 67 |
| 3.14 | Number of terms in the simplified components and corresponding RMS errors along the identification trajectory when choosing different $k_t$ and $k_p$ . . . . .            | 68 |
| 3.15 | Number of terms in components of $M(q)$ and $Ms(q)$ as well as RMS errors along the identification trajectory when $k_t = 3\%$ and $k_p = 1\%$ . . . . .                   | 69 |
| 3.16 | Number of terms in components of $H(q, \dot{q})$ and $Hs(q, \dot{q})$ as well as RMS errors along the identification trajectory when $k_t = 3\%$ and $k_p = 1\%$ . . . . . | 70 |
| 3.17 | Actual intervals of robot states for the linear FSW trajectory . . . . .                                                                                                   | 72 |
| 3.18 | Number of terms in components of $M(q)$ and $Ms(q)$ as well as RMS errors along the linear FSW trajectory when $k_t = 1\%$ and $k_p = 1\%$ . . . . .                       | 74 |
| 3.19 | Number of terms in components of $H(q, \dot{q})$ and $Hs(q, \dot{q})$ as well as RMS errors along the linear FSW trajectory when $k_t = 1\%$ and $k_p = 1\%$ . . . . .     | 74 |
| 3.20 | Actual intervals of robot states for the circular FSW trajectory . . . . .                                                                                                 | 77 |
| 3.21 | Number of terms in components of $M(q)$ and $Ms(q)$ as well as RMS errors along the circular FSW trajectory when $k_t = 1\%$ and $k_p = 1\%$ . . . . .                     | 78 |
| 3.22 | Number of terms in components of $H(q, \dot{q})$ and $Hs(q, \dot{q})$ as well as RMS errors along the circular FSW trajectory when $k_t = 1\%$ and $k_p = 1\%$ . . . . .   | 78 |
| 3.23 | Intervals of torques in robot dynamic equation (axis $i$ ) . . . . .                                                                                                       | 81 |

---

|      |                                                                                                                                                                     |      |
|------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------|------|
| 3.24 | The ratio $p_T(\%)$ of the norm of all torques to the maximum one . . .                                                                                             | 82   |
| 4.1  | Robotic FSW experimental conditions . . . . .                                                                                                                       | 92   |
| 4.2  | Errors of identified models . . . . .                                                                                                                               | 106  |
| 4.3  | Value of identified parameters and their uncertainties for linear model                                                                                             | 1106 |
| 4.4  | Value of identified parameters and their uncertainties for linear model                                                                                             | 2106 |
| 4.5  | Value of identified parameters and their uncertainties for linear model                                                                                             | 3106 |
| 4.6  | Errors of identified nonlinear model . . . . .                                                                                                                      | 112  |
| 4.7  | Value of identified parameters and their uncertainties for nonlinear<br>model . . . . .                                                                             | 112  |
| 5.1  | Parameter uncertainties, fit values and RMS errors of the identified<br>transfer function models in three directions expressed in percentage<br>(%) . . . . .       | 125  |
| 5.2  | Identified parameters of second-order process models in three directions                                                                                            | 127  |
| 5.3  | Parameter uncertainties, fit values and RMS errors of the identified<br>second-order process models in three directions expressed in percent-<br>age (%) . . . . .  | 127  |
| 5.4  | The performance type and corresponding rise time as well as the<br>surface type and corresponding desired phase margin . . . . .                                    | 130  |
| 5.5  | Actual values of gains of robot joint motion controller . . . . .                                                                                                   | 140  |
| 5.6  | Performance comparison of three force controllers in Simulation . . .                                                                                               | 144  |
| B.1  | Number of terms in the simplified components and corresponding<br>RMS errors along the linear FSW trajectory when choosing different<br>$k_t$ and $k_p$ . . . . .   | 237  |
| B.2  | Number of terms in the simplified components and corresponding<br>RMS errors along the circular FSW trajectory when choosing different<br>$k_t$ and $k_p$ . . . . . | 238  |

# List of Abbreviations

| <b>Abbreviations</b> | <b>Definitions</b>                                                                                                     |
|----------------------|------------------------------------------------------------------------------------------------------------------------|
| FSW                  | Friction Stir Welding (Soudage par frottement malaxage)                                                                |
| LCFC                 | Laboratoire de Conception Fabrication Commande (in French)<br>Design Engineering, Manufacturing and Control Laboratory |
| CSC                  | China Scholarship Council                                                                                              |
| ANR                  | Agence Nationale de la Recherche (in French)                                                                           |
| COROUSO              | Modélisation et Commande de RObots d’USinage de pièce<br>composites de grandes dimensions et de SOudage FSW            |
| IS                   | Institut de Soudure à Goin, France (in French)                                                                         |
| IFR                  | International Federation of Robotics                                                                                   |
| AMF                  | American Machine and Foundry                                                                                           |
| PUMA                 | Programmable Universal Machine for Assembly                                                                            |
| SCARA                | Selective Compliance Assembly Robot Arm                                                                                |
| WiTP                 | Wireless Teach Pendant                                                                                                 |
| LWR                  | Light Weight Robot                                                                                                     |
| LCR                  | Learning Control Robot                                                                                                 |
| LVC                  | Learning Vibration Control                                                                                             |
| SPT                  | Singular Perturbation Technique                                                                                        |
| PD                   | Proportion Differentiation                                                                                             |
| PID                  | Proportion Integration Differentiation                                                                                 |
| DOF                  | Degree of Freedom (degrés de liberté, ddl in French)                                                                   |
| DGM                  | Direct Geometric Model                                                                                                 |
| IGM                  | Inverse Geometric Model                                                                                                |
| DKM                  | Direct Kinematic Model                                                                                                 |
| IKM                  | Inverse Kinematic Model                                                                                                |
| DDM                  | Direct Dynamic Model                                                                                                   |
| IDM                  | Inverse Dynamic Model                                                                                                  |
| SYMORO+              | SYmbolic MOdelling of Robots plus                                                                                      |
| MDH                  | Modified Denavit-Hartenberg convention                                                                                 |
| RMS                  | Root Mean Square                                                                                                       |
| XML                  | Extensible Markup Language                                                                                             |
| LS                   | Least Square method                                                                                                    |
| RLS                  | Recursive Least Square method                                                                                          |
| DFT                  | Discrete Fourier Transform                                                                                             |
| FFT                  | Fast Fourier Transform                                                                                                 |



|     |                                                |
|-----|------------------------------------------------|
| TWI | The Welding Institute (UK)                     |
| IRA | Robot Institute of America                     |
| ISO | International Organization for Standardization |

**Part I**  
**English Version**



# Chapter 1

## General Introduction and Literature Review

### 1.1 Research Background and Motivations

Friction stir welding (FSW) is a comparatively new and promising solid state welding technology developed by The Welding Institute (TWI) of UK in 1991 [1]. Compared to conventional fusion welding technologies, FSW is mainly characterized by joining material without reaching the fusion temperature, avoiding the problems caused by melting metals, such as hot cracking, solidification defects and loss of weld mechanical properties [2]. Almost all types of aluminum alloys can be welded through FSW process. Other advantages of FSW includes: energy efficiency, environment friendliness, versatility, excellent weld strength and ductility, low residual stresses, no use of filler metal, relatively high welding speed and repeatability [3].

Due to the above-mentioned benefits, the FSW technology has been widely applied in various industrial applications. Particularly in aeronautic and astronautic industry, this technology is massively used for weight reduction. However, the fact is that the FSW process is generally carried out by some specially developed machines which need a great investment. Consequently, new technologies are required to reduce the cost and improve the productivity, flexibility and quality.

In recent years, robot manipulators have been increasingly used in various industrial applications, such as assembling, spray painting, material handling, machining and welding tasks. As for manufacturing processes, current researches focus on replacing the dedicated machines by industrial robots due to their large working space and low cost. Although the welding performance can be greatly improved by using the industrial manipulators as tool holders, the robotization of FSW process is still an extremely challenging task at present. The limited force capability and stiffness of industrial robots are barriers for expansion of robotic FSW.

During the robotic FSW process, the flexibility of the robot needs to be considered since an extremely strong external force is exerted on the robot and a non-negligible deformation is generated. Thus, for most industrial robots applied to FSW process, a high degree of accuracy and real-time performance can not be achieved. Many studies about industrial robot control have been performed to ensure a better tracking performance. In the research work presented by Qin [4], the tracking

accuracy can be enhanced by using the observer-based compensator. Due to the complicated control structure, a great deal of calculation and memory occupation will be implemented in robot controller, affecting the real-time performance. Besides, the robust control of axial force is also required for good weld quality.

This research work is in the framework of the project COROUSSO of the program ARPEGE sponsored by the French National Agency for Research (ANR). This project aims at realizing the machining of composite materials and the FSW process with industrial robots. In this thesis, the objective is to realize the robotic FSW process with a good tracking accuracy, robust force control and real-time performance, which can be listed in detail as follows:

- 1) Proposition of a new simplification method for robot dynamic model for purpose of improving real-time performance and related case studies;
- 2) Dynamic modeling and identification of the robotic FSW process for the sake of designing the force controller of the robot;
- 3) Design of a robust force controller for industrial robots used in FSW process;
- 4) Development of a simulation system of robotic FSW process and corresponding simulations for analyzing the performance of the force controller;
- 5) Vibration analysis of the axial force in the robotic FSW process.

## 1.2 Literature Review on Industrial Robot Manipulators

According to the official definition of the Robot Institute of America (RIA), a robot is a reprogrammable multifunctional manipulator designed to move materials, parts, tools or specialized devices through variable programmed motions for the performance of a variety of tasks [5]. Furthermore, the industrial robot is also defined by the International Organization for Standardization (ISO) as an automatically controlled, reprogrammable, multipurpose manipulator programmable in three or more axes, which may be either mounted on fixed objects or moving objects for use in industrial automation applications [6]. The typical applications for industrial robots include welding, painting, assembly, pick and place (e.g., packaging, palletizing), product inspection, and testing, etc.

### 1.2.1 History of Industrial Robot Manipulators

The modern history of robotic manipulation dates from the late 1940s. In order to protect technicians handling radioactive materials, the first servoed electric-powered teleoperator was developed in 1947 [7]. Scientists initiated the research on a numerically controlled milling machine almost at the same time [8]. These two important technologies laid a solid foundation for the developments of industrial robots. On the basis of the report of International Federation of Robotics (IFR), a brief history of the industrial robot technology is given below [9].

In 1954, George Devol applied for the first robot patent called "Programmed Article Transfer" [9]. Based on this patent, George Devol and Joe Engelberger established the first robot company known as Unimation in 1956, and developed the world's first industrial robot Unimate in 1959. Two years later, the hydraulically powered robot Unimate was firstly installed in a General Motors (GM) automobile factory in New Jersey to extract die castings [8]. In 1962, the first cylindrical robot Versatran was introduced to the Ford factory (Canton, USA) by American Machine and Foundry (AMF). Stanford Arm invented by Victor Scheinman in 1969 was the first six-axis, electrically powered, computer-controlled robot arm, which became a standard in the design of industrial robots [10].

Industrial robots were developed for new applications as well: GM installed the first spot-welding robot at its assembly plant; Norwegian company Trallfa launched the first commercial robot for spray painting during a domestic labor shortage; Japanese company Hitachi developed the automatic bolting robot which was the first industrial robot with dynamic vision sensors for moving objects; another Japanese company Kawasaki developed the first arc welding robot Hi-T-Hand. In 1973, German company KUKA developed its own robot Famulus, the first industrial robot with six electromechanically driven axes. Cincinnati Milacron introduced the first minicomputer-controlled industrial robot T3 to market (see Figure 1.1).

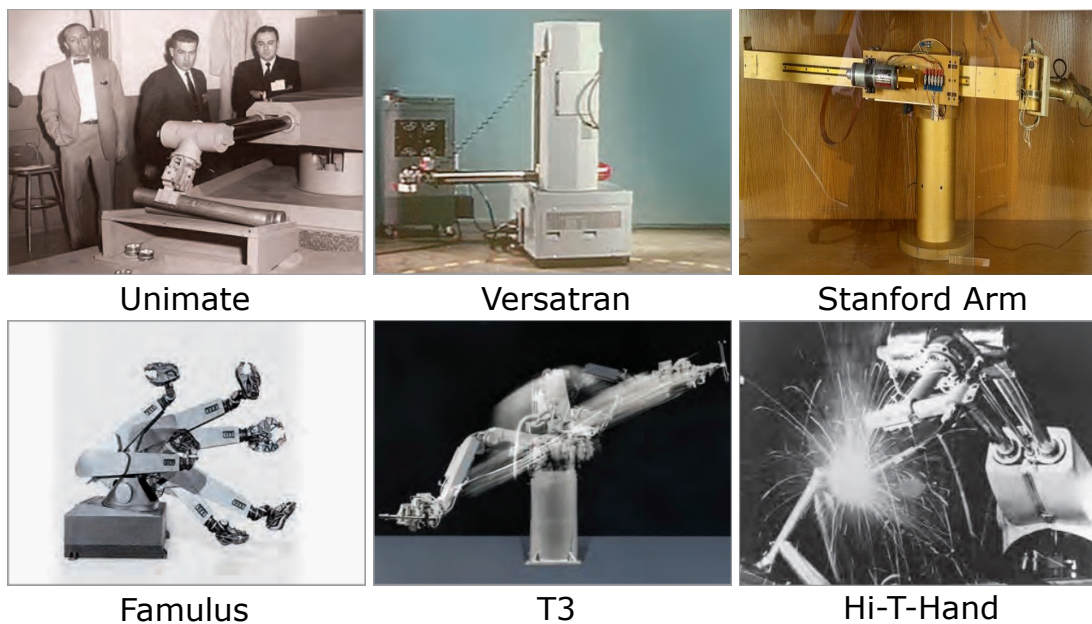


Figure 1.1: Industrial robots from 1959 to 1974 [9]

In 1974, Swedish company ASEA developed the first fully electric, microprocessor-controlled industrial robot IRB-6, which used an Intel 8 bit microprocessor and was delivered to a mechanical engineering company in Sweden for polishing pipe bends. In Italy, the Olivetti SIGMA robot was firstly used for assembly operations with two hands. In 1978, American company Unimation/Vicarm created the PUMA (Programmable Universal Machine for Assembly) robot with support from General Motors. In the same year, the SCARA (Selective Compliance Assembly Robot Arm) robot was designed by Hiroshi Makino at University of Yamanashi in Japan. The

robot arm is slightly compliant in X-Y direction but rigid in Z direction with the help of its parallel-axis joint layout. German company Reis Robotics presented the first six-axis robot with own control system RE15 for loading and unloading of die casting parts into trim presses. In 1979, Japanese company Nachi developed the first motor-driven robot for spot welding.

The industrial robot industry began its rapid growth from 1980. The world's first direct drive arm was built by Takeo Kanade at Carnegie Mellon University in 1981. The motors were installed directly into the robot joints. Due to the removal of transmission mechanisms between the motors and loads, the direct drive arm can move smoothly and freely, making it faster and much more accurate than previous robots. In the same year, Par Systems introduced its first industrial gantry robot which can provide a much larger range of motion than pedestal robots. In 1984, Adept Technology successfully marketed the first direct drive SCARA robot AdeptOne, which is extremely robust during manufacturing processes while maintaining high precision by virtue of the simplicity of the mechanism (see Figure 1.2).

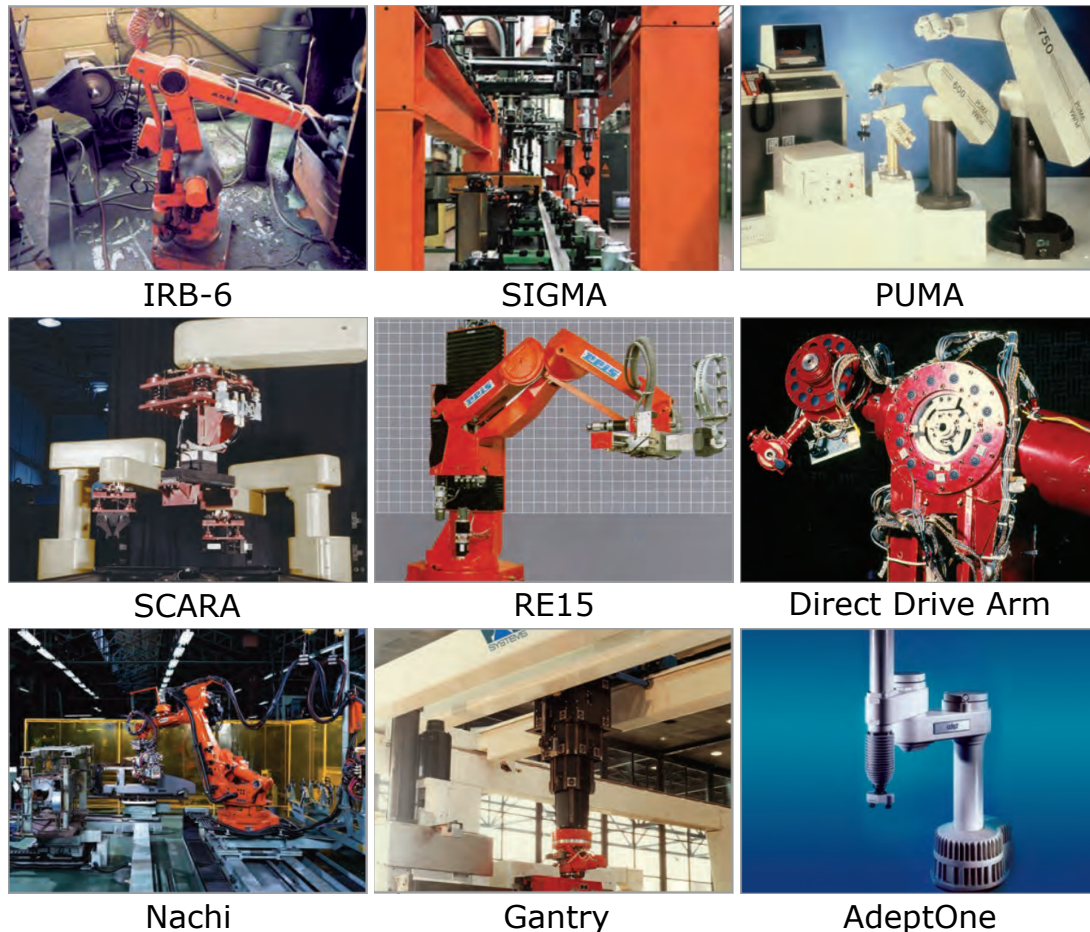


Figure 1.2: Industrial robots from 1974 to 1984 [9]

In the early 1980s, the Delta robot (a type of parallel robot) was invented by Reymond Clavel at the Federal Institute of Technology of Lausanne (EPFL, Switzerland) to meet the industrial need of manipulating small and light objects at a very high speed [11]. After purchasing a license for the Delta robot, the Swiss company

Demaurex sold its first Delta robot in packaging industry market in 1992. Based on the Delta robot and image technology, ABB company developed the world's fastest picking robot FlexPicker in 1998. The robot can pick 120 objects a minute or pick and release at a speed of 10  $m/s$ . In 1999, Reis Robotics introduced the integrated laser beam guiding within the robot arm and launched RV6L-CO2 laser robot, which permits the robot user to operate laser device at high dynamics.

The robot control system also achieved a great development. In 2004, Japanese company Motoman introduced the improved robot control system NX100, which is able to control four robots synchronously (up to 38 axis). Two years later, Italian robot company Comau launched the first Wireless Teach Pendant (WiTP), enabling robot data communication to be implemented without a cable connection.

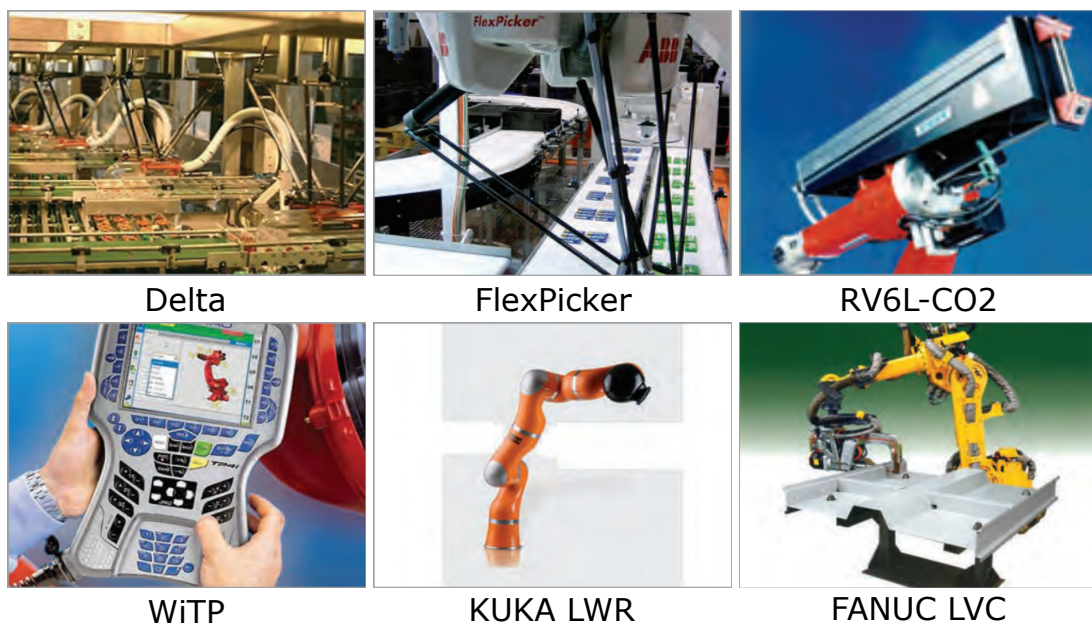


Figure 1.3: Industrial robots from 1985 to present [9]

In 2006, KUKA company presented the first Light Weight Robot (LWR), which weighs only 16  $kg$  but has a payload capacity of 7  $kg$ . The robot is portable and energy-efficient and can carry out a great variety of tasks. The first Learning Control Robot (LCR) was launched by FANUC in 2010. The robot can learn its vibration characteristics through Learning Vibration Control (LVC), and suppress the vibration to reduce the cycle time of robot motion (see Figure 1.3).

### 1.2.2 Current Developments of Industrial Robots

According to the IFR report of World Robotics in 2014, the total worldwide stock of operational industrial robots reached at least 1332000 units at the end of 2013. About 52% of the industrial robots are in Asia, 29% in Europe, and 17% in North America. It is estimated that global robot installations will increase to 1946000 by the end of 2017 [12].

From 2003 to 2013, the worldwide annual installations of industrial robots are summarized in Figure 1.4. Due to the improved global economic situation, the



robot sales in 2013 increased by 12 % to 178132 units, which is the highest number of industrial robots ever sold. In 2013, most of the robot sales (about 70%) were in five countries: China, Japan, the United States, South Korea and Germany. It should be noted that China became the world's biggest and fastest growing robot market with a share of 20% of the total supply. Industrial robot sales to almost all industries gained strong growth, including automotive, electronics, metal, rubber and plastics industries as well as food industry.

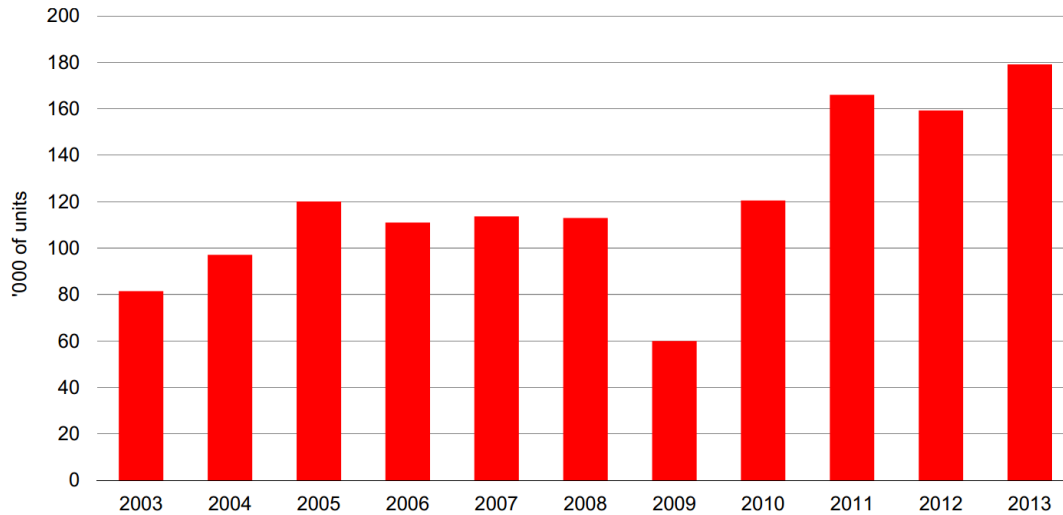


Figure 1.4: Worldwide annual installations of industrial robots [12]

The worldwide market value for industrial robots was up to 9.5 billion US dollars in 2013. Including the cost of software, peripherals and systems engineering, the turnover for robot systems was estimated to be 29 billion US dollars. Most of the industrial robots around the world are designed and made by several famous and influential industrial robot companies, such as ABB Robotics, FANUC Robotics, KUKA Robotics, Yaskawa Electric and Kawasaki Robotics, etc.



Figure 1.5: Industrial robots launched by four different companies

Figure 1.5 shows the industrial robots with high payloads which can carry out a wide range of manufacturing processes. The information about the various industrial robots launched by the four companies can be found in [13, 14, 15, 16].

### 1.3 Literature Review on Modeling of Flexible Joint Robot Manipulators

In most cases, the heavy industrial robots can be considered as rigid since the force exerted on the robot during the manufacturing process is not very large. Consequently, the deformation of the robot due to its flexibility can be neglected. However, the flexibility must be taken into account when the robot carries out some applications which need large working force.

As for industrial manipulators, the flexibility can be supposed as concentrated at the robot joints or distributed along the robot links in different ways from a modeling point of view [7]. According to the research work of Dumas [17], the main sources of flexibilities for the serial industrial robots are the flexibilities located at the robot joints (including motors and transmissions). Based on this study, the links of industrial robots used in this research should be considered as rigid. The modeling and control problems of industrial manipulators with flexible joints will be mainly concerned hereafter.

In fact, the joint flexibility of the current industrial robots comes from the motion transmission/reduction elements, such as belts, cables, long shafts, harmonic drives and cycloidal gears. These components are utilized to allow relocation of the actuators next to the robot base, improving dynamic efficiency. However, these components are intrinsically flexible when subject to the working force during the robot operation [7].

The early study of the problems caused by flexible transmissions in robot manipulators can be found in [18, 19], with first experimental findings on the GE P-50 arm. In 1987, Spong proposed a simplified flexible joint model and considered the joint flexibility as a linear torsional spring with a stiffness constant [20, 21]. Based on the dynamic model of Spong, Khorasani established a single-link rigid-flexible coupled model, in which the influence of the damping term and harmonic gear is taken into account [22].

The simplified flexible joint model proposed by Spong is valid only in the case of large gear ratios because the cross inertial effects could be ignored in that situation. Thus, Tomei proposed a complete flexible joint model by including the couplings between the links and the motors [23].

The precision of a robot model can be greatly influenced by the nonlinear characteristics of the gear transmission, which mainly contain the friction and the nonlinear stiffness. It should be noted that the nonlinear stiffness can be also included in the flexible joint model by replacing the linear stiffness formulation with a nonlinear function describing a nonlinear spring. More nonlinearities can be added to the flexible joint dynamic model, such as backlash, hysteresis and nonlinear damping [24]. A relatively precise flexible joint robot dynamic model was proposed by Bridges and proved to be stable. In this model, the friction, nonlinear flexibility, and kinematic errors are taken into consideration [25].

Moberg proposed an extended flexible joint model where the elasticity is described by a number of localized multi-dimensional spring-damper pairs [24]. This spring-damper pair is able to have up to six degrees of freedom and includes both translational and rotational deflection.

## 1.4 Literature Review on Control of Flexible Joint Robots

Generally, the types of commonly used control laws contain: linear or nonlinear, static or dynamic, feedback dominant or feedforward dominant, discrete-time or continuous-time, robust or adaptive, diagonal or full matrix, etc.

Early researchers mainly focused on studying the impact of motor dynamics on control performance in the field of robot control. However, in 1985 Sweet and Good pointed out that joint flexibility should be taken into account in both modeling and control if high tracking performance is required [18]. In 1987, Spong proposed a simplified flexible joint model [20]. Since then a lot of theoretical and experimental research had been performed on controlling flexible joint robots: singular perturbation and integral manifold, feedback linearization, cascaded system and integral backstepping, PD control, adaptive control, robust control, neural networks, fuzzy control and some other control methods.

### 1.4.1 Singular Perturbation and Integral Manifold

Flexible joint robot can be considered as a weak elasticity system if its joint stiffness is large enough. This feature makes it possible that we transform the flexible joint model into a two-time-scale (FTS: Fast Time Scale and STS: Slow Time Scale) system by using singular perturbation technique. In system dynamics, singular perturbation technique (SPT) is widely applied to the system which can be divided into a fast subsystem and a slow subsystem. With the advantage of model reduction, this technique can decompose a high-order system into two lower-order systems. The core idea is based on adding a simple correction term to the control law for rigid body robots to damp out the elastic oscillation at the joints. However, the approach only can be used in robot systems with weak joint flexibility, where the flexible dynamics is much faster than the rigid-body dynamics.

Singular perturbation model for the same system is not unique. Spong presented a control design modeling the joint elastic forces as the fast variables and the link variables as the slow variables [21]. Ge proposed a new adaptive controller based on SPT and using only position and velocity feedback by modeling the motor tracking error as the fast variables and the link variables as the slow variables [26]. As a consequence, the slow time-scale dynamics and the resulting control laws are also different from those given by Spong.

However, the dynamical systems obtained by decomposition based on SPT are extremely complex to control, and a need for order reduction and simplification is inevitable. In order to solve this problem, some researchers introduced the concept of integral manifold. The integral manifold is a tool for reduced-order modeling because if the system trajectories are on the manifold, then a  $2n^{th}$ -order slow model is an exact representation of the  $4n^{th}$ -order system [27]. This fact will be used to construct corrected reduced-order models. By using this idea to represent the dynamics of the slow subsystem, Spong derived a reduced-order model of the manipulator which incorporates the effects of the joint flexibility on the system [21].

After obtaining the reduced-order systems, various control methods can be directly applied, such as robust control [28], fuzzy control [29], etc. Ghorbel and Spong first proposed an adaptive composite control strategy for flexible joint robot manipulators, which is also called fast/slow control strategy consisting of a slow adaptive controller designed for a rigid robot together with a fast control to damp the elastic oscillations of the joints [30], and gave its local stability analysis [31]. Compared with the method of feedback linearization, this approach, which is based on integral manifold, can overcome the need for feedback of link acceleration and jerk, but lacks of robustness to parametric uncertainty. Then they extended the integral manifold approach from the known parameter case to the adaptive one [32], and clear drawbacks of this adaptive integral manifold is its complexity of expression derivation and computational burden.

Some researchers also studied the robustness issue of control method based on singular perturbation and integral manifold. Taking the unmodeled dynamics and parameter vibrations into account, Khrasani corrected the perturbed equations, and obtained a reduced-order flexible model without parametric uncertainty so that the feedback linearization and robust adaptive control strategies could be applied to control flexible joint robots [33]. Al-Ashoor developed a robust adaptive controller, which was designed to compensate for the effects of unmodeled dynamics and the errors arising from parameter vibrations and neglecting higher order effects of joint flexibility [34].

Although SPT has the advantage of order reduction and can use adaptive control algorithm to get more accurate parameter, it is difficult to find an adaptive control law for a nonlinear system and the computational burden is high. Furthermore, this approach is limited by joint stiffness. In order to solve this problem, Liu designed a joint flexibility compensator which can greatly increase the equivalent joint stiffness so that the approach can be extended to the robot system with normal joint flexibility [35]. But the universality cannot be proved because he only realized this controller in a single joint.

## 1.4.2 Feedback Linearization

Feedback linearization is regarded to be a better method to control a nonlinear system. The core idea of this method is to transform the original nonlinear system to an equivalent linear system by using suitable nonlinear state transformation and correct control law, in order that it is convenient to apply the control methods of the linear system to this equivalent linear system. From a theoretical standpoint, this method is considerably attractive because it is easy to prove the stability of the system, that is to say, the original nonlinear system is stable if the stability of its equivalent linear system is proved. This method is widely used in the control of rigid robots, such as computed torque control and inverse dynamics control. This is because the models of rigid robot can always be feedback-linearized. However, some dynamic models of flexible joint robots cannot meet requirements of feedback linearization.

Spong proposed a simplified flexible joint model where the joint flexibility is considered as a linear torsional spring and the kinetic energy of the rotor is due mainly to

its own rotation. He also pointed out that this simplified model system can be globally linearized by nonlinear coordinate transformation and static state feedback [20]. Then Grimm performed a robustness analysis of feedback-linearized or nonlinear-decoupled control loops for simplified flexible joint model [36]. Sira-Ramirez and Spong proposed an outer-loop control based on the sliding mode theory of variable structure systems for robust tracking [37]. Ge designed a robust adaptive neural network controller for the flexible joint robots using feedback linearization techniques, and it could better handle dynamical model changes and parameter uncertainties than the conventional feedback linearization controller [38]. Ider proposed a trajectory tracking control law for flexible joint robots based on solving the acceleration level inverse dynamics equations which are singular, and the implicit numerical integration method was used to avoid further differentiation of the motion and task equations [39].

As for complete dynamic model of flexible joint robots, it cannot be linearized only by static state feedback. However, De Luca proved that the complete dynamic model can always be fully transformed into a linear, controllable and input-output decoupled system by nonlinear dynamic state feedback if the system is invertible with no zero dynamics [40]. He introduced a general inversion algorithm for the synthesis of a dynamic state feedback law that gives input-output decoupling and full state linearization [41].

The feedback linearization control law requires high-order differential terms, such as acceleration and jerk. It means some state must be estimated from available measurements, for example, the jerk. So we have to construct suitable observers to perform states estimation.

In conclusion, tracking performance of feedback linearization control depends heavily on model accuracy of the system. Meanwhile, a linear or a nonlinear observer needs to be constructed because all states for feedback computation must be obtained, making the system more complex. So one approach is to combine feedback linearization with adaptive control or neural network control to reduce the requirement of model accuracy. Another approach is to use the controller with no need for all states feedback.

### 1.4.3 Cascaded System and Integral Backstepping

Cascaded system method firstly assumes that dynamics related to rigid manipulators can be controlled directly, then decides the actuator control law by taking the flexibility of joints into consideration. In this method, the dynamic model of flexible joint robots is expressed in the form of two cascaded loops: the manipulator loop and the actuator loop regarding the flexibility of the joint. A desired force which is introduced as a synthesized input signal for the manipulator loop, is generated due to the flexibility between the motor and the link. And it must be realized by the output of the actuator loop through designing a suitable controller. Compared with singular perturbation technique, this method is not subject to restrictions on the magnitude of the joint flexibility.

Cascaded system method depend on a better system model, therefore it usually requires a combination of the adaptive control law. Yuan developed a composite

adaptive controller for flexible joint robots without any restrictions on joint stiffness by using this method [42]. The paper [43] proposed adaptive task-space controllers to solve motion control and compliance control problems.

Integral backstepping control technique has recently been developed for the cascaded systems. This systematic method is realized by designing a Lyapunov function for the system whose input side has a series of integrators, and it becomes an important theoretical design method because the method can provide the proof of Lyapunov stability.

Three improvements obtained from integral backstepping technology are (1) the ability to construct globally stable controllers that compensate for parametric uncertainty, (2) the elimination of restrictions on the magnitude of the joint flexibility, and (3) the elimination of the requirement of link acceleration and link jerk measurements. All of the above improvements, however, are not fully exploited at once. For instance, Chen and Fu developed a controller which can eliminate restrictions on the size of joint flexibility, but can only compensate for parametric uncertainty to the extent of proving local stability. In addition, link acceleration measurements are needed, and this is not practical for most industrial robots [44]. Beneallegue and Sirdi used a passivity approach to design a globally asymptotically stable adaptive controller, but it needs both link acceleration and jerk measurements [45].

The above methods have dealt with parametric uncertainty by using an adaptive approach with asymptotic position tracking as the goal but have been plagued by the requirement of either link acceleration or jerk measurements. Dawson used a backstepping approach to design a robust controller that compensates for parametric uncertainty and unknown bounded disturbances while achieving a globally uniformly ultimately bounded position tracking error [46]. Nicosia and Tomei made a further step to design an adaptive controller that can simultaneously satisfy the above three improvements, but it is special for a single-link flexible joint robot still leaving a control problem for general  $n$ -link flexible joint robots [47]. Lozano and Brogliato solved the problem and designed an adaptive control scheme achieving globally asymptotic position tracking while satisfying the three improvements [48]. In their method, inverse inertia matrix is required to eliminate the measurements of the link acceleration, increasing the computational time and complexity.

Kwan and Lewis proposed a robust backstepping controller using neural networks [48]. Compared with adaptive backstepping control, linearity in unknown parameters is not needed. A major problem with backstepping is corrected in that no tedious computation of 'regression matrices' is needed. However, most neural networks need a long training time, and it is not practical for robot real-time control. Macnab and D'Eleuterio developed a neuroadaptive control scheme using a relatively small neural network, which is more practical to implement owing to reduced training times and reduced on-line computational burden [49].

Although integral backstepping technology is not subject to restrictions on the size of joint flexibility, it still needs more complete research. It usually contains an adaptive control law because it is sensitive to model parameters. Another problem with backstepping is that the constructed Lyapunov equation is limited and it does not have a parameter to adjust the system performance directly, while its only

advantage is the guarantee of system stability. More materials can be seen in a survey of backstepping approach for flexible joint robots [50].

#### 1.4.4 PD Control

Traditional PD (or PID) controllers are widely used in practical engineering for simplicity, strong robustness and high control precision. In engineering, a modified PD controller with a joint flexibility compensation can be used in the control of flexible joint robots.

A simple PD controller suffices to stabilize any kind of rigid robot about a reference position if gravitational forces are compensated. Although rigid robots are exactly linearizable by static-state feedback, the stability of the PD controller does not rely on this property, but it is rather related to the intrinsic passivity properties. Tomei proposed a simple PD controller and demonstrated its global asymptotic stability [23]. Albu-Shaffer and Hirzinger developed a joint state feedback PD controller with gravity and nonlinear friction compensation, which can be gradually extended to take account of the full robot dynamics. The global asymptotic stability was proven based on Lyapunov's first method, and effectiveness of the controller was validated through experiments on a 7 d.o.f. lightweight manipulator [51].

Although PD control is simple and useful, the joint flexibility must be taken into account in modeling of the robot in order to improve control accuracy.

#### 1.4.5 Other Control Methods

In addition to the above-mentioned control methods, adaptive control, robust control, neural networks, fuzzy control, sliding mode control [37], etc., are also applied to the control of flexible joint robots.

Adaptive control and robust control are two basic control strategies for solving problems of robot uncertainties, such as uncertainties of model parameters, unknown or varying loads, flexibility of joint, time-varying friction coefficient, etc. When parameters of the controlled system change, adaptive control [26, 30, 32, 33, 34, 38, 42, 43, 45, 47, 49, 52, 53, 54] can achieve certain performance by identification, learning and adjustment of the control law. But it cannot guarantee the stability of the system when strict real-time requirement is proposed and implementation is complex, especially when non-parametric uncertainty exists. However, robust control [28, 33, 34, 37, 38, 46, 48, 55, 56, 57] can guarantee the stability of the system and maintain a better performance when the uncertainties are in a certain range.

Since the accurate model of the flexible joint robot can not be obtained, neural network control [38, 48, 49, 58] is proposed to solve this problem through learning the unknown model information and training the neural network to approximate the system dynamics.

Fuzzy control [29] does not require a precise system model in controller design, and generally it needs the experience from user or expert, information and operational data, etc. It is convenient to obtain the control strategy for nonlinear and uncertain objects, and has a good real-time performance due to its simple algorithm.

For typical robotic tasks (e.g. machining and FSW process) that require interaction with the environment, contact forces must properly be handled by the robot controller. In such cases, a pure motion controller usually gives poor performance and can even cause instability. Chiaverini et al. proposed a stable force/position control scheme for the robots in contact with an elastically compliant surface, and it was tested on the industrial robot COMAU SMART 6.10R [59]. De Luca also designed a robust hybrid control scheme for force and velocity control of manipulators in dynamic contact with the environment [60].

## 1.5 Dissertation Outline

This thesis report is composed of the following six chapters:

**Chapter 1** offers the general introduction of the thesis, including the background of the thesis, the research problems and objectives, a literature review based on a deep and extensive literature study and the outline of the dissertation.

**Chapter 2** presents the modeling of the flexible joint robot manipulators. Several essential models of a 6-DOF flexible joint robot manipulator including direct geometric model (DGM), inverse geometric model (IGM), direct kinematic model (DKM), inverse kinematic model (IKM), direct dynamic model (DDM) and inverse dynamic model (IDM) are established based on the fundamental knowledge concerning the robot modeling, such as the spatial descriptions, the homogeneous transformations, the X-Y-Z fixed angles and the modified Denavit-Hartenberg convention, etc. Besides, the joint flexibility model, the friction model and the torque of gravity compensator at the second axis are studied as well.

**Chapter 3** proposes a novel approach for the simplification of industrial robot dynamic model based on interval method. Firstly, the basic definitions and operations of interval analysis as well as the inclusion function and overestimation are introduced. Then, the symbolic dynamic model of an industrial manipulator KUKA KR500-2MT is generated. Next, a generalized simplification algorithm is proposed and a simple example is provided to explain and describe the simplification method. Meanwhile, the results of the simplified model for the whole workspace and corresponding error analysis as well as the further simplification are also given. Case studies have been carried out on three different test trajectories including the identification trajectory, the linear FSW trajectory and the circular FSW trajectory, in order to demonstrate the performance of the simplification method. Moreover, the torques in the robot dynamic equation has been analyzed in the case of a linear FSW process as well. Finally, some usages of the simplification method are provided.

**Chapter 4** presents the dynamic modeling and identification of the robotic FSW process. First of all, a brief description of the FSW process is given, including its main stages and process forces as well as two static nonlinear power models of the process forces. Then, the experimental setup for the model identification is described. In the meantime, three methods which can calculate the plunge depth and its derivative are proposed, followed by the data filtering for the measurements and calculations. Next, the linear and nonlinear dynamic modeling and identification of the axial force are implemented by using least squares method. Finally, system-



atic studies on the errors of identified models have been carried out to analyze the performance of the model identification.

**Chapter 5** describes the design of robust force controllers for industrial robot manipulators and the development of a simulation system of robotic FSW process. Firstly, a literature review on force control methods of the robot is presented, followed by selecting the force control strategy and the modeling of the robotic FSW process. Then, the parameter identification of rigid robot displacement model is implemented by using two different algorithms. Based on the modeling of robotic FSW process which simultaneously considers the complete kinematics, the rigid robot displacement model, the joint flexibility and the dynamic axial force process model, the robust force controllers have been designed by using the frequency response approach. Next, the simulator of robotic FSW process is established and simulation studies have been carried out to test the performance of force controllers. Finally, the axial force vibration in robotic FSW process is analyzed and a disturbance model of zero force vertical position which can lead to the oscillation of axial force is proposed and used in simulations.

**Chapter 6** concludes the thesis with main contributions of the research and perspectives for the future work.

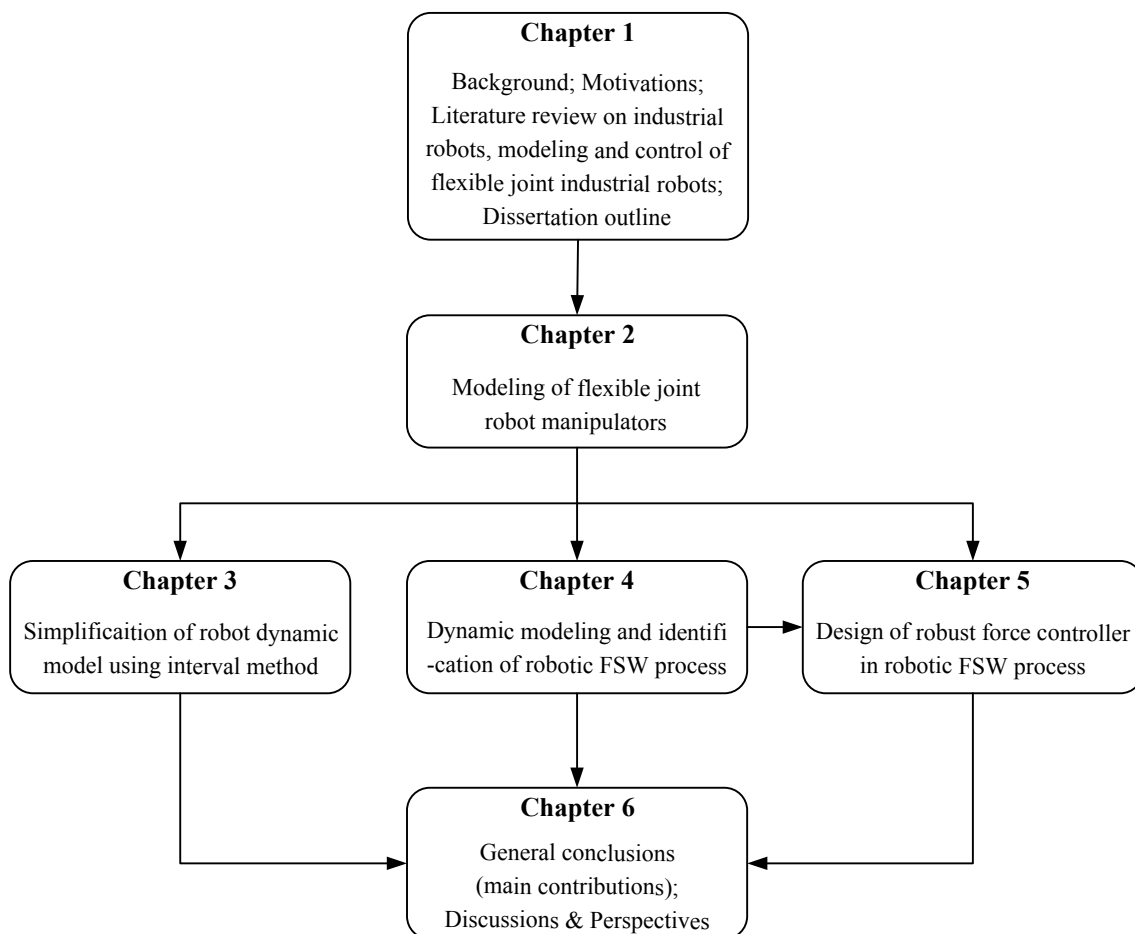


Figure 1.6: The outline and flow of information in this dissertation

# Chapter 2

## Modeling of Flexible Joint Robot Manipulators

### 2.1 Introduction

The work presented in this chapter concerns the modeling of flexible joint robot manipulators. Based on a lot of research work on robot modeling described in the books of Craig [62], Spong [8], and Siciliano [5], the fundamental studies are carried out in order to obtain the different models which we need to use in the model simplification, simulation of robot motion and controller design. Thus, the work in this chapter is an important foundation for the following research.

Section 2.2 gives a brief introduction to the spatial description and coordinate transformation in robotics. Then, section 2.3 focuses on the problem of robot kinematics. The direct and inverse geometric models, the direct and inverse kinematic models are also established. In Section 2.4, the dynamic model, the flexibility model and the friction model of the robot are studied. The software *SYMORO+* for generating symbolic robot models is introduced as well.

### 2.2 Spatial Descriptions and Coordinate Transformations

#### 2.2.1 Descriptions of Positions, Orientations and Frames

Generally, the concepts of position vectors, planes and coordinate frames will be used to describe the relationship between some objects, including parts, tools, and the manipulator itself.

In the selected Cartesian coordinate frame  $\{A\}$ , arbitrary point  $P$  in space can be represented as a  $3 \times 1$  position vector  ${}^A P$ :

$${}^A P = \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} \quad (2.1)$$

where  $p_x$ ,  $p_y$  and  $p_z$  are coordinate components of the point  $P$  in the coordinate

frame  $\{A\}$ .  $\hat{X}_A$ ,  $\hat{Y}_A$  and  $\hat{Z}_A$  are three mutually orthogonal unit vectors along axis  $x$ ,  $y$ ,  $z$  of coordinate frame  $\{A\}$ .  $O_A$  is the origin of coordinate frame  $\{A\}$ , as shown in Figure 2.1.

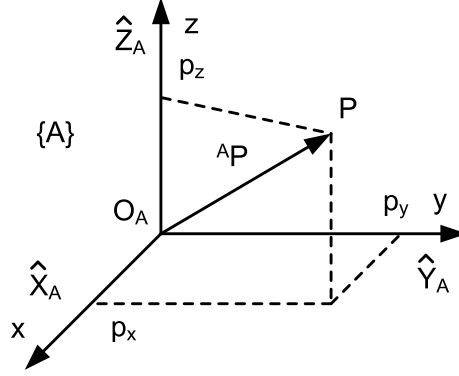


Figure 2.1: Position description of a point in space

In addition to representing a point, the orientation of a body in space is also necessary to describe during the movement and operation of the robot manipulator. After attaching a coordinate frame to a body, the orientation of the body can be described by the coordinate frame relative to the reference frame  $\{A\}$ . In Figure 2.2, the coordinate frame  $\{B\}$  has been attached to the body in a known way.  $\hat{X}_B$ ,  $\hat{Y}_B$ , and  $\hat{Z}_B$  denote the unit vectors giving the principal directions of coordinate frame  $\{B\}$ . These unit vectors can be also written in terms of coordinate frame  $\{A\}$ , yielding the direction cosines  ${}^A\hat{X}_B$ ,  ${}^A\hat{Y}_B$ , and  ${}^A\hat{Z}_B$ .

The orientation of rigid body  $B$  with respect to coordinate frame  $\{A\}$  can be expressed as a  $3 \times 3$  matrix  ${}^A R_B$ :

$${}^A R_B = \begin{bmatrix} {}^A\hat{X}_B & {}^A\hat{Y}_B & {}^A\hat{Z}_B \end{bmatrix} = \begin{bmatrix} \hat{X}_B \cdot \hat{X}_A & \hat{Y}_B \cdot \hat{X}_A & \hat{Z}_B \cdot \hat{X}_A \\ \hat{X}_B \cdot \hat{Y}_A & \hat{Y}_B \cdot \hat{Y}_A & \hat{Z}_B \cdot \hat{Y}_A \\ \hat{X}_B \cdot \hat{Z}_A & \hat{Y}_B \cdot \hat{Z}_A & \hat{Z}_B \cdot \hat{Z}_A \end{bmatrix} \quad (2.2)$$

where  ${}^A R_B$  is referred to as rotation matrix, superscript  $A$  denotes reference coordinate frame  $\{A\}$ , and subscript  $B$  denotes coordinate frame  $\{B\}$ .

Based on the above definitions, the rotation matrix of frame  $\{B\}$  about the  $X_A$  axis, the  $Y_A$  axis and the  $Z_A$  axis by an angle  $\theta$  are:

$$R(X_A, \theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\theta & -s\theta \\ 0 & s\theta & c\theta \end{bmatrix} \quad (2.3)$$

$$R(Y_A, \theta) = \begin{bmatrix} c\theta & 0 & s\theta \\ 0 & 1 & 0 \\ -s\theta & 0 & c\theta \end{bmatrix} \quad (2.4)$$

$$R(Z_A, \theta) = \begin{bmatrix} c\theta & -s\theta & 0 \\ s\theta & c\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.5)$$

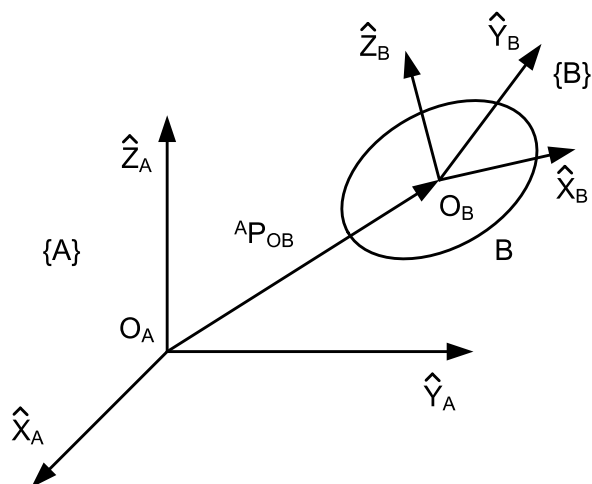


Figure 2.2: Position and orientation description of an object

where  $s\theta$  denotes  $\sin(\theta)$ , and  $c\theta$  denotes  $\cos(\theta)$ . This notation will be adopted in the following sections as well.

In robotics, a frame can be used as a description of one coordinate system relative to another one. The description of a frame can be regarded as a position vector and a rotation matrix. As shown in Figure 2.2, frame  $\{B\}$  can be described by the vector  ${}^A P_{OB}$  and the rotation matrix  ${}^A R_B$ :

$$\{B\} = \{ {}^A R_B \quad {}^A P_{OB} \} \quad (2.6)$$

where  ${}^A P_{OB}$  is the vector that locates the origin of the frame  $\{B\}$  relative to frame  $\{A\}$ .

### 2.2.2 Homogeneous Transformations

Assuming the coordinate of the point  $P$  in frame  $\{B\}$  is known, the coordinate of this point in frame  $\{A\}$  can be solved by homogeneous transformations as follows:

$$\begin{bmatrix} {}^A P \\ 1 \end{bmatrix} = \begin{bmatrix} {}^A R_B & {}^A P_{OB} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} {}^B P \\ 1 \end{bmatrix} = {}^A T_B \begin{bmatrix} {}^B P \\ 1 \end{bmatrix} \quad (2.7)$$

where  ${}^A T_B$  is the  $4 \times 4$  homogeneous transformation matrix which represents both translation transformation and rotation transformation.

It is noted that the homogeneous transformation of a simple rotation about an axis is usually denoted **Rot**, thus a rotation of  $\theta$  about an axis  $z$  is:

$$\mathbf{Rot}(z, \theta) = \begin{bmatrix} c\theta & -s\theta & 0 & 0 \\ s\theta & c\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.8)$$

Similarly, the homogeneous transformation of a simple translation along an axis

is usually denoted **Trans**, thus a translation of  $d$  along an axis  $x$  is:

$$\mathbf{Trans}(x, d) = \begin{bmatrix} 1 & 0 & 0 & d \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.9)$$

### 2.2.3 X-Y-Z fixed angles

In order to obtain a minimal representation, a human operator at a computer terminal prefers to use a simpler representation which requires only three numbers rather than spend a long time inputting a nine-element matrix with orthonormal columns. Thus, there are several such representations which can be applied, such as X-Y-Z fixed angles, Z-Y-X Euler angles, Z-Y-Z Euler angles and so on [5, 8, 62]. Based on the measurements from robot KUKA, the controller of the robot in this research implements X-Y-Z fixed angles convention, which will be described below.

Consider the fixed reference frame  $\{A\}$  and the rotated frame  $\{B\}$  which are shown in Figure 2.3, the orientation of frame  $\{B\}$  relative to the frame  $\{A\}$  can be specified sequentially by three angles  $(\gamma, \beta, \alpha)$ , and obtained by three successive rotations as follows:

- Rotate frame  $\{A\}$  about  $X_A$  by an angle  $\gamma$ , yielding new frame  $\{1\}$ ;
- Rotate frame  $\{1\}$  about  $Y_A$  by an angle  $\beta$ , yielding new frame  $\{2\}$ ;
- Rotate frame  $\{2\}$  about  $Z_A$  by an angle  $\alpha$ , yielding the final frame  $\{B\}$ .

All above three rotations are carried out about an axis of the fixed reference frame  $\{A\}$ , therefore this convention for specifying an orientation is referred to as **X-Y-Z fixed angles** (note that these angles are not Euler angles).

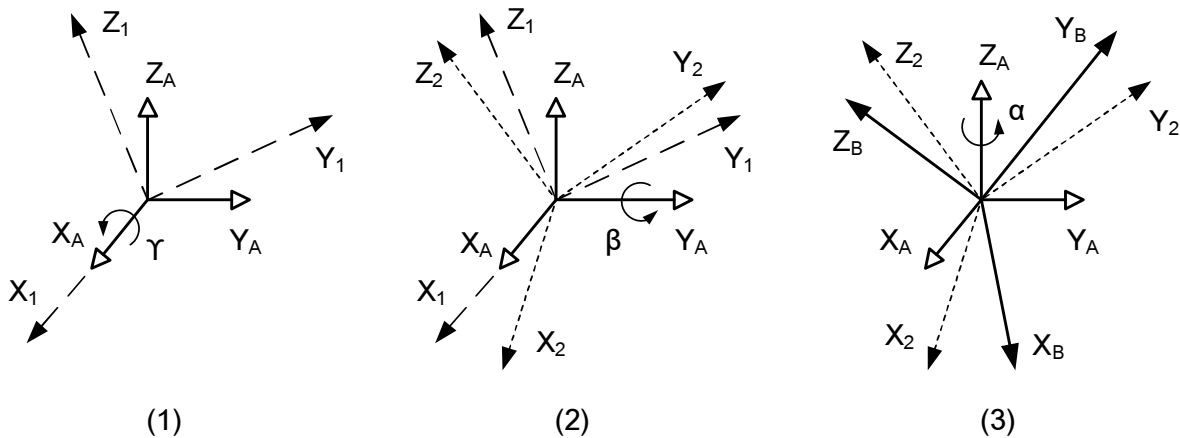


Figure 2.3: X-Y-Z fixed angles

Since all rotations are relative to the fixed frame, the resulting transformation matrix is given by:

$$\begin{aligned} R_{XYZ}(\gamma, \beta, \alpha) &= R(z, \alpha) R(y, \beta) R(x, \gamma) \\ &= \begin{bmatrix} cac\beta & cas\beta s\gamma - sac\gamma & cas\beta c\gamma + sas\gamma \\ sac\beta & sas\beta s\gamma + cac\gamma & sas\beta c\gamma - cas\gamma \\ -s\beta & c\beta s\gamma & c\beta c\gamma \end{bmatrix} \end{aligned} \quad (2.10)$$

where  $s\alpha$ ,  $s\beta$ ,  $s\gamma$ ,  $c\alpha$ ,  $c\beta$ ,  $c\gamma$  represent  $\sin(\alpha)$ ,  $\sin(\beta)$ ,  $\sin(\gamma)$ ,  $\cos(\alpha)$ ,  $\cos(\beta)$  and  $\cos(\gamma)$  respectively.

In the case of real robot coordinate calculation, the resulting transformation matrix is exactly the rotation matrix that expresses the orientation of tool frame with respect to base frame. Therefore, the three angles  $\gamma$ ,  $\beta$ ,  $\alpha$  can be directly obtained for a given rotation matrix, which will be useful for robot kinematics.

## 2.3 Robot Kinematics

Robot kinematics is an important foundation for studying robot manipulators. The position, the velocity, the acceleration and all higher order derivatives of the position variables are described by robot kinematics. This section presents the convention of the modified Denavit-Hartenberg for assigning coordinate frames, forward kinematics, inverse kinematics, forward instantaneous kinematics, and inverse instantaneous kinematics, which are essential for controlling the motion of the robot manipulator.

### 2.3.1 Modified Denavit-Hartenberg Convention

On the basis of the research work of Khalil and Dombre [63], the modified Denavit-Hartenberg convention is widely adopted for selecting frames of reference in robotic applications. Here a brief description for the convention is given.

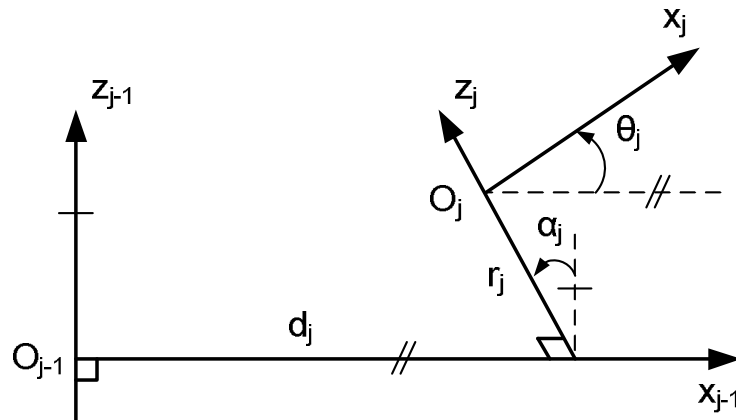


Figure 2.4: Description of modified Denavit-Hartenberg convention

As shown in Figure 2.4, the coordinate frame  $j$ , which is fixed on link  $j$ , and the parameters related to the coordinate transformation can be defined as:

- the  $z_j$  axis is along the axis of joint  $j$ ;
- the  $x_j$  axis is along the common normal between  $z_j$  and  $z_{j+1}$ . If the axes  $z_j$  and  $z_{j+1}$  are parallel, the choice of  $x_j$  is not unique;
- the origin  $O_j$  is the intersection of  $z_j$  and  $x_j$ ;
- $\alpha_j$  is the angle between  $z_{j-1}$  and  $z_j$  about  $x_{j-1}$ ;
- $d_j$  is the distance between  $z_{j-1}$  and  $z_j$  along  $x_{j-1}$ ;
- $\theta_j$  is the angle between  $x_{j-1}$  and  $x_j$  about  $z_j$ ;

-  $r_j$  is the distance between  $x_{j-1}$  and  $x_j$  along  $z_j$ ;

The joint variable of joint  $j$  can be written as:

$$q_j = \bar{\sigma}_j \theta_j + \sigma_j r_j \quad (2.11)$$

where if joint  $j$  is revolute,  $\sigma_j = 0, \bar{\sigma}_j = 1$ ; and if joint  $j$  is prismatic,  $\sigma_j = 1, \bar{\sigma}_j = 0$ . The value  $\sigma_j = 2$  is used to indicate the current frame  $j$  is fixed with respect to frame  $j - 1$ .

In this convention, each homogeneous transformation matrix defining frame  $j$  with respect to frame  $j - 1$  can be represented as a product of four basic transformations:

$$\begin{aligned} {}^{j-1}T_j &= Rot(x, \alpha_j) Trans(x, d_j) Rot(z, \theta_j) Trans(z, r_j) \\ &= \begin{bmatrix} c\theta_j & -s\theta_j & 0 & d_j \\ c\alpha_j s\theta_j & c\alpha_j c\theta_j & -s\alpha_j & -r_j s\alpha_j \\ s\alpha_j s\theta_j & s\alpha_j c\theta_j & c\alpha_j & r_j c\alpha_j \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned} \quad (2.12)$$

The industrial robot considered in this study is a 6-degree-of-freedom serial manipulator KUKA KR500-2MT, which is composed of six moving links and six revolute joints (see Figure 2.5). In order to simplify the modeling of this robot, it is assumed that the tool is directly fixed on the link 6 and the spindle axis of the tool is coincident with the axis of joint 6.



Number of axes : 6  
 Payload : 500 kg  
 Repeatability :  $\pm 0.08$  mm  
 Max. reach : 2826mm  
 Weight : 2400 kg  
 Controller : KR C2  
 Mounting position : Floor

Figure 2.5: Robot KUKA KR500-2MT

Based on the above-mentioned modified Denavit-Hartenberg convention, a geometric description of the robot manipulator including the assignment of coordinate frames is given in Figure 2.6. The corresponding geometric parameters are listed in Table 2.1.

### 2.3.2 Forward Kinematics: Direct Geometric Model

The forward kinematics problem is to determine the position and orientation of the end-effector relative to the base frame in terms of the joint variables, which are the

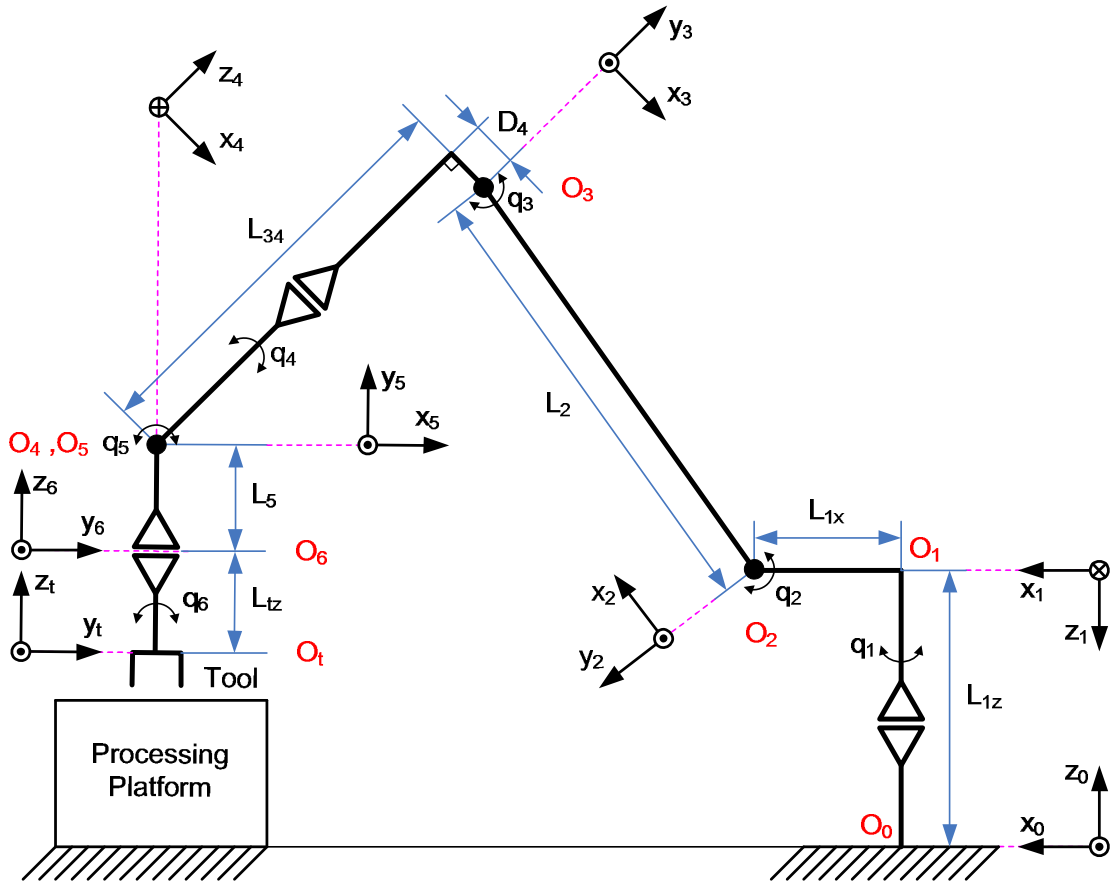


Figure 2.6: Geometric description of the robot

Table 2.1: Geometric parameters of robot model

| $j$ | $\mu_j$ | $\sigma_j$ | $\alpha_j$ | $d_j$    | $\theta_j$     | $r_j$     |
|-----|---------|------------|------------|----------|----------------|-----------|
| 1   | 1       | 0          | $\pi$      | 0        | $q_1$          | $-L_{1z}$ |
| 2   | 1       | 0          | $\pi/2$    | $L_{1x}$ | $q_2$          | 0         |
| 3   | 1       | 0          | 0          | $L_2$    | $\pi/2 + q_3$  | 0         |
| 4   | 1       | 0          | $-\pi/2$   | $-D_4$   | $q_4$          | $-L_{34}$ |
| 5   | 1       | 0          | $\pi/2$    | 0        | $q_5$          | 0         |
| 6   | 1       | 0          | $-\pi/2$   | 0        | $-\pi/2 + q_6$ | $-L_5$    |
| t   | 0       | 2          | 0          | 0        | 0              | $-L_{tz}$ |

link extensions in the case of sliding or prismatic joints, and the angles between the links in the case of rotational or revolute joints. The frame fixed in the end-effector is usually referred to as the tool frame. The direct geometric model (DGM) can be used to calculate the position and orientation of the tool frame by using the joint positions:

$$X = f_{DGM}(q) \quad (2.13)$$

where  $q$  denotes the vector of the joint positions.  $X = [P_{tx} \ P_{ty} \ P_{tz} \ A_{tx} \ A_{ty} \ A_{tz}]^T$  is the vector defining the position and orientation of the tool frame with respect to the base frame. In this vector,  $P_{tx}$ ,  $P_{ty}$  and  $P_{tz}$  represent the tool position,  $A_{tx}$ ,  $A_{ty}$ ,





In fact, the rotation matrix  ${}^0R_t(q)$  is represented in terms of X-Y-Z fixed angles  $\gamma$ ,  $\beta$ , and  $\alpha$  in the real controller of robot KUKA, that is:

$$R_{XYZ}(\gamma, \beta, \alpha) = {}^0R_t(q) \quad (2.18)$$

By comparing equation (2.10) and (2.17), the X-Y-Z fixed angles  $\gamma$ ,  $\beta$ , and  $\alpha$  can be calculated as follows:

If  $\sqrt{r_{11}^2 + r_{21}^2} \neq 0$  ( $\cos(\beta) \neq 0$ ), that is to say,  $\beta \neq \pm\pi/2$ , then

$$\begin{aligned} \beta &= \text{atan2}(-r_{31}, \pm\sqrt{r_{11}^2 + r_{21}^2}) \\ \alpha &= \text{atan2}(r_{21}, r_{11}) \\ \gamma &= \text{atan2}(r_{32}, r_{33}) \end{aligned} \quad (2.19)$$

where  $\text{atan2}(y, x)$  is a two-argument arc tangent function. This function computes  $\arctan(y/x)$  but uses the signs of both  $x$  and  $y$  to identify the quadrant in which the resulting angle lies. Note that if both  $x$  and  $y$  are zero,  $\text{atan2}(y, x)$  is undefined. The plus and minus sign in the expression of  $\beta$  can be chosen when the value of  $\alpha$  is calculated.

Otherwise, we will obtain the following two solutions:

$$\begin{aligned} \text{if } r_{31} &= -1, \\ \beta &= \pi/2 \\ \alpha &= 0 \\ \gamma &= \text{atan2}(r_{12}, r_{22}) \end{aligned} \quad (2.20)$$

$$\begin{aligned} \text{if } r_{31} &= 1, \\ \beta &= -\pi/2 \\ \alpha &= 0 \\ \gamma &= -\text{atan2}(r_{12}, r_{22}) \end{aligned} \quad (2.21)$$

### 2.3.3 Inverse Kinematics: Inverse Geometric Model

The inverse kinematics problem is to find the value of joint positions in terms of the position and orientation of the end-effector.

The solution strategies of inverse kinematics problem can be divided into two broad classes: closed-form solutions and numerical solutions. Closed-form solutions, namely analytic solutions, are commonly used since they are generally much faster than numerical solutions and readily identify all possible solutions. However, the existence of closed-form solutions is dependent on robot structure. Pieper studied manipulators with six degrees of freedom, and the sufficient conditions to obtain closed-form solutions were given: three consecutive joint axes intersect at a common point as in a spherical wrist, or three consecutive joint axes are parallel [64]. Paul method can be used to obtain a closed-form explicit solution for each joint variable here [65].

To solve this problem, the inverse geometric model (IGM) is established as:

$$q = f_{IGM}(X) \quad (2.22)$$

where vector  $X = [P_{tx} \ P_{ty} \ P_{tz} \ \gamma \ \beta \ \alpha]^T$  is given and defines the desired position and orientation of the tool frame with respect to the base frame.  $q$  is the vector of joint position to be calculated.

Thus, the homogeneous transformation matrix defining the tool frame with respect to the base frame is calculated as:

$${}^0T_t = \begin{bmatrix} {}^0R_t & {}^0P_t \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} cac\beta & cas\beta s\gamma - sac\gamma & cas\beta c\gamma + sas\gamma & P_{tx} \\ sac\beta & sas\beta s\gamma + cac\gamma & sas\beta c\gamma - cas\gamma & P_{ty} \\ -s\beta & c\beta s\gamma & c\beta c\gamma & P_{tz} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.23)$$

Assuming we use matrix  $U_0$  to represent the above transformation matrix  ${}^0T_t$ :

$$U_0 = {}^0T_1 {}^1T_2 {}^2T_3 {}^3T_4 {}^4T_5 {}^5T_6 {}^6T_t \quad (2.24)$$

and:

$$U_0 = \begin{bmatrix} s_x & n_x & a_x & P_{tx} \\ s_y & n_y & a_y & P_{ty} \\ s_z & n_z & a_z & P_{tz} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.25)$$

It is obvious that all components of the matrix  $U_0$  are known according to equation (2.23). In fact, calculating the joint position  $q$  directly from equation (2.24) is very difficult. By premultiplying successively the two sides of equation (2.24) by the matrices  $({}^{j-1}T_j)^{-1}$ , for  $j = 1, \dots, 6$ , new set of equations can be obtained which enable us to isolate and identify the joint variables individually. For example, premultiplying the two sides of equation (2.24) by  $({}^0T_1)^{-1}$  gives:

$$({}^0T_1)^{-1} U_0 = {}^1T_2 {}^2T_3 {}^3T_4 {}^4T_5 {}^5T_6 {}^6T_t \quad (2.26)$$

where the left hand side is a function of the variable  $q_1$ , and the right hand side is a function of  $q_2, \dots, q_6$ . The variable  $q_1$  can be calculated by identifying some elements of both sides of this equation.

In a similar way, new set of equations are given by:

$$\begin{aligned} U_0 &= {}^0T_1 {}^1T_2 {}^2T_3 {}^3T_4 {}^4T_5 {}^5T_6 {}^6T_t \\ U_1 &= {}^1T_2 {}^2T_3 {}^3T_4 {}^4T_5 {}^5T_6 {}^6T_t \\ U_2 &= {}^2T_3 {}^3T_4 {}^4T_5 {}^5T_6 {}^6T_t \\ U_3 &= {}^3T_4 {}^4T_5 {}^5T_6 {}^6T_t \\ U_4 &= {}^4T_5 {}^5T_6 {}^6T_t \\ U_5 &= {}^5T_6 {}^6T_t \\ U_6 &= {}^6T_t \end{aligned} \quad (2.27)$$

where  $U_j = ({}^{j-1}T_j)^{-1}U_{j-1}$ , and these equations are named forward equations. In some cases, it is appropriate to use backward equations which are obtained by post-multiplying successively the two sides of equation (2.24) by  $({}^{j-1}T_j)^{-1}$ , for  $j = 7, \dots, 2$

( $t = 7$ ):

$$\begin{aligned}
 U'_0 &= {}^0T_1 {}^1T_2 {}^2T_3 {}^3T_4 {}^4T_5 {}^5T_6 {}^6T_t \\
 U'_1 &= {}^0T_1 {}^1T_2 {}^2T_3 {}^3T_4 {}^4T_5 {}^5T_6 \\
 U'_2 &= {}^0T_1 {}^1T_2 {}^2T_3 {}^3T_4 {}^4T_5 \\
 U'_3 &= {}^0T_1 {}^1T_2 {}^2T_3 {}^3T_4 \\
 U'_4 &= {}^0T_1 {}^1T_2 {}^2T_3 \\
 U'_5 &= {}^0T_1 {}^1T_2 \\
 U'_6 &= {}^0T_1
 \end{aligned} \tag{2.28}$$

with  $U'_j = U'_{j-1}({}^{j-1}T_j)^{-1}$ , and  $U'_0 = U_0$ .

### Step 1. Calculate $q_1$

By respectively using  $T_{Rj}$  and  $T'_{Rj}$  to denote the matrices on the right hand side corresponding to  $U_j$  and  $U'_j$ , the forward equations  $U_1(2, 3) = T_{R1}(2, 3)$  and  $U_1(2, 4) = T_{R1}(2, 4)$  can be written as:

$$\begin{aligned}
 -a_y c_1 - a_x s_1 &= -s_4 s_5 \\
 -P_{ty} c_1 - P_{tx} s_1 &= L_{5tz} s_4 s_5
 \end{aligned} \tag{2.29}$$

where  $L_{5tz} = L_5 + L_{tz}$ , and the variable  $q_1$  is deduced as:

$$q_1 = \text{atan2}(-L_{5tz} a_y - P_{ty}, L_{5tz} a_x + P_{tx}) \tag{2.30}$$

In fact, the above solution is selected according to the range of joint motion, although two solutions can be obtained here.

### Step 2. Calculate $q_2$ and $q_3$

In backward equations, if we choose  $U'_1(1, 4) = T'_{R1}(1, 4)$ ,  $U'_1(2, 4) = T'_{R1}(2, 4)$  and  $U'_1(3, 4) = T'_{R1}(3, 4)$ , then:

$$\begin{aligned}
 a_x L_{5tz} + P_{tx} &= c_1 (L_{1x} + D_4 s_{23} + L_{34} c_{23} + L_2 c_2) \\
 a_y L_{5tz} + P_{ty} &= -s_1 (L_{1x} + D_4 s_{23} + L_{34} c_{23} + L_2 c_2) \\
 a_z L_{5tz} + P_{tz} &= L_{1z} + D_4 c_{23} - L_{34} s_{23} - L_2 s_2
 \end{aligned} \tag{2.31}$$

Let:

$$\begin{aligned}
 \mathbb{M} &= D_4 s_{23} + L_{34} c_{23} + L_2 c_2 \\
 \mathbb{N} &= D_4 c_{23} - L_{34} s_{23} - L_2 s_2
 \end{aligned} \tag{2.32}$$

where  $\mathbb{M}$  and  $\mathbb{N}$  can be calculated as:

$$\begin{aligned}
 \mathbb{M} &= (a_x L_{5tz} + P_{tx}) / c_1 - L_{1x} \quad (\text{if } q_1 \neq \pm\pi/2) \\
 \mathbb{M} &= (a_y L_{5tz} + P_{ty}) / s_1 - L_{1x} \quad (\text{if } q_1 = \pm\pi/2) \\
 \mathbb{N} &= a_z L_{5tz} + P_{tz} - L_{1z}
 \end{aligned} \tag{2.33}$$

According to equation (2.32), we can get:

$$D_4s_3 + L_{34}c_3 = (\mathbb{M}^2 + \mathbb{N}^2 - L_2^2 - L_{34}^2 - D_4^2)/(2L_2) \quad (2.34)$$

Thus, if we define  $\Delta = (\mathbb{M}^2 + \mathbb{N}^2 - L_2^2 - L_{34}^2 - D_4^2)/(2L_2)$ , the variable  $q_3$  can be deduced as:

$$q_3 = \text{atan2}(\Delta, \pm\sqrt{L_{34}^2 + D_4^2 - \Delta^2}) - \text{atan2}(L_{34}, D_4) \quad (2.35)$$

where sign  $\pm$  depends on the value of  $q_3$ , we choose the negative sign as  $q_3 > 0$ , otherwise the positive sign is selected.

By expanding the trigonometric function in equation (2.32), we have:

$$\begin{aligned} (D_4c_3 - L_{34}s_3)s_2 + (L_2 + L_{34}c_3 + D_4s_3)c_2 &= \mathbb{M} \\ (D_4c_3 - L_{34}s_3)c_2 - (L_2 + L_{34}c_3 + D_4s_3)s_2 &= \mathbb{N} \end{aligned} \quad (2.36)$$

If we define:

$$\begin{aligned} \mathbb{U} &= L_2 + L_{34}c_3 + D_4s_3 \\ \mathbb{V} &= D_4c_3 - L_{34}s_3 \end{aligned} \quad (2.37)$$

then:

$$\begin{aligned} \mathbb{U}c_2 + \mathbb{V}s_2 &= \mathbb{M} \\ \mathbb{V}c_2 - \mathbb{U}s_2 &= \mathbb{N} \end{aligned} \quad (2.38)$$

Obviously,  $\sin(q_2)$  and  $\cos(q_2)$  can be calculated from the above equation, and the variable  $q_2$  can be obtained by:

$$q_2 = \text{atan2}(\mathbb{M}\mathbb{V} - \mathbb{N}\mathbb{U}, \mathbb{M}\mathbb{U} + \mathbb{N}\mathbb{V}) \quad (2.39)$$

### Step 3. Calculate $q_4$ , $q_5$ and $q_6$

In forward equations, if we choose  $U_3(2, 1) = T_{R3}(2, 1)$ ,  $U_3(2, 2) = T_{R3}(2, 2)$ ,  $U_3(1, 3) = T_{R3}(1, 3)$ ,  $U_3(2, 3) = T_{R3}(2, 3)$  and  $U_3(3, 3) = T_{R3}(3, 3)$ , and:

$$\begin{aligned} U_3(2, 1) &= -s_x c_1 c_{23} + s_y c_{23} s_1 + s_z s_{23} \\ U_3(2, 2) &= -n_x c_1 c_{23} + n_y c_{23} s_1 + n_z s_{23} \\ U_3(1, 3) &= -a_z c_{23} - a_x c_1 s_{23} + a_y s_1 s_{23} \\ U_3(2, 3) &= -a_x c_1 c_{23} + a_y c_{23} s_1 + a_z s_{23} \\ U_3(3, 3) &= a_y c_1 + a_x s_1 \\ T_{R3}(2, 1) &= s_5 s_6 \\ T_{R3}(2, 2) &= s_5 c_6 \\ T_{R3}(1, 3) &= -c_4 s_5 \\ T_{R3}(2, 3) &= c_5 \\ T_{R3}(3, 3) &= s_4 s_5 \end{aligned} \quad (2.40)$$

As a result, the variable  $q_5$  can be calculated according to above equations:

$$q_5 = \text{atan2}(\pm\sqrt{1 - U_3(2, 3)^2}, U_3(2, 3)) \quad (2.41)$$

Then,  $q_4$  and  $q_6$  are obtained by:

$$\begin{aligned} q_4 &= \text{atan2}(\epsilon U_3(3, 3), -\epsilon U_3(1, 3)) \\ q_6 &= \text{atan2}(\epsilon U_3(2, 1), \epsilon U_3(2, 2)) \end{aligned} \quad (2.42)$$

where  $\epsilon = \text{sign}(q_5)$ . If  $q_5 = 0$ , there are innumerable solutions for  $q_4$  and  $q_6$ , and we need to consider the range and convenience of joint motion.

### 2.3.4 Forward Instantaneous Kinematics: Direct Kinematic Model

The forward instantaneous kinematics problem is to derive the velocity relationships between the operational coordinates and joint coordinates. The direct kinematic model (DKM), which can express the linear velocities ( $v_x, v_y, v_z$ ) and angular velocities ( $\omega_x, \omega_y, \omega_z$ ) of the tool frame in terms of the Jacobian matrix and the derivatives of the joint variables, is yielded by the differentiation with respect to time of the forward position kinematics equation (2.13):

$$\dot{X} = J(q)\dot{q} \quad (2.43)$$

where  $\dot{X} = [v_x \ v_y \ v_z \ \omega_x \ \omega_y \ \omega_z]^T$  denotes the vector of operational velocity, which is the velocity of the origin of the tool frame combined with its angular velocity with respect to the fixed Cartesian coordinate frame.  $\dot{q} = [\dot{q}_1 \ \dot{q}_2 \ \dot{q}_3 \ \dot{q}_4 \ \dot{q}_5 \ \dot{q}_6]^T$  is the vector of angular velocity. The  $6 \times 6$  matrix  $J(q)$  is the Jacobian matrix of the tool frame with respect to the base frame.

The Jacobian matrix  $J(q)$  is composed of two parts: the upper half of the Jacobian  $J_v(q)$  can calculate the linear velocity  $v$ , and the lower half of the Jacobian  $J_\omega(q)$  is able to calculate the angular velocity  $\omega$ . Thus, the equation (2.43) can be rewritten as:

$$\begin{bmatrix} v \\ \omega \end{bmatrix} = \begin{bmatrix} J_v(q)_{(3 \times 6)} \\ J_\omega(q)_{(3 \times 6)} \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \vdots \\ \dot{q}_6 \end{bmatrix} \quad (2.44)$$

The elements of the Jacobian matrix can be obtained by differentiating the direct geometric model  $X = f_{DGM}(q)$  with respect to joint position  $q$  as:

$$J_{ij} = \frac{\partial f_i(q)}{\partial q_j} \quad (2.45)$$

where  $i = \{1, \dots, 6\}$ ,  $j = \{1, \dots, 6\}$ , and  $J_{ij}$  is the  $i^{\text{th}}$  row,  $j^{\text{th}}$  column element of Jacobian matrix  $J$ .

The expressions of the Jacobian matrix are detailed in Appendix A.2. In addition to calculating the total velocity of the tool frame, the Jacobian matrix can also be used to study the singularity which is a robot configuration where the Jacobian

loses rank. The transformation of tool forces and torques to the corresponding joint torques can be implemented by Jacobian matrix as well. The relation can be derived by the principle of virtual work as:

$$\tau = J^T(q)F \quad (2.46)$$

where  $J(q)$  is the Jacobian matrix of the tool frame with respect to the base frame,  $F$  is the vector of tool forces and torques expressed in the tool frame,  $\tau$  denotes the vector of joint torques.

The acceleration equations can be obtained by differentiating equation (2.43):

$$\ddot{X} = J(q)\ddot{q} + \left(\frac{d}{dt}J(q)\right)\dot{q} \quad (2.47)$$

### 2.3.5 Inverse Instantaneous Kinematics: Inverse Kinematic Model

In order to solve the inverse instantaneous kinematics problem, the inverse kinematic model (IKM) is established to find the joint velocities ( $\dot{q}$ ) by using the total velocities of the tool ( $\dot{X}$ ) and the joint positions ( $q$ ), that is:

$$\dot{q} = J(q)^{-1}\dot{X} \quad (2.48)$$

It should be noted that the inverse of the Jacobian matrix is used in the inverse kinematic model. Thus, the singular points must be avoided during the calculations.

The joint acceleration  $\ddot{q}$  can be derived from equation (2.47):

$$\ddot{q} = J(q)^{-1} \left[ \ddot{X} - \left(\frac{d}{dt}J(q)\right)\dot{q} \right] \quad (2.49)$$

## 2.4 Robot Dynamics

The dynamic model of the robot manipulator plays an important role for simulating robot motion, analyzing the robot structure and designing the robot controller. The methods for building a dynamic model of the industrial robot are presented in several books [5, 8, 62]. This section presents the dynamic modeling of robot, the flexibility and the friction model as well as the direct and inverse dynamic model of the robot.

### 2.4.1 Dynamic Modeling of the Robot

As for obtaining the dynamic model of the robot, the most common approaches are the Lagrange formulation, the Newton-Euler formulation, and Kane's method. Compared with the Newton-Euler formulation, the Lagrange formulation is able to consider the robot as a whole rather than treat each axis of the robot respectively.

The Lagrange method is based on describing scalar energy functions of the system, including the kinetic energy  $K(q, \dot{q})$  and the potential energy  $V(q)$ . The two

energy functions can be expressed in terms of the joint positions  $q$  and the joint velocities  $\dot{q}$ . Thus, the Lagrangian function  $L$  of the robot can be defined as:

$$L(q, \dot{q}) = K(q, \dot{q}) - V(q) \quad (2.50)$$

By applying the least action principle, the Lagrange equation can be formulated as:

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{q}_i} - \frac{\partial L}{\partial q_i} = \tau_i \quad (2.51)$$

where  $i = \{1, \dots, 6\}$ ,  $\tau$  is the vector of the joint torques. As a result, the dynamic model can be derived in a straightforward way.

Applying the Lagrange method to the robot body and the motor respectively, the dynamic model of the flexible joint robot can be obtained as follows [8]:

$$M(q)\ddot{q} = \Gamma - H(q, \dot{q}) - F_{fr}(\dot{q}) + J^T(q)F_e \quad (2.52)$$

$$I_a\ddot{\theta} = \Gamma_m - N^{-T}\Gamma - F_{fm}(\dot{\theta}) \quad (2.53)$$

- $M(q)_{(6 \times 6)}$  is the symmetric, uniformly positive definite and bounded inertia matrix of the robot;
- $H(q, \dot{q})_{(6 \times 1)}$  represents the contribution due to centrifugal, Coriolis and gravitational torques as well as the gas accumulator of axis two;
- $F_{fr}(\dot{q})_{(6 \times 1)}$  is the vector of friction at the robot joints;
- $F_e_{(6 \times 1)}$  is the vector of efforts applied by external environment on the robot expressed in the tool frame;
- $J^T(q)_{(6 \times 6)}$  is the transposed Jacobian matrix of the tool frame with respect to the base frame;
- $\Gamma_{(6 \times 1)}$  is the vector of gearbox output torque;
- $q$ ,  $\dot{q}$  and  $\ddot{q}$  represent the robot angular position, velocity and acceleration vectors respectively;
- $I_a_{(6 \times 6)}$  is the inertia matrix of the motor axes;
- $\Gamma_m_{(6 \times 1)}$  is the vector of motor torques;
- $F_{fm}(\dot{q})_{(6 \times 1)}$  is the vector of friction at the motor axes;
- $N^{-1}$  is the inverse of the gear transmissions ratio matrix  $N$ ;
- $\theta$ ,  $\dot{\theta}$  and  $\ddot{\theta}$  denote the vector of angular positions, velocities and accelerations of motor axes respectively;

Besides,  $H(q, \dot{q})_{(6 \times 1)}$  is the vector of centrifugal, Coriolis and gravitational torques on the robot side which can be expressed by:

$$H(q, \dot{q})_{(6 \times 1)} = B(q)_{(6 \times 15)}(\dot{q}\dot{q})_{(15 \times 1)} + C(q)_{(6 \times 6)}(\dot{q}^2)_{(6 \times 1)} + G(q)_{(6 \times 1)} \quad (2.54)$$



in which  $B(q)_{(6 \times 15)}$  denotes the coefficient matrix of Coriolis torques,  $C(q)_{(6 \times 6)}$  is the coefficient matrix of centrifugal torques,  $G(q)_{(6 \times 1)}$  is the vector of the gravitational torques,  $(\dot{q}\dot{q})_{(15 \times 1)} = [\dot{q}_1\dot{q}_2 \cdots \dot{q}_1\dot{q}_6 \dot{q}_2\dot{q}_3 \cdots \dot{q}_5\dot{q}_6]^T$ , and  $(\dot{q}^2)_{(6 \times 1)} = [\dot{q}_1^2 \cdots \dot{q}_6^2]^T$ .

For the robot KUKA KR500-2MT considered in this research, the gear transmissions ratio matrix  $N$  is not a diagonal matrix, and the couplings exist among axes 4, 5 and 6, as shown in the following form [4]:

$$N = \begin{bmatrix} N_{11} & 0 & 0 & 0 & 0 & 0 \\ 0 & N_{22} & 0 & 0 & 0 & 0 \\ 0 & 0 & N_{33} & 0 & 0 & 0 \\ 0 & 0 & 0 & N_{44} & 0 & 0 \\ 0 & 0 & 0 & N_{54} & N_{55} & 0 \\ 0 & 0 & 0 & N_{64} & N_{65} & N_{66} \end{bmatrix} \quad (2.55)$$

In addition, it should be noted that the torque of gravity compensator at the 2nd axis ( $\Gamma_{com2}$ ) must be considered in the dynamic model of the robot as well, and the detailed presentation of the gravity compensator can be found in [4].

## 2.4.2 Flexibility Model

Industrial robots have two main sources of flexibilities: the flexibilities of arms and those located at robot joints. According to the research work of Dumas [17], the dominating sources of flexibilities for the serial industrial robots are those located at the robot joints (including motors and transmissions). Thus, the robot links are considered as rigid and it is assumed that the flexibilities are only localized at gearboxes.

In this thesis, the concept of considering the joint flexibility as a linear torsional spring [20] is utilized for establishing the flexible joint model. By introducing a stiffness matrix  $K_r$ , the flexibility model can be given by:

$$\Gamma = K_r \tilde{q} \quad (2.56)$$

where  $\tilde{q}$  denotes the deformation of the robot defined in the joint space, and  $K_r$  is a  $6 \times 6$  diagonal matrix which includes the equivalent stiffness of six robot joints.

In the schematic of the flexible joint model shown in Figure 2.7,  $\theta_l$  is defined as the vector of angular positions of motor axes after the gear reduction, expressed as:

$$\theta_l = N^{-1}\theta \quad (2.57)$$

As a result, the robot deformation in joint space can be given by:

$$\tilde{q} = N^{-1}\theta - q \quad (2.58)$$

with  $\theta$  being the vector of angular positions of the motor axes, and  $q$  is the vector of angular positions of the robot axes.

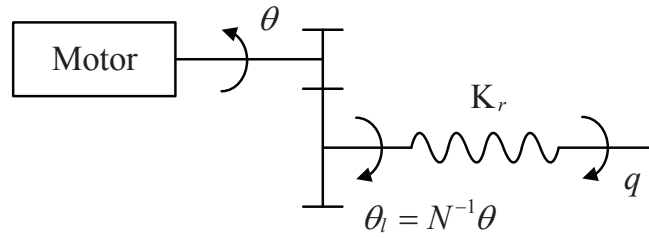


Figure 2.7: Schematic of the flexible joint model

### 2.4.3 Friction Model

Friction is universally present in the robot motion, and a lot of studies are carried out on the modeling of the friction at the robot and the motor axes, including the transmissions and motors. Advanced friction models are described in [66]. One dynamic model of friction called LuGre model was presented in [67]. Although the model is able to capture most of the friction behavior such as the Stribeck effect, hysteresis, spring-like characteristics for stiction, and varying break-away force, it is too complicated.

In our research, a classical friction model including the viscous and Coulomb friction is chosen for establishing the robot dynamic model. The friction at the robot and motor axes can be expressed as follows:

$$F_{fr}(\dot{q}) = B_s \dot{q} + F_s \text{sign}(\dot{q}) \quad (2.59)$$

$$F_{fm}(\dot{\theta}) = B_m \dot{\theta} + F_m \text{sign}(\dot{\theta}) \quad (2.60)$$

where  $B_m$  and  $B_s$  are the viscous friction coefficient matrices of the motor and joint axes;  $F_m$  and  $F_s$  are the matrices of Coulomb friction of the motor and joint axes.

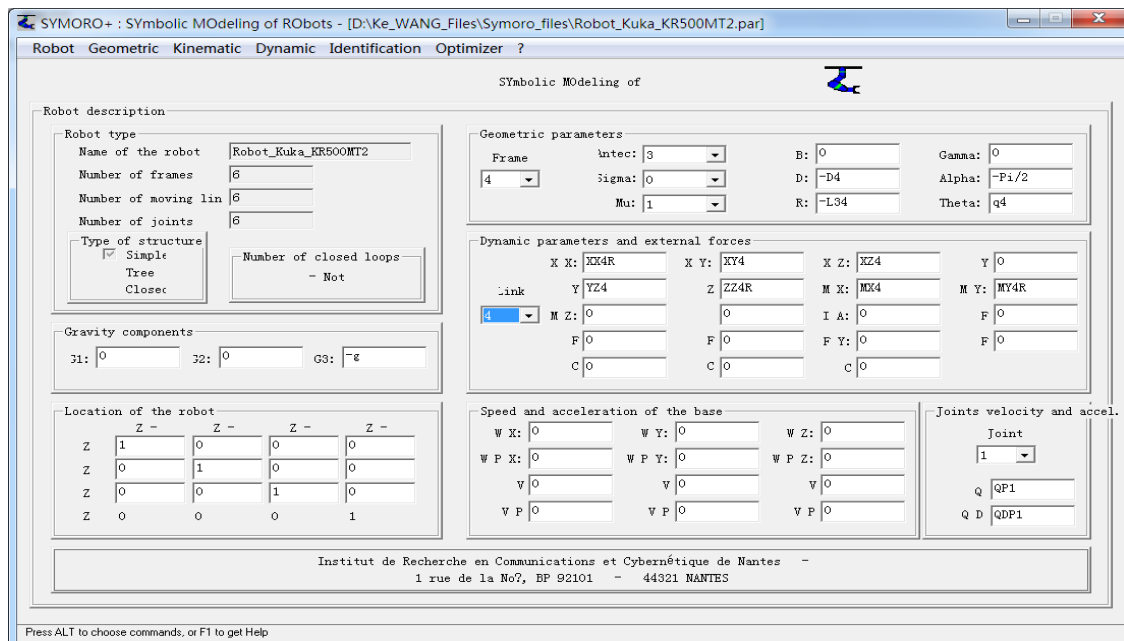


Figure 2.8: Interface of the software SYMORO+

#### 2.4.4 Introduction of SYMORO+

The software package *SYMORO+* (SYmbolic MOdelling of Robots) is an interactive software for the automatic generation of symbolic models of robots [68, 69]. It is based on the software *Mathematica* which has a strong capacity of symbolic calculations. *SYMORO+* provides the user with fast and efficient tools to generate various models which are needed in the control, simulation and robot design, such as the direct and inverse geometric models, the direct and inverse kinematic models, the direct and inverse dynamic models, Jacobian matrix, inertia matrix, Centrifugal, Coriolis and gravity torques, etc.

The software uses the modified Denavit-Hartenberg convention. After entering the necessary parameters into the interactive interface, the software is able to provide the symbolic models which can be directly used in Matlab, Mathematica, even C environment. The interface of the software *SYMORO+* can be seen in Figure 2.8.

## 2.5 Conclusion

In this chapter, several essential models of a 6-DOF flexible joint robot manipulator including direct geometric model (DGM), inverse geometric model (IGM), direct kinematic model (DKM), inverse kinematic model (IKM), direct dynamic model (DDM) and inverse dynamic model (IDM) are established based on the fundamental knowledge concerning the robot modeling, such as the spatial descriptions, the homogeneous transformations, the X-Y-Z fixed angles and the modified Denavit-Hartenberg convention, etc. Besides, the joint flexibility model, the friction model and the torque of gravity compensator at the second axis are studied as well. These models are extremely important in the robot model simplification, simulation of robotic FSW process and the design of robot force controller.

# Chapter 3

## Simplification of Robot Dynamic Model Using Interval Method

### 3.1 Introduction

This chapter focuses on the proposition of a novel simplification approach for the dynamic model of industrial robots by means of interval method [70]. Generally, low complexity models are highly preferred in the model-based control design. However, due to strong nonlinearities and complexities, some components of robot dynamic model such as the inertia matrix and the vector of centrifugal, Coriolis and gravitational torques, are very complicated to calculate for real-time control. By applying the obtained simplified model to practical control system, the computation time and memory occupation will be greatly reduced, particularly in the observer-based control presented by Qin et al. [4, 71, 72, 73]. What is more important is that this simplification is suitable for arbitrary trajectories in whole robot workspace. Moreover, the simplification contributes to finding negligible inertia parameters, which is very useful for robot model identification.

For most industrial robots applied to machining or Friction Stir Welding (FSW) processes, a high degree of accuracy and real-time performance can not be achieved. Many studies about industrial robot control have been performed to ensure a better tracking performance. Although the processing precision can be enhanced by using observer-based control methods [72], it will inevitably bring a more complicated control structure. As a result, a great deal of calculation and memory occupation will be implemented in robot controller, affecting the real-time performance. Therefore, model reduction methods are essential and helpful tools.

Model reduction of linear systems is a mature research topic and well-known methods featuring error bounds and preserved stability are available. However, in practice, one is often confronted with nonlinear systems and model reduction for this model class is so far an open research problem [74].

Industrial robots are highly coupled, nonlinear and time-varying systems. The currently available model reduction methods for nonlinear systems [74] include heuristic methods, proper orthogonal decomposition, balancing using energy functions, balancing empirical Gramians [75], linearization around equilibrium point, trajectory piecewise-linear approach [76], model reduction through system identification.

A computationally efficient reduction method relating to balanced truncation was also proposed in [77]. All these methods can reduce the number of states and improve calculation speed, while the limitation is that they only apply to specific trajectories or in a part of the workspace.

Interval method has been widely used in various applications, such as robotics, robust control, parameter and state estimation, etc [78, 79]. To solve actual tracking problems, a guaranteed recursive nonlinear state estimator based on interval analysis was proposed by Kieffer and Jaulin [80]. This technique was applied to the robust autonomous localization and tracking of mobile robots in [81, 82]. The main limitation lies in the explosion of complexity with the number of state variables. A similar robust navigation method was used for sailboat robots, and an interval-based method for the validation of reliable and robust navigation rules was presented [83]. In [84], the interval computation and constraint propagation were combined to deal with some difficult problems in robust control and nonlinear identification.

In the field of industrial robots, [85] presented a new interval-based approach to find the global minimum-jerk trajectory of a robot manipulator within a joint space scheme using cubic splines. The geometric design issue of serial-link robot manipulators with three revolute joints was solved for the first time by using interval analysis [86]. In [87], a novel approach based on interval method was used to tackle problems of dynamic self-collision detection and prevention for 2-DOF robot manipulator. An approach using interval analysis was proposed to the wrench-feasible workspace determination of  $n$ -DOF parallel robots driven by  $n$  or more cables [88]. The forward kinematic map of a serial manipulator was extended to intervals using the product of exponential formulation to analyze the kinematic errors [89].

As for interval-based model simplification, Martini presented an approach to simplify a 7-DOF helicopter model [90]. In this model reduction, the terms related to the components of the inertia matrix and the vector of centrifugal, Coriolis and gravitational torques were first studied from the point of view of specific trajectories, including the normal and helical ones. Then all the terms were analyzed by using interval method, which could be generalized to any trajectory. A simulation on the two trajectories was carried out to demonstrate the effectiveness of the method. However, this method only applies to some specific flight trajectories.

This chapter is organized as follows: Section 3.2 firstly introduces the basic definitions and operations of interval analysis as well as the inclusion function and overestimation. Then, the generation of the symbolic dynamic model of an industrial manipulator KUKA KR500-2MT is presented in Section 3.3. In Section 3.4, a generalized simplification algorithm is proposed, followed by a simple example of the simplification. The results of the simplified model for the whole workspace and corresponding error analysis as well as the further simplification are also given. In Section 3.5, case studies have been carried out on three different test trajectories including the identification trajectory, the linear FSW trajectory and the circular FSW trajectory, in order to verify the performance of the simplification algorithm. Besides, the torques in the robot dynamic equation has been analyzed by using the linear FSW trajectory as well. Section 3.6 discusses some usages of the simplification method. Finally, a conclusion is provided in Section 3.7.

## 3.2 Interval Analysis

Interval analysis, which was born in the 1960s, is a very useful mathematical tool for rigorous computations of lower and upper bounds of solutions to model equations when the intervals express the input arguments of the model instead of point values. With the aid of this theory, the equations can be extended to the interval domain, allowing the uncertainty propagation without specifying a distribution for the intervals [89]. In this section, the basic definitions and operations of intervals are introduced, and more details concerning intervals are presented in the books of Moore et al. [78] and Jaulin [79, 91].

### 3.2.1 Basic Definitions and Operations of Intervals

A real interval  $[x]$  (or scalar interval) is a closed, bounded and connected set of real numbers given by:

$$[x] = \{x \in \mathbb{R} \mid x^- \leq x \leq x^+\} \quad (3.1)$$

in which the left and right endpoints of the interval  $[x]$  are denoted by  $x^-$  and  $x^+$  respectively, and the set of all real intervals of  $\mathbb{R}$  is denoted by  $\mathbb{IR}$ . For simplicity, the interval  $[x]$  can be also given by:

$$[x] = [x^-, x^+] \quad (3.2)$$

Besides, the *width* of the interval  $[x]$  can be defined and denoted by:

$$w([x]) = x^+ - x^- \quad (3.3)$$

The *midpoint* of  $[x]$  is given by:

$$mid([x]) = \frac{x^- + x^+}{2} \quad (3.4)$$

The *absolute value* (also called *norm*) of  $[x]$ , is the maximum of the absolute values of its endpoints:

$$|[x]| = \max\{|x^-|, |x^+|\} \quad (3.5)$$

and it should be noted that  $|x| \leq |[x]|$  for every  $x \in [x]$ .

Let  $[x]$  and  $[y]$  be real compact intervals and the general form of the interval arithmetic operations are defined as:

$$[x] \odot [y] = \{x \odot y \mid x \in [x], y \in [y]\} \quad (3.6)$$

where  $\odot \in \{+, -, *, /, \max, \min, \dots\}$ , standing for one of the basic operations including addition, subtraction, multiplication, division, etc. Here we assume  $0 \notin [y]$  in the case of division.

As a consequence, these operations can be expressed in terms of the endpoints of intervals, and the following rules hold:

$$[x] + [y] = [x^- + y^-, x^+ + y^+] \quad (3.7)$$

$$[x] - [y] = [x^- - y^+, x^+ - y^-] \quad (3.8)$$

$$[x] * [y] = [\min\{x^-y^-, x^+y^-, x^-y^+, x^+y^+\}, \max\{x^-y^-, x^+y^-, x^-y^+, x^+y^+\}] \quad (3.9)$$

$$[x]/[y] = [x] * \frac{1}{[y]}, \quad \text{where } \frac{1}{[y]} = [1/y^+, 1/y^-], \quad \text{if } 0 \notin [y] \quad (3.10)$$

$$\max([x], [y]) = [\max(x^-, y^-), \max(x^+, y^+)] \quad (3.11)$$

Similarly, the notion of real intervals and their arithmetic operations can be extended to include interval vectors and interval matrices as well. The interval vectors, interval matrices and interval functions are denoted by boldface letters.

An  $n$ -dimensional vector interval (also called a box)  $[\mathbf{x}]$  is an interval of  $\mathbb{R}^n$ . It corresponds to the Cartesian product of  $n$  intervals of  $\mathbb{IR}$ :

$$\begin{aligned} [\mathbf{x}] &= [\mathbf{x}^-, \mathbf{x}^+] \\ &= [x_1] \times [x_2] \times \cdots \times [x_n] \\ &= [x_1^-, x_1^+] \times [x_2^-, x_2^+] \times \cdots \times [x_n^-, x_n^+] \end{aligned} \quad (3.12)$$

The set of all boxes of  $\mathbb{R}^n$  will be denoted by  $\mathbb{IR}^n$ . The *width*, *midpoint* and *norm* of a box  $[\mathbf{x}]$  can be defined as follows:

$$w([\mathbf{x}]) = \max( w([x_1]), w([x_2]), \dots, w([x_n]) ) \quad (3.13)$$

$$\text{mid}([\mathbf{x}]) = (\text{mid}([x_1]), \text{mid}([x_2]), \dots, \text{mid}([x_n])) \quad (3.14)$$

$$\|[\mathbf{x}]\| = \max( |[x_1]|, |[x_2]|, \dots, |[x_n]| ) \quad (3.15)$$

### 3.2.2 Inclusion Functions and Overestimation

As a matter of fact, the essential part of the interval analysis is the extension of real functions to intervals which is able to calculate outer approximations of the image of intervals by real functions [92]. Thus, it is of importance to present the definitions of inclusion function and minimal inclusion function.

#### Definition 1: Inclusion Function $[\mathbf{f}]$

Considering the case of a function  $\mathbf{f}$  from  $\mathbb{R}^n$  to  $\mathbb{R}^m$  and an interval function denoted by  $[\mathbf{f}]$  from  $\mathbb{IR}^n$  to  $\mathbb{IR}^m$ , the interval function  $[\mathbf{f}]$  is an inclusion function of  $\mathbf{f}$  if and only if [93]:

$$\forall [\mathbf{x}] \in \mathbb{IR}^n, \mathbf{f}([\mathbf{x}]) \subseteq [\mathbf{f}]([\mathbf{x}]) \quad (3.16)$$

#### Definition 2: Minimal Inclusion Function $[\mathbf{f}]^*$

For the given function  $\mathbf{f}$  from  $\mathbb{R}^n$  to  $\mathbb{R}^m$ , the minimal inclusion function of  $\mathbf{f}$ , expressed by  $[\mathbf{f}]^*$ , can be defined as [93]:

$$[\mathbf{f}]^* : \mathbb{IR}^n \rightarrow \mathbb{IR}^m, [\mathbf{x}] \rightarrow \{f(\mathbf{x}) \mid \mathbf{x} \in [\mathbf{x}]\} \quad (3.17)$$

Therefore, the minimal inclusion function  $[\mathbf{f}]^*([\mathbf{x}])$  is the smallest box of  $\mathbb{IR}^m$  which include  $\mathbf{f}([\mathbf{x}])$ . Compared with the minimal inclusion function  $[\mathbf{f}]^*$ , the inclusion function  $[\mathbf{f}]$  is not unique. According to the above definitions, we can infer that

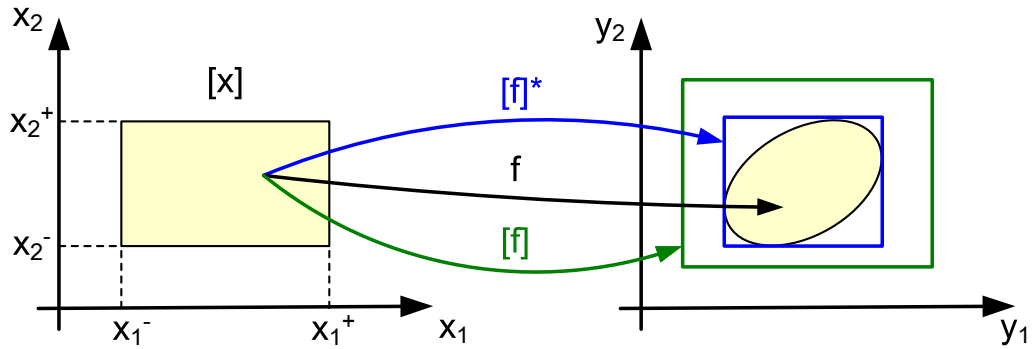


Figure 3.1: The function  $\mathbf{f}$  and its inclusion function  $[\mathbf{f}]$  as well as minimal inclusion function  $[\mathbf{f}]^*$  from  $\mathbb{X}$  to  $\mathbb{Y}$  ( $\mathbb{X}$  and  $\mathbb{Y}$  correspond to  $\mathbb{R}^2$ )

$[\mathbf{f}]^* \subset [\mathbf{f}]$ . Figure 3.1 illustrates the function  $\mathbf{f}$  and its inclusion function  $[\mathbf{f}]$  as well as the minimal inclusion function  $[\mathbf{f}]^*$  in the case where  $\mathbb{X}$  and  $\mathbb{Y}$  correspond to  $\mathbb{R}^2$ . It can be seen that the minimal inclusion function  $[\mathbf{f}]^*$  is the most accurate interval extension that one would like to construct.

For elementary functions  $f \in \{+, -, *, /, exp, sin, cos, \dots\}$ , their inclusion functions can be obtained by directly computing the exact ranges of the functions. However, computing the minimal inclusion function of a polynomial function  $\mathbf{x}$  is proven to be a NP-hard problem [92]. For arbitrary function  $\mathbf{x}$  which is composed of elementary operators, a commonly-used approach to compute its inclusion function is to replace each of the elementary operators by its minimal inclusion function in the formal expression of  $\mathbf{x}$ . The resulting inclusion function is referred to be natural interval extension [93].

For instance, for a given function expressed by:

$$f(x) = x^2 - 2x + 1 \quad (3.18)$$

its natural inclusion function is:

$$[f]_1([x]) = [x]^2 - 2 * [x] + 1 \quad (3.19)$$

It should be noted that the overestimation for the range of the function may occur during the process of interval extensions, which is due to the correlation among the variables. For example, the function  $f(x)$  in equation (3.18) can be rewritten as:

$$f(x) = (x - 1)^2 \quad (3.20)$$

Thus, the corresponding natural inclusion function of the above function is:

$$[f]_2([x]) = ([x] - 1)^2 \quad (3.21)$$

Supposing that the interval  $[x] = [1, 2]$ , equation (3.19) and (3.21) can be calculated as follows:

$$\begin{aligned} [f]_1([1, 2]) &= [1, 2]^2 - 2 * [1, 2] + 1 \\ &= [1, 4] - [2, 4] + 1 \\ &= [-2, 3] \end{aligned} \quad (3.22)$$



$$[f]_2([1, 2]) = ([1, 2] - 1)^2 = [0, 1]^2 = [0, 1] \quad (3.23)$$

Apparently, it can be seen that substitution of minimal inclusion functions for elementary operators in different expressions of  $f(x)$  yields different inclusion functions, leading to different results for the range of function. In this case, the correct range of the function  $f(x)$  is  $[0, 1]$  when  $x \in [1, 2]$ . Since  $[x]$  appears only once in  $[f_2]$ , that is to say,  $[x]$  can be considered as the independent variable, the overestimation for the range of the function is nonexistent. On the contrary,  $[x]^2$  and  $[x]$  are related in  $[f_1]$ , thus the overestimation occurs. As a consequence, although both of  $[f_1]$  and  $[f_2]$  are inclusion functions of the function  $f$ , the minimal inclusion function is  $[f_2]$ .

In order to reduce overestimation, other interval extensions like centered form, mean value form, slope form, monotonicity test form, can be constructed other than the natural interval extension, and the related algorithms based on refinements are discussed in [78, 79]. In the first step of simplification algorithm presented in Section 3.4.1, some trigonometric transformations are implemented so as to remove the correlation among the variables, avoiding the occurrence of the overestimation.

### 3.3 Symbolic Robot Dynamic Model

In the previous chapter, the modeling of the 6-DOF industrial robot manipulator KUKA KR500-2MT (as shown in Figure 2.5) has already been accomplished, and we know that the dynamic model of the flexible joint robot can be obtained in the following form [8]:

$$M(q)\ddot{q} + H(q, \dot{q}) + F_{fr}(\dot{q}) - J^T(q)F_e = \Gamma \quad (3.24)$$

$$I_a\ddot{\theta} + N^{-T}\Gamma + F_{fm}(\dot{\theta}) = \Gamma_m \quad (3.25)$$

where  $M(q)_{(6 \times 6)}$  is the symmetric, positive definite inertia matrix,  $F_{fr}(\dot{q})_{(6 \times 1)}$  is the vector of friction at the robot joints,  $F_e_{(6 \times 1)}$  is the vector of efforts applied by external environment on the robot expressed in the tool frame,  $J^T(q)_{(6 \times 6)}$  is the transposed Jacobian matrix of the tool frame with respect to the base frame,  $\Gamma_{(6 \times 1)}$  is the vector of gearbox output torque,  $q$ ,  $\dot{q}$  and  $\ddot{q}$  represent the robot angular position, velocity and acceleration vectors respectively.  $I_{a(6 \times 6)}$  is the inertia matrix of the motor axes,  $\Gamma_{m(6 \times 1)}$  is the vector of motor torques,  $F_{fm}(\dot{q})_{(6 \times 1)}$  is the vector of friction at the motor axes,  $N$  is the gear transmission ratio matrix,  $\theta$ ,  $\dot{\theta}$  and  $\ddot{\theta}$  denote the vector of angular positions, velocities and accelerations of motor axes respectively.

Besides,  $H(q, \dot{q})_{(6 \times 1)}$  is the vector of centrifugal, Coriolis and gravitational torques on the robot side which can be expressed by:

$$H(q, \dot{q})_{(6 \times 1)} = B(q)_{(6 \times 15)}(\dot{q}\dot{q})_{(15 \times 1)} + C(q)_{(6 \times 6)}(\dot{q}^2)_{(6 \times 1)} + G(q)_{(6 \times 1)} \quad (3.26)$$

in which  $B(q)_{(6 \times 15)}$  denotes the coefficient matrix of Coriolis torques,  $C(q)_{(6 \times 6)}$  is the coefficient matrix of centrifugal torques,  $G(q)_{(6 \times 1)}$  is the vector of the gravitational torques,  $(\dot{q}\dot{q})_{(15 \times 1)} = [\dot{q}_1\dot{q}_2 \cdots \dot{q}_1\dot{q}_6 \dot{q}_2\dot{q}_3 \cdots \dot{q}_5\dot{q}_6]^T$ , and  $(\dot{q}^2)_{(6 \times 1)} = [\dot{q}_1^2 \cdots \dot{q}_6^2]^T$ .

In addition, a geometric description of the robot manipulator based on the modified Denavit-Hartenberg convention and the corresponding geometric parameters are given in Figure 2.6 and Table 2.1 respectively. The description facilitates to

calculate symbolic expressions of the geometric, kinematic and dynamic model of the robot with the help of the software *SYMORO+* [68, 69] or other calculation techniques. The values of the robot geometric parameters used for the simplification are provided in Table 3.1.

Table 3.1: Values of the robot geometric parameters

| Length | $L_{1x}$ | $L_{1z}$ | $L_2$ | $L_{34}$ | $D_4$ | $L_5$ | $L_{tz}$ |
|--------|----------|----------|-------|----------|-------|-------|----------|
| (m)    | 0.500    | 1.045    | 1.300 | 1.025    | 0.055 | 0.235 | 0.435    |

Actually, the symbolic robot dynamic model can be generated by the software *SYMORO+* based on the Lagrange method or Newton-Euler method [68, 69]. Compared with the Newton-Euler method, the Lagrange approach is able to consider the robot as a whole rather than treat each axis of the robot respectively. However, the Newton-Euler method is more efficient than the Lagrange method in computation. Thus, low computation time can be achieved by implementing the Newton-Euler method. The vector  $H(q, \dot{q})$  in equation (3.26) can be calculated by using Newton Euler algorithm when putting the joint accelerations  $\ddot{q}$  equal to zero. The schematic for generating the symbolic robot dynamic model is shown in Figure 3.2, and each link of the robot is modeled by 11 inertia parameters. The simplification method will be applied to the resulting symbolic robot dynamic model.

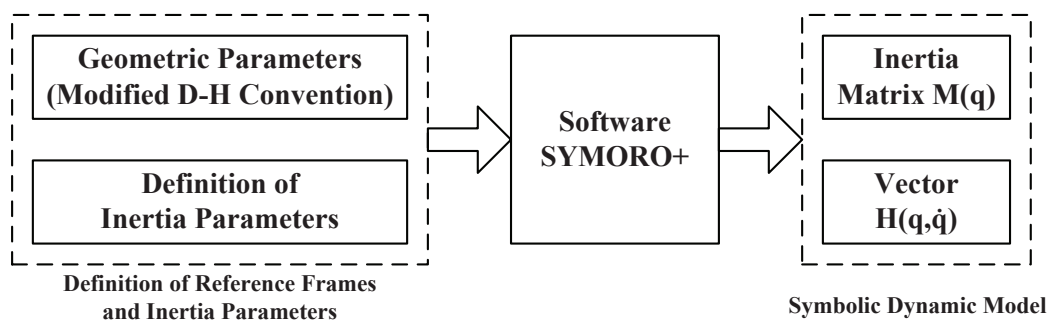


Figure 3.2: The schematic of generation of the symbolic robot dynamic model

### 3.4 Simplification Using Interval Method

The expressions of some components in equation (3.24) are extremely complicated, such as the inertia matrix  $M(q)$  and the vector of centrifugal, Coriolis and gravitational torques  $H(q, \dot{q})$ . For the sake of showing the complexity, the original expression of one component  $M_{21}$  of inertia matrix  $M(q)$  is given in Appendix B.1. All expressions of these components show strong nonlinearities and a high coupling between the control inputs, which makes it difficult to design the control law of the robot. Therefore, it is interesting to simplify these expressions in order to acquire a simple model containing essential terms and characteristics of the robot manipulator.

There exist various model reduction methods for nonlinear systems, however, most of them are only applicable to specific trajectories. According to the specification of robot KUKA KR500-2MT, the range of angular motion, velocity and

acceleration for each axis are given in Table 3.2. As shown in this table, all the variables about trajectories in operational workspace are intervals. As a result, interval method will be an easier and more effective way to simplify robot dynamic model compared with other methods. More importantly, the interval-method-based simplification can be definitely applied to arbitrary trajectories in the robot workspace.

Table 3.2: Range of robot motion, velocity and acceleration

| Axis | Angular Motion              | Velocity          | Acceleration       |
|------|-----------------------------|-------------------|--------------------|
| 1    | $\pm 185^\circ$             | $\pm 42^\circ/s$  | $\pm 59^\circ/s^2$ |
| 2    | $-130^\circ$ to $+20^\circ$ | $\pm 42^\circ/s$  | $\pm 43^\circ/s^2$ |
| 3    | $-94^\circ$ to $+150^\circ$ | $\pm 42^\circ/s$  | $\pm 78^\circ/s^2$ |
| 4    | $\pm 350^\circ$             | $\pm 76^\circ/s$  | $\pm 42^\circ/s^2$ |
| 5    | $\pm 118^\circ$             | $\pm 74^\circ/s$  | $\pm 92^\circ/s^2$ |
| 6    | $\pm 350^\circ$             | $\pm 123^\circ/s$ | $\pm 68^\circ/s^2$ |

### 3.4.1 Simplification Algorithm

The simplification algorithm can be applied to the symbolic expression of the robot dynamic model which includes the inertia matrix  $M(q)_{(6 \times 6)}$  and the vector of centrifugal, Coriolis and gravitational torques  $H(q, \dot{q})_{(6 \times 1)}$ . Besides, the coefficient matrix of Coriolis torques  $B(q)_{(6 \times 15)}$ , the coefficient matrix of centrifugal torques  $C(q)_{(6 \times 6)}$  and the vector of the gravitational torques  $G(q)_{(6 \times 1)}$  can be simplified by the algorithm as well.

For every component of the above matrices and vectors in equation (3.24) and (3.26), its expression can be described as:

$$\begin{aligned}
 M_{ij} &= f_1(sq, cq) \\
 H_i &= f_2(sq, cq, \dot{q}) \\
 B_{ik} &= f_3(sq, cq) \\
 C_{ij} &= f_4(sq, cq) \\
 G_i &= f_5(sq, cq)
 \end{aligned} \tag{3.27}$$

with

$$\begin{aligned}
 sq &= [\sin(q_1), \sin(q_2), \sin(q_3), \sin(q_4), \sin(q_5), \sin(q_6)]^T \\
 cq &= [\cos(q_1), \cos(q_2), \cos(q_3), \cos(q_4), \cos(q_5), \cos(q_6)]^T \\
 \dot{q} &= [\dot{q}_1, \dot{q}_2, \dot{q}_3, \dot{q}_4, \dot{q}_5, \dot{q}_6]^T
 \end{aligned}$$

where  $i, j \in \{1, 2, 3, 4, 5, 6\}$ ,  $k \in \{1, 2, \dots, 15\}$ , and  $f_1$  to  $f_5$  are the corresponding mapping functions.

Since the variables  $sq$ ,  $cq$ ,  $\dot{q}$  and  $\ddot{q}$  are all intervals, the target object of the simplification algorithm can be generalized as:

$$\begin{aligned}
 \Psi &= f(sq, cq, \dot{q}, \ddot{q}) \\
 &= \sum_{i=1}^n T_i
 \end{aligned} \tag{3.28}$$

where  $\ddot{q} = [\ddot{q}_1, \ddot{q}_2, \ddot{q}_3, \ddot{q}_4, \ddot{q}_5, \ddot{q}_6]^T$ ,  $\Psi$  is the target object to be simplified,  $f$  is the corresponding mapping function,  $T_i$  denotes the  $i^{th}$  term of  $\Psi$  and  $i \in \{1, 2, \dots, n\}$ ,  $n$  is the total number of terms in the expression of  $\Psi$ .

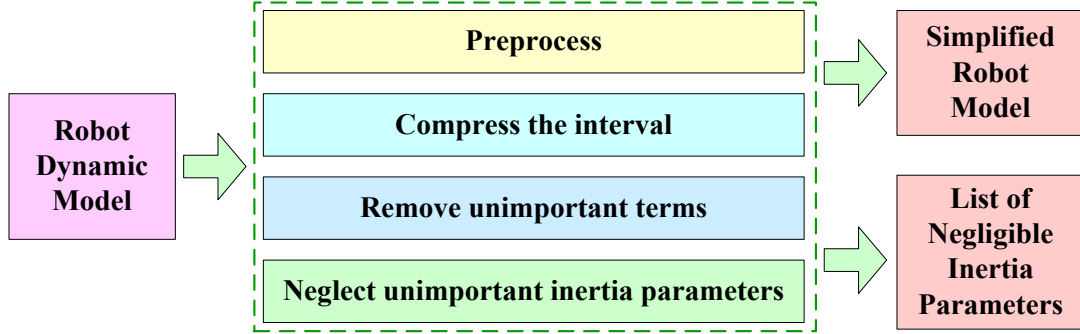


Figure 3.3: Diagram of the simplification algorithm process

As shown in Figure 3.3, the simplification algorithm proposed in this thesis is aiming at obtaining the simplified robot dynamic model and finding the negligible inertia parameters, and the whole process can be divided into 3 steps:

### Step 1. Compress the interval

First of all, some trigonometric transformations can be used to rewrite the products and powers of sine and cosine functions in the expression of target object  $\Psi$ , in terms of trigonometric functions with combined arguments. For example:

$$\begin{aligned}
 2 \sin^2(x) &= 1 - \cos(2x) \\
 2 \cos^2(x) &= 1 + \cos(2x) \\
 2 \sin(x) \cos(x) &= \sin(2x) \\
 2 \sin(x) \sin(y) &= \cos(x - y) - \cos(x + y) \\
 2 \cos(x) \cos(y) &= \cos(x - y) + \cos(x + y) \\
 2 \sin(x) \cos(y) &= \sin(x + y) + \sin(x - y)
 \end{aligned} \tag{3.29}$$

After combining similar terms, the following transformation can be carried out:

$$a \cos(x) + b \sin(x) = \sqrt{a^2 + b^2} \cos(x - \arctan(a, b)) \tag{3.30}$$

The above trigonometric transformations will be repeatedly performed until the expression of  $\Psi$  no longer changes. The transformations can be realized by using the functions "*TrigReduce*", "*TrigFactor*" and "*ReplaceRepeated*" of the software *Mathematica*. After this step, a linear expression involving trigonometric functions with more complicated arguments is yielded.

By means of the above trigonometric transformations, the correlations among some terms are removed, reducing the overestimation to a certain extent. For instance, if we assume that the interval  $[x] \in [-\pi, \pi]$ , the interval of left side of equation (3.30) is  $[-|a| - |b|, |a| + |b|]$ , and the interval of right side is  $[-\sqrt{a^2 + b^2}, \sqrt{a^2 + b^2}]$ . It can be clearly seen that tighter bounds are obtained. As a result, the interval of  $\Psi$  will be compressed greatly, which is extremely necessary for the following steps.

### Step 2. Remove unimportant terms

The norm of an interval, which is also called the absolute value, can be gained from equation (3.5). Since most of terms in the expression of  $\Psi$  are intervals, it is possible to compare the terms by calculating their norms. Thus, the unimportant terms of  $\Psi$  can be removed in this step.

By defining  $k_t$  as a proportional factor, any term  $T_i$  in the expression of  $\Psi$  can be removed if it satisfies:

$$\frac{|T_i|}{g(|T_1|, |T_2|, \dots, |T_n|)} \leq k_t \quad (i = 1, \dots, n) \quad (3.31)$$

where  $g \in \{max, sum, rms, \dots\}$ , denoting one of the mapping functions which can find the maximum value, sum and root mean square value of the inputs,  $n$  is the total number of terms in the expression of  $\Psi$ .

### Step 3. Neglect unimportant inertia parameters

As a matter of fact, for a given term  $T_i$  in the expression of  $\Psi$ , not every inertia parameter in  $T_i$  has a great influence on the norm of  $T_i$ . By defining  $k_p$  as a new proportional factor, any inertia parameter  $P$  can be neglected if it meets the following condition:

$$e_p = \frac{||T_i|_{P=0} - |T_i||}{|T_i|} \leq k_p \quad (3.32)$$

where  $e_p$  denotes the percentage of the norm error of  $T_i$  when putting the inertia parameter  $P$  equal to zero from the original norm of  $T_i$ .

Finally, the target object  $\Psi$  can be simplified as  $\Psi_s$ , and the flow chart of the simplification algorithm for programming is given in Figure 3.4.

## 3.4.2 Example of Simplification of Component $M_{44}$

For the sake of simplicity, a relatively simple component  $M_{44}$  of inertia matrix  $M(q)$  will be taken as an instance to present how to implement the above simplification algorithm. The original component  $M_{44}$  is formulated by:

$$\begin{aligned} M_{44} = & ZZ_{4R} + ZZ_6c_5^2 + 2XY_5c_5s_5 + 2L_5MY_6c_5c_6s_5 \\ & + 2YZ_6c_5c_6s_5 + XX_{5R}s_5^2 + 2L_5MX_6c_5s_5s_6 \\ & + 2XZ_6c_5s_5s_6 + 2XY_6c_6s_5^2s_6 + XX_{6R}s_5^2s_6^2 \end{aligned} \quad (3.33)$$

where  $s_i$ ,  $c_i$  denote  $\sin(q_i)$  and  $\cos(q_i)$  respectively,  $ZZ_{4R}$ ,  $ZZ_6$ ,  $XY_5$ ,  $MY_6$ ,  $YZ_6$ ,  $XX_{5R}$ ,  $MX_6$ ,  $XZ_6$ ,  $XY_6$ , and  $XX_{6R}$  are 10 of 36 regrouped scalar base inertia parameters used in the robot modeling. Comparing to  $M_{44}$ , the component  $M_{21}$  is extremely complicated since it is a function of almost all the 36 inertia parameters which is shown in Appendix B.1.

In addition, the values of the inertia parameters used in the simplification are provided in Table 3.3. In fact, these values of inertia parameters are obtained through the identification. More details related to the inertia parameters and their identification can be found in [4, 94].

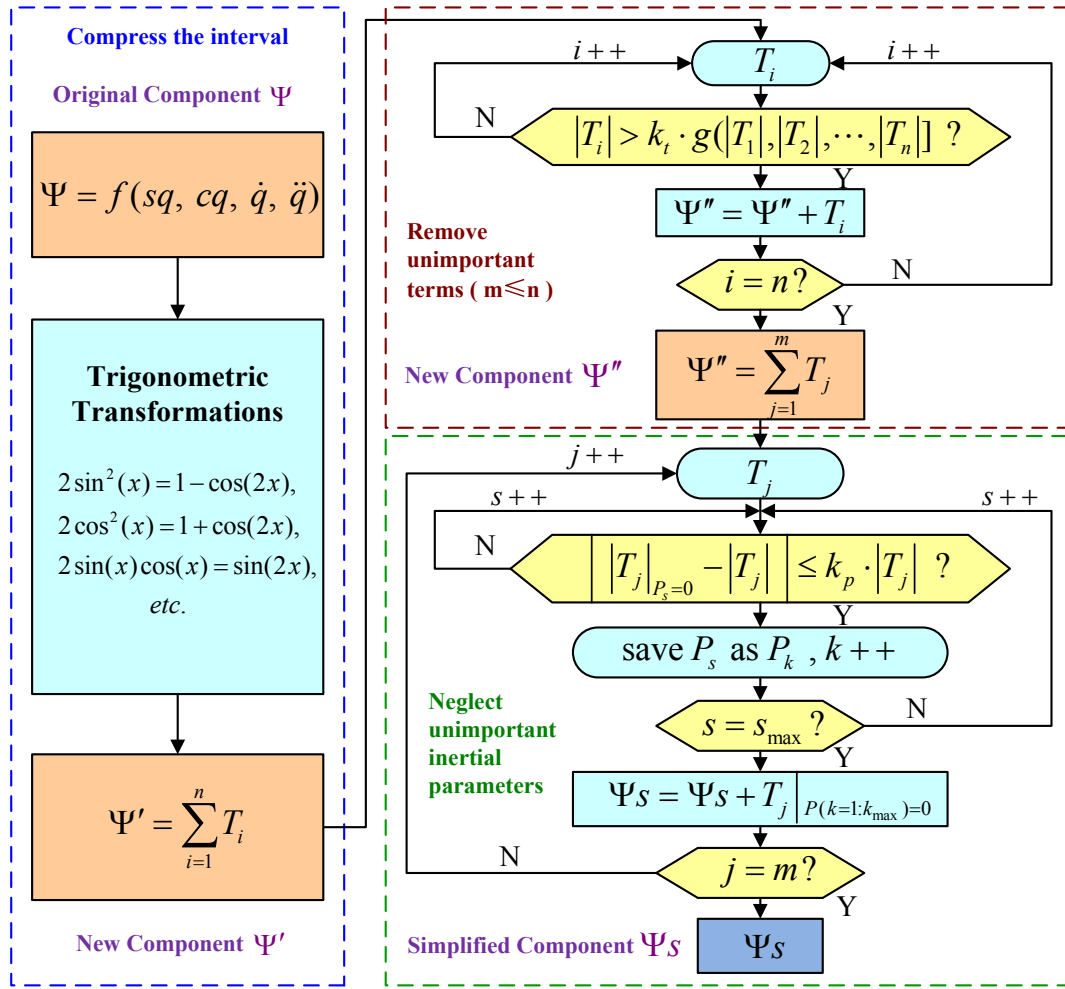


Figure 3.4: Flow chart of the simplification program

Table 3.3: Values of the inertia parameters used in the simplification

| Parameter | Value (s.i.) | Parameter | Value (s.i.) | Parameter | Value (s.i.) |
|-----------|--------------|-----------|--------------|-----------|--------------|
| $XX_{2R}$ | -1100.9      | $XZ_{2R}$ | -265.7       | $MX_{2R}$ | 117.3        |
| $XX_{3R}$ | 265.7        | $XZ_3$    | 2.97         | $MX_{3R}$ | -45.1        |
| $XX_{4R}$ | 13.6         | $XZ_4$    | 70.3         | $MX_4$    | -2.59        |
| $XX_{5R}$ | -16.8        | $XZ_5$    | 55.5         | $MX_5$    | 10.9         |
| $XX_{6R}$ | -40.5        | $XZ_6$    | -2.2         | $MX_6$    | 6.42         |
| $XY_2$    | -787.3       | $YZ_2$    | 673.2        | $MY_2$    | -395.3       |
| $XY_{3R}$ | 16.5         | $YZ_3$    | 18.9         | $MY_{3R}$ | -113.6       |
| $XY_4$    | 50           | $YZ_4$    | 11.6         | $MY_{4R}$ | 20.2         |
| $XY_5$    | -17.1        | $YZ_5$    | -70.1        | $MY_{5R}$ | -23.4        |
| $XY_6$    | 11.6         | $YZ_6$    | -3.84        | $MY_6$    | 22.5         |
| $ZZ_{1R}$ | 5729.4       | $ZZ_{3R}$ | 319.5        | $ZZ_{5R}$ | 79.8         |
| $ZZ_{2R}$ | 5731.4       | $ZZ_{4R}$ | 142.5        | $ZZ_6$    | 23.4         |

Firstly, by performing the trigonometric transformations in the first step, the component  $M_{44}$  can be rewritten as:

$$\begin{aligned}
 M'_{44} &= \sum_{i=1}^{10} T_i \\
 &= ZZ_{4R} + \frac{1}{2}ZZ_6 + \frac{1}{2}XX_{5R} + \frac{1}{4}XX_{6R} \\
 &\quad + \frac{1}{4}t_0 \cos(2q_6 - \arctan(-XX_{6R}, 2XY_6)) \\
 &\quad + \frac{1}{8}t_0 \cos(2q_5 + 2q_6 - \arctan(XX_{6R}, -2XY_6)) \\
 &\quad + \frac{1}{8}t_0 \cos(2q_5 - 2q_6 - \arctan(XX_{6R}, 2XY_6)) \\
 &\quad + \frac{1}{2}\sqrt{t_1^2 + t_2^2} \cos(2q_5 + q_6 - \arctan(-t_1, t_2)) \\
 &\quad + \frac{1}{2}\sqrt{t_1^2 + t_2^2} \cos(2q_5 - q_6 - \arctan(t_1, t_2)) \\
 &\quad + \frac{1}{4}\sqrt{t_3^2 + t_4^2} \cos(2q_5 - \arctan(t_3, t_4))
 \end{aligned} \tag{3.34}$$

where  $T_i$  denotes the  $i^{\text{th}}$  term in above expression of  $M'_{44}$ ,  $t_0 = \sqrt{XX_{6R}^2 + 4XY_6^2}$ ,  $t_1 = L_5MX_6 + XZ_6$ ,  $t_2 = L_5MY_6 + YZ_6$ ,  $t_3 = 2ZZ_6 - 2XX_{5R} - XX_{6R}$ ,  $t_4 = 4XY_5$ .

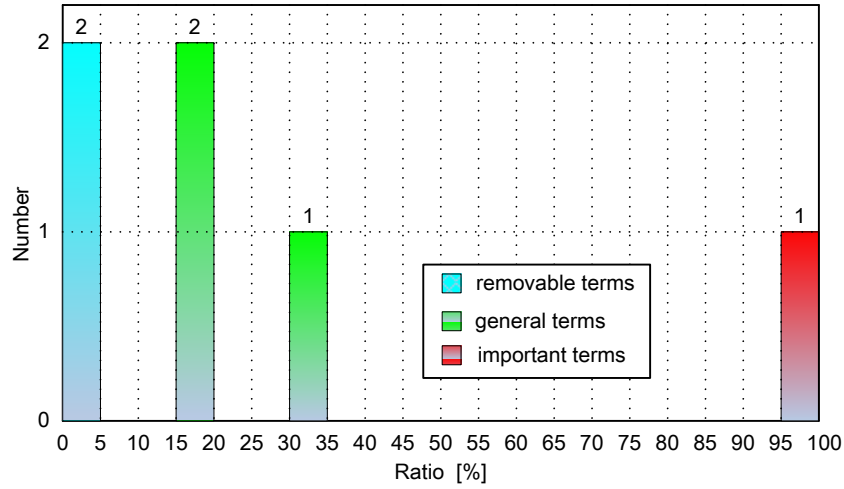
By developing a *Mathematica* program based on the above algorithm, the tighter bounds of  $M_{44}$  have been computed. The results show that the interval of  $M_{44}$  is compressed greatly from  $[-8.8580, 306.27]$  to  $[76.007, 195.34]$ , and the total number of terms in the expression of  $M_{44}$  is reduced from 16 to 10. It should be noted that the trigonometric transformations in this step are implemented without any approximation although the total number of terms is reduced. Actually, it is proved that more exact range of  $M_{44}$  can be obtained by using the natural interval extension of  $M'_{44}$  and these operations have a high capability of reducing overestimation.

Next, the second step of the simplification algorithm will be applied to  $M'_{44}$  in equation (3.34) in order to remove unimportant terms. In this study, the maximum value function *max* is adopted for the mapping function  $g$  in equation (3.31). As a consequence, the ratio of the norm of each term ( $|T_i|$ ) to the maximum one ( $|T_i|_{\max}$ ) can be calculated, as shown in Table 3.4, except the first 4 constant terms.

Table 3.4: The ratio of  $|T_i|$  to  $|T_i|_{\max}$

| Term ( $T_i$ )   | $T_5$  | $T_6$ | $T_7$ | $T_8$ | $T_9$ | $T_{10}$ |
|------------------|--------|-------|-------|-------|-------|----------|
| Norm ( $ T_i $ ) | 11.669 | 5.834 | 5.834 | 0.802 | 0.802 | 34.727   |
| Ratio (%)        | 33.6   | 16.8  | 16.8  | 2.30  | 2.30  | 100      |

The histogram statistics in each 5% section can be obtained in Figure 3.5 as well. The red part and the green part represent the important terms and the general terms in the component respectively, and the blue part denotes the unimportant terms which can be removed in the simplification. According to Table 3.4 and Figure 3.5, the appropriate proportional factor  $k_t$  can be chosen as 5%. As a result, the term  $T_8$  and  $T_9$  in expression (3.34) will be removed. The simplified component  $M''_{44}$  is in the range of  $[77.611, 193.74]$ , approximately equal to the interval of the component before this step of the simplification ( $M'_{44}$ ).


 Figure 3.5: Histogram statistics of term norm ratio for  $M_{44}$ 

Then, the third step of the simplification algorithm is implemented to the newly generated component  $M''_{44}$  so as to find unimportant inertia parameters. Take the term  $T_{10}$  in equation (3.34) for an instance, the inertia parameters including  $XX_{5R}$ ,  $XY_5$ ,  $XX_{6R}$ , and  $ZZ_6$  can be found in its expression formulated by:

$$T_{10} = \frac{1}{4} \sqrt{(2ZZ_6 - 2XX_{5R} - XX_{6R})^2 + (4XY_5)^2} \cos(2q_5 - \arctan(2ZZ_6 - 2XX_{5R} - XX_{6R}, 4XY_5)) \quad (3.35)$$

Based on equation (3.32), the percentage errors  $e_p$  can be calculated when putting the four inertia parameters equal to zero respectively, and the results are given in Table 3.5. If the proportional factor  $k_p$  is assigned to 15%, the inertia parameter  $XY_5$  can be neglected.

 Table 3.5: Norm of term  $T_{10}$  and error  $e_p(\%)$  when  $P = 0$ 

| $P$              | <i>none</i> | $XX_{5R}$ | $XY_5$ | $XX_{6R}$ | $ZZ_6$ |
|------------------|-------------|-----------|--------|-----------|--------|
| $ T_{10} _{P=0}$ | 34.727      | 27.726    | 30.225 | 26.390    | 25.211 |
| $e_p(\%)$        | 0           | 20.16     | 12.96  | 24.01     | 27.40  |

Finally, the simplified component  $M_{s44}$  can be obtained, which is in the range of [73.355, 197.99]. The norms of all the generated components during the simplification process are provided in Table 3.6. Obviously, the change of norm from 306.27 to 195.34 represents the reduction of overestimation, and the norm of simplified component  $M_{s44}$  is very close to the norm of  $M'_{44}$ .

Table 3.6: Norms of all generated components during simplification process

| Components | $M_{44}$ | $M'_{44}$ | $M''_{44}$ | $M_{s44}$ |
|------------|----------|-----------|------------|-----------|
| Norm       | 306.27   | 195.34    | 193.74     | 197.99    |



### 3.4.3 Results of Simplified Model for Whole Workspace

In accordance with the flow chart of the simplification algorithm (see Figure 3.4), programs are developed by using the software Mathematica which is well-known for the symbolic and interval calculations.

Owing to interval approach, the proportional factors  $k_t$  and  $k_p$  do not depend on initial states of the robot, and the corresponding results are valid for the whole robot workspace. By choosing  $k_t = 5\%$  and  $k_p = 1\%$ , the number of terms in the components of inertia matrix  $M(q)$  and vector  $H(q, \dot{q})$  as well as the simplified inertia matrix  $M_s(q)$  and vector  $H_s(q, \dot{q})$  are provided in Table 3.7 and Table 3.8.

Table 3.7: Number of terms in  $M(q)$  and  $M_s(q)$  when  $k_t = 5\%$  and  $k_p = 1\%$

| $M(q)$   | $N_{terms}$ | $M_s(q)$  | $N_{terms}$ | $M(q)$   | $N_{terms}$ | $M_s(q)$  | $N_{terms}$ |
|----------|-------------|-----------|-------------|----------|-------------|-----------|-------------|
| $M_{11}$ | 238         | $M_{s11}$ | 11          | $M_{52}$ | 35          | $M_{s52}$ | 24          |
| $M_{21}$ | 112         | $M_{s21}$ | 4           | $M_{53}$ | 17          | $M_{s53}$ | 14          |
| $M_{22}$ | 73          | $M_{s22}$ | 18          | $M_{54}$ | 5           | $M_{s54}$ | 4           |
| $M_{31}$ | 93          | $M_{s31}$ | 53          | $M_{55}$ | 3           | $M_{s55}$ | 3           |
| $M_{32}$ | 72          | $M_{s32}$ | 26          | $M_{61}$ | 40          | $M_{s61}$ | 36          |
| $M_{33}$ | 47          | $M_{s33}$ | 24          | $M_{62}$ | 26          | $M_{s62}$ | 26          |
| $M_{41}$ | 89          | $M_{s41}$ | 27          | $M_{63}$ | 10          | $M_{s63}$ | 10          |
| $M_{42}$ | 41          | $M_{s42}$ | 33          | $M_{64}$ | 3           | $M_{s64}$ | 1           |
| $M_{43}$ | 23          | $M_{s43}$ | 15          | $M_{65}$ | 1           | $M_{s65}$ | 1           |
| $M_{44}$ | 10          | $M_{s44}$ | 8           | $M_{66}$ | 1           | $M_{s66}$ | 1           |
| $M_{51}$ | 56          | $M_{s51}$ | 52          |          |             |           |             |

Table 3.8: Number of terms in  $H(q, \dot{q})$  and  $H_s(q, \dot{q})$  when  $k_t = 5\%$  and  $k_p = 1\%$

| $H(q, \dot{q})$ | $N_{terms}$ | $H_s(q, \dot{q})$ | $N_{terms}$ |
|-----------------|-------------|-------------------|-------------|
| $H_1$           | 341         | $H_{s1}$          | 97          |
| $H_2$           | 357         | $H_{s2}$          | 10          |
| $H_3$           | 337         | $H_{s3}$          | 46          |
| $H_4$           | 321         | $H_{s4}$          | 107         |
| $H_5$           | 318         | $H_{s5}$          | 88          |
| $H_6$           | 302         | $H_{s6}$          | 136         |

According to above tables, 61% of terms in matrix  $M(q)$  and 76% of terms in vector  $H(q, \dot{q})$  can be removed. In order to give more intuitive explanation for the above simplification, the original component  $M_{21}$  of inertia matrix  $M(q)$  and its simplified form  $M_{s21}$  are provided in Appendix B.1. It can be seen that the original expression of the component  $M_{21}$  is greatly complicated and Table 3.7 shows that  $M_{21}$  has as many as 112 terms. In fact, this figure is even bigger (up to 162) before the trigonometric transformations. By contrast, the simplified component  $M_{s21}$  is given by:

$$M_{s21} = Y Z_2 \cos(q_2) + X Z_{2R} \sin(q_2)$$

$$\begin{aligned}
&+ A_1 \cos(q_2 + q_3 + q_4 - \varphi_1) \\
&+ A_2 \cos(q_2 + q_3 - q_4 - \varphi_2)
\end{aligned}$$

with

$$\begin{aligned}
A_1 &= \frac{1}{2} \sqrt{XZ_4^2 + YZ_4^2}, \\
A_2 &= \frac{1}{2} \sqrt{XZ_4^2 + (L_{34}MY_{4R} + YZ_4)^2}, \\
\varphi_1 &= \arctan(-XZ_4, YZ_4), \\
\varphi_2 &= \arctan(XZ_4, L_{34}MY_{4R} + YZ_4).
\end{aligned}$$

Compared with the original component  $M_{21}$ , the simplified component  $M_{s21}$  is extremely simple and has only 4 terms. Besides, many related parameters (e.g.  $A_1$ ,  $A_2$ ,  $\varphi_1$  and  $\varphi_2$ ) are constants which can be computed offline. Thus, the practical computation time will be greatly reduced by the simplification.

In addition, by using different  $k_t$  and  $k_p$ , some negligible inertia parameters for  $M(q)$  and  $H(q, \dot{q})$  can be obtained, which are listed in Table 3.9. For example, if  $k_t = 1\%$  and  $k_p = 3\%$ , the inertia parameters ( $XY_{3R}$ ,  $XZ_3$ ) of axis 3 have negligible influence on the movement of the KUKA KR500-2MT robot.

Table 3.9: Negligible Inertia Parameters for  $M(q)$  and  $H(q, \dot{q})$

| $k_t$ | $k_p$ | $M(q)$                         | $H(q, \dot{q})$       |
|-------|-------|--------------------------------|-----------------------|
| 1%    | 3%    | $XY_{3R}, XZ_3$                | $XY_{3R}, XZ_3$       |
| 1%    | 5%    | $MX_{2R}, XY_{3R}, XZ_3, MX_4$ | $XY_{3R}, XZ_3, MX_4$ |
| 5%    | 3%    | $XY_{3R}, XZ_3$                | $XY_{3R}, XZ_3$       |
| 5%    | 5%    | $MX_{2R}, XY_{3R}, XZ_3, MX_4$ | $XY_{3R}, XZ_3, MX_4$ |

### 3.4.4 Error Analysis of Simplified Model for Whole Workspace

In the previous section, if we select different proportional factors  $k_t$  and  $k_p$ , the simplified models with different accuracies can be obtained. In order to prove the effectiveness and performance of the above simplification, it is necessary to analyze the errors of the simplified models for the whole robot workspace. Thus, a lot of points in the operational workspace need to be chosen to evaluate the accuracy of the simplified model. Based on the balance between accuracy and simplicity, the number of terms after simplification and the root mean square (RMS) error are mainly discussed.

According to Table 3.2, the range of angular motion for each axis of the robot KUKA KR500-2MT are given by:

$$\begin{aligned}
q_1 &= [-37\pi/36, 37\pi/36] \\
q_2 &= [-13\pi/18, 1\pi/9] \\
q_3 &= [-47\pi/90, 5\pi/6] \\
q_4 &= [-35\pi/18, 35\pi/18] \\
q_5 &= [-59\pi/90, 59\pi/90] \\
q_6 &= [-35\pi/18, 35\pi/18]
\end{aligned} \tag{3.36}$$

The selected values of the angular positions  $q_1, q_2, q_3, q_4, q_5$  and  $q_6$  should be uniformly distributed across their intervals. Since  $q_1$  is not included in the expression of the inertia matrix, a total of  $1 \times 10^5$  points in the operational workspace can be obtained by choosing 10 values for each angular position except  $q_1$ . As a result, the root mean square (RMS) errors of the simplified components of inertia matrix can be calculated as:

$$e_{RMS}(\%) = 100 \times \frac{RMS[ M_{ij}(q) - Ms_{ij}(q) ]}{RMS[ M_{ij}(q) ]} \quad (3.37)$$

As  $k_t = 5\%$  and  $k_p = 1\%$ , the number of terms in the components of  $M(q)$  and  $Ms(q)$  as well as the corresponding RMS errors are provided in Table 3.10. The RMS errors show good precision of the simplified inertia matrix, about 61% of terms in matrix  $M(q)$  can be removed while the corresponding global RMS error for the  $1 \times 10^5$  test points is only 1.60%. However, for certain points, the maximum errors of some components are extremely large. Consequently, these points should be avoided in the trajectory planning of the robot.

Table 3.10: Number of terms in components of  $M(q)$  and  $Ms(q)$  as well as RMS errors ( $e_{RMS}$ ) when  $k_t = 5\%$  and  $k_p = 1\%$

| $M(q)$   | $N_t$ | $Ms(q)$   | $N_t$ | $e_{RMS}(\%)$ | $M(q)$   | $N_t$ | $Ms(q)$   | $N_t$ | $e_{RMS}(\%)$ |
|----------|-------|-----------|-------|---------------|----------|-------|-----------|-------|---------------|
| $M_{11}$ | 238   | $Ms_{11}$ | 11    | 1.86          | $M_{52}$ | 35    | $Ms_{52}$ | 24    | 11.3          |
| $M_{21}$ | 112   | $Ms_{21}$ | 4     | 11.8          | $M_{53}$ | 17    | $Ms_{53}$ | 14    | 1.80          |
| $M_{22}$ | 73    | $Ms_{22}$ | 18    | 0.81          | $M_{54}$ | 5     | $Ms_{54}$ | 4     | 1.08          |
| $M_{31}$ | 93    | $Ms_{31}$ | 53    | 4.69          | $M_{55}$ | 3     | $Ms_{55}$ | 3     | 0             |
| $M_{32}$ | 72    | $Ms_{32}$ | 26    | 4.53          | $M_{61}$ | 40    | $Ms_{61}$ | 36    | 3.13          |
| $M_{33}$ | 47    | $Ms_{33}$ | 24    | 0.96          | $M_{62}$ | 26    | $Ms_{62}$ | 26    | 0             |
| $M_{41}$ | 89    | $Ms_{41}$ | 27    | 18.3          | $M_{63}$ | 10    | $Ms_{63}$ | 10    | 0             |
| $M_{42}$ | 41    | $Ms_{42}$ | 33    | 4.98          | $M_{64}$ | 3     | $Ms_{64}$ | 1     | 6.24          |
| $M_{43}$ | 23    | $Ms_{43}$ | 15    | 5.04          | $M_{65}$ | 1     | $Ms_{65}$ | 1     | 0             |
| $M_{44}$ | 10    | $Ms_{44}$ | 8     | 0.61          | $M_{66}$ | 1     | $Ms_{66}$ | 1     | 0             |
| $M_{51}$ | 56    | $Ms_{51}$ | 52    | 1.08          |          |       |           |       |               |

### 3.4.5 Further Simplification and Results

In addition to inertia matrix  $Ms(q)$  and vector  $Hs(q, \dot{q})$ , the product of  $M(q)$  and the angular acceleration  $\ddot{q}$  in equation (3.24) can be further simplified. Above all, the product  $M(q)\ddot{q}$  is formulated by:

$$M(q)\ddot{q} = \begin{bmatrix} M_{11} & M_{12} & M_{13} & M_{14} & M_{15} & M_{16} \\ M_{21} & M_{22} & M_{23} & M_{24} & M_{25} & M_{26} \\ M_{31} & M_{32} & M_{33} & M_{34} & M_{35} & M_{36} \\ M_{41} & M_{42} & M_{43} & M_{44} & M_{45} & M_{46} \\ M_{51} & M_{52} & M_{53} & M_{54} & M_{55} & M_{56} \\ M_{61} & M_{62} & M_{63} & M_{64} & M_{65} & M_{66} \end{bmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \\ \ddot{q}_3 \\ \ddot{q}_4 \\ \ddot{q}_5 \\ \ddot{q}_6 \end{bmatrix} = \begin{bmatrix} W_1 \\ W_2 \\ W_3 \\ W_4 \\ W_5 \\ W_6 \end{bmatrix} \quad (3.38)$$

For the vector of  $M(q)\ddot{q}$ , the component  $W_k$  ( $k \in \{1, \dots, 6\}$ ) can be considered as the target object  $\Psi$  of the simplification:

$$W_k = M_{k1}\ddot{q}_1 + M_{k2}\ddot{q}_2 + M_{k3}\ddot{q}_3 + M_{k4}\ddot{q}_4 + M_{k5}\ddot{q}_5 + M_{k6}\ddot{q}_6 \quad (3.39)$$

Similarly, the second step of simplification algorithm can be applied to  $W_k$  by means of comparing the norms of its terms. Define  $k_w$  as a proportional factor, then any term  $M_{ki}\ddot{q}_i$  can be neglected if it satisfies:

$$\frac{|M_{ki}\ddot{q}_i|}{g(|M_{k1}\ddot{q}_1|, |M_{k2}\ddot{q}_2|, |M_{k3}\ddot{q}_3|, |M_{k4}\ddot{q}_4|, |M_{k5}\ddot{q}_5|, |M_{k6}\ddot{q}_6|)} \leq k_w \quad (i \in \{1, \dots, 6\}) \quad (3.40)$$

where  $g$  is the mapping function and  $g \in \{max, sum, rms, \dots\}$ . Here the function *max* which can find the maximum value of inputs is adopted. By taking  $W_1$  as an example, the ratio of the norm of each term ( $|M_{1i}\ddot{q}_i|$ ) to the maximum one ( $|M_{ki}\ddot{q}_i|_{max}$ ) can be computed, as shown in Table 3.11.

Table 3.11: The ratio of  $|M_{1i}\ddot{q}_i|$  to  $|M_{1i}\ddot{q}_i|_{max}$  in  $W_1$

| Term ( $M_{1i}\ddot{q}_i$ )   | $M_{11}\ddot{q}_1$ | $M_{12}\ddot{q}_2$ | $M_{13}\ddot{q}_3$ | $M_{14}\ddot{q}_4$ | $M_{15}\ddot{q}_5$ | $M_{16}\ddot{q}_6$ |
|-------------------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|
| Norm ( $ M_{1i}\ddot{q}_i $ ) | 7386.8             | 760.4              | 667.3              | 369.3              | 735.0              | 206.8              |
| Ratio (%)                     | 100                | 10.3               | 9.03               | 4.99               | 9.95               | 2.80               |

If  $k_w$  is chosen as 5%, correspondingly the term  $M_{14}\ddot{q}_4$  and  $M_{16}\ddot{q}_6$  in  $W_1$  can be neglected, that is to say, the component  $M_{14}$  and  $M_{16}$  of inertia matrix have little influence on the norm of  $M(q)\ddot{q}$ . The results of the further simplification are provided in Table 3.12.

Table 3.12: Results of further simplification for  $M(q)\ddot{q}$

| $k_w$ | Simplified Components of Inertia Matrix                                                  |
|-------|------------------------------------------------------------------------------------------|
| 1%    | $M_{56}$                                                                                 |
| 3%    | $M_{16}, M_{56}, M_{65}$                                                                 |
| 5%    | $M_{14}, M_{16}, M_{26}, M_{56}, M_{65}$                                                 |
| 10%   | $M_{13}, M_{14}, M_{15}, M_{16}, M_{24}, M_{26}, M_{36}, M_{46}, M_{56}, M_{64}, M_{65}$ |

### 3.5 Case Study: Simplification on Three Different Test Trajectories

In practical applications, the simplification can be performed based on the given trajectory although the simplified model obtained in the previous section is valid for the whole robot workspace. In this section, simulation studies have been carried out on three different test trajectories including an identification trajectory, a linear FSW trajectory and a circular FSW trajectory, in order to analyze the effectiveness and performance of the simplification method. All of the three trajectories are obtained from the tests of the identification and FSW experiments for the robot KUKA KR500-2MT.

### 3.5.1 Evaluation Indexes of Simplification

Generally, the proportional factors  $k_t$  and  $k_p$  can be selected intuitively. Calculating the root mean square (RMS) errors can also be considered as a good approach. In this case, several representative components are chosen to find the most appropriate proportional factors, such as  $M_{11}$ ,  $M_{21}$ ,  $M_{22}$ , and  $M_{41}$ .

Based on the balance between accuracy and simplicity, the following evaluation indexes of the simplification are taken into account:

- 1)  $N_t$ : the number of terms in the components of the simplified model;
- 2)  $e_{RMS}(\Psi s_i)$  or  $e_{RMS}(\Psi s_{ij})$ : the RMS errors of the simplified components, expressed in percent of the RMS value of the original ones;
- 3)  $e_{RMS}(\Psi s)_{global}$ : the global RMS errors for the entire simplified matrix or vector, which will be defined in the following equations;
- 4) the comparison between the value range (or the norm) of the simplified component and that of the original component.

To begin with, the RMS errors of the simplified components  $M s_{ij}$  and  $H s_i$  can be calculated by:

$$\begin{aligned} e_{RMS}(M s_{ij}) &= \frac{RMS(M s_{ij} - M_{ij})}{RMS(M_{ij})} \\ &= \frac{\sqrt{(1/n) \sum_{k=1}^n (M s_{ij}(k) - M_{ij}(k))^2}}{\sqrt{(1/n) \sum_{k=1}^n M_{ij}(k)^2}} \end{aligned} \quad (3.41)$$

$$\begin{aligned} e_{RMS}(H s_i) &= \frac{RMS(H s_i - H_i)}{RMS(H_i)} \\ &= \frac{\sqrt{(1/n) \sum_{k=1}^n (H s_i(k) - H_i(k))^2}}{\sqrt{(1/n) \sum_{k=1}^n H_i(k)^2}} \end{aligned} \quad (3.42)$$

Besides, the global RMS errors of inertia matrix  $M(q)$  and vector  $H(q, \dot{q})$  are defined by:

$$e_{RMS}(M s)_{global} = \frac{\sqrt{(1/36) \sum_{i=1}^6 \sum_{j=1}^6 (RMS(M s_{ij} - M_{ij}))^2}}{\sqrt{(1/36) \sum_{i=1}^6 \sum_{j=1}^6 (RMS(M_{ij}))^2}} \quad (3.43)$$

$$e_{RMS}(H s)_{global} = \frac{\sqrt{(1/6) \sum_{i=1}^6 (RMS(H s_i - H_i))^2}}{\sqrt{(1/6) \sum_{i=1}^6 (RMS(H_i))^2}} \quad (3.44)$$

In addition to the inertia matrix  $M(q)$  and vector  $H(q, \dot{q})$ , the torque  $T = M(q)\ddot{q} + H(q, \dot{q}) + F_{fr}(\dot{q})$  in equation (3.24) is frequently calculated in the practical control by supposing that the robot works without any external force. Thus, the simplified torque  $T s$  can be obtained by:

$$T s = M s(q)\ddot{q} + H s(q, \dot{q}) + F_{fr}(\dot{q}) \quad (3.45)$$

and the calculation methods for the RMS errors of simplified components  $T s_i$  and the global RMS errors of simplified torque  $T s$  are similar to  $H s_i$  and  $H s$  respectively.

### 3.5.2 Case I: Simplification on an Identification Trajectory

The identification trajectory obtained from test *IS0811A* is firstly utilized to implement the simplification, as shown in Figure 3.6. The actual intervals of robot states including the tool position, joint angular position, velocity and acceleration can be found in Table 3.13. It should be noted that these values have comparatively large variations so as to improve the accuracy of parameter identification.

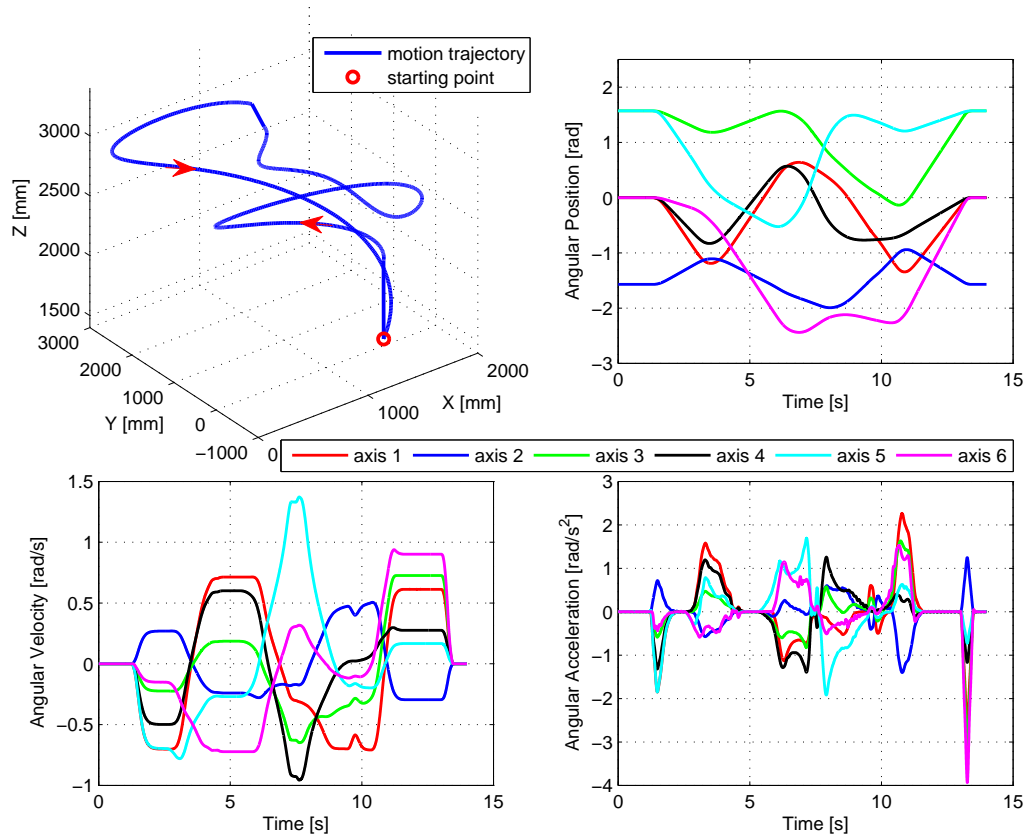


Figure 3.6: The tool position, joint angular position, velocity and acceleration along the identification trajectory (test *IS0811A*)

Table 3.13: Actual intervals of robot states for the identification trajectory

| Axis | Position ( <i>rad</i> ) | Velocity ( <i>rad/s</i> ) | Acceleration ( <i>rad/s</i> <sup>2</sup> ) |
|------|-------------------------|---------------------------|--------------------------------------------|
| 1    | $[-1.346, 0.639]$       | $[-0.709, 0.714]$         | $[-2.581, 2.260]$                          |
| 2    | $[-1.993, -0.941]$      | $[-0.295, 0.506]$         | $[-1.404, 1.248]$                          |
| 3    | $[-0.136, 1.571]$       | $[-0.649, 0.726]$         | $[-3.062, 1.634]$                          |
| 4    | $[-0.829, 0.570]$       | $[-0.957, 0.601]$         | $[-1.396, 1.258]$                          |
| 5    | $[-0.525, 1.571]$       | $[-0.781, 1.372]$         | $[-1.920, 1.693]$                          |
| 6    | $[-2.475, 0]$           | $[-0.733, 0.939]$         | $[-3.937, 1.522]$                          |

In order to find the most appropriate proportional factors  $k_t$  and  $k_p$  for the trajectory, the components  $M_{11}$ ,  $M_{21}$ ,  $M_{22}$  and  $M_{41}$  are illustrated to calculate the evaluation indexes. By choosing different  $k_t$  and  $k_p$ , the number of terms in the simplified components and the corresponding RMS errors along the identification trajectory are given in Table 3.14.

Table 3.14: Number of terms in the simplified components and corresponding RMS errors along the identification trajectory when choosing different  $k_t$  and  $k_p$ 

| $k_t$ | $k_p$ | $M_{11}$ |               | $M_{21}$ |               | $M_{22}$ |               | $M_{41}$ |               |
|-------|-------|----------|---------------|----------|---------------|----------|---------------|----------|---------------|
|       |       | $N_t$    | $e_{RMS}$ (%) | $N_t$    | $e_{RMS}$ (%) | $N_t$    | $e_{RMS}$ (%) | $N_t$    | $e_{RMS}$ (%) |
| 0     | 0     | 238      | 0             | 112      | 0             | 73       | 0             | 89       | 0             |
| 0.5   | 0     | 99       | 0.38          | 53       | 3.02          | 50       | 0.05          | 77       | 0.77          |
|       | 0.5   | 94       | 0.40          | 53       | 2.98          | 46       | 0.13          | 77       | 0.79          |
|       | 1     | 94       | 0.39          | 53       | 3.09          | 45       | 0.59          | 77       | 1.08          |
|       | 3     | 92       | 4.45          | 53       | 3.11          | 45       | 0.59          | 77       | 1.88          |
|       | 5     | 92       | 4.75          | 53       | 5.38          | 45       | 0.80          | 77       | 5.37          |
| 1     | 0     | 50       | 0.63          | 29       | 3.16          | 48       | 0.05          | 77       | 0.77          |
|       | 0.5   | 45       | 0.43          | 29       | 3.18          | 44       | 0.14          | 77       | 0.79          |
|       | 1     | 45       | 0.41          | 29       | 3.21          | 43       | 0.57          | 77       | 1.08          |
|       | 3     | 43       | 4.05          | 29       | 3.38          | 43       | 0.57          | 77       | 1.88          |
|       | 5     | 43       | 4.35          | 29       | 4.34          | 43       | 0.78          | 77       | 5.37          |
| 3     | 0     | 20       | 1.26          | 13       | 4.44          | 31       | 0.55          | 37       | 10.7          |
|       | 0.5   | 15       | 1.08          | 13       | 4.46          | 27       | 0.57          | 37       | 10.7          |
|       | 1     | 15       | 1.08          | 13       | 4.46          | 26       | 0.77          | 37       | 10.5          |
|       | 3     | 13       | 3.77          | 13       | 4.54          | 26       | 0.77          | 37       | 10.5          |
|       | 5     | 13       | 4.06          | 13       | 4.22          | 26       | 0.87          | 37       | 11.4          |
| 5     | 0     | 45       | 0.43          | 29       | 3.18          | 44       | 0.14          | 26       | 12.2          |
|       | 0.5   | 11       | 1.22          | 4        | 10.76         | 19       | 0.59          | 26       | 12.1          |
|       | 1     | 11       | 1.22          | 4        | 10.76         | 18       | 0.95          | 26       | 11.9          |
|       | 3     | 9        | 4.47          | 4        | 10.68         | 18       | 0.95          | 26       | 11.8          |
|       | 5     | 9        | 4.80          | 4        | 9.37          | 18       | 1.00          | 26       | 13.5          |

By making a good trade-off between simplicity and accuracy, the factors can be chosen as  $k_t = 3\%$ ,  $k_p = 1\%$ . The change in the number of terms for each component of  $M(q)$  and the corresponding RMS errors are provided in Table 3.15. Although the term numbers of the original component  $M_{61}$  and its simplified form  $M_{s61}$  are same (40), the error between them still exists since some negligible inertia parameters are removed, and similar situations may occur in the following simplifications.

The results indicate that the simplification causes very small RMS errors for most of the simplified components while the quantities of terms in these components are significantly decreased. By taking the component  $M_{11}$  as an instance, about 95% of terms are removed (from 238 to 15) through the simplification, however, the RMS error of the simplified component  $M_{s11}$  is only 1.08%. Besides, the component  $M_{41}$  has the maximum RMS error which is about 10.5%.

Figures 3.7 and 3.8 show the comparison between the original component of  $M(q)$  and the simplified component of  $M_s(q)$  along the identification trajectory. From these figures, it can be seen that the value ranges of all simplified components are nearly the same as those of original components. On the whole, the simplified components are found to agree well with the original ones, even though some components may not match exactly, such as  $M_{41}$ . However, it should be mentioned that all of the inertia parameters, which are obtained through the model identification,

have a certain degree of errors [4]. Compared with the uncertainties of inertia parameters, the errors caused by the above simplification are within the acceptable range, and the global RMS error of inertia matrix  $M(q)$  is 0.95%.

Table 3.15: Number of terms in components of  $M(q)$  and  $M_s(q)$  as well as RMS errors along the identification trajectory when  $k_t = 3\%$  and  $k_p = 1\%$

| $M(q)$   | $N_t$ | $M_s(q)$  | $N_t$ | $e_{RMS}(\%)$ | $M(q)$   | $N_t$ | $M_s(q)$  | $N_t$ | $e_{RMS}(\%)$ |
|----------|-------|-----------|-------|---------------|----------|-------|-----------|-------|---------------|
| $M_{11}$ | 238   | $M_{s11}$ | 15    | 1.08          | $M_{52}$ | 35    | $M_{s52}$ | 32    | 1.60          |
| $M_{21}$ | 112   | $M_{s21}$ | 13    | 4.46          | $M_{53}$ | 17    | $M_{s53}$ | 14    | 1.56          |
| $M_{22}$ | 73    | $M_{s22}$ | 26    | 0.77          | $M_{54}$ | 5     | $M_{s54}$ | 4     | 1.29          |
| $M_{31}$ | 93    | $M_{s31}$ | 61    | 3.07          | $M_{55}$ | 3     | $M_{s55}$ | 3     | 0             |
| $M_{32}$ | 72    | $M_{s32}$ | 39    | 2.48          | $M_{61}$ | 40    | $M_{s61}$ | 40    | 3.64          |
| $M_{33}$ | 47    | $M_{s33}$ | 24    | 0.83          | $M_{62}$ | 26    | $M_{s62}$ | 26    | 0             |
| $M_{41}$ | 89    | $M_{s41}$ | 37    | 10.5          | $M_{63}$ | 10    | $M_{s63}$ | 10    | 0             |
| $M_{42}$ | 41    | $M_{s42}$ | 37    | 1.57          | $M_{64}$ | 3     | $M_{s64}$ | 3     | 0             |
| $M_{43}$ | 23    | $M_{s43}$ | 19    | 0.55          | $M_{65}$ | 1     | $M_{s65}$ | 1     | 0             |
| $M_{44}$ | 10    | $M_{s44}$ | 8     | 0.38          | $M_{66}$ | 1     | $M_{s66}$ | 1     | 0             |
| $M_{51}$ | 56    | $M_{s51}$ | 52    | 0.91          |          |       |           |       |               |

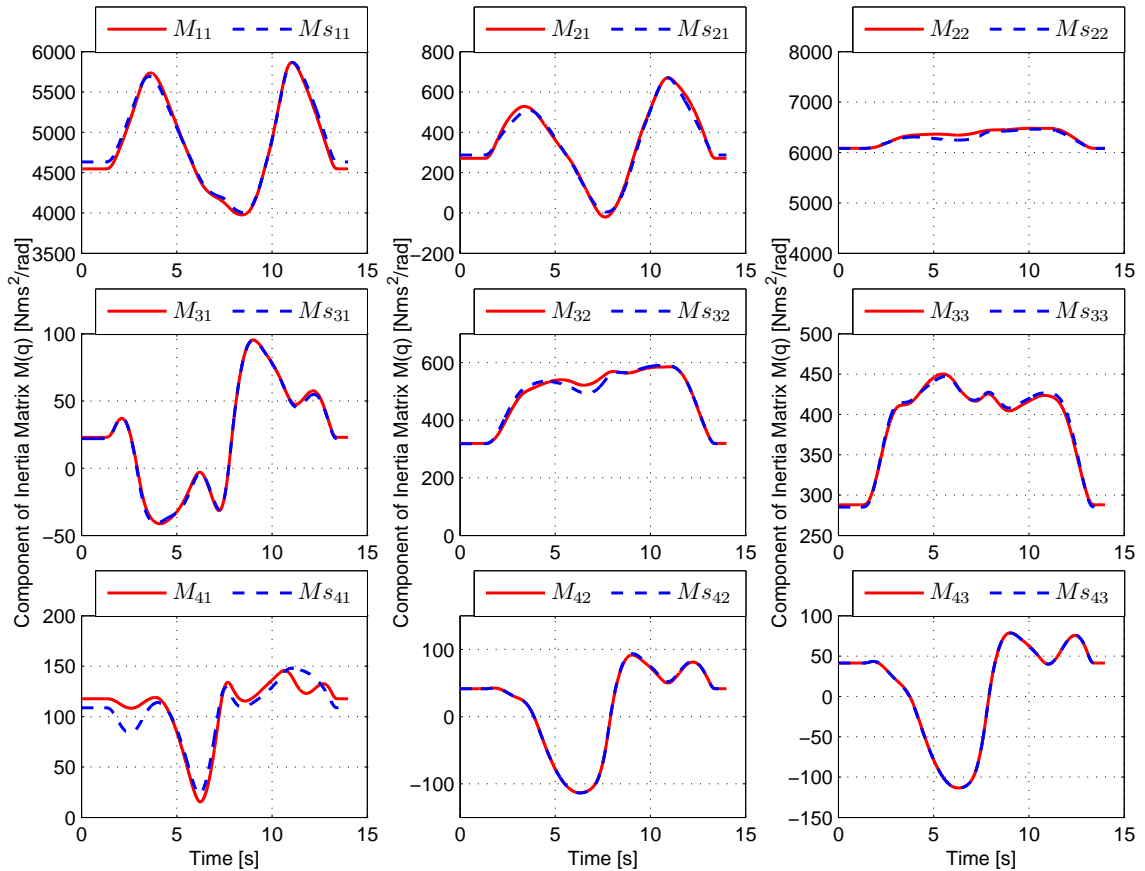


Figure 3.7: Comparison between original component of  $M(q)$  and simplified component of  $M_s(q)$  along the identification trajectory (I)



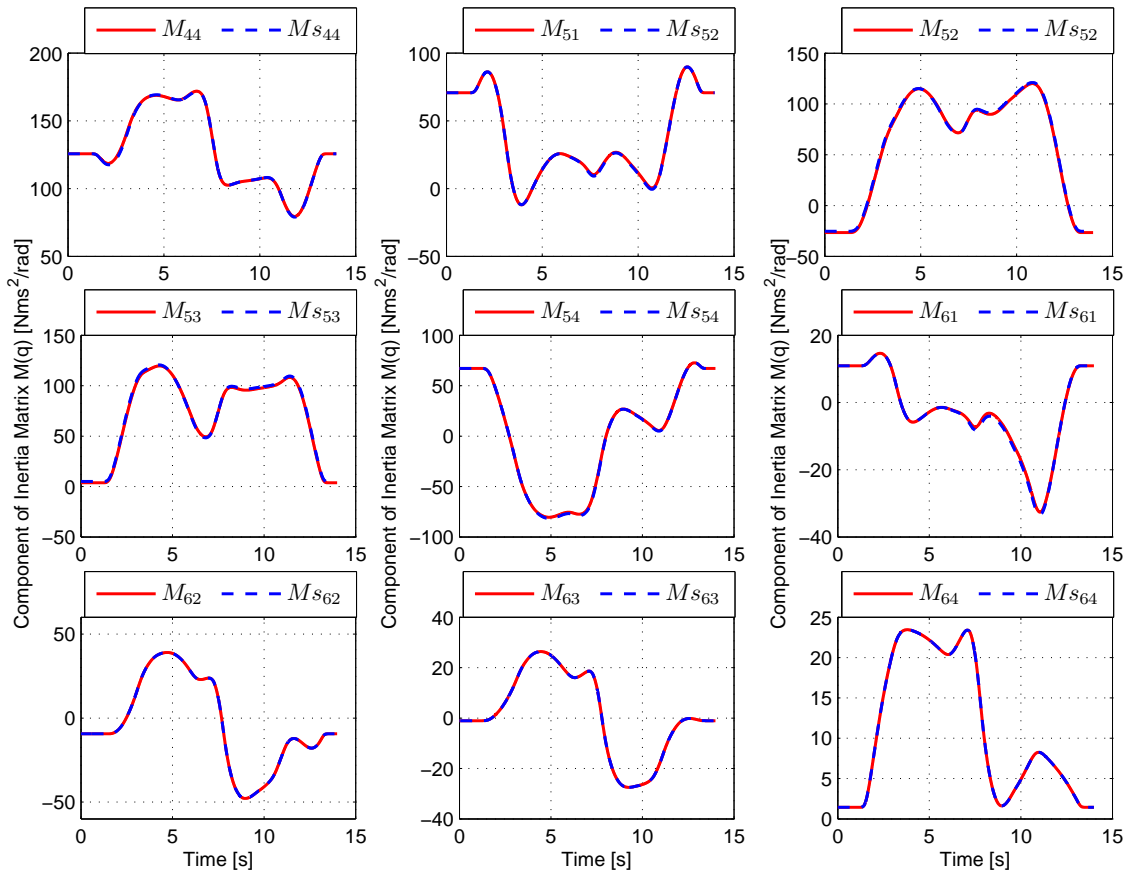


Figure 3.8: Comparison between original component of  $M(q)$  and simplified component of  $M_s(q)$  along the identification trajectory (II)

Table 3.16: Number of terms in components of  $H(q, \dot{q})$  and  $H_s(q, \dot{q})$  as well as RMS errors along the identification trajectory when  $k_t = 3\%$  and  $k_p = 1\%$

| $H$   | $N_t$ | $H_s$    | $N_t$ | $e_{RMS}(\%)$ | $H$   | $N_t$ | $H_s$    | $N_t$ | $e_{RMS}(\%)$ |
|-------|-------|----------|-------|---------------|-------|-------|----------|-------|---------------|
| $H_1$ | 341   | $H_{s1}$ | 147   | 5.89          | $H_4$ | 321   | $H_{s4}$ | 153   | 8.38          |
| $H_2$ | 357   | $H_{s2}$ | 46    | 4.41          | $H_5$ | 318   | $H_{s5}$ | 116   | 4.99          |
| $H_3$ | 337   | $H_{s3}$ | 66    | 4.11          | $H_6$ | 302   | $H_{s6}$ | 170   | 2.83          |

Similarly, by choosing  $k_t = 3\%$  and  $k_p = 1\%$ , the change in the number of terms for each component of vector  $H(q, \dot{q})$  and the corresponding RMS errors are given in Table 3.16. The comparison between the original component of  $H(q, \dot{q})$  and simplified component of  $H_s(q, \dot{q})$  along the identification trajectory is shown in Figure 3.7. The global RMS error of vector  $H(q, \dot{q})$  is 4.41%.

After obtaining  $M_s(q)$  and  $H_s(q, \dot{q})$ , the torque  $T$  can be also simplified according to equation (3.45). Figure 3.10 gives the comparison between original torque  $T$  and simplified torque  $T_s$  along the identification trajectory, and the RMS errors for each axis are 0.89%, 3.05%, 1.58%, 2.52%, 1.57%, and 0.93% respectively. Furthermore, the global RMS error of torque  $T$  is 2.10%. The simulation results demonstrate a good performance of simplification for the given identification trajectory.

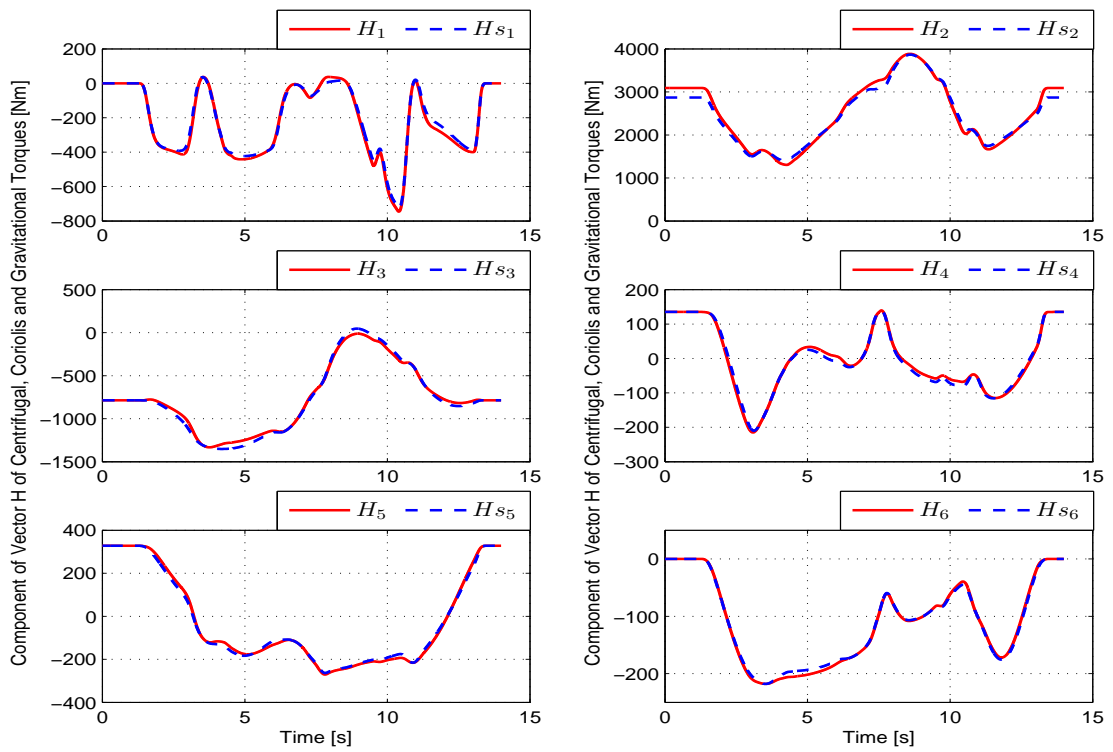


Figure 3.9: Comparison between original component of  $H(q, \dot{q})$  and simplified component of  $H_s(q, \dot{q})$  along the identification trajectory

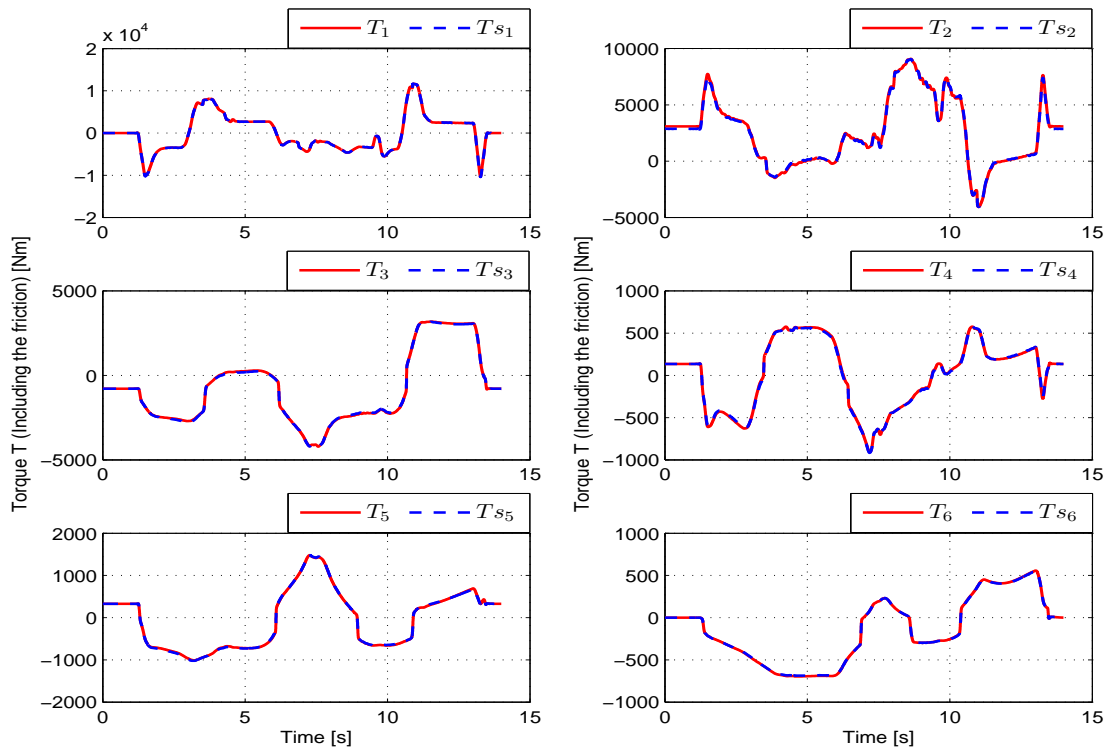


Figure 3.10: Comparison between original torque  $T$  and simplified torque  $T_s$  along the identification trajectory

### 3.5.3 Case II: Simplification on a Linear FSW Trajectory

In addition to the identification trajectory, the simplification algorithm is applicable to a linear FSW trajectory as well. The trajectory considered here is acquired from the linear FSW test *IS1709J*, as shown in Figure 3.11.

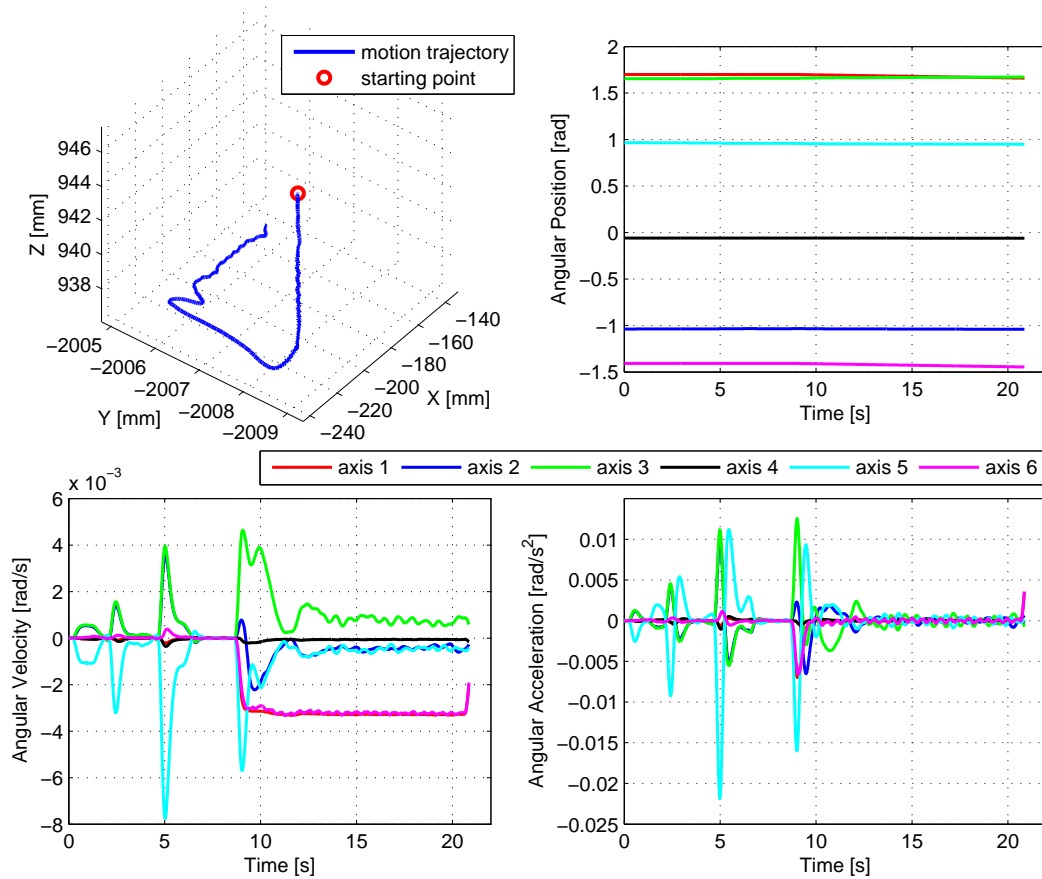


Figure 3.11: The tool position, joint angular position, velocity and acceleration along the linear FSW trajectory (test IS1709J)

Compared with the case of identification trajectory, all the robot states during the linear FSW process have much smaller variations, which can be obviously shown in the figure. The actual intervals of the robot states including the tool position, joint angular position, velocity and acceleration during the linear FSW process are provided in Table 3.17.

Table 3.17: Actual intervals of robot states for the linear FSW trajectory

| Axis | Position (rad)     | Velocity (rad/s)  | Acceleration (rad/s <sup>2</sup> ) |
|------|--------------------|-------------------|------------------------------------|
| 1    | [1.6609, 1.7000]   | [-0.0033, 0]      | [-0.0070, 0.0035]                  |
| 2    | [-1.0405, -1.0336] | [-0.0022, 0.0036] | [-0.0065, 0.0100]                  |
| 3    | [1.6541, 1.6725]   | [-0.0002, 0.0046] | [-0.0055, 0.0126]                  |
| 4    | [-0.0606, -0.0593] | [-0.0004, 0]      | [-0.0011, 0.0005]                  |
| 5    | [0.9491, 0.9664]   | [-0.0078, 0.0003] | [-0.0219, 0.0112]                  |
| 6    | [-1.4458, -1.4079] | [-0.0033, 0.0004] | [-0.0066, 0.0036]                  |

As shown in Table B.1 of Appendix, the number of terms in some simplified representative components (e.g.,  $M_{s11}$ ,  $M_{s21}$ ,  $M_{s22}$  and  $M_{s41}$ ) and the corresponding RMS errors along the linear FSW trajectory can be calculated while the different proportional factors  $k_t$  and  $k_p$  are selected. In addition to choosing the most appropriate proportional factors intuitively, the criteria of accuracy and simplicity can be used to draw a Pareto diagram so as to make a good choice as well.

**Definition: Criterion of Accuracy and Simplicity for the Simplification**

In this simplification method, the definitions of the accuracy criterion and the simplicity criterion can be given as follows:

- Criterion of Simplicity: the total number of terms in the expression of the target object  $\Psi$ , for example, the inertia matrix  $M(q)$ , the vector  $H(q, \dot{q})$ , the coefficient matrix of Coriolis torques  $B(q)_{(6 \times 15)}$  and so on;
- Criterion of Accuracy: the global RMS error of the target object  $\Psi$ , and the corresponding calculation method can be found in (3.43) and (3.44);

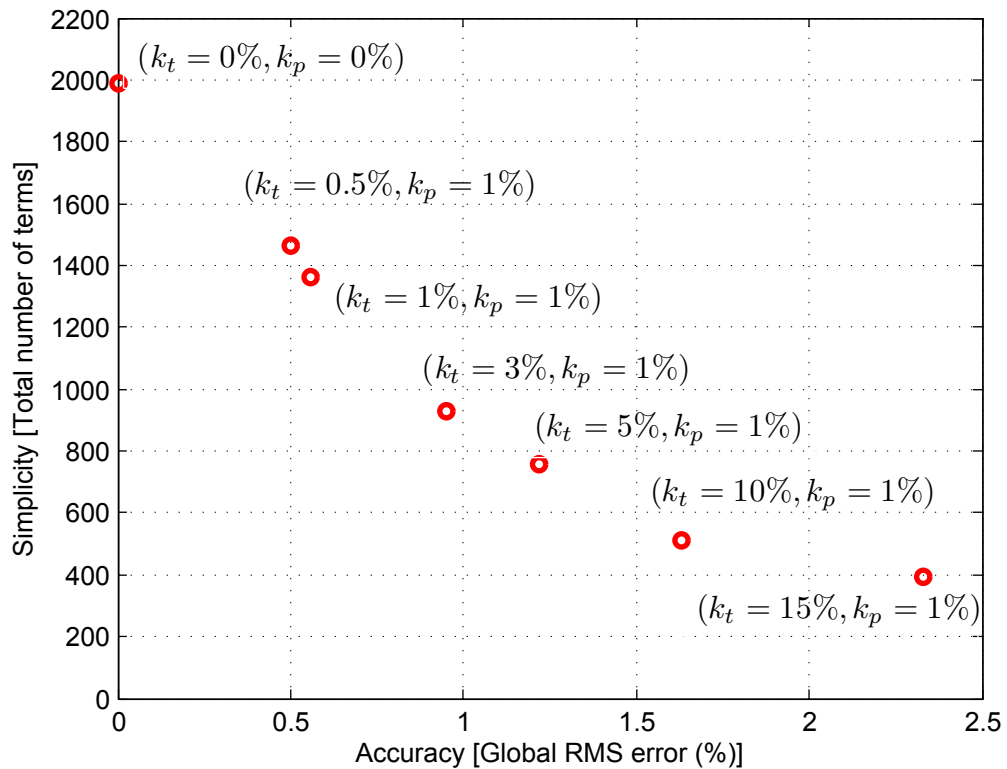


Figure 3.12: Pareto diagram: choose the most appropriate proportional factors for simplifying the inertia matrix  $M(q)$  in the linear FSW case based on the criteria of accuracy and simplicity

In the light of the above criteria of accuracy and simplicity, the Pareto diagram can be given in Figure 3.13, and the proportional factors for inertia matrix  $M(q)$  and vector  $H(q, \dot{q})$  can be chosen as:  $k_t = 1\%$  and  $k_p = 1\%$ .

Accordingly, the change in the number of terms for each component of  $M(q)$  and  $H(q, \dot{q})$  as well as the corresponding RMS errors are provided in Tables 3.18 and 3.19. Compared with simplification results for identification trajectory, it is observed that, by using a smaller  $k_t = 1\%$ , fewer terms are simplified for  $M(q)$  while more terms are removed for  $H(q, \dot{q})$ . This may be explained by that the intervals of the robot states are different for the two trajectories. About 32% of terms in matrix  $M(q)$  and vector  $H(q, \dot{q})$  can be reduced with a good precision of 3%. The component  $M_{41}$  has the maximum RMS error which is about 12.05%.

Figures 3.13, 3.14 and 3.15 provide the comparison between the simplified component and the original one along the linear FSW trajectory for inertia matrix  $M(q)$  and vector  $H(q, \dot{q})$ . Due to small variations of robot states, the values of all the components in these figures have changed slightly compared with Figures 3.7, 3.8 and 3.9. On the whole, the simplified components are essentially identical with original ones and the global RMS errors of  $M(q)$  and  $H(q, \dot{q})$  are 0.56% and 3.16% respectively.

Figure 3.16 gives the comparison between simplified and original component for torque  $T$  along the linear FSW trajectory, and the RMS errors for each axis are 0.03%, 3.39%, 2.76%, 0.01%, 0.001%, and 0.005% respectively. The global RMS error of the torque  $T$  is 3.21%. All the above results show a good performance of simplification for the given linear FSW trajectory.

Table 3.18: Number of terms in components of  $M(q)$  and  $M_s(q)$  as well as RMS errors along the linear FSW trajectory when  $k_t = 1\%$  and  $k_p = 1\%$

| $M(q)$   | $N_t$ | $M_s(q)$  | $N_t$ | $e_{RMS}(\%)$ | $M(q)$   | $N_t$ | $M_s(q)$  | $N_t$ | $e_{RMS}(\%)$ |
|----------|-------|-----------|-------|---------------|----------|-------|-----------|-------|---------------|
| $M_{11}$ | 238   | $M_{s11}$ | 69    | 0.53          | $M_{52}$ | 35    | $M_{s52}$ | 33    | 0.15          |
| $M_{21}$ | 112   | $M_{s21}$ | 24    | 2.93          | $M_{53}$ | 17    | $M_{s53}$ | 15    | 0.10          |
| $M_{22}$ | 73    | $M_{s22}$ | 45    | 0.53          | $M_{54}$ | 5     | $M_{s54}$ | 5     | 1.05          |
| $M_{31}$ | 93    | $M_{s31}$ | 65    | 1.33          | $M_{55}$ | 3     | $M_{s55}$ | 3     | 0             |
| $M_{32}$ | 72    | $M_{s32}$ | 54    | 0.18          | $M_{61}$ | 40    | $M_{s61}$ | 40    | 5.09          |
| $M_{33}$ | 47    | $M_{s33}$ | 37    | 0.08          | $M_{62}$ | 26    | $M_{s62}$ | 26    | 0.76          |
| $M_{41}$ | 89    | $M_{s41}$ | 66    | 12.05         | $M_{63}$ | 10    | $M_{s63}$ | 10    | 0.81          |
| $M_{42}$ | 41    | $M_{s42}$ | 39    | 5.10          | $M_{64}$ | 3     | $M_{s64}$ | 2     | 0.14          |
| $M_{43}$ | 23    | $M_{s43}$ | 21    | 5.83          | $M_{65}$ | 1     | $M_{s65}$ | 1     | 0             |
| $M_{44}$ | 10    | $M_{s44}$ | 9     | 1.53          | $M_{66}$ | 1     | $M_{s66}$ | 1     | 0             |
| $M_{51}$ | 56    | $M_{s51}$ | 50    | 0.61          |          |       |           |       |               |

Table 3.19: Number of terms in components of  $H(q, \dot{q})$  and  $H_s(q, \dot{q})$  as well as RMS errors along the linear FSW trajectory when  $k_t = 1\%$  and  $k_p = 1\%$

| $H$   | $N_t$ | $H_s$    | $N_t$ | $e_{RMS}(\%)$ | $H$   | $N_t$ | $H_s$    | $N_t$ | $e_{RMS}(\%)$ |
|-------|-------|----------|-------|---------------|-------|-------|----------|-------|---------------|
| $H_1$ | 341   | $H_{s1}$ | 180   | 2.50          | $H_4$ | 321   | $H_{s4}$ | 18    | 0             |
| $H_2$ | 357   | $H_{s2}$ | 17    | 3.37          | $H_5$ | 318   | $H_{s5}$ | 18    | 0             |
| $H_3$ | 337   | $H_{s3}$ | 19    | 2.61          | $H_6$ | 302   | $H_{s6}$ | 16    | 0.01          |

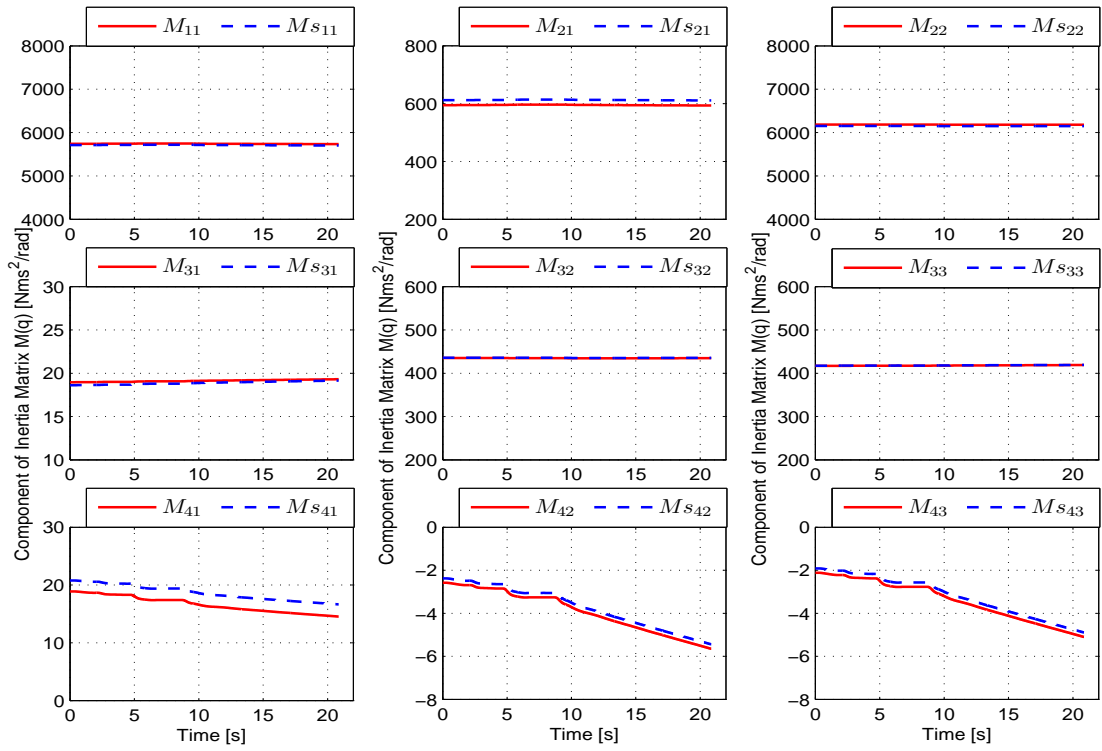


Figure 3.13: Comparison between original component of  $M(q)$  and simplified component of  $M_s(q)$  along the linear FSW trajectory (I)

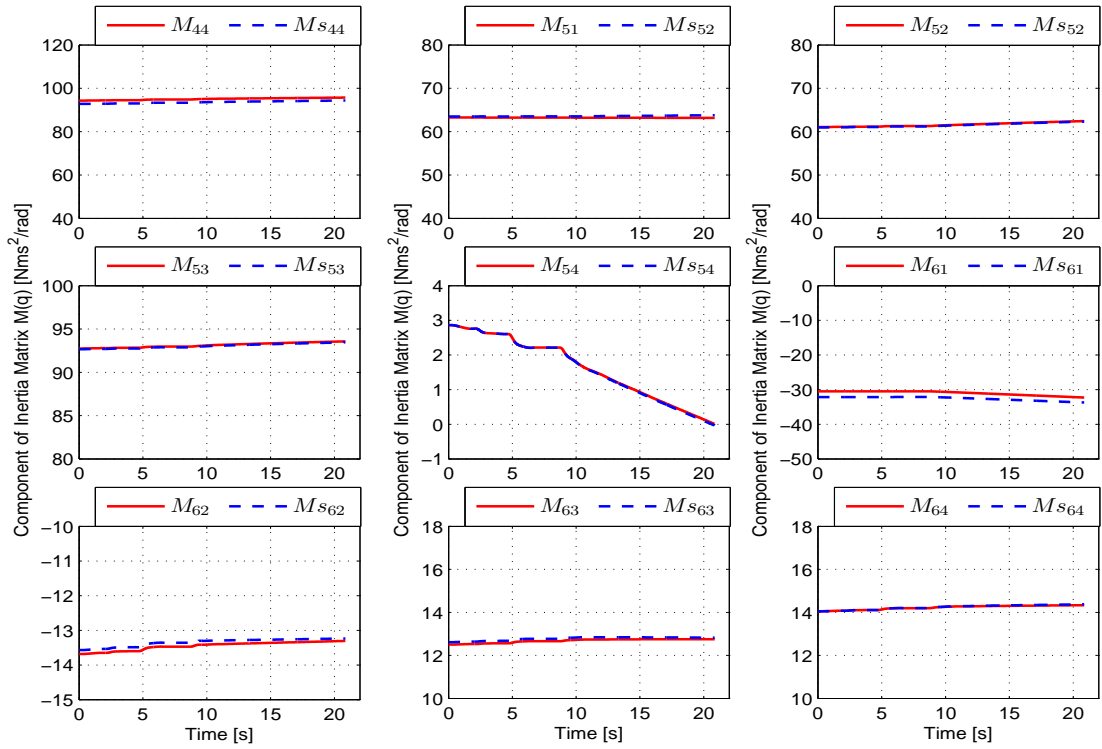


Figure 3.14: Comparison between original component of  $M(q)$  and simplified component of  $M_s(q)$  along the linear FSW trajectory (II)

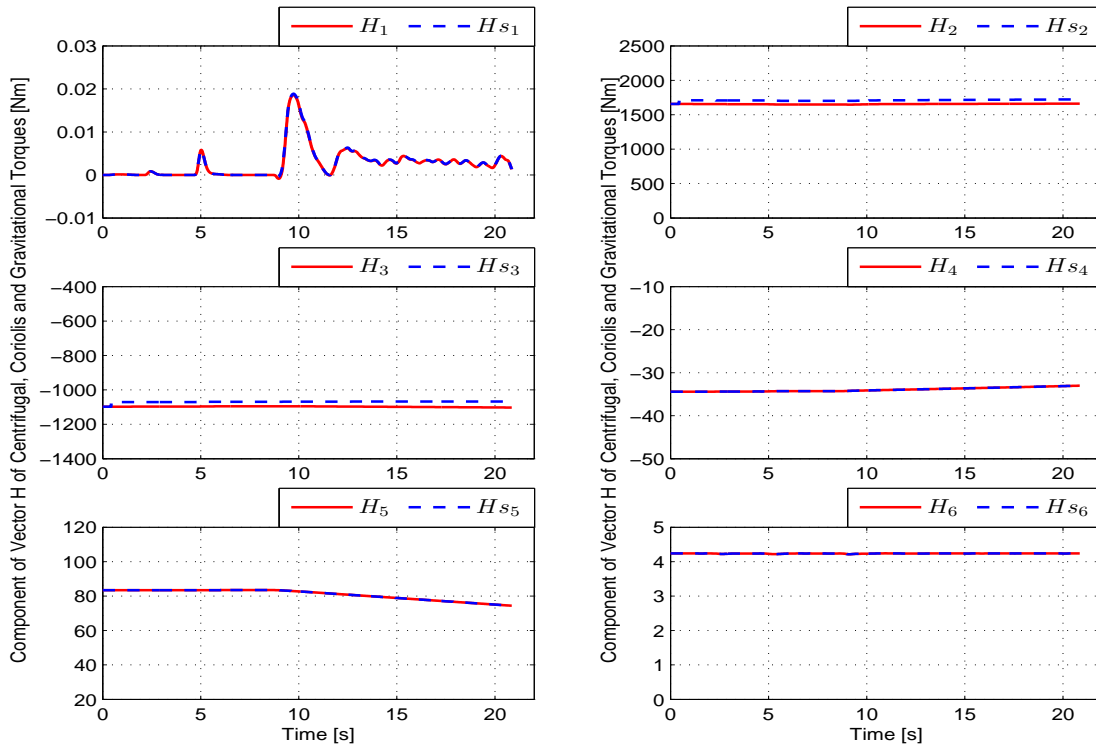


Figure 3.15: Comparison between original component of  $H(q, \dot{q})$  and simplified component of  $H_s(q, \dot{q})$  along the linear FSW trajectory

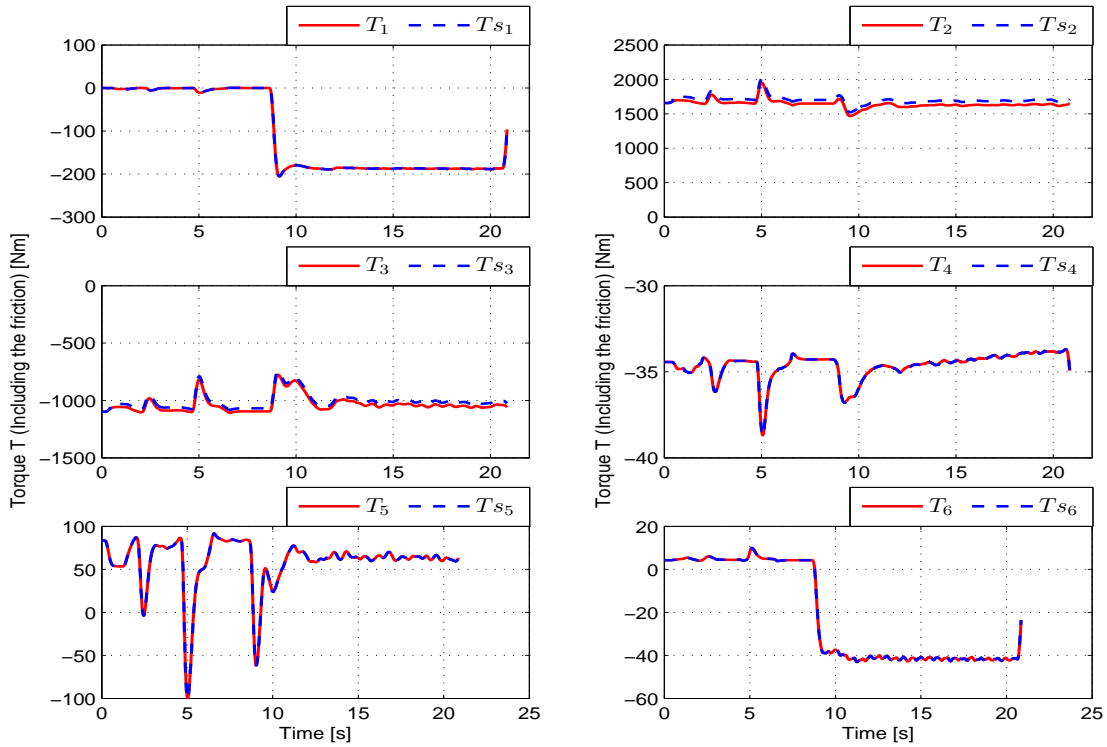


Figure 3.16: Comparison between original torque  $T$  and simplified torque  $T_s$  along the linear FSW trajectory

### 3.5.4 Case III: Simplification on a Circular FSW Trajectory

During the FSW process, the circular trajectories are also frequently used for FSW tests of industrial robots other than the linear trajectories. In this part, the circular FSW trajectory obtained from the test *IS1006P* is adopted for performing the simplification algorithm, as shown in Figure 3.17. In this figure, it can be observed that the joint angular position, velocity and acceleration of the axis 6 has bigger variations than other axes. According to the trajectory data of the circular FSW process, the actual intervals of the robot states including the tool position, joint angular position, velocity and acceleration can be found in Table 3.20.

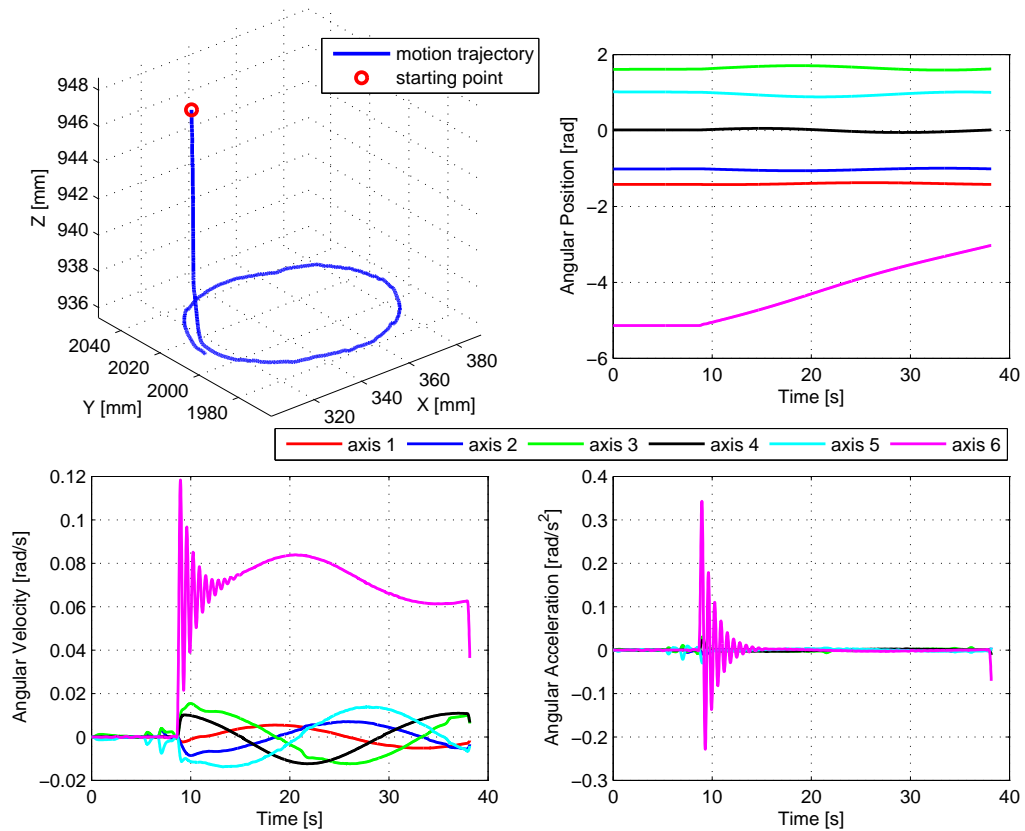


Figure 3.17: The tool position, joint angular position, velocity and acceleration along the circular FSW trajectory (test IS1006P)

Table 3.20: Actual intervals of robot states for the circular FSW trajectory

| Axis | Position ( <i>rad</i> ) | Velocity ( <i>rad/s</i> ) | Acceleration ( <i>rad/s<sup>2</sup></i> ) |
|------|-------------------------|---------------------------|-------------------------------------------|
| 1    | $[-1.4245, -1.3750]$    | $[-0.0052, 0.0055]$       | $[-0.0054, 0.0035]$                       |
| 2    | $[-1.0604, -0.9944]$    | $[-0.0085, 0.0071]$       | $[-0.0094, 0.0094]$                       |
| 3    | $[1.5906, 1.7068]$      | $[-0.0123, 0.0154]$       | $[-0.0100, 0.0306]$                       |
| 4    | $[-0.0533, 0.0546]$     | $[-0.0124, 0.0109]$       | $[-0.0111, 0.0227]$                       |
| 5    | $[0.8845, 1.0166]$      | $[-0.0138, 0.0138]$       | $[-0.0299, 0.0095]$                       |
| 6    | $[-5.1373, -3.0271]$    | $[-0.0004, 0.1184]$       | $[-0.2285, 0.3430]$                       |



A good trade-off between simplicity and accuracy has been made based on Table B.2, which provides the number of terms in the simplified components (e.g.,  $M_{s11}$ ,  $M_{s21}$ ,  $M_{s22}$  and  $M_{s41}$ ) and the corresponding RMS errors along the circular FSW trajectory when selecting different proportional factors. As a result, the proportional factors for  $M(q)$  and  $H(q, \dot{q})$  can be selected to be:  $k_t = 1\%$  and  $k_p = 1\%$ . The change in the number of terms for each component of  $M(q)$  and  $H(q, \dot{q})$  as well as the corresponding RMS errors are given in Tables 3.21 and 3.22.

The results indicate that the terms removed in simplification for the circular FSW trajectory are fewer than those removed in simplification for the linear FSW trajectory when using same proportional factors. Furthermore, the simplification for the circular FSW trajectory produces smaller RMS errors for nearly all simplified components. Similar to the previous cases, the component  $M_{41}$  has the maximum RMS error which is about 4.04%.

Figures 3.18, 3.19, 3.20 and 3.21 provide the comparison between the simplified component and the original one in inertia matrix  $M(q)$ , vector  $H(q, \dot{q})$  and torque  $T$  along the circular FSW trajectory. The RMS errors for each axis of torque  $T$  are 0.01%, 0.01%, 0.003%, 0.005%, 0.01%, and 0.001% respectively. In general, the simplified components are found to coincide essentially with the original ones, and the global RMS errors of  $M(q)$ ,  $H(q, \dot{q})$  and torque  $T$  are 0.54%, 0.004% and 0.005% respectively. The above simulation results indicate a good performance of simplification for the given circular FSW trajectory.

Table 3.21: Number of terms in components of  $M(q)$  and  $M_s(q)$  as well as RMS errors along the circular FSW trajectory when  $k_t = 1\%$  and  $k_p = 1\%$

| $M(q)$   | $N_t$ | $M_s(q)$  | $N_t$ | $e_{RMS}(\%)$ | $M(q)$   | $N_t$ | $M_s(q)$  | $N_t$ | $e_{RMS}(\%)$ |
|----------|-------|-----------|-------|---------------|----------|-------|-----------|-------|---------------|
| $M_{11}$ | 238   | $M_{s11}$ | 87    | 0.54          | $M_{52}$ | 35    | $M_{s52}$ | 33    | 1.01          |
| $M_{21}$ | 112   | $M_{s21}$ | 29    | 0.22          | $M_{53}$ | 17    | $M_{s53}$ | 15    | 0.62          |
| $M_{22}$ | 73    | $M_{s22}$ | 53    | 0.54          | $M_{54}$ | 5     | $M_{s54}$ | 5     | 0             |
| $M_{31}$ | 93    | $M_{s31}$ | 69    | 1.29          | $M_{55}$ | 3     | $M_{s55}$ | 3     | 0             |
| $M_{32}$ | 72    | $M_{s32}$ | 61    | 0.06          | $M_{61}$ | 40    | $M_{s61}$ | 40    | 1.11          |
| $M_{33}$ | 47    | $M_{s33}$ | 43    | 0.13          | $M_{62}$ | 26    | $M_{s62}$ | 26    | 0             |
| $M_{41}$ | 89    | $M_{s41}$ | 77    | 4.04          | $M_{63}$ | 10    | $M_{s63}$ | 10    | 0             |
| $M_{42}$ | 41    | $M_{s42}$ | 41    | 0.20          | $M_{64}$ | 3     | $M_{s64}$ | 2     | 0             |
| $M_{43}$ | 23    | $M_{s43}$ | 23    | 0.18          | $M_{65}$ | 1     | $M_{s65}$ | 1     | 0             |
| $M_{44}$ | 10    | $M_{s44}$ | 10    | 0             | $M_{66}$ | 1     | $M_{s66}$ | 1     | 0             |
| $M_{51}$ | 56    | $M_{s51}$ | 52    | 0.84          |          |       |           |       |               |

Table 3.22: Number of terms in components of  $H(q, \dot{q})$  and  $H_s(q, \dot{q})$  as well as RMS errors along the circular FSW trajectory when  $k_t = 1\%$  and  $k_p = 1\%$

| $H$   | $N_t$ | $H_s$    | $N_t$ | $e_{RMS}(\%)$ | $H$   | $N_t$ | $H_s$    | $N_t$ | $e_{RMS}(\%)$ |
|-------|-------|----------|-------|---------------|-------|-------|----------|-------|---------------|
| $H_1$ | 341   | $H_{s1}$ | 296   | 2.44          | $H_4$ | 321   | $H_{s4}$ | 18    | 0.005         |
| $H_2$ | 357   | $H_{s2}$ | 26    | 0.004         | $H_5$ | 318   | $H_{s5}$ | 18    | 0.005         |
| $H_3$ | 337   | $H_{s3}$ | 25    | 0.004         | $H_6$ | 302   | $H_{s6}$ | 16    | 0.03          |

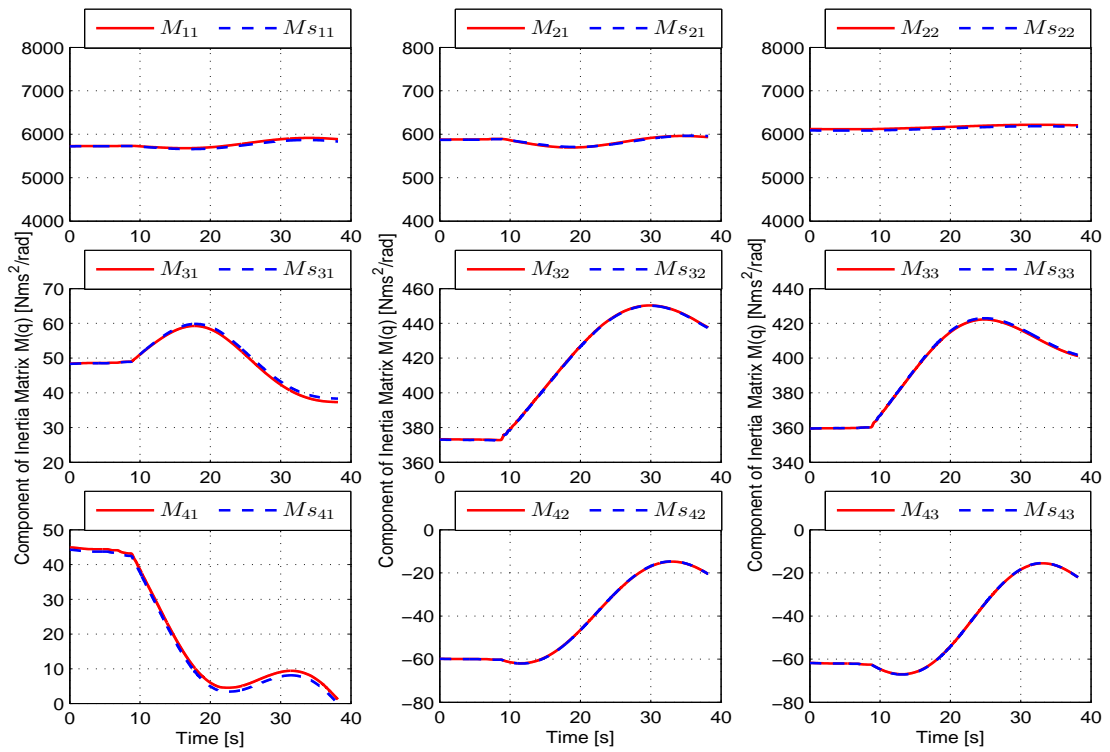


Figure 3.18: Comparison between original component of  $M(q)$  and simplified component of  $M_s(q)$  along the circular FSW trajectory (I)

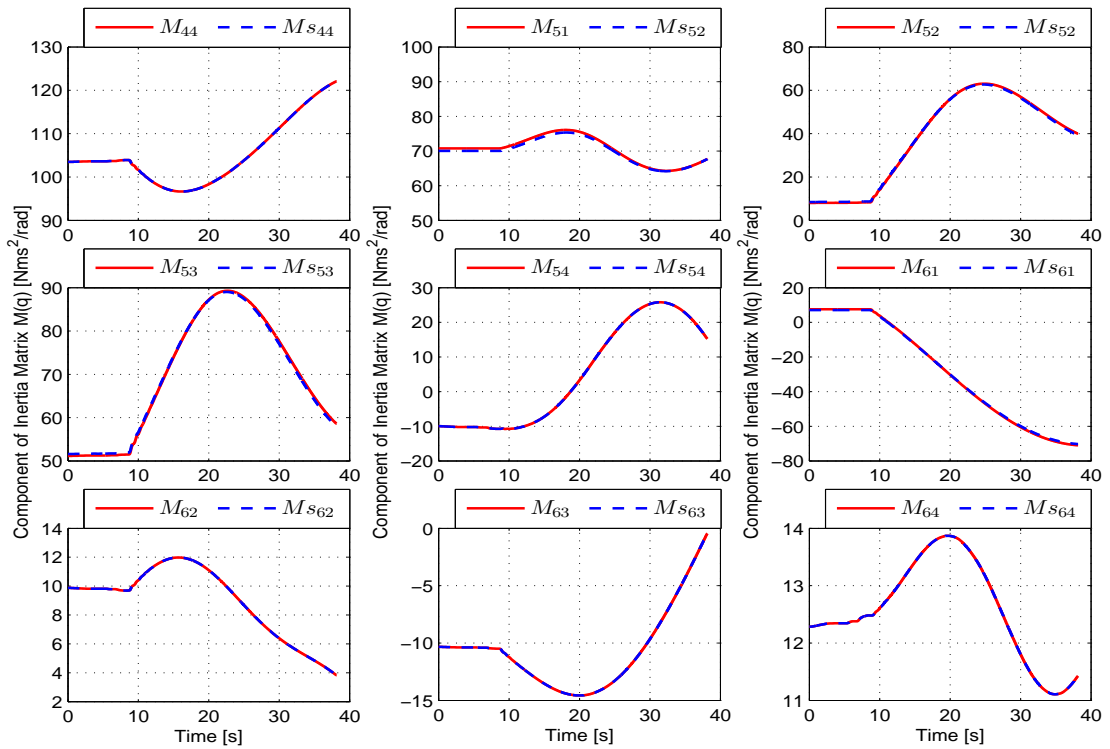


Figure 3.19: Comparison between original component of  $M(q)$  and simplified component of  $M_s(q)$  along the circular FSW trajectory (II)

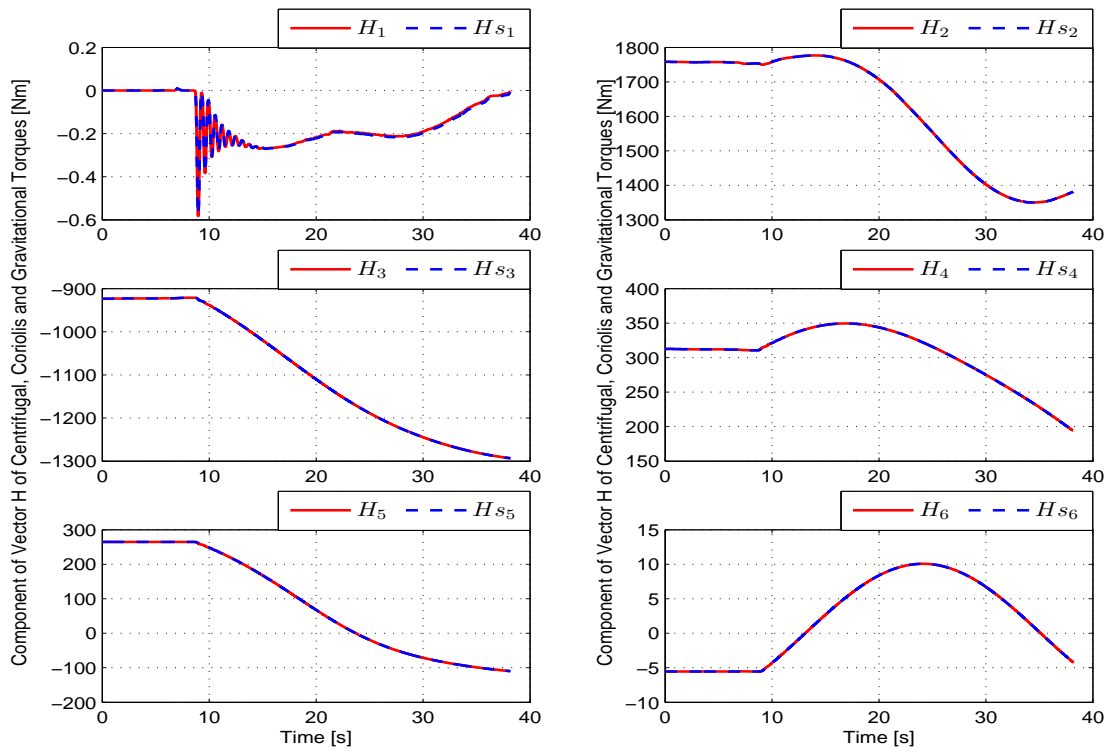


Figure 3.20: Comparison between original component of  $H(q, \dot{q})$  and simplified component of  $H_s(q, \dot{q})$  along the circular FSW trajectory

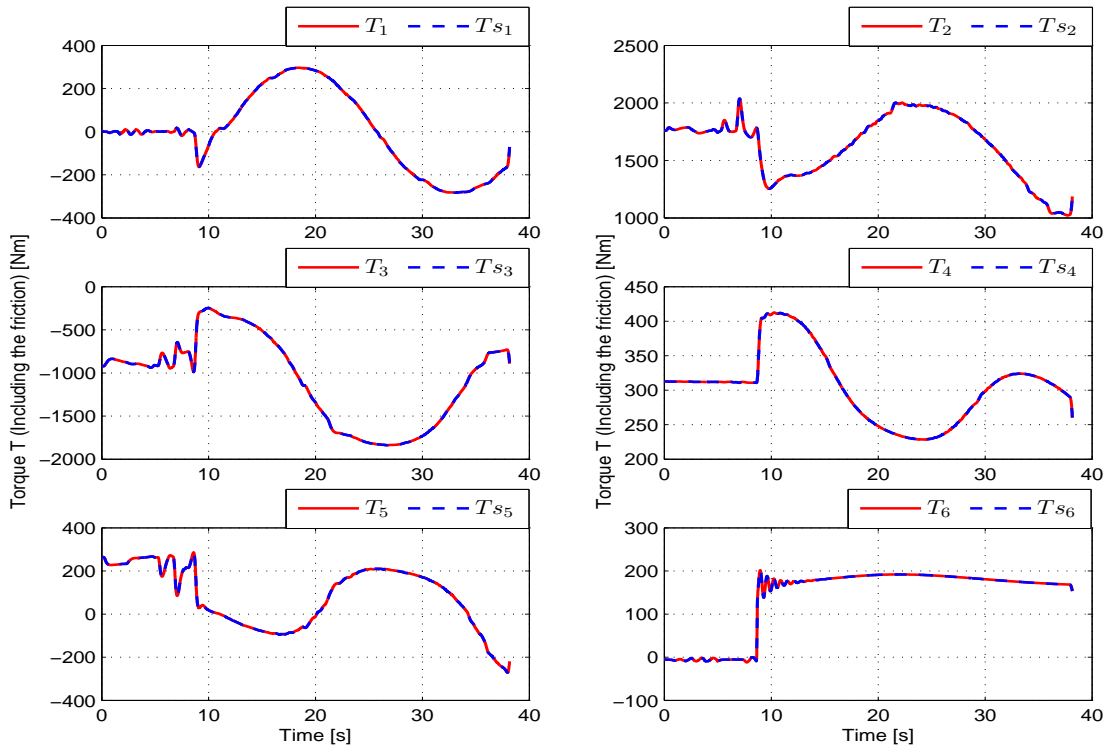


Figure 3.21: Comparison between original torque  $T$  and simplified torque  $T_s$  along the circular FSW trajectory

### 3.5.5 Case IV: Torques Analysis in Robot Dynamic Equation

By means of interval calculations, the torques in the robot dynamic equation have been analyzed by using the linear FSW trajectory (test *IS1709J*) which is shown in Figure 3.11. According to equation (3.24), the following torques can be analyzed:

- 1)  $T_{Racc}$  : the vector of torque of angular acceleration  $M(q)\ddot{q}$  on the robot side;
- 2)  $H_c(q, \dot{q})$  : the vector of sum of the Coriolis torques  $B(q)(\dot{q}\dot{q})$  and the centrifugal torques  $C(q)(\dot{q}^2)$  on the robot side, which can be calculated by putting the gravitational acceleration equal zero in  $H(q, \dot{q})$ ;
- 3)  $G(q)$ : the vector of torque of gravity;
- 4)  $F_{fr}(\dot{q})$  : the vector of torque of friction at the robot joints;
- 5)  $\Gamma$  : the vector of gearbox output torque;

Besides, the external torque ( $J^T(q)F_e$ ) applied by the environment on the robot is typically very large during the FSW process which will be discussed in the next chapter. Since the torque of angular acceleration on the motor side  $I_a\ddot{\theta}$  and the torque of friction on the motor side  $F_{fm}(\theta)$  are much smaller than the motor torque  $\Gamma_m$  in equation (3.25), the gearbox output torque can be approximately calculated by:

$$\Gamma \approx N^T(k_t I) \quad (3.46)$$

where  $k_t$  is the diagonal matrix of the torque constants of all motors,  $I$  is the vector of the measured currents of all axes. Figure 3.22 provides the measured currents of all the motors during the linear FSW process.

Based on the actual intervals of the measured currents ( $I$ ) and the robot states ( $q$ ,  $\dot{q}$  and  $\ddot{q}$ ), the interval calculations of the torques in the robot dynamic equation can be implemented by using the natural interval extensions, and the results are given in Table 3.23.

Table 3.23: Intervals of torques in robot dynamic equation (axis  $i$ )

| $i$ | $T_{Racc}(Nm)$    | $H_c(q, \dot{q})(Nm)$ | $G(q)(Nm)$       | $F_{fr}(\dot{q})(Nm)$ | $\Gamma(Nm)$    |
|-----|-------------------|-----------------------|------------------|-----------------------|-----------------|
| 1   | [-45.91, 27.22]   | [-0.149, 0.146]       | 0                | [-60.57, 0.2019]      | [-6720, -3.148] |
| 2   | [-48.42, 70.52]   | [-0.059, 0.039]       | [1630, 1678]     | [-233.1, 366.3]       | [425.7, 15587]  |
| 3   | [-7.482, 10.87]   | [-0.026, 0.029]       | [-1118, -1081]   | [-14.86, 390.9]       | [-382.9, 10069] |
| 4   | [-0.5367, 0.2755] | [-0.023, 0.032]       | [-35.76, 31.65]  | [-3.451, 0.2526]      | [-1643, 11.43]  |
| 5   | [-3.218, 2.986]   | [-0.013, 0.016]       | [68.46, 90.52]   | [-125.7, 5.645]       | [-1014, 1151]   |
| 6   | [-0.5325, 0.5973] | [-0.005, 0.006]       | [-0.7041, 10.12] | [-33.51, 4.026]       | [-238.2, 186.1] |

It can be observed that the gearbox output torque  $\Gamma$  has the largest variations among the torques in robot dynamic equation. According to the second step of the simplification algorithm, all these torques can be analyzed by comparing their norms. Accordingly, the ratio  $p_T$  of the norm of any torque  $T$  to the maximum one ( $|\Gamma|$ ), expressed as a percentage, can be calculated by:

$$p_T(T) = \frac{|T|}{|\Gamma|} \times 100\% \quad (3.47)$$

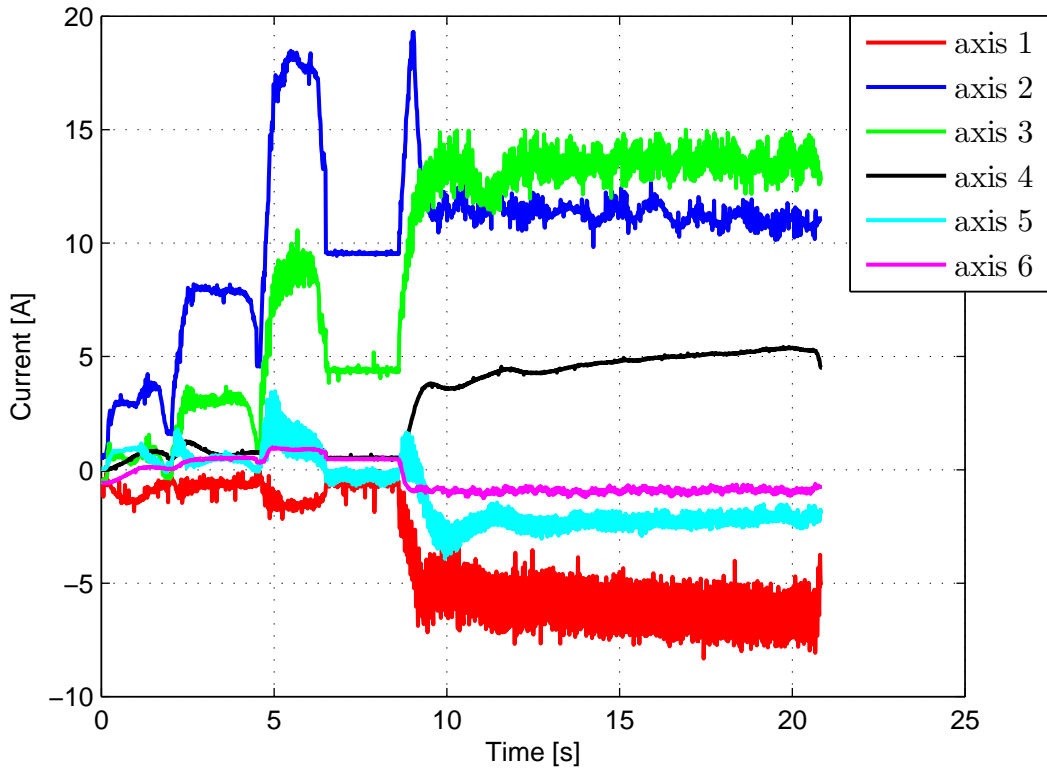


Figure 3.22: Measured currents of all the motors during the linear FSW process (test 1709J)

Table 3.24: The ratio  $p_T(\%)$  of the norm of all torques to the maximum one

| Axis | $p_T(T_{Racc})$ | $p_T(H_c(q, \dot{q}))$ | $p_T(G(q))$ | $p_T(F_{fr}(\dot{q}))$ | $p_T(\Gamma)$ |
|------|-----------------|------------------------|-------------|------------------------|---------------|
| 1    | 0.68            | $2.2 \times 10^{-3}$   | 0           | 0.90                   | 100           |
| 2    | 0.45            | $3.8 \times 10^{-4}$   | 10.77       | 2.35                   | 100           |
| 3    | 0.11            | $2.9 \times 10^{-4}$   | 11.10       | 3.88                   | 100           |
| 4    | 0.033           | $1.9 \times 10^{-3}$   | 2.18        | 0.21                   | 100           |
| 5    | 0.28            | $1.4 \times 10^{-3}$   | 7.86        | 10.92                  | 100           |
| 6    | 0.25            | $2.5 \times 10^{-3}$   | 4.25        | 14.07                  | 100           |

and the results can be found in Table 3.24.

The results indicate that all the torques except the external one in the robot dynamic equation are much smaller than the torque of the gearbox output  $\Gamma$  during the linear FSW process. Among these torques, the centrifugal torque ( $C(q)(\dot{q}^2)$ ), the Coriolis torque ( $B(q)(\dot{q}\dot{q})$ ) and the torque of acceleration on the robot side ( $T_{Racc}$ ) are negligible. The components of torque  $T_{Racc}$  for all axes are smaller than 1% of  $\Gamma$ , and the result of  $p_T$  for the centrifugal and Coriolis torques even reach the magnitude order of  $1 \times 10^{-4}$ . Besides, the torque of gravity and the friction on robot side can not be neglected, and some components of them are within the range of 10% to 15% of the torque of gearbox output. Similar results can be obtained for the motor dynamic equation as well.

### 3.6 Discussion on Usage of the Simplified Model

Seen from the above simulation results, generally the simplified model has not only relatively simple components but also sufficient accuracy compared with the original model. As a consequence, the simplified model can be applied to many control methods so as to simplify the control structure and improve the real-time performance, such as the computed torque controller or the observer-based control.

In the research work presented by Qin et al. [4, 71, 72, 73], when a strong external force is exerted on the robot during the FSW process, the natural stiffness of the industrial robot is not sufficient, resulting in a deformation of the robot. In order to compensate the manipulator deformation, a nonlinear observer has been proposed to estimate the robot states  $(q, \dot{q}, \ddot{q})$  and the external force  $F$ .

In this observer-based control for robot KUKA KR500-2MT, a corrected target position is given to the interpolator of the robot controller every 12 ms so as to compensate the error between the desired position and so-called real position estimated by a discrete observer, whose sampling time is 1.2 ms [73]. The diagram of external deformation compensation of KUKA robot is provided in Figure 3.23, which can explain the reason why the simplification is needed for this control.

With regard to the estimation of external force in the above observer-based control, it can be obtained by:

$$\widehat{F} = -J^{-T}(q)M(q)\widehat{z}_4 \quad (3.48)$$

in which  $\widehat{F}$  denotes the estimated external force, and  $\widehat{z}_4$  represents the fourth state of the observer [72]. Obviously, the inertia matrix  $M(q)$  can be replaced by the simplified one  $M_s(q)$  in equation (3.48).

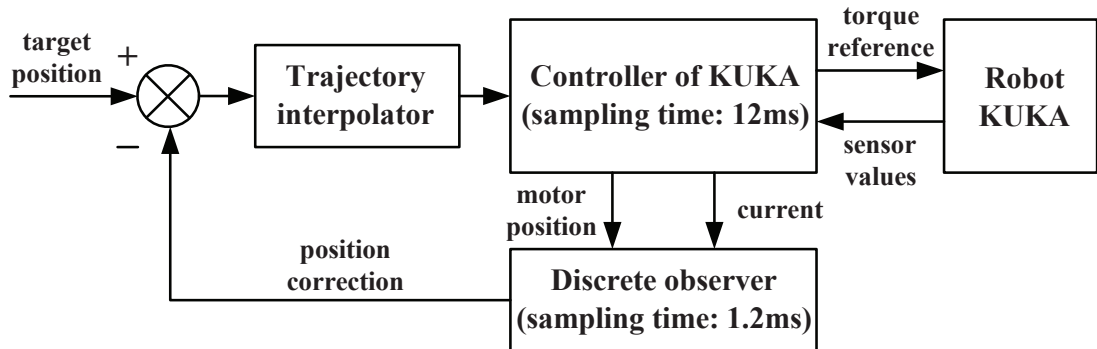


Figure 3.23: External deformation compensation of KUKA robot [73]

In addition, the resulting negligible inertia parameters have a significant meaning for the robot model identification. The identified inertia parameters will be more precise as the negligible inertia parameters are removed [94].

By using the simplification method for the dynamic model of the robot, the important parameters of the robot can be found, which is extremely beneficial for the design of new robots. For example, during the FSW process, it is supposed that a new axis is added to the industrial robot in order to press the workpiece on the platform tightly, the influence of the parameters of the new axis can be analyzed

by the simplification method. Furthermore, the simplification algorithm can be also applied to other dynamic models such as planes and helicopters.

## 3.7 Conclusion

In this chapter, a new approach using interval method for simplification of industrial robot dynamic model is presented. From the symbolic dynamic model of a 6-DOF industrial manipulator, all the expressions of components in inertia matrix  $M(q)$  and vector  $H(q, \dot{q})$  show complexities and strong nonlinearities, besides, a high coupling between control inputs also exists. Thus, an interval-based simplification algorithm is proposed to make the robot model much simpler.

First of all, the interval analysis theory is briefly presented, followed by the generation of the symbolic robot dynamic model. Next, the general simplification algorithm is proposed. The entire simplification process can be divided into 3 steps. After the simplification example, the results and error analysis of the simplified model for the whole robot workspace are given, which are suitable for arbitrary trajectories. As  $k_t = 5\%$  and  $k_p = 1\%$ , about 61% of terms in matrix  $M(q)$  and 76% of terms in vector  $H(q, \dot{q})$  can be removed while the corresponding global RMS errors for the  $1 \times 10^5$  test points are less than 5%, showing an excellent precision of the simplified model. The original expression of one component  $M_{21}$  of inertia matrix and its simplified form  $M_{s21}$  were provided so as to explain and describe the simplification method intuitively. Besides, the further simplification has been implemented as well.

Then, case studies have been carried out on three different test trajectories including the identification trajectory, the linear FSW trajectory and the circular FSW trajectory. The criteria of accuracy and simplicity for the simplification are given and the Pareto diagram is drawn to choose the most appropriate proportional factors. For the linear and circular FSW trajectories, about 32% of terms in matrix  $M(q)$  and vector  $H(q, \dot{q})$  can be reduced with a good precision of 3%. Comparisons between simplified and original components of  $M(q)$ ,  $H(q, \dot{q})$  and  $T$  are performed. A good accuracy of the simplified model is shown, demonstrating the effectiveness and good performance of the simplification. Besides, the torques in the robot dynamic equation are analyzed by means of the linear FSW trajectory as well.

The simplification algorithm proposed in this thesis can be generalized to other dynamic models like planes and helicopters. The simplified model can be applied to the observer-based control, and it also contributes to finding negligible inertia parameters, which is very useful for robot model identification. The identified inertia parameters will be more precise as the negligible inertia parameters are removed.

# Chapter 4

## Dynamic Modeling and Identification of Robotic FSW Process

### 4.1 Introduction

The work presented in this chapter focuses on dynamic modeling and identification of the total forces acting on the tool during the robotic FSW process [97]. Such dynamic models can be used to calculate the forces by using the robot states without the implementation of real welding operation, which contributes to simulating the FSW process in the simulator of the entire robot-controlled FSW system. Besides, although the static models can describe the steady-state relationships between the process forces and the process parameters (including plunge depth, travel speed and rotation speed), the dynamic models of the process forces are highly required for the design of the control strategies in robotic FSW process, especially for the design of the robot force controller which will be introduced in the next chapter. Due to insufficient natural stiffness of the industrial robot, the large external forces exerted on the robot leads to nonnegligible deformation which can be calculated by the identified models. Thus, the models can be used to establish the control laws for compensating the positioning errors of the robot. In addition, the dynamic models of the process forces can be utilized for the stability analysis of the robot force controller.

Friction stir welding (FSW) is a relatively new and promising solid state welding technology developed by The Welding Institute (TWI) of UK in 1991 [1]. Compared to conventional fusion welding technologies, FSW is mainly characterized by joining material without reaching the fusion temperature, avoiding the problems caused by melting metals, such as hot cracking, solidification defects and loss of weld mechanical properties [2]. Almost all types of aluminum alloys, even those that are classified as non-weldable materials by traditional fusion welding techniques (e.g., the highly alloyed 2XXX and 7XXX series aluminum), can be welded through FSW process [98]. Besides, this joining technology can also be used to weld similar or dissimilar alloys and metals like steel, copper and magnesium alloys [99]. In addition to the energy efficiency, environment friendliness and versatility, the main advantages of FSW technology contains: excellent weld strength and ductility, low residual stresses, no use of filler metal, relatively high welding speed and repeatability [3]. Owing to



above benefits, the technology has been widely applied in various industrial applications in recent years, such as shipbuilding, railways, automotive and aerospace industries.

In recent years, the dedicated machines which are commonly used to perform the FSW process have been increasingly replaced by the industrial robots because of their excellent repeatability, production flexibility and low cost [100]. However, the limited force capability and stiffness of industrial robots are barriers for expansion of robotic FSW. Smith et al. [101] developed a FSW system which was integrated into an ABB IRB 7600 articulated robot and the capability of implementing three-dimensional contours welding in various positions with excellent force feedback control was proved. Bres et al. [102] established a model-based framework which allows the simulation, analysis and optimization of FSW processes of metallic structures using industrial robots. De Backer et al. [103] applied on-line sensing and path compensation methods to obtain a strong and defect-free welding in robotic FSW process. Fehrenbacher et al. [104] demonstrated a successful development and evaluation of a closed-loop control system for robotic FSW that controls plunge force and tool interface temperature by varying spindle speed and commanded vertical tool position. A lot of research have been carried out to solve the control problems in robotic FSW and dynamic modeling of the process is extremely important for the design of the control strategies.

In the field of modeling and identification of FSW process, Ulysse [105] developed a three-dimensional viscoplastic model using the finite element method. The author also studied the influence of travel speed and rotation speed on the welding force. A thermomechanical finite element model was presented in [106], and numerical calculations of the force was in good agreement with experimental results. Nevertheless, the applications of above models are limited in real-time process control because of large computations. Therefore, an empirical dynamic model was developed in [107], which constructed the relationships between the process parameters (i.e., plunge depth, travel speed, and rotation speed) and the process variables (i.e., axial, path, normal forces). Experiments were also implemented to identify the dynamic characteristics of the process variables using recursive least squares methods. By analyzing FSW process forces and their impact on machine structure, Zaeh and Gebhard [96] proposed an experimental process force model composed of the static force components and the superposition of wave equations with different amplitudes, phases and the frequency of the rotation speed and its first harmonic.

This chapter is organized as follows. Section 4.2 gives a brief description of the FSW process including its main stages and process forces. Section 4.3 introduces the static modeling and identification of process forces in FSW. Section 4.4 describes the experimental setup for the identification of the dynamic model of axial force. In Section 4.5, three methods for calculating the plunge depth and its derivative are given, followed by the data filtering for the measurements and calculations. Section 4.6 and Section 4.7 present the linear and nonlinear dynamic modeling and identification of the axial force by using least squares method. Systematic studies on the errors of identified models have been carried out to analyze the performance of the model identification. Finally, a conclusion is given in Section 4.8.

## 4.2 Description of the FSW Process

The basic concept of FSW process can be described as follows: a non-consumable tool, consisting of a specially designed pin and shoulder, is inserted into the interface of two rigidly clamped workpieces at a specified rotation speed and plunge rate. Then the tool is traversed along the line of joint once the shoulder is in contact with the surfaces of workpieces to be jointed. During the process, the tool contributes to two primary functions: heating of workpieces and movement of material to produce the joint. The heating can be accomplished by the friction between the tool and the workpieces and the severe plastic deformation of workpieces generated around the tool, simultaneously, the material around the pin is softened. Due to the combination of tool rotation and translation, the softened material is stirred from the front of the pin to the back of the pin, joining the workpieces together [3].

Besides, it is worth noting that the generated heat is always not sufficient to melt the material (the fusion point of material is not reached) during the entire welding process. As a consequence, the tool is traversed along the weld interface to mix the joining members in a forging action along the joining line to produce a joint in "solid state" [99, 108]. The tool is retracted when the welding process is finished.

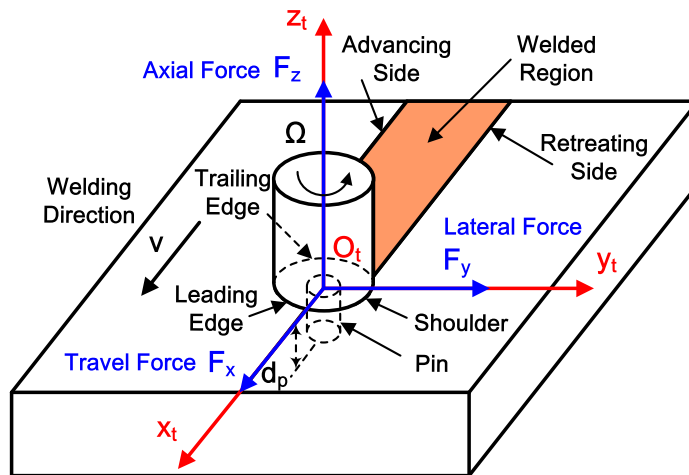


Figure 4.1: Schematic drawing of friction stir welding

A schematic drawing of a linear FSW process where two workpieces are being welded is shown in Figure 4.1. Several model parameters of the process forces are described in this figure as well, including the plunge depth  $d_p$ , travel speed  $v$  and rotation speed of the tool  $\Omega$ .

As for the process forces, the axial force  $F_z$  is generated by the pressure acting on the end of the pin and shoulder of the tool, causing a forging action that produces good microstructure of the welded parts. It is supposed that the tool is always perpendicular to the manufacturing planar during the process so as to simplify the force model. Typically, the axial force is the largest one of the three process forces and has an important contribution to the weld quality. In order to perform a good welding, the axial force should be controlled as a desired constant value.

The travel force  $F_x$  is the force imposed on the tool along the welding direction, which is caused by the stirred material imbalance from the leading to the trailing edge in the plastic deformation zone and the movement of the plastic deformation zone toward the unprocessed material (reaction force).

The lateral force  $F_y$  is the force exerted on the tool in the plane of the part and perpendicular to the welding direction. The direction of the force is usually from the weld retreating side to the advancing side and is produced by the unbalance of the material flow on the two sides [107].

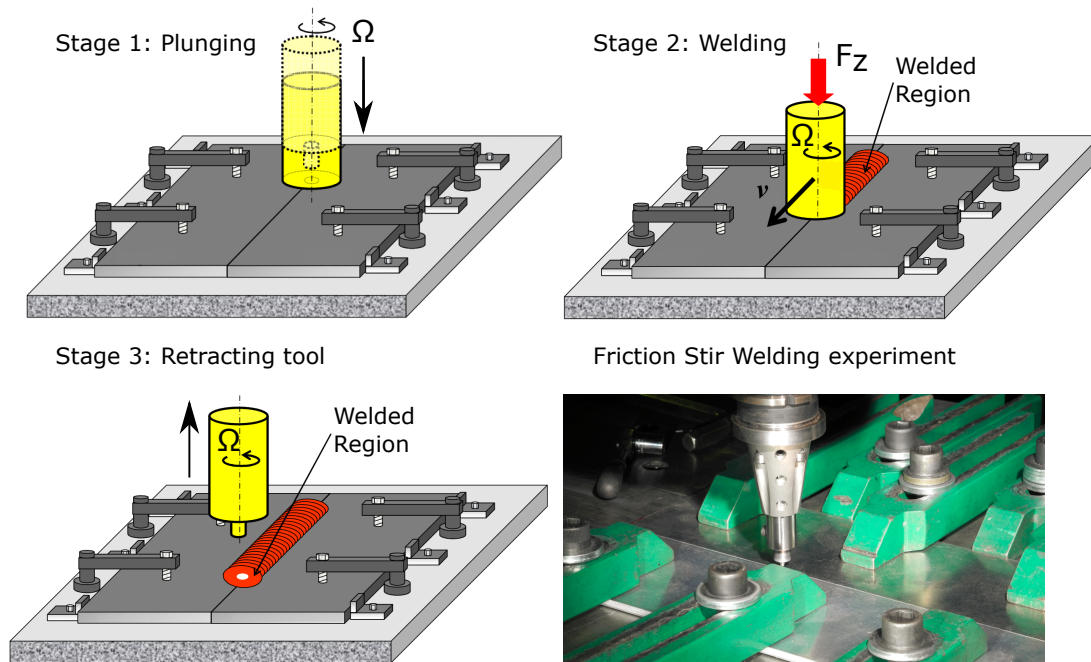


Figure 4.2: Friction stir welding experiment and its three main stages [2]

The FSW process can be divided into three main stages : plunging, welding and retracting of the tool, as illustrated in Figure 4.2. The photo in Figure 4.2 shows a friction stir welding of two aluminum alloy parts.

In addition to the above three main stages, other stages including the dwell time, acceleration and deceleration can also be seen in Figure 4.3, which provides an evolution of the forces and torque applied on the tool during the FSW process of 6000 aluminum alloy with a welding depth of 6 mm [109]. The test was carried out by an instrumented MTS-ISTIR-10 Friction Stir Welder with position controlled plunging stage and force controlled welding stage. In the figure, two peaks of the axial force  $F_z$  can be found during the plunging stage: the first one is caused by the insertion of the pin into the workpieces, and the second one is related to the contact between the shoulder and the material [110]. During the stage of "dwell time", the tool rotates at the desired welding depth without any travel speed so that the material around the pin can reach the viscoplastic state, preparing for the welding stage. The stages of "acceleration" serves for increasing the travel speed  $v$  to a constant value which is designed for the welding stage, and "deceleration" stage is prepared for the retracting of the tool. It can be seen that the axial force  $F_z$  and

spindle torque  $C_z$  are much larger than the travel force  $F_x$  and the lateral force  $F_y$  through the whole process. In addition, by analyzing the mechanical interactions between the tool and material through the process forces and the spindle torque, Zimmer et al. [111] conclude that the forces and torque generated during the stage of plunging and welding at constant speed are characteristic for a static qualification of a FSW means of production.

Thus, the modeling and identification of the process forces mainly consider the plunging stage and the welding stage in this thesis.

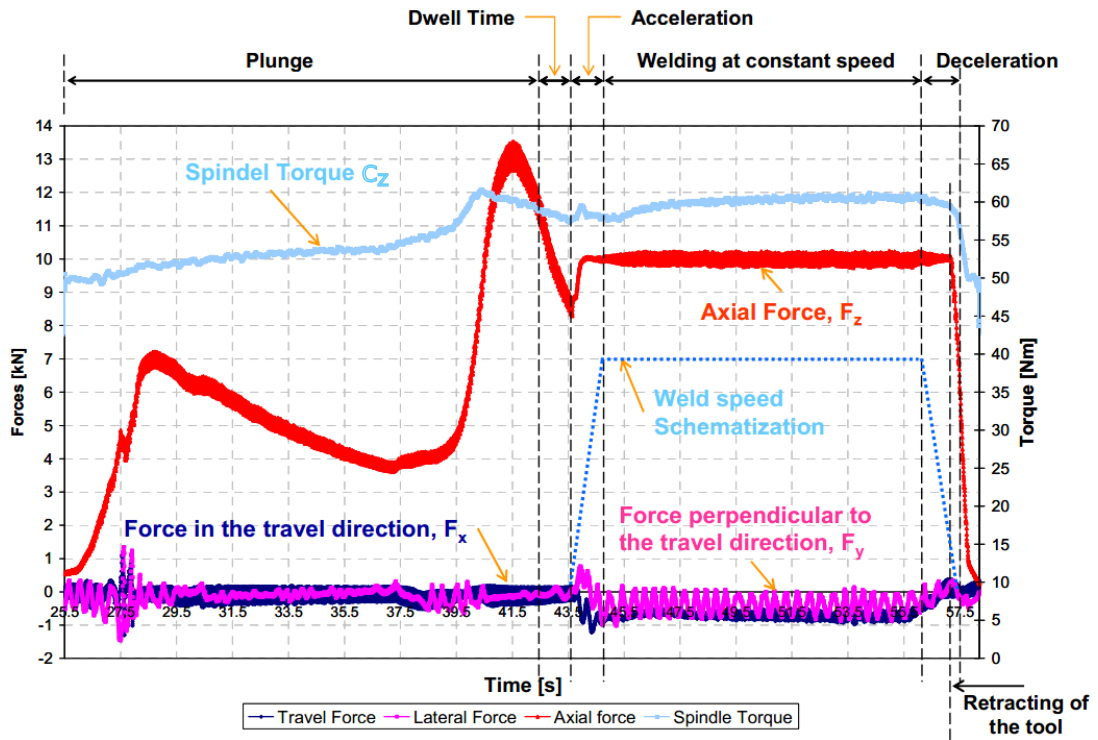


Figure 4.3: Measurements of the forces and torque applied on the tool during the FSW process implemented by an instrumented MTS-ISTIR-10 Friction Stir Welder on 6000 aluminum alloy with a welding depth of 6 mm [111]

### 4.3 Static Modeling of Process Forces in FSW

The FSW process can be affected by numerous parameters and factors such as plunge depth, rotation speed, travel speed, tool geometry, material thermomechanical properties, etc. The process parameters for modeling the forces mainly include plunge depth, rotation speed and travel speed of the tool. Although the factors like weld material properties, tool geometry, fixturing and travel angles have a significant influence on the process forces other than the process parameters, these factors can be considered as constant for the modeling.

Based on the research of [107], the nonlinear power models which are used to describe the steady-state relationships between the process forces and the process

parameters during the welding stage can be given by:

$$\begin{aligned} F_z &= K_z d_p^{\alpha_z} v^{\beta_z} \Omega^{\lambda_z} \\ F_x &= K_x d_p^{\alpha_x} v^{\beta_x} \Omega^{\lambda_x} \\ F_y &= K_y d_p^{\alpha_y} v^{\beta_y} \Omega^{\lambda_y} \end{aligned} \quad (4.1)$$

where  $F_x$ ,  $F_y$  and  $F_z$  are the travel force, lateral force and axial force respectively. The plunge depth  $d_p$ , travel speed  $v$  and rotation speed  $\Omega$  are input process parameters. The parameters  $K_i$ ,  $\alpha_i$ ,  $\beta_i$ ,  $\lambda_i$  ( $i \in x, y, z$ ) are the unknown constant parameters to be identified. It should be noted that the plunge depth  $d_p$  is defined as zero when the bottom of the pin is touching the top surface of the workpiece.

As for parameters identification, these nonlinear power models can be transformed to a linear form by taking the natural log of equation (4.1). For example, the axial force model can be transformed as:

$$\ln(F_z) = \ln(K_z) + \alpha_z \ln(d_p) + \beta_z \ln(v) + \lambda_z \ln(\Omega) \quad (4.2)$$

As a result, the previous nonlinear models can be built and evaluated by means of linear regression analysis, and the unknown parameters can be simply estimated by using the least-square (LS) method.

It should be noted that the influence of the plunge depth  $d_p$  on the travel force  $F_x$  is negligible, as compared with the influence of the travel speed  $v$  and rotation speed  $\Omega$ . Thus, the resulting static model of process forces are:

$$\begin{aligned} F_z &= 0.131 d_p^{2.207} v^{0.097} \Omega^{-0.230} \\ F_x &= 0.151 v^{0.999} \Omega^{-1.232} \\ F_y &= 20.8 d_p^{-3.93} v^{0.654} \Omega^{-0.277} \end{aligned} \quad (4.3)$$

where the units of  $F_x$ ,  $F_y$ ,  $F_z$ ,  $d_p$ ,  $v$  and  $\Omega$  are  $kN$ ,  $kN$ ,  $kN$ ,  $mm$ ,  $mm/s$  and  $rpm$ .

Based on the static model of process forces, the empirical dynamic models for the FSW process have been proposed by [107] by using the plunge depth  $d_p$  as the input. The parameters of the dynamic models have been estimated by using the recursive least-square method (RSL) on three groups of experiments. Accordingly, the transfer functions of the FSW process forces are obtained as:

$$\begin{aligned} F_z(s) &= \frac{3.86 v^{0.097} \Omega^{-0.230} (s + 2.25)}{(s + 25.5)(s + 2.65)} d_p^{2.207}(s) \\ F_x(s) &= \frac{0.151}{0.635s + 1} (v^{0.999} \Omega^{-1.232})(s) \\ F_y(s) &= \frac{20.8 d_p^{-3.93} \Omega^{-0.277}}{0.421s + 1} v^{0.654} \end{aligned} \quad (4.4)$$

where the units of  $F_x$ ,  $F_y$ ,  $F_z$ ,  $d_p$ ,  $v$  and  $\Omega$  are  $kN$ ,  $kN$ ,  $kN$ ,  $mm$ ,  $mm/s$  and  $rpm$ . Thus, the identification obtains a second order dynamic model for axial force  $F_z$  and two first order dynamic models for travel force  $F_x$  and lateral force  $F_y$ .

As a matter of fact, the axial force  $F_z$  is commonly measured by the force sensor mounted on the welding equipment included in the industrial robots. Therefore, the

axial force  $F_z$  can be considered as an input parameter when identifying other models of process forces and torques (e.g., spindle torque  $C_z$ ). Besides, the experiments show that there is not an obvious correlation between the other process forces and torques ( $F_x$ ,  $F_y$ ,  $C_z$ ) and the plunge depth  $d_p$ , meanwhile, the same is true for the relationship between the torque  $C_z$  and travel speed  $v$  [112]. According to equation (4.1), the modified static models for  $F_x$ ,  $F_y$  and  $C_z$  can be written as follows:

$$\begin{aligned} F_x &= K_x F_z^{\alpha_x} v^{\beta_x} \Omega^{\lambda_x} \\ F_y &= K_y F_z^{\alpha_y} v^{\beta_y} \Omega^{\lambda_y} \\ C_z &= K_z F_z^{\alpha_z} \Omega^{\lambda_z} \end{aligned} \quad (4.5)$$

The research for identifying the model parameters of process forces and torques in equation (4.5) has been carried out by our project partners, and the measurements of the process forces and torques ( $F_x$ ,  $F_y$ ,  $F_z$  and  $C_z$ ) have been implemented in a lot of friction stir welding tests for the static model identification (same as Figure 4.3). The identified static models of the process forces and torques show good accuracy in the experimental validations [2, 113].

## 4.4 Experimental Setup

The basic welding system used for conducting the FSW process is composed of the industrial robot KUKA, an electrically driven FSW spindle, the control system of the robot and the sensors (see Figure 4.4). The FSW spindle, which is coincident with the axis 6 of the robot, is installed on the robot flange via the mounting kit.

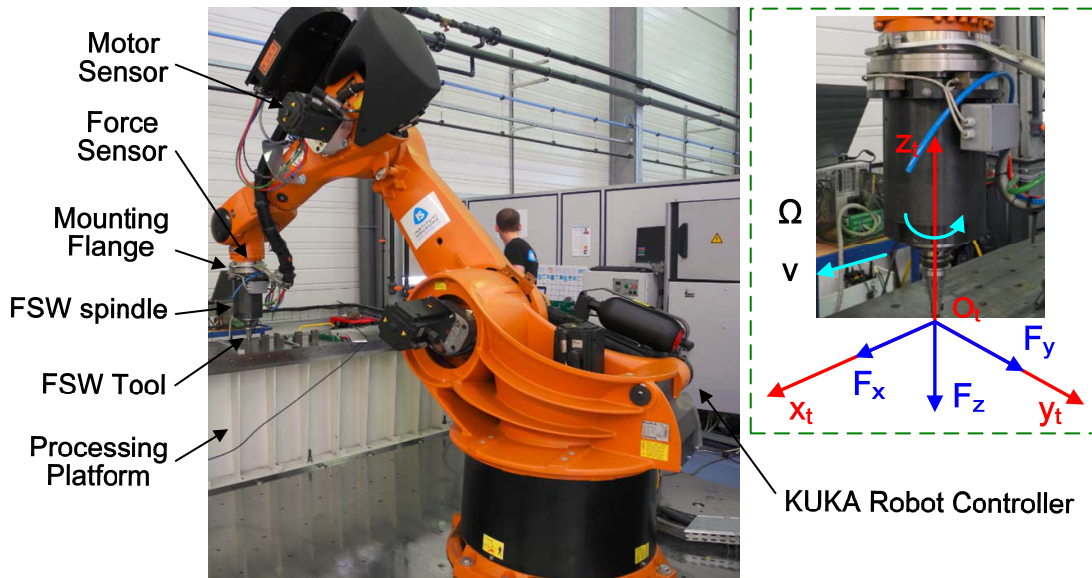


Figure 4.4: Experimental setup: FSW system based on robot KUKA KR500-2MT (by courtesy of Institut de Soudure, Goin, France)

The friction stir welding of some aluminum alloy plates (type Al6082) is implemented by the robotic welding system with the FSW tool called COR-12-6 which is

specially developed by Institut de Soudure. During the welding stage of the FSW process, the spindle is controlled by the robot controller to weld in  $x$  direction of the tool frame with constant travel speed ( $v = 400 \text{ mm/min}$ ) and rotation speed ( $\Omega = 1100 \text{ tr/min}$ ). The desired axial force is  $9000 \text{ N}$ . All the welding conditions for the experimental tests are listed in Table 4.1. The desired welding trajectory in 3D space, measured (or calculated) angular position, velocity and acceleration of motor axes after the gear reduction ( $\theta_i$ ,  $\dot{\theta}_i$  and  $\ddot{\theta}_i$ ) along the trajectory can be found in Figure 4.5 (the excitation in  $z$  direction of the tool frame is added).

Table 4.1: Robotic FSW experimental conditions

| Parameters                       | Name/Value    | Unit     |
|----------------------------------|---------------|----------|
| weld material                    | Al6082        |          |
| FSW tool type                    | COR-12-6 (IS) |          |
| material thickness ( $d_m$ )     | 10            | $mm$     |
| travel speed ( $v$ )             | 400           | $mm/min$ |
| rotation speed ( $\Omega$ )      | 1100          | $tr/min$ |
| desired axial force ( $F_{zd}$ ) | 9000          | $N$      |

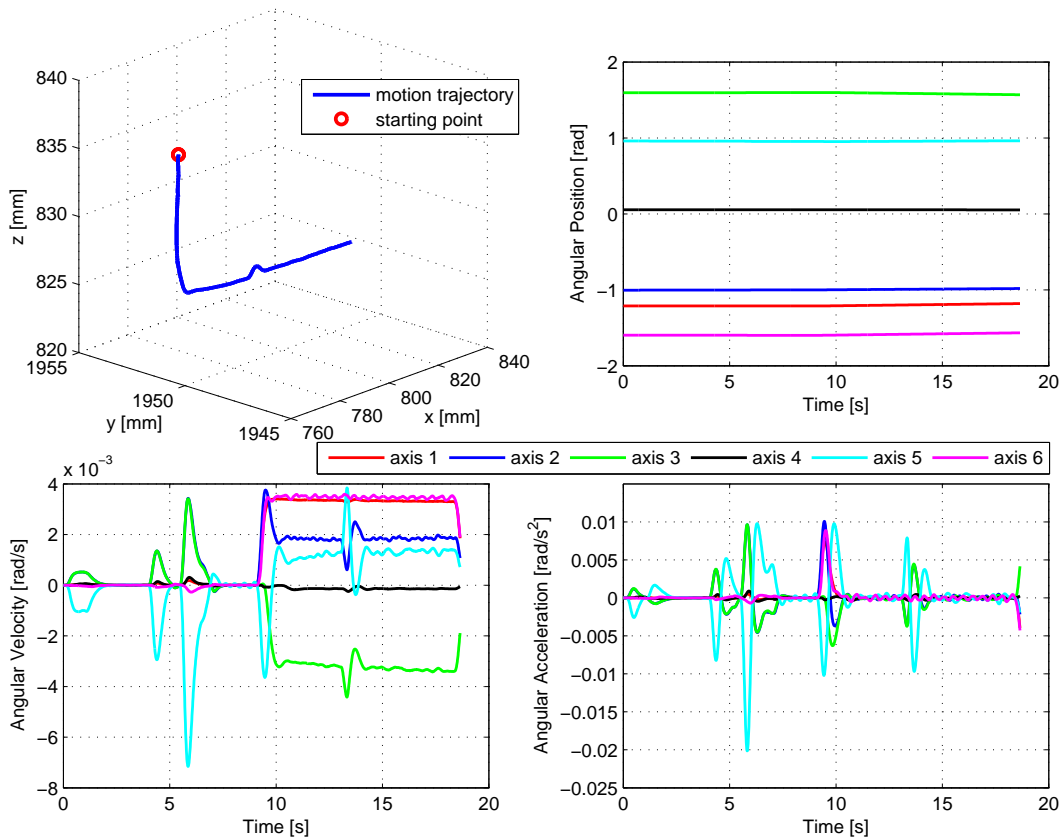


Figure 4.5: The desired welding trajectory in 3D space, measured (or calculated) joint angular position, velocity and acceleration of motor axes after the gear reduction along the welding trajectory (test IS1505I)

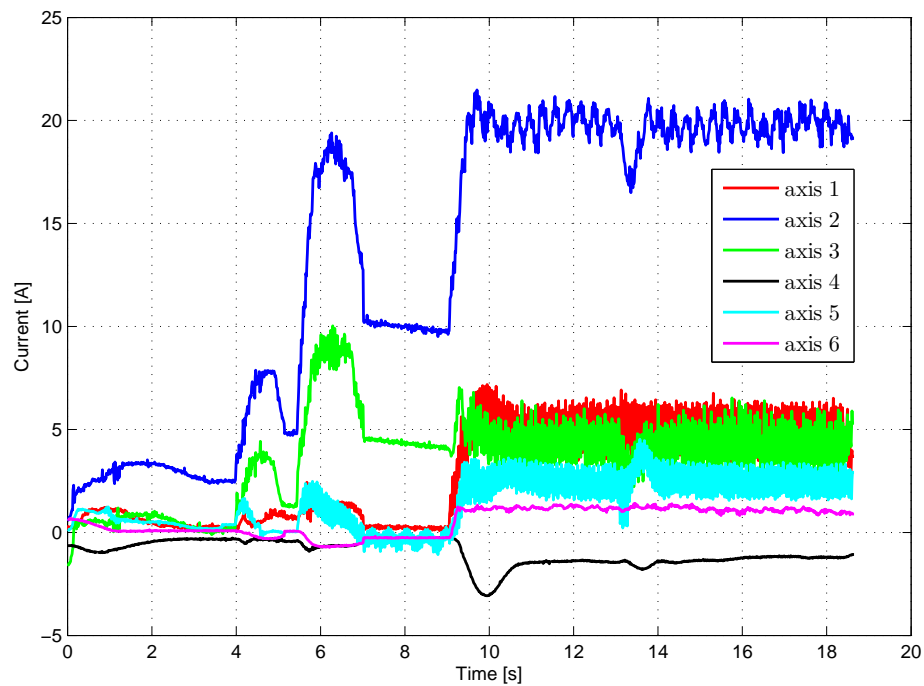


Figure 4.6: Measured currents of all the motors during the FSW test (test IS1505I)

As for the robot sensors, the angular positions of motor axes after the gear reduction can be obtained by the incremental motor encoders and the currents of all motors can be acquired by the actuators. The axial force can be measured by the Schunk FTC force sensor which is mounted on the axis 6 of the robot.

Furthermore, the desired welding trajectory which represents the commanded tool position with respect to the base frame can be generated by the direct geometric model (DGM) and the measured motor angular position after the gear reduction ( $\theta_l$ ). The currents of all the motors during the FSW test are shown in Figure 4.6. Besides, the measurements of the axial force and corresponding data filtering will be detailed in the next section.

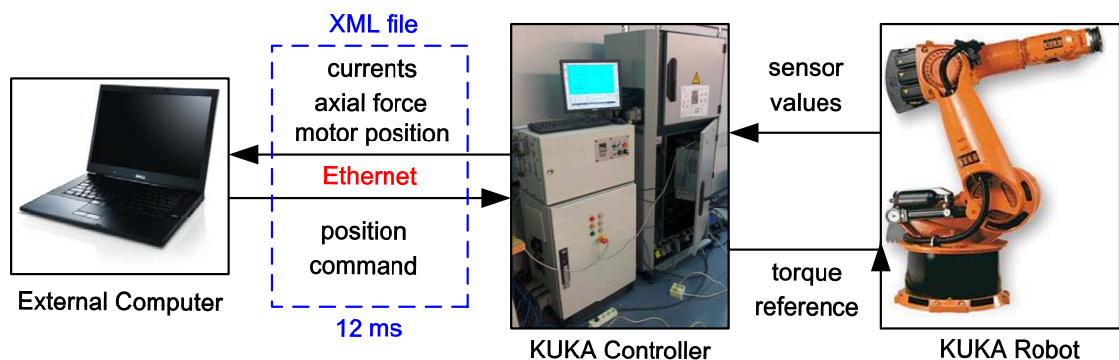


Figure 4.7: Signal flow schematic of the experimental testbed

The transfer of the measurements can be realized by connecting an external computer with the control cabinet of KUKA robot via an Ethernet cable. During



the FSW tests, the KUKA controller sends an *XML* file to the external computer in every 12 *ms*. This file contains all the measurements including the measured angular positions of robot axes after the gear reduction ( $\theta_l$ ), the currents of all motors ( $I$ ) as well as the force in  $z$  direction of the tool frame ( $F_{zm}$ ). Besides, the position command can be given to the KUKA controller through the *XML* file as well. The signal flow schematic of the experimental testbed is provided in Figure 4.7.

In our studies, the dynamic modeling and identification of the force in  $z$  direction (hereafter, referred as axial force  $F_z$ ) are mainly considered. This mainly results from two reasons: Firstly, the axial force is measured by most of the industrial robots because of the need for the force control. By contrast, installing additional force sensors on the robot will lead to high cost. Secondly, as mentioned in the description of the FSW process, the axial force  $F_z$  is much larger than the travel force  $F_x$  and the lateral force  $F_y$ . In order to obtain good weld quality, the axial force during the welding stage should be controlled as a desired constant value. As for dynamic modeling of the travel force  $F_x$  and the lateral force  $F_y$ , the combination of the identified dynamic model of axial force and the modified static models in terms of axial force in equation (4.5) can be applied.

In addition to the axial force  $F_z$ , the tool position in Cartesian space is required for the identification. Typically, the external force exerted on the robot is very large during the robotic FSW. In this case, the deformation of the robot can not be neglected and the flexibility of the robot needs to be considered. Thus, the angular positions of robot axes ( $q$ ) is different from the measured angular positions of robot axes ( $\theta_l$ ) due to the joint flexibility. As a result, the tool position which is calculated by using the direct geometric model (DGM) and  $\theta_l$  will be inaccurate. However, the direct measurement of the tool position is extremely complicated and costly for the robotic FSW process. Thus, it should be a good choice to use the measured angular positions of motor axes after the gear reduction ( $\theta_l$ ) and the currents of all motors ( $I$ ) to estimate the actual tool position with the help of an observer.

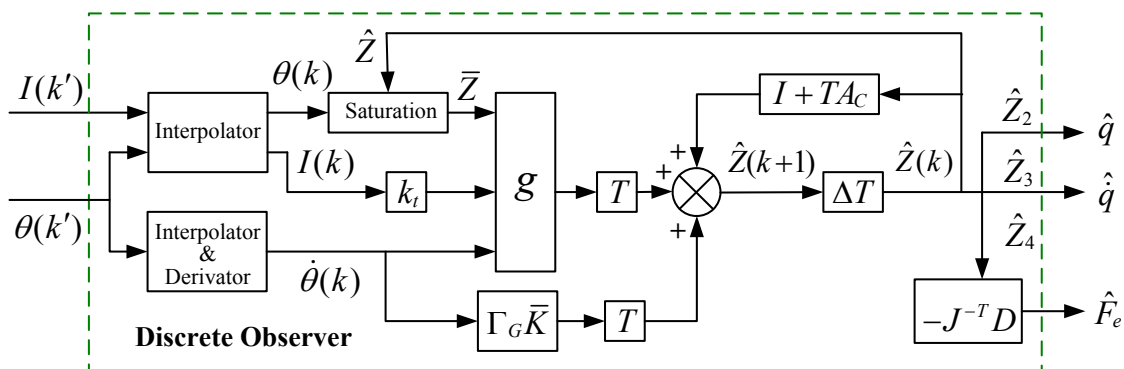


Figure 4.8: Diagram of the discrete observer for robot state estimation [73]

In the research work presented by [4, 72, 73], a nonlinear high-gain discrete observer is proposed to compensate the tracking error due to the flexibility of robot manipulators. The proposed observer is used to estimate the actual angular positions and velocities of robot axes ( $q$ ,  $\dot{q}$ ) as well as the external force ( $F_e$ ) by using only

the motor side measurements including the motor angular positions and velocities ( $\theta, \dot{\theta}$ ) and the currents of the actuators ( $I$ ). The block diagram of the observer is shown in Figure 4.8, and the detailed presentation can be found in [4].

In addition, the observer has been experimentally validated on a 6 DOF industrial manipulator during the FSW process, and the observer-based compensation has succeeded to correct the positioning error in real-time implementation [4, 73]. Therefore, the estimated angular positions and velocities of robot axes ( $\hat{q}, \hat{\dot{q}}$ ) as well as the estimated external force ( $\hat{F}_e$ ) can be used for calculating the actual tool position in the next section. The diagram of data acquisition and processing in the external observer-based test of KUKA robot is given in Figure 4.9. The excitation in  $z$  direction of the tool frame is added during the FSW process in order to obtain more accurate model parameters for the term of  $\frac{d}{dt}d_p$  in the following identification.

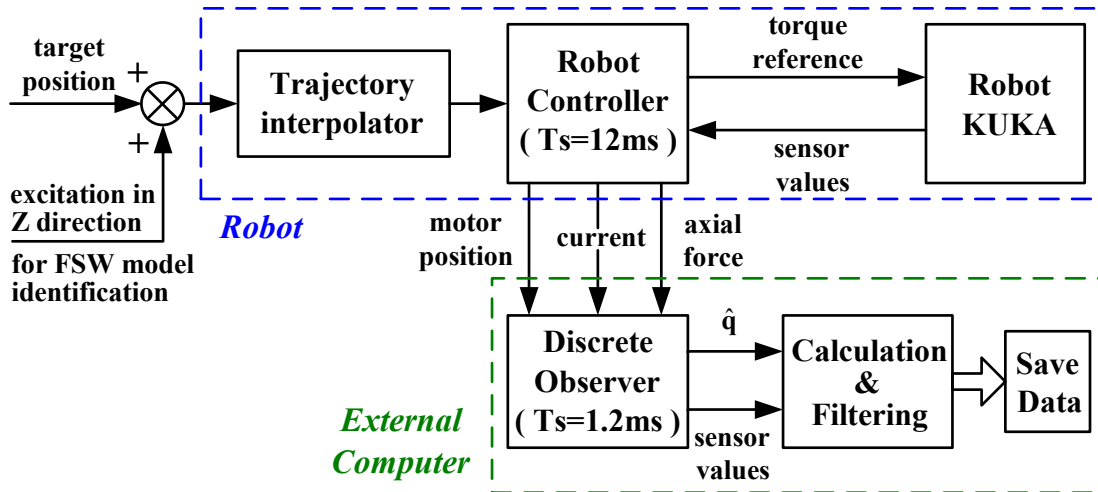


Figure 4.9: Diagram of data acquisition and processing for model identification in the external observer-based test of KUKA robot

## 4.5 Calculation of Plunge Depth and Data Filtering

As mentioned in the experimental setup, the travel speed  $v$  and rotation speed  $\Omega$  are constant in our FSW tests. In this chapter, the dynamic modeling and identification of axial force  $F_z$  will be carried out by using the plunge depth ( $d_p$ ) and its derivative ( $\frac{d}{dt}d_p$ ), which can not be obtained from the experimental measurements directly. Accordingly, the calculation of plunge depth  $d_p$  will be a critical step for the following identification. Besides, the calculation of the derivative of plunge depth and data filtering will be presented in this section as well.

### 4.5.1 Definition of Plunge Depth

As shown in Figure 4.10, the initial tool position of the robot in  $z$  direction ( $Z_0$ ) is supposed to be the specific position when the bottom of the pin is touching the

top surface of the workpiece. In this case, the plunge depth  $d_p$  is actually the tool displacement in  $z$  direction ( $\Delta Z$ ).

During FSW process, for the given tool position in  $z$  direction ( $Z$ ), the plunge depth  $d_p$  can be defined by:

$$d_p = \Delta Z = Z_0 - Z \quad (4.6)$$

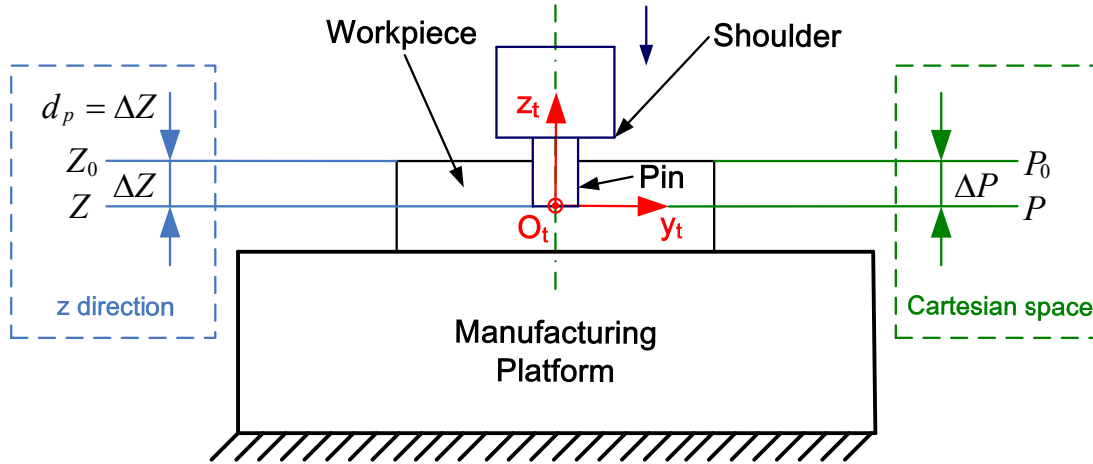


Figure 4.10: Schematic of definition of the plunge depth  $d_p$

Generally, in Cartesian space, the tool displacement from the initial position, denoted by  $\Delta P$ , can be calculated by:

$$\Delta P = P_0 - P \quad (4.7)$$

and

$$\Delta P = [\Delta X \ \Delta Y \ \Delta Z \ \Delta \gamma \ \Delta \beta \ \Delta \alpha]^T \quad (4.8)$$

where  $P$  is the vector of the actual tool position, and  $P_0$  denotes the vector of the initial tool position when the bottom of the pin just touches the top surface of the workpiece. Obviously, the plunge depth  $d_p$  is the third component ( $\Delta Z$ ) of the vector of tool displacement  $\Delta P$ .

However, since large external forces are exerted on the end-effector of the industrial robot in robotic FSW process, the actual tool position  $P$  is different from the commanded tool position  $P_c$  due to the robot joint flexibility. As a consequence, the actual tool position  $P$  can not be directly acquired and the calculation of the plunge depth  $d_p$  will be extremely complicated. According to the above definition, three methods for calculating the plunge depth  $d_p$  are proposed here.

## 4.5.2 Three Methods for Calculating Plunge Depth

### 1. First method for calculating plunge depth $d_p$

The first method is based on the estimated angular positions of robot axes, which are obtained from the discrete nonlinear observer presented in the previous section

[4, 73]. According to the direct geometric model (DGM) of the robot, the estimated tool position  $\hat{P}$  can be given by:

$$\hat{P} = DGM(\hat{q}) \quad (4.9)$$

with  $\hat{q}$  denoting the estimated angular positions of robot axes. Then, the estimated tool displacement  $\widehat{\Delta P}$  can be calculated by:

$$\begin{aligned} \widehat{\Delta P} &= \hat{P}_0 - \hat{P} \\ &= DGM(\hat{q}_0) - DGM(\hat{q}) \end{aligned} \quad (4.10)$$

in which  $\hat{q}_0$  represents the estimated initial angular positions of robot axes, and  $\hat{P}_0$  is the corresponding estimated initial tool position calculated by using  $\hat{q}_0$ .

In fact, when the tool starts to contact with the top surface of the workpiece, the commanded tool position can be considered to be equal to the actual tool position, accordingly:

$$\widehat{\Delta P} = DGM(\theta_{i0}) - DGM(\hat{q}) \quad (4.11)$$

where  $\theta_{i0}$  denotes the vector of measured initial motor angular position after the gear reduction.

Although this method based on the observer is extremely simple for calculating the plunge depth, the disadvantages are obvious as well:

- 1) the discrete nonlinear observer used in the FSW experiments produces a phase shift due to the usage of on-line filters;
- 2) the friction model used in the observer is not precise, affecting the accuracy of the calculation results.

## 2. Second method for calculating plunge depth $d_p$

Different from the first method, the second method is mainly based on the flexibility model of the flexible joint robot presented in Section 2.4. As shown in Figure 4.11, the actual tool position  $P$  can be given by:

$$P = P_c - \tilde{P} \quad (4.12)$$

where  $P_c$  is the commanded tool position and  $\tilde{P}$  denotes the deformation of the robot expressed in the tool frame in Cartesian space.

First of all, the commanded tool position  $P_c$  for the industrial robots can be easily obtained by the direct geometric model (DGM):

$$P_c = DGM(\theta_l) \quad (4.13)$$

where  $\theta_l = N^{-1}\theta$  represents the vector of angular positions of motor axes after the gear reduction,  $\theta$  denotes the vector of angular positions of motor axes and  $N$  is the gear transmission ratio matrix.

In the joint space of the robot, according to equations (2.56) and (2.58), the deformation of the robot joint  $\tilde{q}$  can be given by:

$$\begin{aligned} \tilde{q} &= \theta_l - q \\ &= K_r^{-1}\Gamma \end{aligned} \quad (4.14)$$

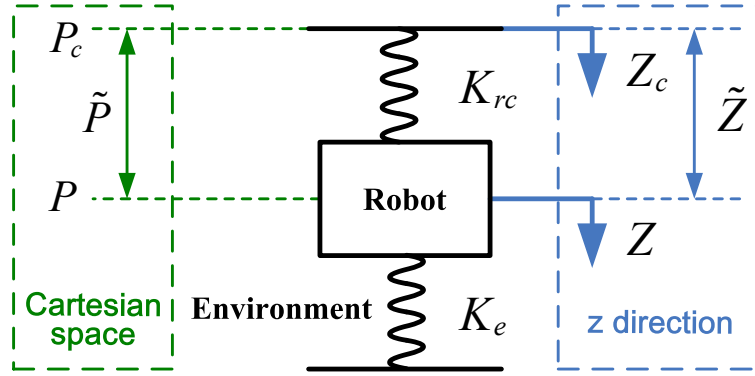


Figure 4.11: Schematic of the flexibility model in Cartesian space

where  $q$  is the vector of actual angular positions of robot axes,  $K_r$  denotes the stiffness matrix of robot joints, and  $\Gamma$  is the vector of the gearbox output torque.

Then, the deformation of the robot expressed in the tool frame in Cartesian space  $\tilde{P}$  can be calculated by:

$$\begin{aligned}\tilde{P} &= J(q)\tilde{q} \\ &= J(\theta_l - \tilde{q})K_r^{-1}\Gamma\end{aligned}\quad (4.15)$$

in which  $J$  is the Jacobian matrix of the tool frame with respect to the fixed frame. In order to simplify the calculation, the term  $J(\theta_l - \tilde{q})$  can be approximately replaced by  $J(\theta_l)$ . In this study, the actual angular position ( $q = \theta_l - \tilde{q}$ ) is used as the input variable of Jacobian calculation.

Based on equation (2.52), which represents the robot dynamic equation on robot side, the gearbox output torque  $\Gamma$  can be obtained by:

$$\Gamma = J^T(q)T_e + H(q, \dot{q}) + F_{fr}(\dot{q}) + M(q)\ddot{q}\quad (4.16)$$

where  $T_e$  is the vector of the external torque in the tool frame,  $q$  denotes the vector of actual angular positions of robot axes, which can be approximately replaced by the measured motor angular position after the gear reduction ( $\theta_l$ ).

As a consequence, the deformation of the robot in the tool frame  $\tilde{P}$  can be calculated by:

$$\tilde{P} = J(\theta_l - \tilde{q})K_r^{-1}[J^T(\theta_l)T_e + H(\theta_l, \dot{\theta}_l) + F_{fr}(\dot{\theta}_l) + M(\theta_l)\ddot{\theta}_l]\quad (4.17)$$

Thus, the tool displacement  $\Delta P$  can be obtained as follows:

$$\begin{aligned}\Delta P &= P_0 - P \\ &= P_0 - (P_c - \tilde{P}) \\ &= DGM(\theta_{l0}) - DGM(\theta_l) \\ &\quad + J(\theta_l - \tilde{q})K_r^{-1}[J^T(\theta_l)T_e + H(\theta_l, \dot{\theta}_l) + F_{fr}(\dot{\theta}_l) + M(\theta_l)\ddot{\theta}_l]\end{aligned}\quad (4.18)$$

In addition, it should be noted that the vector of the external torque  $T_e$  can be given approximately by supposing all its components equal to zero except the one in  $z$  direction:

$$T_e = [0 \ 0 \ F_{zm} \ 0 \ 0 \ 0]^T\quad (4.19)$$

with  $F_{zm}$  being the measured axial force. In order to calculate the results more precisely, the estimated external forces can be utilized as other components of the external torque  $T_e$  as well:

$$T_e = [\hat{F}_x \hat{F}_y F_{zm} \hat{C}_x \hat{C}_y \hat{C}_z]^T \quad (4.20)$$

where  $\hat{F}_x$ ,  $\hat{F}_y$ ,  $\hat{C}_x$ ,  $\hat{C}_y$  and  $\hat{C}_z$  are the components of the estimated external torque which are obtained from the discrete nonlinear observer.

### 3. Third method for calculating plunge depth d

Similar to the second method, the third method is on the basis of the flexibility model of the robot as well. The only difference is the calculation of the gearbox output torque  $\Gamma$ .

In accordance with equation (2.53) which represents the robot dynamic equation on motor side, the gearbox output torque  $\Gamma$  can be calculated by:

$$\Gamma = k_t I - F_{fm}(\dot{\theta}) \quad (4.21)$$

where  $k_t$  represents the vector of the motor torque constants,  $I$  is the vector of the currents of all motors, and  $\theta$  denotes the vector of the angular positions of motor axes, which can be calculated by the vector of measured motor angular position after the gear reduction ( $\theta_l$ ):

$$\theta = N\theta_l \quad (4.22)$$

As a result, according to equation (2.56), the deformation of the robot defined in the joint space  $\tilde{q}$  can be given by:

$$\tilde{q} = K_r^{-1} N^T [k_t I - F_{fm}(N\dot{\theta}_l)] \quad (4.23)$$

Consequently, the deformation of the robot in the tool frame  $\tilde{P}$  can be obtained by substituting  $\tilde{q}$  in equation (4.15), formulated by:

$$\tilde{P} = J(\theta_l - \tilde{q}) K_r^{-1} N^T [k_t I - F_{fm}(N\dot{\theta}_l)] \quad (4.24)$$

Finally, the tool displacement  $\Delta P$  can be obtained on the basis of the above calculations:

$$\begin{aligned} \Delta P &= P_0 - P \\ &= P_0 - (P_c - \tilde{P}) \\ &= DGM(\theta_{l0}) - DGM(\theta_l) \\ &\quad + J(\theta_l - \tilde{q}) K_r^{-1} N^T [k_t I - F_{fm}(N\dot{\theta}_l)] \end{aligned} \quad (4.25)$$

### 4.5.3 Derivative of Plunge Depth and Data Filtering

The plunge depth  $d_p$  can be calculated by the above three methods. Since the calculation is implemented off-line, the derivative of the plunge depth, denoted by  $\frac{d}{dt}d_p$ , can be approximated by:

$$\frac{d}{dt}d_p(i) = \frac{d_p(i+1) - d_p(i-1)}{2T_s} \quad (4.26)$$

where  $\frac{d}{dt}d_p(i)$  is the calculated derivative of plunge depth at the  $i^{th}$  iteration,  $d_p(i+1)$  is the calculated plunge depth at the  $i^{th}$  iteration, and  $T_s = 12 \text{ ms}$ , representing the sampling time of the measurements.

As a matter of fact, the data obtained from measurements and calculations generally contain significant noise, therefore, the data filtering is essential to achieve a better identification performance.

To begin with, all the data including the measured axial force  $F_{zm}$ , the plunge depth  $d_p$  and its derivative  $\frac{d}{dt}d_p$  are simply processed by a median filter which can remove the sudden variations in the measurements and calculations without producing any phase shift. Thus, a five-point moving average algorithm is applied to the measured (calculated) data to filter the noise, and the filtered force signal is:

$$F_{zf}(i) = \frac{F_{zm}(i-2) + F_{zm}(i-1) + F_{zm}(i) + F_{zm}(i+1) + F_{zm}(i+2)}{5} \quad (4.27)$$

where  $F_{zf}(i)$  denotes the filtered force data at the  $i^{th}$  iteration and  $F_{zm}(i)$  is the measured force data at the  $i^{th}$  iteration.

Next, in order to eliminate the noise at the frequency of the spindle rotation during the FSW process, a notch filter is applied to the original measured force  $F_{zm}$ , and the specific interfering frequency to be removed is:

$$f_r = \frac{\Omega}{60} = \frac{1100 \text{ r/min}}{60 \text{ s/min}} = 18.3 \text{ Hz} \quad (4.28)$$

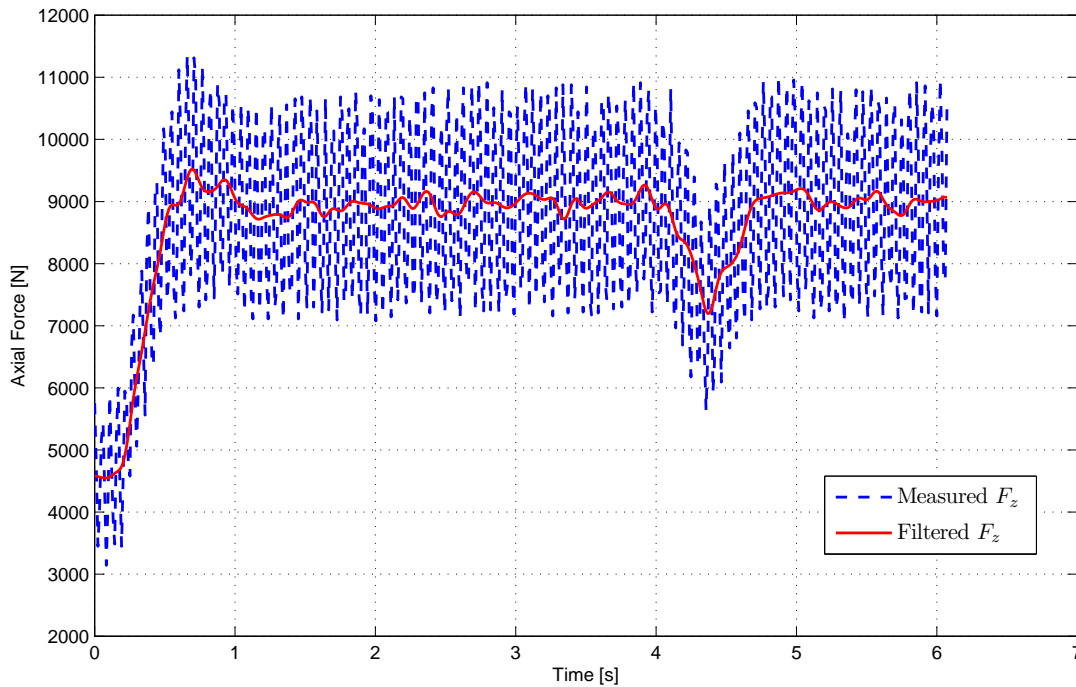


Figure 4.12: Comparison between filtered and original measured axial force  $F_z$  during the FSW process (test IS15051)

Furthermore, since the terms which represent the high frequency phenomena will not be considered in the model identification, a fifth-order Butterworth low-pass filter (the cutoff frequency  $f_c = 8.3 \text{ Hz}$ ) should be utilized for all the data to prevent the high frequency noise which probably disturbs the model identification.

Figure 4.12 shows the comparison between the original measured and filtered axial force  $F_z$ . Besides, it should be mentioned that the plunge depth  $d_p$  can be changed during the welding phase of FSW process by using the position control in  $z$  direction. More precisely, the reference force changes from  $4000 \text{ N}$  to  $9000 \text{ N}$  and after about  $4.1 \text{ s}$  a reference speed of  $16.7 \text{ mm/s}$  in  $z$  direction of the tool frame is added during the following  $72 \text{ ms}$ , leading to the significant change of the axial force. This operation is aiming at obtaining more accurate model parameters for the term of  $\frac{d}{dt}d_p$  in the following identification.

In addition, by using three different methods for calculating the plunge depth  $d_p$  and data filtering, the comparison between the filtered and the calculated plunge depth  $d_p$  and its derivative  $\frac{d}{dt}d_p$  are shown in Figures 4.13, 4.14 and 4.15. These filtered values will be used as the inputs for the identification. Besides, the filtered and the calculated data in Figure 4.14 are obviously different from the corresponding data in Figure 4.13 and Figure 4.15. Such a big difference could be due to the vector of the external torque  $T_e$  used in the second method. In this method, the external torque  $T_e$  is approximately given by supposing all its components equal to zero except the measured axial force  $F_{zm}$  or by using certain components of the estimated external torque based on the observer. As a result, the data calculated by using this method may be less accurate than those calculated by other methods.

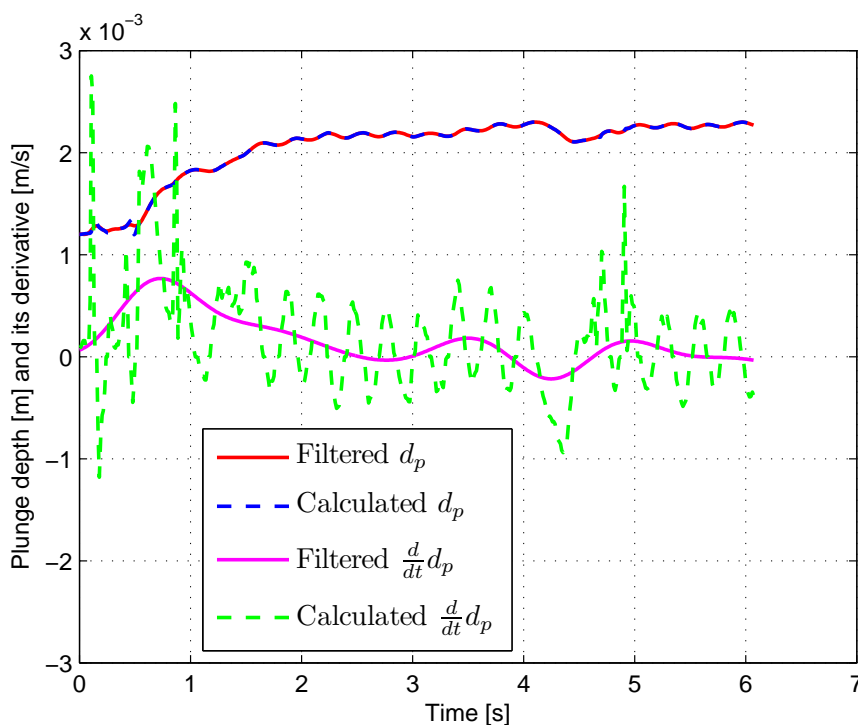


Figure 4.13: Comparison between filtered and calculated plunge depth  $d_p$  and its derivative  $\frac{d}{dt}d_p$  (Method 1)



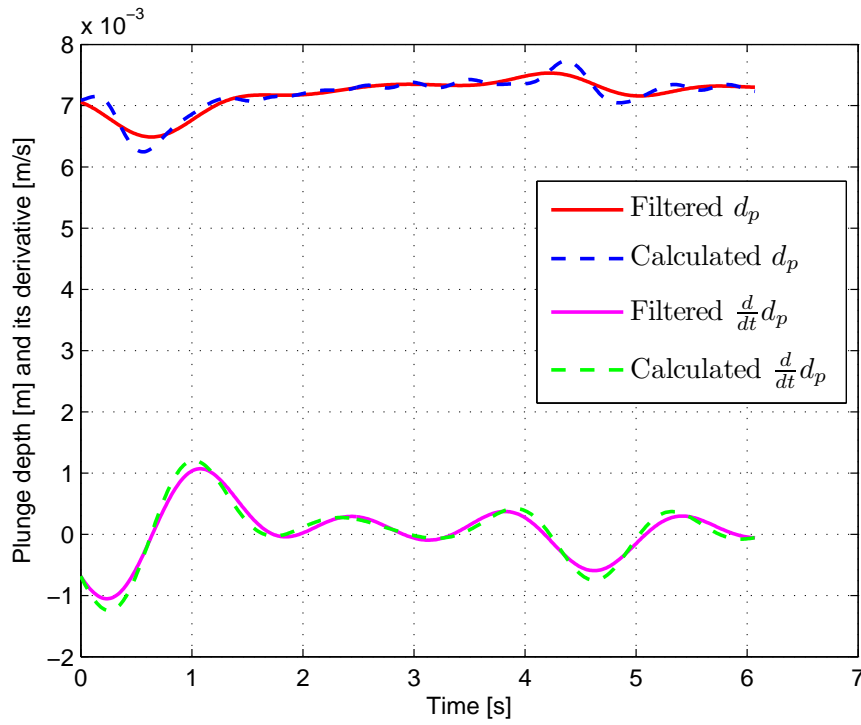


Figure 4.14: Comparison between filtered and calculated plunge depth  $d_p$  and its derivative  $\frac{d}{dt}d_p$  (Method 2)

## 4.6 Linear Dynamic Modeling and Identification of Axial Force

### 4.6.1 Linear Dynamic Model of Axial Force

According to the research work of [104], the relationship between the measured axial force  $F_z$  and the actual vertical position of the tool  $Z$  during the robotic FSW process can be described by using a simple analytical 1-DOF lumped parameter model with one mass ( $m$ ), two springs with different stiffnesses ( $K_{rz}, K_e$ ) and one damper with damping factor ( $f_e$ ), as shown in Figure 4.16.

Then, the following equations that govern the physics of the above simple lumped parameter model are given by:

$$\begin{aligned}
 m \frac{d^2}{dt^2} Z &= F_{rz} - F_z \\
 F_{rz} &= K_{rz}(Z_c - Z) \\
 F_z &= K_e(Z - Z_0) + f_e \frac{d}{dt} Z
 \end{aligned} \tag{4.29}$$

Thus, a linear dynamic model describing the axial force by using the plunge depth and its derivative can be defined as follows:

$$F_z = K_e d_p + f_e \frac{d}{dt} d_p + C \tag{4.30}$$

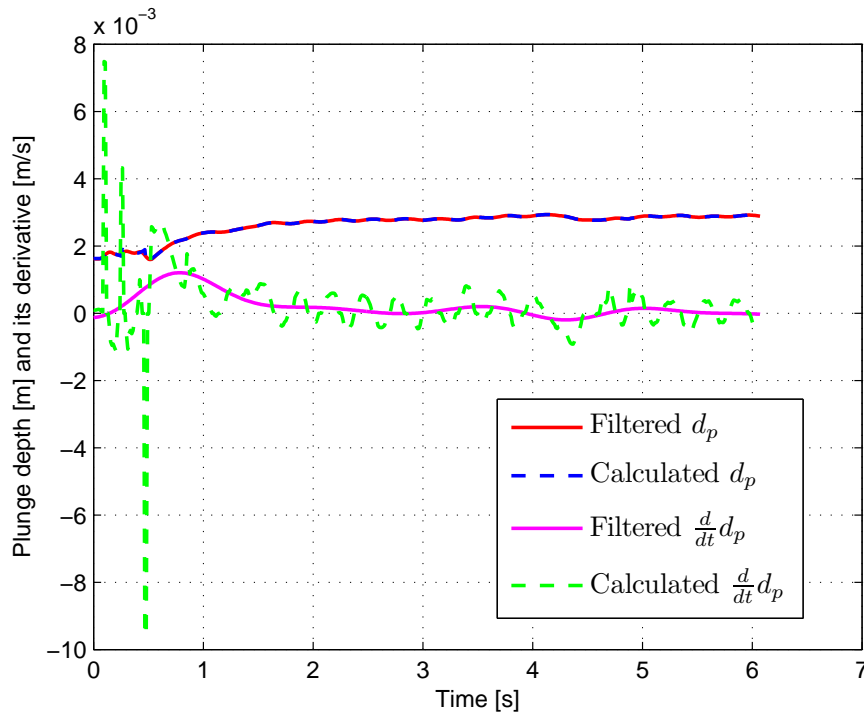


Figure 4.15: Comparison between filtered and calculated plunge depth  $d_p$  and its derivative  $\frac{d}{dt}d_p$  (Method 3)

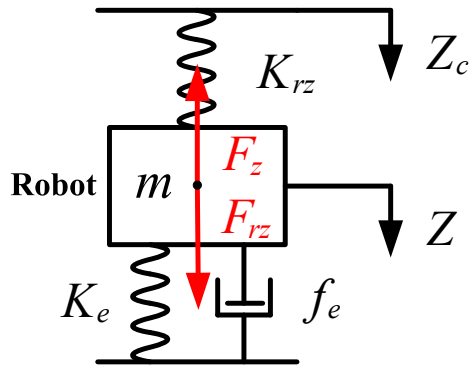


Figure 4.16: Free-body diagram of simple lumped parameter model in  $z$  direction

where  $F_{rz}$  is the force generated by the flexibility of the robot in  $z$  direction,  $F_z$ ,  $d_p$  and  $\frac{d}{dt}d_p$  are the axial force, plunge depth and its derivative respectively,  $K_e$  denotes the stiffness coefficient of the environment,  $f_e$  is the damping factor of the environment, and the constant  $C$  represents the offset of the initial axial force, which is added for a better identification.

#### 4.6.2 Least Squares Method and Error Analysis Method

The least squares (LS) method is a standard approach to find the approximate solution, which can minimize the sum of the squares of the errors for the model

identification problems. This method is extensively used in regression analysis and estimation.

First of all, the linear dynamic model of axial force in equation (4.30) should be transformed to the following matrix form:

$$y = X\gamma \quad (4.31)$$

in which the parameters can be chosen as:

$$y = \begin{bmatrix} \vdots \\ F_z(i) \\ \vdots \end{bmatrix}; X = \begin{bmatrix} \vdots & \vdots & \vdots \\ d_p(i) & \frac{d}{dt}d_p(i) & 1 \\ \vdots & \vdots & \vdots \end{bmatrix}; \gamma = \begin{bmatrix} K_e \\ f_e \\ C \end{bmatrix} \quad (4.32)$$

where  $i$  denotes the serial number of the measurements,  $y$  is the vector of output, which includes the values of the filtered axial force,  $X$  is the matrix of input containing the data from the measurements and calculations, such as the plunge depth  $d_p$  and its derivative  $\frac{d}{dt}d_p$ , and  $\gamma$  represents the vector of unknown parameters including  $K_e$ ,  $f_e$  and  $C$ , which need to be identified.

Based on the least squares (LS) method, the optimal solution of the vector  $\gamma$  can be estimated by the following formula:

$$\hat{\gamma} = (X^T X)^{-1} X^T y \quad (4.33)$$

Thus, the unknown parameters  $K_e$ ,  $f_e$  and  $C$  in the linear dynamic model of the axial force can be identified by using the above equations.

In order to analyze the accuracy of the identified model and the performance of model identification, the following evaluation indexes are mainly concerned:

- 1) the errors of the identified model, including the RMS error, maximum error and mean error;
- 2) the uncertainties of the identified parameters;
- 3) the normality test of the errors.

Generally, calculating the root mean square (RMS) error can be considered as an effective method to evaluate the accuracy of the identified model. The error of the identified axial force model can be defined as:

$$error = F_{zmes} - F_{zfit} \quad (4.34)$$

where  $F_{zmes}$  denotes the measured axial force with data filtering, and  $F_{zfit}$  is the fitted axial force.

Then, the RMS errors of the identified axial force model, expressed as a percentage, can be given by:

$$e_{RMS}(\%) = 100 \frac{\sqrt{(1/n) \sum_{i=1}^n error(i)^2}}{\sqrt{(1/n) \sum_{i=1}^n F_{zmes}(i)^2}} \quad (4.35)$$

Besides, the uncertainties of identified parameters can be used to evaluate the precision of the model identification as well.

For the given identified linear dynamic model, the vector of the identification error  $\varepsilon$  can be calculated by:

$$\varepsilon = y - X\gamma \quad (4.36)$$

As a result, the parameter estimations of the model identification can be considered as unbiased when the bias equals zero, which can be calculated by using the following equation:

$$bias = E((X^T X)^{-1} X^T \varepsilon) \quad (4.37)$$

In order to calculate the uncertainties of the identified parameters, the square of the standard deviation should be firstly computed as follows:

$$\sigma^2 = \frac{\varepsilon^T \varepsilon}{N - N_\gamma} \quad (4.38)$$

where  $\sigma$  represents the standard deviation, which measures the amount of variation or dispersion from the average.  $N$  denotes the number of the measured data used in the identification, and  $N_\gamma$  is the number of parameters to be identified.

After the computation of the standard deviation, the variance can be determined by:

$$Variance(\gamma) = Diag((X^T X)^{-1} \sigma^2) \quad (4.39)$$

Then, by using the estimated solution  $\hat{\gamma}$  of the identified model, the vector of uncertainties of the identified parameters can be calculated by:

$$Uncertainty(\gamma) = \frac{\sqrt{Variance(\gamma)}}{\hat{\gamma}} \quad (4.40)$$

Finally, the normality test of errors is implemented with the help of the *Matlab* function *lillietest* which can perform a Lilliefors test for the errors of the identified model, determining whether the errors come from a normal distribution.

### 4.6.3 Results and Error Analysis of Identified Linear Model

In the light of the above algorithm, the program of model identification can be developed by means of the software *Matlab*. Then, three linear dynamic models of the axial force can be identified by using the data of measurements and calculations in Figure 4.12 to 4.15. By defining the units of  $F_z$ ,  $d_p$  and  $\frac{d}{dt}d_p$  as  $N$ ,  $m$  and  $m/s$  respectively, the three identified linear models are given as follows:

#### 1) Linear Model 1 based on Calculation Method 1

$$F_z = 3.93 \times 10^6 d_p + 3.64 \times 10^6 \frac{d}{dt}d_p + 16.4 \quad (4.41)$$

#### 2) Linear Model 2 based on Calculation Method 2

$$F_z = 1.24 \times 10^6 d_p + 1.69 \times 10^6 \frac{d}{dt}d_p - 380.5 \quad (4.42)$$

### 3) Linear Model 3 based on Calculation Method 3

$$F_z = 3.02 \times 10^6 d_p + 2.44 \times 10^6 \frac{d}{dt} d_p + 180.8 \quad (4.43)$$

Systematic studies on the errors of the identified linear axial force models have been carried out in order to analyze the effectiveness and performance of the model identification. The accuracy of the axial force model, the uncertainties of identified parameters and the normality test of errors are taken into consideration.

By means of the equation (4.35), the RMS errors of the three linear models are calculated to be 4.47%, 7.75% and 4.75% respectively. All other results of error calculations including the standard deviation, maximum errors and mean errors for the three linear identified models are shown in Table 4.2.

Table 4.2: Errors of identified models

| Name                       | Method 1 | Method 2 | Method 3 |
|----------------------------|----------|----------|----------|
| standard deviation ( $N$ ) | 389.33   | 669.93   | 413.38   |
| maximum error (%)          | 24.70    | 54.67    | 24.09    |
| mean error (%)             | 3.72     | 6.51     | 3.82     |
| RMS error (%)              | 4.47     | 7.75     | 4.75     |

The values of the identified parameters and their uncertainties for the three linear models are listed in Table 4.3, 4.4 and 4.5.

Table 4.3: Value of identified parameters and their uncertainties for linear model 1

| Parameters (Model 1) | $K_e$ ( $N/m$ )    | $f_e$ ( $Ns/m$ )   | $C$ ( $N$ ) |
|----------------------|--------------------|--------------------|-------------|
| Value                | $3.93 \times 10^6$ | $3.64 \times 10^6$ | 16.4        |
| Uncertainty (%)      | 1.92               | 2.67               | —           |

Table 4.4: Value of identified parameters and their uncertainties for linear model 2

| Parameters (Model 2) | $K_e$ ( $N/m$ )    | $f_e$ ( $Ns/m$ )   | $C$ ( $N$ ) |
|----------------------|--------------------|--------------------|-------------|
| Value                | $1.24 \times 10^6$ | $1.69 \times 10^6$ | -380.5      |
| Uncertainty (%)      | 9.15               | 3.88               | —           |

Table 4.5: Value of identified parameters and their uncertainties for linear model 3

| Parameters (Model 3) | $K_e$ ( $N/m$ )    | $f_e$ ( $Ns/m$ )   | $C$ ( $N$ ) |
|----------------------|--------------------|--------------------|-------------|
| Value                | $3.02 \times 10^6$ | $2.44 \times 10^6$ | 180.8       |
| Uncertainty (%)      | 2.10               | 2.73               | —           |

Figures 4.17, 4.18 and 4.19 provide the comparisons between filtered and fitted axial force  $F_z$  as well as the generated error for the three identified linear models. As shown in Figure 4.17, the fitted axial forces are found to coincide essentially with the filtered axial forces on the whole, although some parts do not match exactly because of the phase shifts brought by the discrete nonlinear observer. However, it should be noted that the phase shifts can be eliminated if the offline filter can be applied to the measured data used in the observer, improving the accuracy of the identified model. In contrast, the fitted axial forces are not in good agreement with the filtered axial forces in Figures 4.18 and 4.19, which are possibly due to the inaccurate friction model and the approximate calculation used in the second and the third method. As mentioned previously, the input data calculated by the second method are not accurate due to the specified external torque or the estimated one obtained from the observer, as a result, the corresponding identified model will not be precise. According to the results of the identification, the identified linear model by using the first calculation method will be used in the following study.

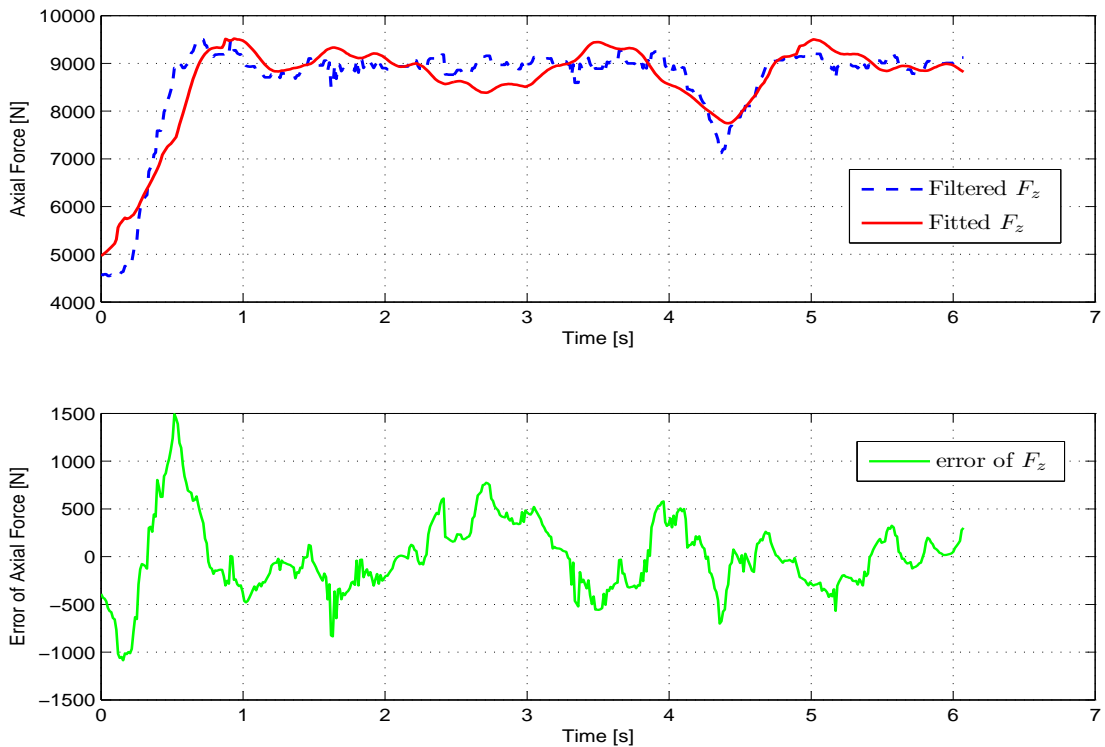


Figure 4.17: Comparison between filtered and fitted axial force  $F_z$  and generated error for Linear Model 1

Finally, the normality tests of the errors are performed by using the *Matlab* function *lillietest*, and the histograms of error and the estimated Gaussian for the three linear models with data filtering are shown in Figures 4.20, 4.21 and 4.22. All the above analysis and results show a good performance of the model identification.

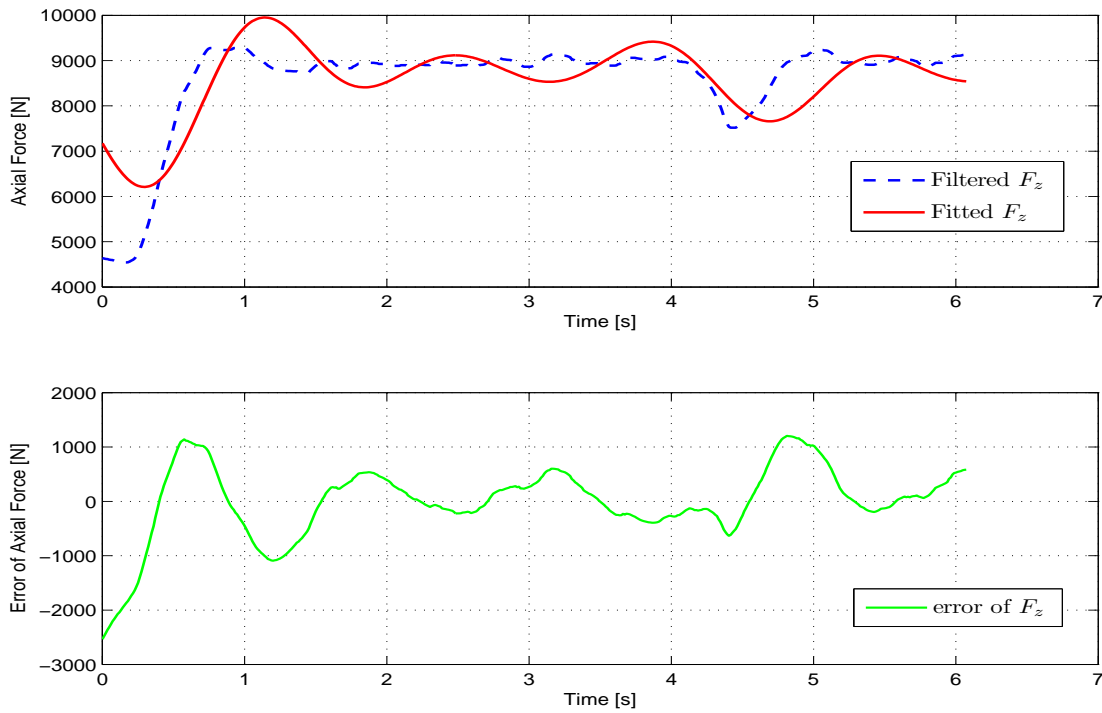


Figure 4.18: Comparison between filtered and fitted axial force  $F_z$  and generated error for Linear Model 2

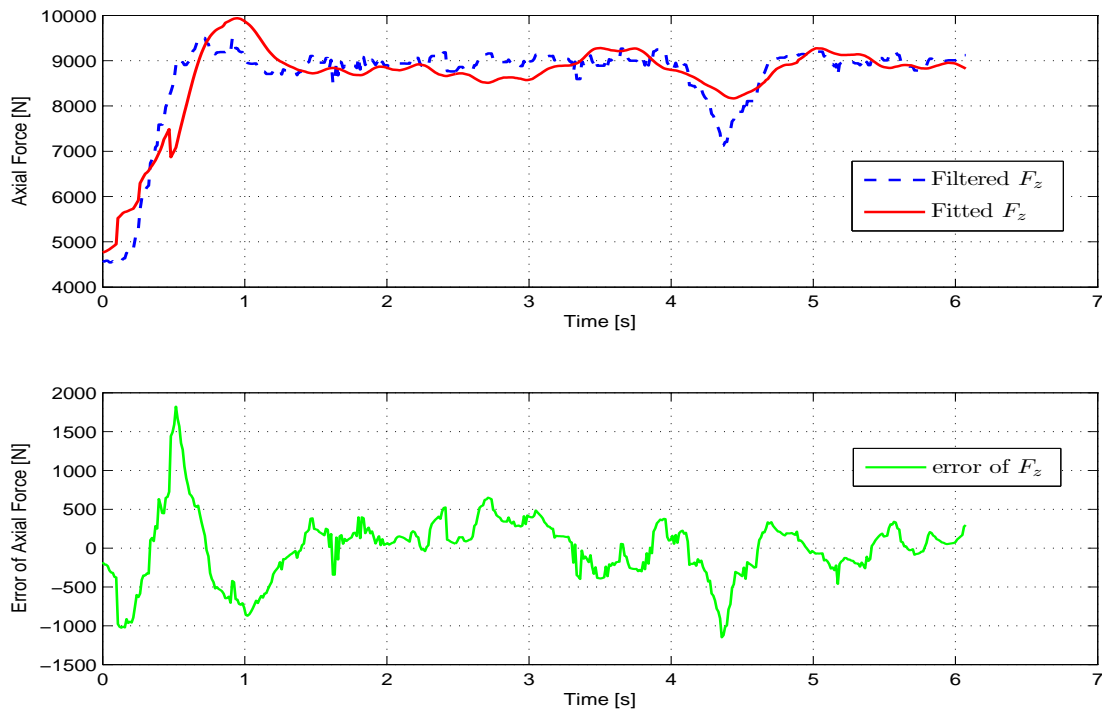


Figure 4.19: Comparison between filtered and fitted axial force  $F_z$  and generated error for Linear Model 3

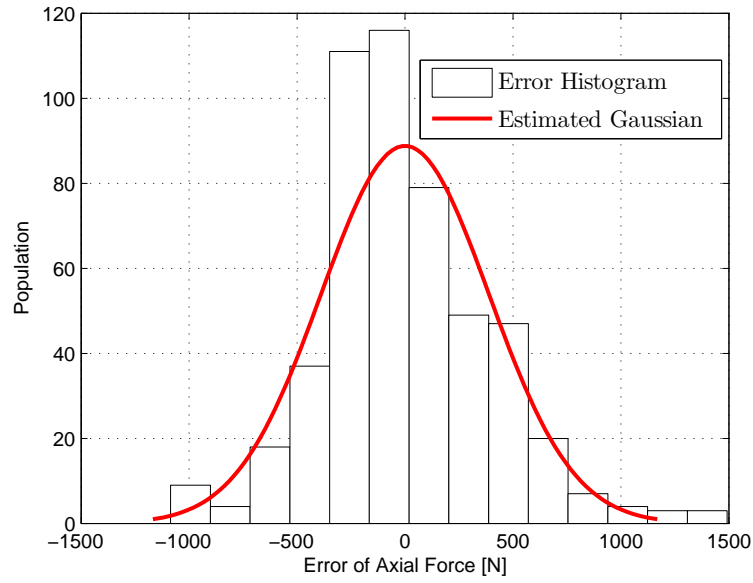


Figure 4.20: Histogram of error and estimated Gaussian for Linear Model 1 with data filtering

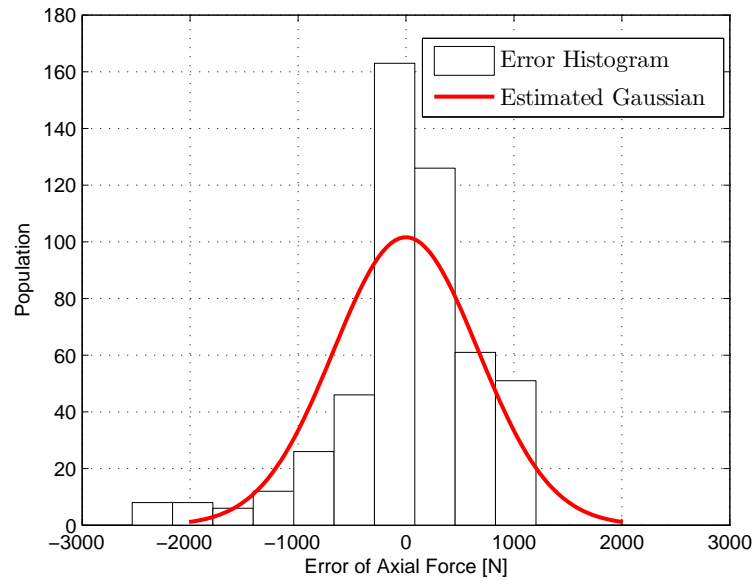


Figure 4.21: Histogram of error and estimated Gaussian for Linear Model 2 with data filtering

## 4.7 Nonlinear Dynamic Modeling and Identification of Axial Force

### 4.7.1 Nonlinear Dynamic Model and Identification Method

According to equation (4.4), a nonlinear dynamic model of the axial force during the FSW process has been proposed by [107] using the plunge depth  $d_p$  as the input.



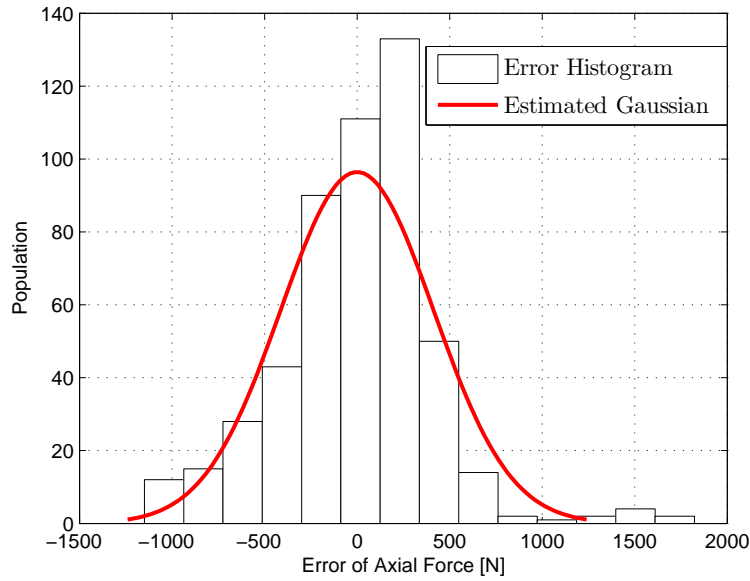


Figure 4.22: Histogram of error and estimated Gaussian for Linear Model 3 with data filtering

The transfer function of the second order dynamic model can be expressed by:

$$F_z(s) = \frac{3.86v^{0.097}\omega^{-0.230}(s + 2.25)}{(s + 25.5)(s + 2.65)}d_p^{2.207}(s) \quad (4.44)$$

where the units of  $F_x$ ,  $F_y$ ,  $F_z$ ,  $d_p$ ,  $v$  and  $\Omega$  are  $kN$ ,  $kN$ ,  $kN$ ,  $mm$ ,  $mm/s$  and  $rpm$ .

Besides, since the elastic and plastic deformations of the material are unevenly distributed in the  $xy$  plane during the FSW process, the axial force  $F_z$  can be considered to be the resultant of all the component forces in  $z$  direction. As a consequence, a nonlinear dynamic model can be defined by the following expression:

$$F_z = A d_p^\alpha \left(1 + \beta \frac{d}{dt} d_p\right) \quad (4.45)$$

where  $A$  denotes the coefficient of the nonlinear power term,  $\beta$  is the coefficient of the derivative of plunge depth (related to the damping), and  $\alpha$  is the power of the plunge depth (dimensionless) for identifying the high frequency phenomena. The units of  $F_z$ ,  $d_p$ ,  $\frac{d}{dt} d_p$ ,  $A$  and  $\beta$  are  $N$ ,  $m$ ,  $m/s$ ,  $N/(mm^\alpha)$  and  $s/mm$  respectively.

In order to use the least squares method for parameter identification, the nonlinear dynamic model can be transformed to a linear form by taking the natural log of equation (4.45) at first, formulated by:

$$\ln(F_z) = \ln A + \alpha \ln(d_p) + \ln\left(1 + \beta \frac{d}{dt} d_p\right) \quad (4.46)$$

In this case, the parameter identification of the nonlinear model can be divided into two steps. First of all, a fixed value should be given to the model parameter  $\beta$ . After the value assignment, the term  $\ln\left(1 + \beta \frac{d}{dt} d_p\right)$  can be calculated, thus, equation (4.46) can be rewritten by:

$$\ln\left(\frac{F_z}{1 + \beta \frac{d}{dt} d_p}\right) = \ln A + \alpha \ln(d_p) \quad (4.47)$$

Obviously, the above equation can be transformed to be the matrix form according to equation (4.31) and the corresponding parameters are given by:

$$y = \begin{bmatrix} \vdots \\ \ln\left(\frac{F_z(i)}{1+\beta\frac{d}{dt}d_p(i)}\right) \\ \vdots \end{bmatrix}; \quad X = \begin{bmatrix} \vdots \\ 1 \quad \ln(d_p(i)) \\ \vdots \end{bmatrix}; \quad \gamma = \begin{bmatrix} \ln A \\ \alpha \end{bmatrix} \quad (4.48)$$

Similarly, the unknown parameters  $A$  and  $\alpha$  can be calculated directly by using equation (4.33), and the corresponding nonlinear axial force model can be obtained as well.

In the second place, by using test and error method for the model parameter  $\beta$ , many approximate solutions for the nonlinear model identification can be obtained so as to find a better one with sufficient precision. The most appropriate model parameter  $\beta$  can be chosen based on the accuracy of the nonlinear axial force model and the model parameters. Thus, the root mean square (RMS) errors, the uncertainties of the identified model parameters and the normality test of errors can be taken into account. Then, the nonlinear dynamic model of the axial force can finally be identified.

As for the error analysis method for the identified nonlinear model, the RMS error can be calculated according to equation (4.35) which is presented for the linear model as well.

As a matter of fact, the method of uncertainty calculation for the linear model is not suitable for the nonlinear model. In order to solve this problem, it is necessary to calculate the partial derivatives of the axial force  $F_z$  with respect to the identified model parameters  $A$ ,  $\alpha$  and  $\beta$ , which can be denoted as the matrix  $J_a$ :

$$J_a = \frac{\partial F_z}{\partial \gamma} = \left[ \frac{\partial F_z}{\partial A} \quad \frac{\partial F_z}{\partial \alpha} \quad \frac{\partial F_z}{\partial \beta} \right] \quad (4.49)$$

For calculating uncertainties of the identified parameters of the nonlinear model, the error of the axial force and the matrix  $J_a$  should be used to respectively replace the vector of identification error  $\varepsilon$  and the measurement matrix  $X$  in equations (4.38) and (4.39), as formulated in the following equations:

$$\sigma^2 = \frac{(F_{zmes} - F_{zfit})^T (F_{zmes} - F_{zfit})}{N - N_\gamma} \quad (4.50)$$

$$Variance(\gamma) = |Diag((J_a^T J_a)^{-1} \sigma^2)| \quad (4.51)$$

Eventually, the uncertainties of the identified parameters of the nonlinear model can be calculated by using equation (4.40).

## 4.7.2 Results and Error Analysis of Identified Nonlinear Model

On the basis of the above identification method for the nonlinear dynamic model, the model parameter  $\beta$  can be selected as 423.1 ( $s/mm$ ). Since the units of  $F_z$ ,  $d_p$ ,

$\frac{d}{dt}d_p$ ,  $A$  are  $\beta$  as  $N$ ,  $m$ ,  $m/s$ ,  $N/(mm^\alpha)$  and  $s/mm$  respectively and the coefficient  $\alpha$  is dimensionless, the resulting nonlinear model of axial force can be calculated by:

$$F_z = 1.781 \times 10^6 d_p^{0.870} (1 + 423.1 \frac{d}{dt}d_p) \quad (4.52)$$

By using equation (4.35), the RMS errors of the nonlinear identified model is computed as 5.35%. All other results of error calculations including the standard deviation, maximum errors and mean errors for the nonlinear identified model are given in Table 4.6. Besides, the value of identified parameters and their uncertainties for the nonlinear model are provided in Table 4.7.

Table 4.6: Errors of identified nonlinear model

| Name                       | Nonlinear Model |
|----------------------------|-----------------|
| standard deviation ( $N$ ) | 465.09          |
| maximum error (%)          | 26.88           |
| mean error (%)             | 4.18            |
| RMS error (%)              | 5.35            |

Table 4.7: Value of identified parameters and their uncertainties for nonlinear model

| Parameters (Nonlinear) | $A$ ( $N/(mm^\alpha)$ ) | $\alpha$ | $\beta$ ( $s/mm$ ) |
|------------------------|-------------------------|----------|--------------------|
| Value                  | $1.781 \times 10^6$     | 0.870    | 423.1              |
| Uncertainty (%)        | 13.83                   | 2.60     | 3.97               |

Figure 4.23 shows the comparison between the filtered and the fitted axial force  $F_z$  as well as the generated error for the identified nonlinear model. In general, the fitted axial forces are in good agreement with the filtered axial forces. Besides, the normality test of the errors is implemented by means of the *Matlab* function *lillietest*, and the histogram of error and the estimated Gaussian for the nonlinear model with data filtering is given in Figure 4.24.

## 4.8 Conclusion

In this chapter, the dynamic modeling and identification of the robotic FSW process have been presented. The axial force  $F_z$  is modeled by using the plunge depth  $d_p$  and its derivative  $\frac{d}{dt}d_p$  in this study. During this operation, the flexibility of the robot needs to be considered since an extremely strong external force is exerted on the robot and a nonnegligible deformation is generated. Hence, the actual tool position is different from the commanded tool position and the plunge depth can not be directly obtained by the robot measurements. The identified dynamic model can be used to calculate the forces by using the robot states without the implementation of real welding operation, which contributes to establishing the simulator for the

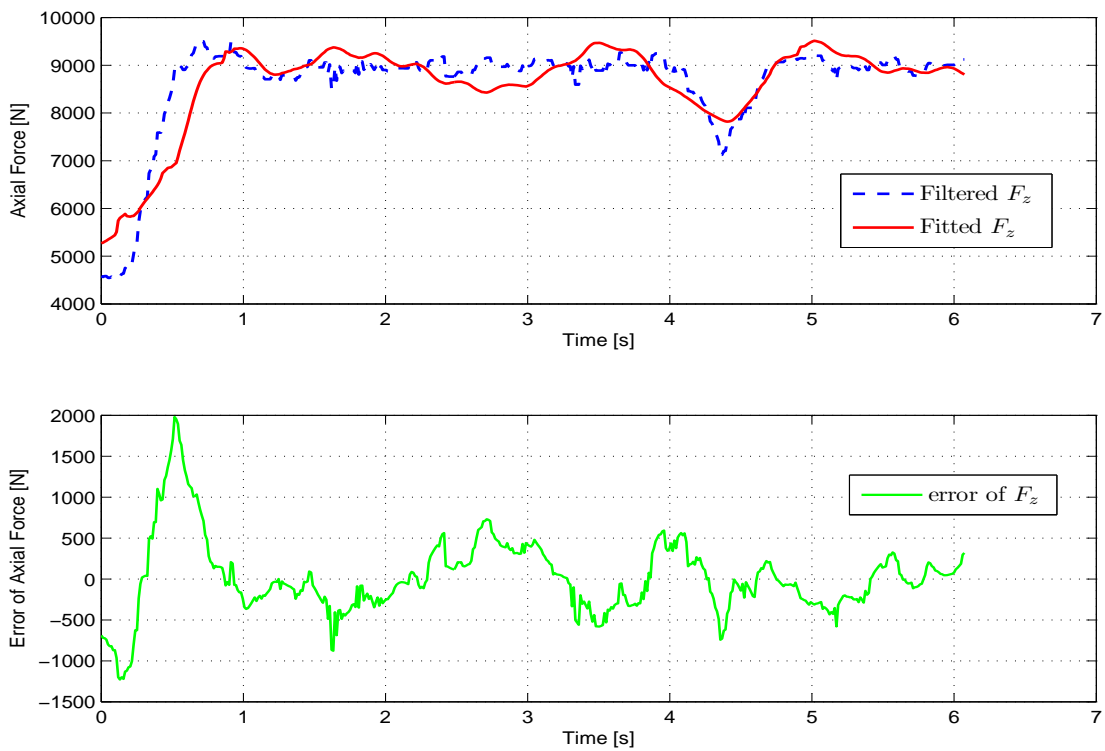


Figure 4.23: Comparison between filtered and fitted axial force  $F_z$  and generated error for the nonlinear model

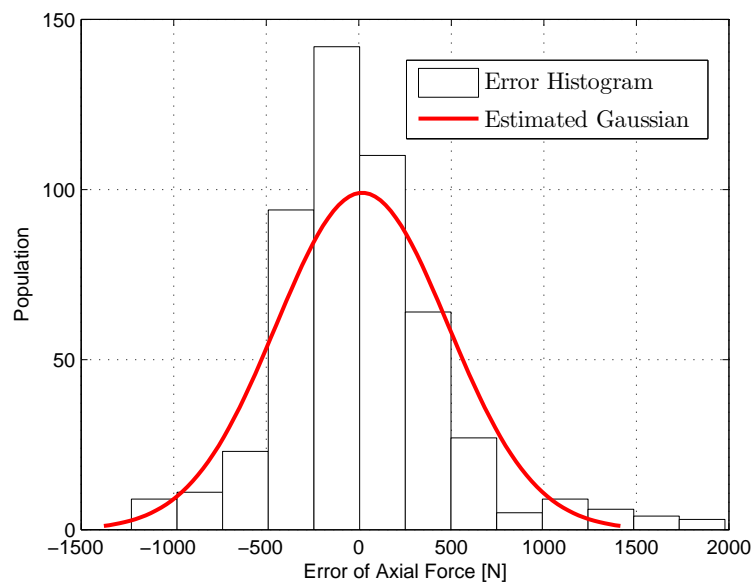


Figure 4.24: Histogram of error and estimated Gaussian for the nonlinear model with data filtering

FSW process. Since the dynamic models of the process forces are greatly required for the design of the control strategies in robotic FSW process, especially for the

design of the robot force controller, the models can be used to establish the control laws for compensating the positioning errors of the robot. Besides, the dynamic models of the process forces can be utilized for the stability analysis of the robot controller. A brief description of the FSW process including its three main stages and process forces is given firstly. Then the two static nonlinear power models of process forces in FSW has been discussed. With the help of the robotic FSW system, the experimental tests for the identification have been implemented. A nonlinear observer presented by [4, 72, 73] is used to estimate the actual joint angular position in the experiments so as to obtain the required plunge depth. By means of data filtering, the proposed linear and nonlinear models are identified based on the least squares method. The systematic studies on the errors of identified models have been carried out and the comparisons between filtered and fitted axial forces are performed. As a result, acceptable accuracy of the identified models can be seen according to the RMS errors, the uncertainties of identified parameters and the normality tests, demonstrating good performance of the identification.

# Chapter 5

## Design of Robust Force Controller in Robotic FSW Process

### 5.1 Introduction

This chapter focuses on the proposition of robust force controllers for industrial robot manipulators and the development of a simulation system of robotic FSW process. On the basis of the identification of rigid robot displacement model and the dynamic axial force process model, the system modeling of robotic FSW process can be accomplished and the design of robust force controllers can be performed by means of a systematic design procedure [95]. Besides, the corresponding simulator is established in order to simulate the robotic FSW process and test the performance of force controllers. In addition, the oscillation of axial force in FSW process is experimentally studied as well, in the meantime, based on the height measurements of the weld surfaces, the related disturbance model of zero force vertical position ( $Z_0(t)$ ) is proposed and adopted in the simulation studies.

This chapter is organized as follows. Section 5.2 presents a literature review on the force control of robot manipulator. Section 5.3 describes the force control strategy of the robot manipulator and the system modeling of the robotic FSW process, including the interaction model of robot and environment. In Section 5.4, the parameter identification of rigid robot displacement model is implemented by using two different algorithms of the System Identification Toolbox provided by the software MATLAB including transfer function estimation and process model estimation. In Section 5.5, based on a systematic design procedure [95], robust force controllers are designed, simultaneously taking the complete system kinematics, the position response of the industrial robot controller and the joint flexibility of robot into consideration. Section 5.6 introduces the establishment of the simulator of robotic FSW process, and simulation studies have been carried out to test the performance of the designed force controllers. In Section 5.7, the axial force vibration during robotic FSW process is firstly analyzed by the identification of an experimental axial force model, and a disturbance model of the vertical position of zero force is proposed and used in the simulator. The disturbance model and the performance of the force controller has been verified by the simulation results. Finally, a conclusion is given in Section 5.8.

## 5.2 Force Control of Robot Manipulator

The control of the physical interaction between the robot manipulator and the environment is crucial in order to successfully implement a variety of tasks in industrial applications where the end-effector of the robot is required to be in contact with the environment, such as pushing, drilling, insertion, deburring, scraping, grinding, polishing, cutting, assembly, etc [7]. The quantity that describes the state of the interaction more effectively is the contact force at the end-effector of the robot [5]. In the past three decades, a lot of theoretical and experimental research have been carried out on the force control strategies of the robot, including several high quality reviews of Whitney [114], Patarinski and Botev [115], Zeng and Hemami [116], De Schutter et al. [117], Chiaverini et al. [118], and Yoshikawa [119]. A monograph contributed to this field was written by Siciliano and Villani [120]. Furthermore, the fundamental knowledge of the robot force control methods can be found in the books of Siciliano et al. [5], Sciciliano and Khatib [7], Craig [62], Khalil and Dombre [63] as well.

According to the book of Siciliano and Villani [120], the force control strategies can be categorized as indirect force control methods and direct force control methods. In the first category, the force control is accomplished by using motion control without explicit closure of a force feedback loop. To this category belong stiffness control, impedance control and admittance control. On the contrary, the contact force can be controlled to a desired value by means of the closure of a force feedback loop in the second category, such as hybrid position/force control, inner position/outer force control, parallel force/position control. In addition, the force control methods can be also classified as basic force control approaches and advanced force control approaches which will be presented below.

### 5.2.1 Basic Force Control Approaches

The basic force control schemes include stiffness control (or compliance control), impedance control (or admittance control), hybrid position/force control, hybrid velocity/force control [121], hybrid impedance control [122], parallel force/position control [123], explicit force control [124, 125], implicit force control [126] and so on.

#### Stiffness/Compliance Control

Under the stiffness control (also referred to be compliance control), the position error of the robot due to the interaction with the environment is related to the contact force through a mechanical stiffness of adjustable parameters. The stiffness control can be considered as a programmable spring since the stiffness of the closed-loop system is alterable through the force feedback [116]. Although the implementation of stiffness control is extremely easy, the stability and accuracy issues may occur since the dynamics of the industrial robot is not considered. In [127], a method of active stiffness control of a manipulator end-effector in Cartesian coordinates was proposed, and the corresponding force control and kinematic issues were discussed in [128]. An external force control scheme based on the stiffness control was proposed and its implementation in the UNIMATE controller of an industrial PUMA 560

robot is presented in [129]. In [130], the stability analysis of the robot manipulator under the stiffness control was investigated and the bound for stable manipulation was derived. Besides, the kinematic instability problem associated with stiffness control was investigated in [131], and the qualitative evaluations of the stability of stiffness control were established.

### **Impedance/Admittance Control**

The original idea of impedance control proposed by Hogan [132, 133] is that the control system should not be designed to regulate the force or position directly, but rather to regulate the mechanical impedance of the robot. Hence, in the impedance control (or admittance control), the deviation of the end-effector position from the desired position is related to the contact force through a mechanical impedance or admittance with adjustable parameters. The industrial robot under this control scheme can be described as an equivalent mass-spring-damper system with adjustable parameters. The relationship is an impedance if the robot control reacts to the position deviation by generating forces, while it corresponds to an admittance if the robot control reacts to interaction forces by imposing a deviation from the desired position [7].

An impedance controller design methodology to develop a robust compliant motion for robot manipulators was described in [134, 135]. Two adaptive schemes for controlling the end-effector impedance of dexterous space manipulators were described in [136], and the impedance controllers can provide accurate force regulation and global stability in the presence of bounded disturbances. In [137], two impedance controllers were simulated numerically and implemented on a direct drive robot manipulator. As for the stability of impedance control, the stability problem of impedance control system of a manipulator during contact tasks was discussed in [138]. Besides, the impedance control stability properties in common implementations were systematically analyzed in [139], and the stability boundaries for the impedance parameters were obtained. In [140], Bigras proposed a new formulation for the design of an impedance controller for the industrial robots. The complete kinematics of the system and the position response of the industrial robot controller was taken into account for the system modeling and good results were obtained for the impedance controller design.

### **Hybrid Position/Force Control**

The concept of hybrid position/force control was firstly developed by Mason [141]. The hybrid position/force control is based on the decomposition of the workspace into purely position controlled directions and purely force controlled directions. The objective of this control scheme is to split up simultaneous control of both end-effector position and contact forces into two separate decoupled subproblems [7].

In [142], Raibert and Craig implemented the hybrid position/force control scheme to make the position and force follow the given desired trajectory simultaneously, and the results showed that the control strategy achieved stable, accurate control of force and position trajectories for a variety of test conditions. A unified approach for



motion and force control of the robot manipulators in the operational space was presented in [143]. In [144], the author proposed a method for designing a motion/force controller for a robot manipulator, leading to decoupled controllers that are capable of ensuring a global stability and are applicable to stationary as well as to mobile contact surfaces. Besides, the problem of simultaneous motion and force control for manipulator tasks with continuous contact between the end-effector and the environment was solved by using the hybrid position/force control [145]. A force/position controller for robot manipulators in contact with an elasticity compliant surface was presented in [146], the asymptotic stability of the system in the neighborhood of the equilibrium state was demonstrated by using the classical Lyapunov method. In [59], the author proposed another stable force/position controller of robot manipulators, which is composed of a PD position loop, a PI force loop, the gravity compensation and the desired contact force feedforward.

### 5.2.2 Advanced Force Control Approaches

The advanced force control schemes are based on adaptive control, robust control and learning methods integrated or combined with the basic force control schemes.

The adaptive force control can be applied in order to maintain consistent performance of a control system when parameters of the robot and the environment are unknown. Based on the existing definitions of the basic force control schemes, an adaptive force controller combined with certain control strategy can be used to maintain the desired stiffness, impedance, force, etc [147, 148, 149, 150].

Besides, the robust force control is aimed at achieving the target dynamics (e.g., target impedance) and at preserving the stability robustness in the presence of the bounded model uncertainties (also referred to as modeling errors) of the robot and the environment. The robust force control mainly include robust hybrid position/force control [151, 152] and robust impedance control [134, 135].

Furthermore, the learning algorithm can be introduced to hybrid position/force control when a robot manipulator implements the same task repeatedly, and the performance of the controlled robotic system can be enhanced significantly. The algorithm utilizes position, velocity and acceleration errors or force error for learning the command input required to perform tasks. It guarantees the convergence of both position and force tracking errors, as well as robustness, for sufficiently small parameter uncertainties and disturbances [153, 154, 155, 156].

## 5.3 Force Control Strategy and System Modeling

### 5.3.1 Force Control Strategy

During the robotic FSW process, the tool of the FSW spindle installed on the robot manipulator is required to be in contact with the workpieces, besides, the contact force should be controlled to guarantee appropriate welding performance. Particularly, the axial force  $F_z$  should be controlled as a desired constant value since it has an important contribution to obtaining good weld quality.

In order to ensure regulation of the contact force to a desired value during the interaction of the industrial robot with the environment, a lot of control strategies can be taken into account. Generally, in most of the control schemes, the measured force errors are directly converted to actuator forces or torques which will be applied at the robot joints [117]. However, these control schemes are difficult to implement on the industrial robots with a closed controller architecture where only the position or velocity of the tool can be controlled. In fact, most commercial manipulators have built-in position controllers and direct access to the motor torques is not allowed. Compared with other control methods, in the inner motion/outer force control, the measured force errors are converted to desired position or velocity, which is executed by a position or velocity control loop. Consequently, most of the force controllers implemented on the industrial robots are based on the cascade structure of outer force control loop closed around inner motion loop [125, 126, 151, 157, 158, 159, 160].

In addition, if the bandwidth of the inner motion control loop is much faster than the bandwidth of the outer force control loop, the force disturbances will be rejected more efficiently by using the inner/outer loop implementation [95]. Since the industrial robot KUKA KR500-2MT considered in this study can be controlled only by using its positional interface, the force control strategy of the robot system can be chosen as inner position/outer force control, as shown in Figure 5.1. Simultaneously, the industrial robot will be position-controlled in  $x$  and  $y$  direction, and the force regulation will be implemented in  $z$  direction in the above control scheme which is also referred as hybrid position/force control.

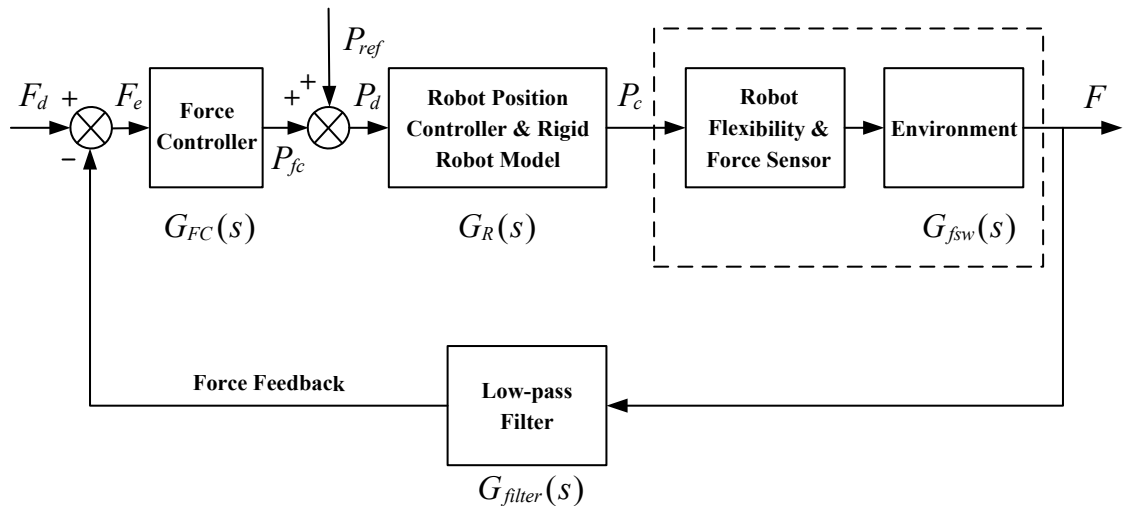


Figure 5.1: Inner position/outer force control scheme for industrial robot KUKA

### 5.3.2 System Modeling in Cartesian Space

As shown in Figure 5.2, the system of the interaction between the robot manipulator and the environment during FSW process can be described by using a simple lumped mass model with mass, stiffness and damping parameters in the Cartesian space.

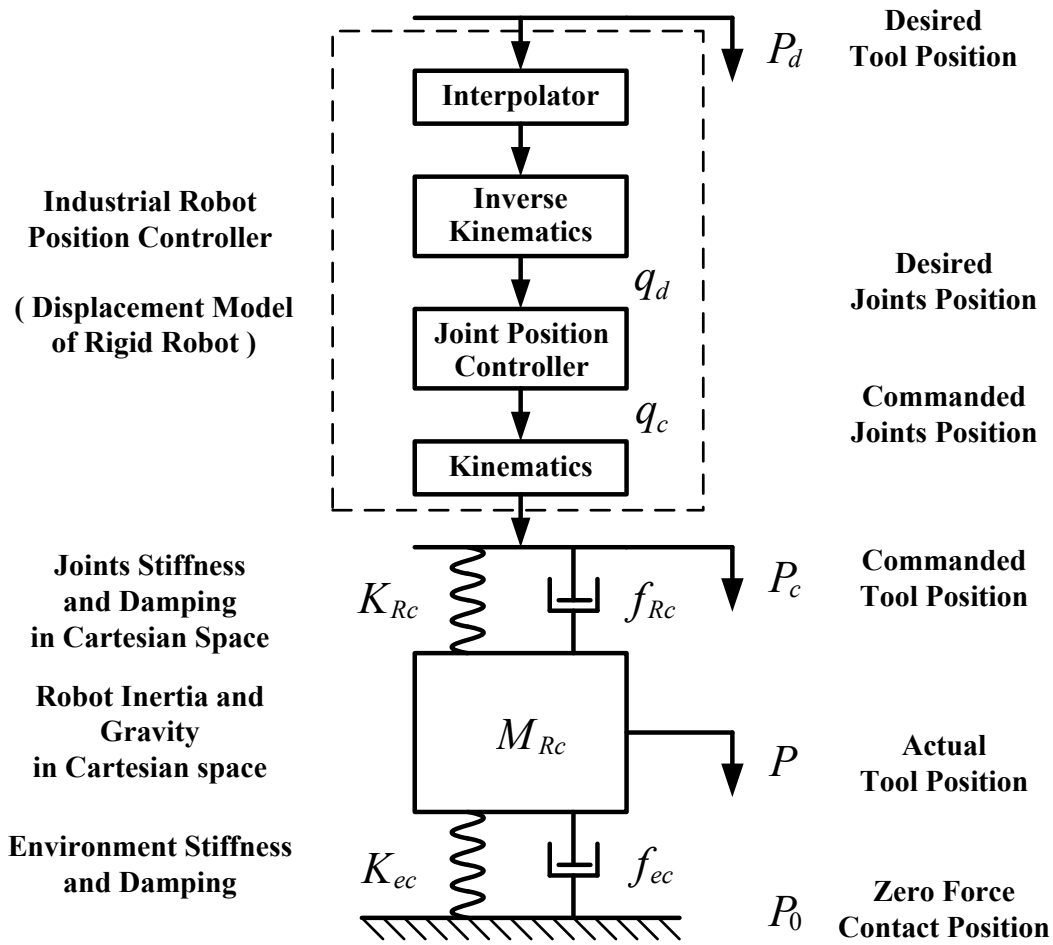


Figure 5.2: System modeling in Cartesian space

In this figure, the industrial robot position controller includes the interpolator, the inverse kinematics used to transform the desired trajectory from the Cartesian space to the joint space, the joint position controller of the industrial robot and the kinematics of the robot.  $P_d$  is the desired tool position which is the input of the industrial robot position controller.  $q_d$  and  $q_c$  are the vector of desired joint position and commanded joints position respectively.  $P_c$  is the commanded tool position which is also the output of the industrial robot position controller.  $P$  is the actual tool position and  $P_0$  is the zero force contact position.  $K_{Rc}$  and  $f_{Rc}$  are the stiffness and damping of robot joints transformed to the Cartesian space.  $M_{Rc}$  represents the robot inertia and gravity in Cartesian space.  $K_{ec}$  and  $f_{ec}$  are the stiffness and damping of the environment in the Cartesian space.

As a consequence, the system model can be divided into two parts: the first part is the displacement model of rigid robot which can transform the desired tool position  $P_d$  to the commanded tool position  $P_c$ , and second part is the robot-environment interaction model which can be used to obtain the actual tool position  $P$  from the commanded tool position  $P_c$ .

Since the axial force  $F_z$  should be controlled as a desired constant value during the FSW process, we will only consider the lumped mass model in  $z$  direction.

Let parameters  $Z_d, Z_c, Z, Z_0, M_{Rz}, K_{Rz}, f_{Rz}, K_e$  and  $f_e$  be  $z$ -direction components of  $P_d, P_c, P, P_0, M_{Rc}, K_{Rc}, f_{Rc}, K_{ec}$  and  $f_{ec}$  respectively. Then, the following equations that govern the physics of the above simple lumped mass model are given by:

$$\begin{aligned} M_{Rz} \frac{d^2}{dt^2} Z(t) + (f_{Rz} + f_e) \frac{d}{dt} Z(t) + (K_{Rz} + K_e) Z(t) \\ = K_{Rz} Z_c(t) + f_{Rz} \frac{d}{dt} Z_c(t) + K_e Z_0(t) + f_e \frac{d}{dt} Z_0(t) \end{aligned} \quad (5.1)$$

$$F_z(t) = K_e [Z(t) - Z_0(t)] + f_e \left[ \frac{d}{dt} Z(t) - \frac{d}{dt} Z_0(t) \right] \quad (5.2)$$

By using equation (5.1) and (5.2) and applying a Laplace transform, the following equations will be yielded:

$$Z(s) = \frac{(K_{Rz} + f_{Rz}s)Z_c(s) + (K_e + f_e s)Z_0(s)}{M_{Rz}s^2 + (f_{Rz} + f_e)s + K_{Rz} + K_e} \quad (5.3)$$

$$F_z(s) = (K_e + f_e s)[Z(s) - Z_0(s)] \quad (5.4)$$

Then,  $Z(s)$  in equation (5.4) can be replaced by equation (5.3). As a result, the transfer function relating the axial force  $F_z$  and the commanded vertical tool position  $Z_c$  and the vertical tool position of zero force  $Z_0$  can be given by:

$$F_z(s) = G_{Pc}(s)Z_c(s) + G_{P0}(s)Z_0(s) \quad (5.5)$$

where

$$G_{Pc}(s) = \frac{(K_e + f_e s)(K_{Rz} + f_{Rz}s)}{M_{Rz}s^2 + (f_{Rz} + f_e)s + K_{Rz} + K_e} \quad (5.6)$$

$$G_{P0}(s) = - \frac{(K_e + f_e s)(M_{Rz}s^2 + f_{Rz}s + K_{Rz})}{M_{Rz}s^2 + (f_{Rz} + f_e)s + K_{Rz} + K_e} \quad (5.7)$$

The above transfer functions are extremely important for the design of robust force controller, especially for  $G_{Pc}(s)$ . In the axial force control system, the vertical tool position of zero force  $Z_0$  can be regarded as the position disturbance in  $z$  direction. The parameters of dynamic axial force model  $K_e$  and  $f_e$  have already been identified in the previous chapter.

According to the research work of Abele [161] and Dumas [162], the lumped mass model parameters  $K_{Rc}, f_{Rc}$  and  $M_{Rc}$  can be calculated by the following equations:

$$K_{Rc}(q) = J(q)^{-T}(q)K_r J(q)^{-1}(q) \quad (5.8)$$

$$f_{Rc}(q) = J(q)^{-T}(q)B_s J(q)^{-1}(q) \quad (5.9)$$

$$M_{Rc}(q) = J(q)^{-T}(q)M(q)J(q)^{-1}(q) \quad (5.10)$$

where  $K_r$  is the  $6 \times 6$  diagonal matrix which includes the equivalent stiffness of six robot joints,  $B_s$  is the  $6 \times 6$  diagonal matrix of the viscous friction coefficients of joint axes,  $M(q)$  denotes the  $6 \times 6$  inertia matrix of the robot in joint space.

It is noteworthy that approximations exist in the above parameter calculations, for example, only the viscous friction is considered in the damping factor calculation and the Coulomb friction is not taken into account.

In addition, the parameters of the lumped mass model is continuously changing during the FSW process due to the different configuration of the robot. The time history of the model parameters during the circular FSW process are shown in Figure 5.3. Since the variations of the model parameters are comparatively small, the parameters  $K_{Rz}$ ,  $f_{Rz}$  and  $M_{Rz}$  can be calculated by using the vector of measured initial angular positions of joint axes ( $q_0$ ), and their  $z$ -direction components can be expressed by:

$$\begin{cases} K_{Rz} = K_{Rc}(q_0)(3) \\ f_{Rz} = f_{Rc}(q_0)(3) \\ M_{Rz} = M_{Rc}(q_0)(3, 3) \end{cases} \quad (5.11)$$

Considering the transfer function  $G_{Pc}(s)$ , it can be transformed as the standard form:

$$G_{Pc}(s) = \frac{K_P \omega_P^2 (1 + \tau_e s)(1 + \tau_{Rz} s)}{s^2 + 2\xi_P \omega_P s + \omega_P^2} \quad (5.12)$$

with

$$\begin{cases} \tau_e = \frac{f_e}{K_e} \\ \tau_{Rz} = \frac{f_{Rz}}{K_{Rz}} \\ K_P = \frac{K_{Rz} K_e}{K_{Rz} + K_e} \\ \omega_P = \sqrt{\frac{K_{Rz} + K_e}{M_{Rz}}} \\ \xi_P = \frac{f_{Rz} + f_e}{2\sqrt{M_{Rz}(K_{Rz} + K_e)}} \end{cases} \quad (5.13)$$

where  $\tau_e$  and  $\tau_{Rz}$  are time constants of the first-order differentiation elements in  $G_{Pc}(s)$ ,  $K_P$  denotes the gain of  $G_{Pc}(s)$ ,  $\omega_P$  is the natural frequency of the second-order oscillatory element in  $G_{Pc}(s)$  and  $\xi_P$  is the corresponding damping ratio. The time history of parameters  $K_P$ ,  $\omega_P$  and  $\xi_P$  are provided in Figure 5.3 as well. As can be seen, all these parameters have small variations during the circular FSW process.

Similarly, if we consider the transfer function  $G_{P0}(s)$ , it can be transformed as the standard form as well:

$$G_{P0}(s) = - \frac{K_P \omega_P^2 (1 + \tau_e s)(s^2 + 2\xi_D \omega_D s + \omega_D^2)}{\omega_D^2 (s^2 + 2\xi_P \omega_P s + \omega_P^2)} \quad (5.14)$$

with

$$\begin{cases} \omega_D = \sqrt{\frac{K_{Rz}}{M_{Rz}}} \\ \xi_D = \frac{f_{Rz}}{2\sqrt{M_{Rz}K_{Rz}}} \end{cases} \quad (5.15)$$

where  $\omega_D$  is the natural frequency of the new second-order oscillatory element in  $G_{P0}(s)$  and  $\xi_D$  is the corresponding damping ratio.

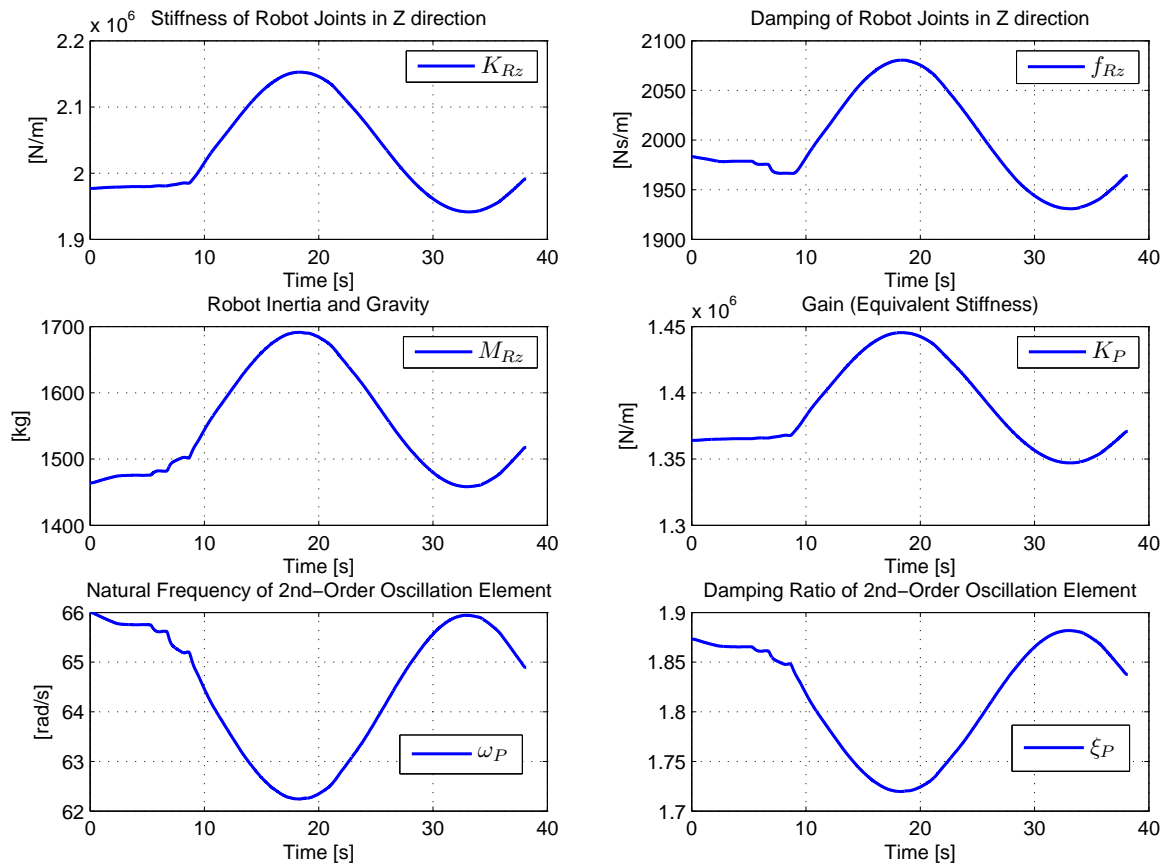


Figure 5.3: Time history of model parameters during circular FSW process (test IS1006P)

## 5.4 Parameter Identification of Displacement Model of Rigid Robot

In order to design the robust force controller for the actual industrial robot KUKA, the parameter identification of the displacement model of the rigid robot should be performed at first.

As for the model identification, the experimental tests have been carried out without contacting the workpieces, that is to say, the industrial robot can be regarded

as rigid in the tests. Accordingly, the identified model will be the displacement model of the rigid robot. The input data used for this identification, also referred to be the desired tool displacement ( $\Delta P_d$ ), is a 2 mm step input in  $z$ ,  $y$  and  $x$  direction respectively, which is performed by sending the position correction from an external computer to robot KUKA every 12 ms. The output data, also called the commanded tool displacement ( $\Delta P_c$ ), is the 2 mm step response in the above three directions which can be calculated by using the direct geometric model (DGM) and the measured motor angular position after the gear reduction ( $\theta_l$ ), expressed by:

$$\Delta P_c = DGM(\theta_l) \quad (5.16)$$

In this section, the parameter identification of the rigid robot displacement model will be implemented by using two different algorithms of the System Identification Toolbox provided by the software MATLAB : transfer function estimation and process model estimation.

### 5.4.1 Identification using Transfer Function Estimation

On the basis of the system modeling in Figure 5.2, the commanded tool displacement  $\Delta P_c$  is the response of the industrial robot position controller to the desired tool displacement  $\Delta P_d$ . In [125], a continuous second order linear model was proposed by supposing that the industrial robot controller is a proportional derivative controller. Besides, a time delay in the model can be considered as well [151].

According to the research of Bigras [140], a third-order transfer function model that constitutes a relatively good approximation of the experimental data for industrial robot KUKA KR270 is obtained by:

$$\frac{B(s)}{A(s)} = \frac{0.3070s^2 + 14.42s + 4495}{s^3 + 32.82s^2 + 594.4s + 4502} \quad (5.17)$$

where the units of  $B$ ,  $A$  and  $s$  are  $m$ ,  $m$  and  $rad/s$  respectively.

Similarly, as for the position controller of industrial robot KUKA KR500-2MT, the displacement can be modeled by a general  $n^{th}$ -order linear continuous system in the  $s$  domain, formulated by:

$$\frac{\Delta P_c(s)}{\Delta P_d(s)} = \frac{b_{n-1}s^{n-1} + b_{n-2}s^{n-2} + \dots + b_0}{s^n + a_{n-1}s^{n-1} + \dots + a_0} \quad (5.18)$$

where  $b_i$  and  $a_i$  are the coefficients of the numerator and denominator of the transfer function respectively, and  $n$  denotes the order of the transfer function.

Then, the parameter identification of the rigid robot displacement model is implemented by using the transfer function estimation algorithm of the toolbox. By defining the units of  $\Delta Z_c$ ,  $\Delta Y_c$ ,  $\Delta X_c$ ,  $\Delta Z_d$ ,  $\Delta Y_d$ ,  $\Delta X_d$  as  $m$  and  $s$  as  $rad/s$ , the best third-order transfer function models in  $z$  direction,  $y$  direction and  $x$  direction are:

$$\frac{\Delta Z_c(s)}{\Delta Z_d(s)} = \frac{1.389s^2 + 26.28s + 2800}{s^3 + 27.27s^2 + 430.5s + 2776} \quad (5.19)$$

$$\frac{\Delta Y_c(s)}{\Delta Y_d(s)} = \frac{2.301s^2 + 28.1s + 3002}{s^3 + 26.73s^2 + 453.7s + 2977} \quad (5.20)$$

$$\frac{\Delta X_c(s)}{\Delta X_d(s)} = \frac{1.771s^2 + 19.19s + 2919}{s^3 + 27.31s^2 + 446.2s + 2890} \quad (5.21)$$

Furthermore, the uncertainties of the model parameters, the fit values and the RMS errors of the identified models in three directions can be calculated in Table 5.1. Based on the algorithm provided by the toolbox, the fit value is defined by:

$$Fit = 100 \times \left( 1 - \frac{|y_m - y_s|}{|y_m - \text{mean}(y_m)|} \right) \quad (5.22)$$

where  $y_m$  is the measured output data for the validation,  $y_s$  is the simulated output data obtained by using the identified model to the input data. The fit value is calculated in order to find the best model which can minimize the sum of the squares of the difference between the measured output data for the validation and the simulated output data by using the identified model.

In addition, the response to the transfer functions in three directions as well as the comparison with the experimental data set are provided in Figures 5.4, 5.5 and 5.6. The bode diagram of three transfer function models can be found in Figure 5.7. Although the results show good accuracies of identified models based on fit values and RMS errors, the uncertainties of model parameters are not satisfactory. Thus, the transfer function models can not be used in the design of force controller, and a new identification of the rigid robot displacement model will be implemented.

Table 5.1: Parameter uncertainties, fit values and RMS errors of the identified transfer function models in three directions expressed in percentage (%)

| Dir. | $U(b_2)$ | $U(b_1)$ | $U(b_0)$ | $U(a_2)$ | $U(a_1)$ | $U(a_0)$ | $Fit$ | $ERMS$ |
|------|----------|----------|----------|----------|----------|----------|-------|--------|
| Z    | 114.86   | 144.09   | 36.86    | 20.02    | 26.19    | 37.21    | 96.76 | 0.40   |
| Y    | 80.53    | 130.02   | 32.87    | 16.59    | 23.25    | 33.18    | 96.15 | 0.57   |
| X    | 90.79    | 201.34   | 35.75    | 19.93    | 26.08    | 36.10    | 96.73 | 0.48   |

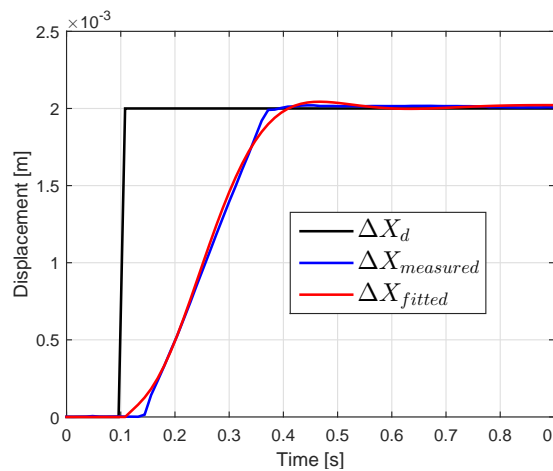


Figure 5.4: Measured and fitted displacements in  $x$  direction

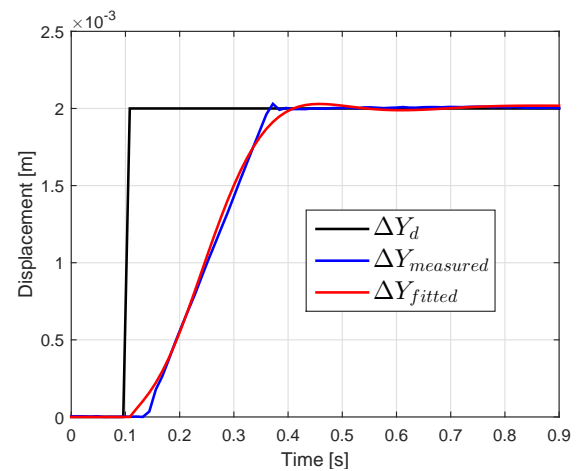


Figure 5.5: Measured and fitted displacements in  $y$  direction



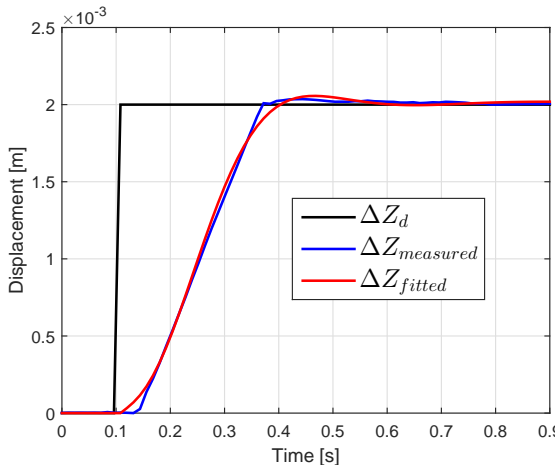
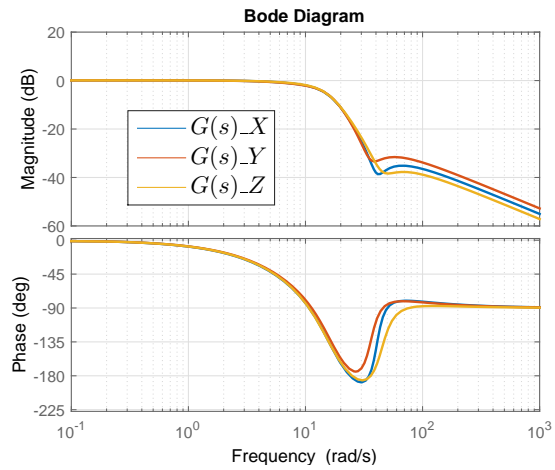

 Figure 5.6: Measured and fitted displacements in  $z$  direction


Figure 5.7: Bode diagram of three third-order transfer function models

### 5.4.2 Identification using Process Model Estimation

On the other hand, the parameter identification of the rigid robot displacement model can be implemented by using the process model estimation algorithm of the System Identification Toolbox as well. Based on [95], the relationship between the desired tool position and the commanded tool position can be approximated by the first-order process model:

$$G(s) = \frac{1}{1 + sT_R} e^{-sT_D} \quad (5.23)$$

where  $T_R$  denotes the time constant,  $T_D$  is the time delay due to the signal processing of the cascaded controller.

However, the rigid robot displacement model in this study will be approximated by a second-order or third-order process model, formulated by:

$$G(s) = \frac{K_{dis}}{1 + 2\xi_R T_R s + T_R^2 s^2} \quad (5.24)$$

$$G(s) = \frac{K_{dis}}{(1 + 2\xi_R T_R s + T_R^2 s^2)(1 + T_{P3}s)} \quad (5.25)$$

Then, by using the same experimental data set in the previous section, the parameters of the second-order process models in  $z$  direction,  $y$  direction and  $x$  direction can be identified in Table 5.2.

As for the accuracy of the identified model, the uncertainties of the model parameters, the fit values and the RMS errors of the identified second-order process models in three directions can be calculated, as shown in Table 5.3.

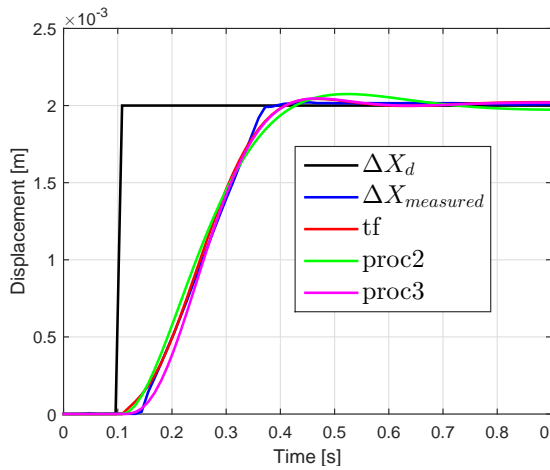
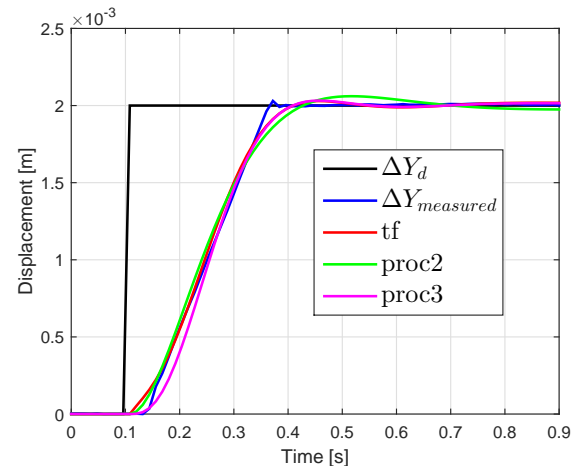
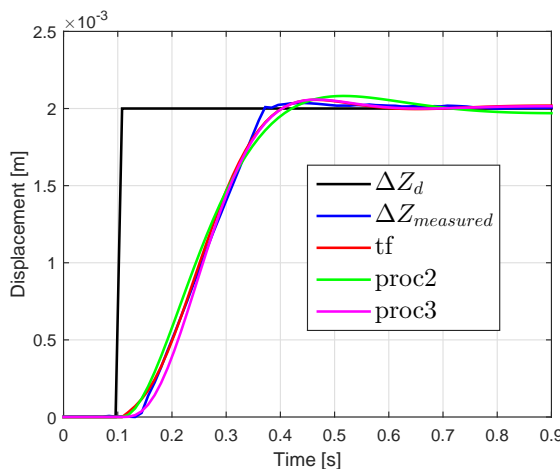
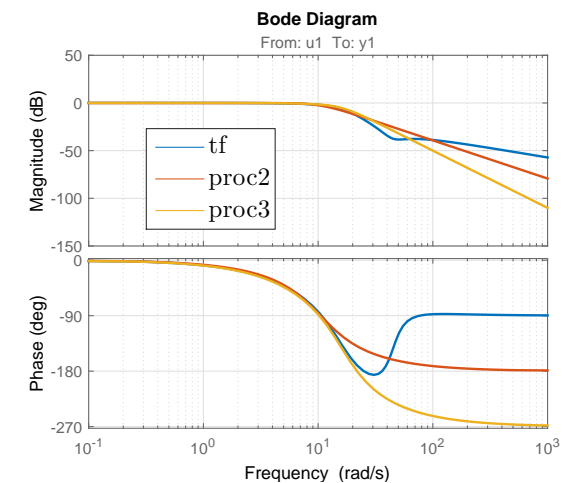
Finally, the response to the transfer function models, the second-order process models and the third-order process models in three directions as well as the comparison with the experimental data set are illustrated in Figures 5.8, 5.9 and 5.10. The corresponding bode diagram of the three kinds of models in  $z$  direction are given in Figure 5.11. The above results demonstrate an excellent performance of the model identification, besides, the identified process models are of importance in the design of a robust force controller which will be presented in the next section.

Table 5.2: Identified parameters of second-order process models in three directions

| Direction | $K_{dis}$ | $T_R$  | $\xi_R$ |
|-----------|-----------|--------|---------|
| $Z$       | 0.9876    | 0.0952 | 0.6811  |
| $Y$       | 0.9890    | 0.0916 | 0.7109  |
| $X$       | 0.9888    | 0.0954 | 0.6921  |

Table 5.3: Parameter uncertainties, fit values and RMS errors of the identified second-order process models in three directions expressed in percentage (%)

| Direction | $U(K_{dis})$ | $U(T_R)$ | $U(\xi_R)$ | $Fit$ | $ERMS$ |
|-----------|--------------|----------|------------|-------|--------|
| $Z$       | 0.47         | 1.63     | 2.42       | 94.24 | 0.81   |
| $Y$       | 0.47         | 1.79     | 2.57       | 94.05 | 0.63   |
| $X$       | 0.48         | 1.70     | 2.52       | 94.11 | 0.82   |


 Figure 5.8: Measured and fitted displacements in  $x$  direction

 Figure 5.9: Measured and fitted displacements in  $y$  direction

 Figure 5.10: Measured and fitted displacements in  $z$  direction

 Figure 5.11: Bode diagram of three kinds of models in  $z$  direction

## 5.5 Design of Robust Force Controller

After the parameter identification of the rigid robot displacement model, the design of the robust force controller can be performed by three main steps, which are based on the systematic design procedure of force controllers for industrial robots presented in [95]. In the design of force controller, the complete system kinematics, the position response of the industrial robot controller and the joint flexibility of the robot are taken into account. The resulting force controller can guarantee high gain and sufficient phase margins for the closed-loop system, which imply the robustness of the force control system.

As for the design procedure, the first step is to specify the desired performance of the force controller, including the rapidness and the robustness of the control system. In the successive two steps, the structure of the force controller will be designed and the corresponding controller parameters can be calculated.

### 5.5.1 Description of Entire Control System

According to the inner position/outer force control scheme illustrated in Figure 5.1 and equation (5.5) to (5.7), a block diagram of the axial force control system is sketched in Figure 5.12.

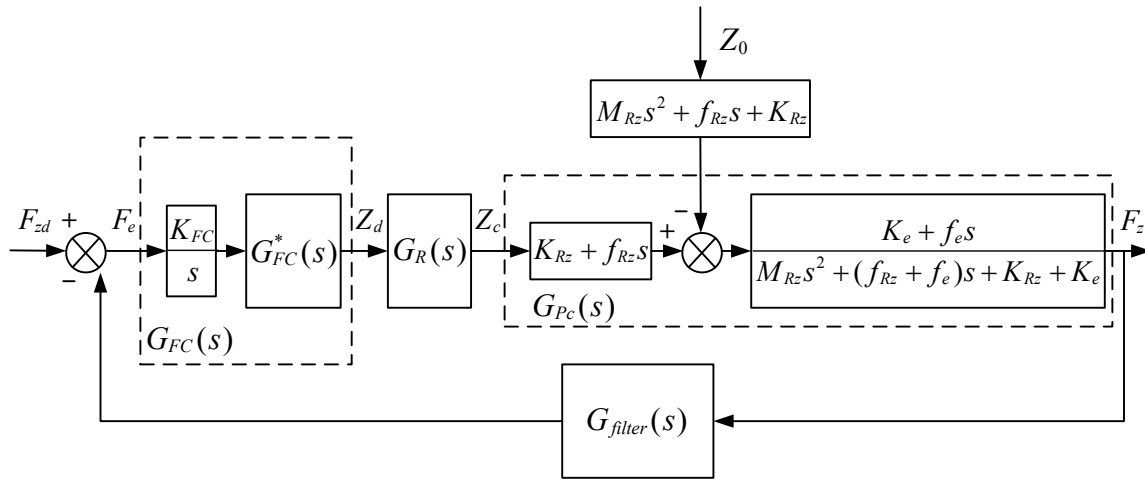


Figure 5.12: Block diagram of the axial force control system

As mentioned in the previous section, the relationship between the desired vertical tool position  $Z_d$  and the commanded vertical tool position  $Z_c$  can be approximately expressed by the following second-order process model:

$$G_R(s) = \frac{K_{dis}}{1 + 2\xi_R T_R s + T_R^2 s^2} \quad (5.26)$$

where  $K_{dis}$  denotes the gain of  $G_R(s)$ , and  $\omega_R = 1/T_R$  is the natural frequency of the second-order oscillatory system,  $\xi_R$  is the corresponding damping ratio.

By considering the vertical tool position of zero force  $Z_0$  as the position disturbance in  $z$  direction, the transfer function relating the axial force  $F_z$  and the

commanded vertical tool position  $Z_c$  and the position disturbance  $Z_0$  can be provided based on equations (5.5) to (5.7):

$$F_z(s) = G_{Pc}(s)[Z_c(s) - \frac{(M_{Rz}s^2 + f_{Rz}s + K_{Rz})}{K_{Rz} + f_{Rz}s}Z_0(s)] \quad (5.27)$$

Obviously, according to the above equation, the transfer function  $G_{Pc}(s)$  in the closed-loop system is considered for the design of the force controller. Based on equation (5.12), the transfer function  $G_{Pc}(s)$  can be expressed as the following standard form:

$$G_{Pc}(s) = \frac{K_P(1 + \tau_e s)(1 + \tau_{Rz}s)}{1 + 2\xi_P T_P s + T_P^2 s^2} \quad (5.28)$$

where  $\tau_e$  and  $\tau_{Rz}$  are time constants of two first-order differentiation elements,  $K_P$  denotes the gain of  $G_{Pc}(s)$ , representing the equivalent stiffness of the robot-environment interaction model,  $\omega_P = 1/T_P$  is the natural frequency of the second-order oscillatory element and  $\xi_P$  is the corresponding damping ratio. All the above parameters can be calculated by equation (5.13).

The block on the feedback branch to be described is the low-pass filter which can handle the force measurements. The existence of the filter depends on the signal-to-noise ratio of the force sensor, influencing the phase margin of the control system. The transfer function of the filter is formulated by:

$$G_{filter}(s) = \frac{1}{1 + sT_F} \quad (5.29)$$

with  $T_F$  being the time constant of the filter.

As shown in the left dashed block of Figure 5.12, the transfer function of the force controller to be designed can be given by:

$$G_{FC}(s) = \frac{K_{FC}}{s} G_{FC}^*(s) \quad (5.30)$$

where  $K_{FC}$  denotes the static gain of the force controller, the pole in the origin ( $1/s$ ) corresponds to the integral action on the force error, and  $G_{FC}^*(s)$  is the transfer function of rest part of the controller with unitary static gain.

In order to design the force controller, it is extremely useful to provide the transfer function of the open-loop system, which is expressed by:

$$G_{open}(s) = \frac{K_{FC}K_PK_{dis}(1 + \tau_e s)(1 + \tau_{Rz}s)}{s(1 + sT_F)(1 + 2\xi_R T_R s + T_R^2 s^2)(1 + 2\xi_P T_P s + T_P^2 s^2)} G_{FC}^*(s) \quad (5.31)$$

### 5.5.2 Specifying Desired Performance of Controller

The first step of the design procedure is to specify the desired performance of the force controller, including the rapidness and the robustness of the control system.

First of all, the primary parameter characterizing the rapidness of the control system when responding to a reference input is the rise time  $t_r$ . In order to guarantee the stability of the control system, the values of the rise time for different performance types during the robotic FSW process can be defined according to the

Table 5.4: The performance type and corresponding rise time as well as the surface type and corresponding desired phase margin

| Performance type | Rise time ( $t_r$ ) | Surface type | Phase margin ( $m_\varphi$ ) |
|------------------|---------------------|--------------|------------------------------|
| Slow             | $7T_R$              | Soft         | $55^\circ$                   |
| Medium fast      | $5T_R$              | Medium stiff | $60^\circ$                   |
| Fast             | $3T_R$              | Stiff        | $65^\circ$                   |

research work of Natale [95], as shown in Table 5.4. As a matter of fact, if the outer force loop is required to be faster than the inner position loop, the system can not be stable.

In addition to the rise time  $t_r$ , the phase margin  $m_\varphi$  for different surface types are provided in Table 5.4. The stiffer the environment is, the bigger the phase margin should be used for the design of the force controller. For instance, by choosing the *Fast* performance type and *Soft* surface type in the table, the desired rise time  $t_r$  can be obtained as  $3T_R$  and the desired phase margin  $m_\varphi$  is chosen as  $55^\circ$ , which will be used in the following controller design procedures.

### 5.5.3 Designing Structure of Controller

In order to obtain a suitable force controller which can be used to achieve the desired performance including the desired rise time and phase margin, an appropriate structure of the force controller should be designed. Moreover, both of the steady-state performance and the transient performance should be taken into account.

Firstly, the steady-state response is considered. The desired steady-state performance can be specified so that the static gain of the force controller can be determined. It should be noted that the steady-state error is nonzero and finite only for the response to a ramp input since a pole in the origin exists in the forward path of the control system. Thus, the static gain  $K_{FC}$  can be obtained by:

$$K_{FC} = \frac{1}{K_P K_{dis} e_{ss}} \quad (5.32)$$

where  $e_{ss}$  is the desired steady-state error for the response to a unitary ramp input. The value of  $e_{ss}$  is expressed in percent of the input magnitude and can be chosen as 5% here. As a result, only  $G_{FC}^*(s)$  needs to be designed.

Secondly, the transient response will be taken into consideration. The desired gain-crossover frequency can be calculated by using the desired rise time and the following approximate relationship [163]:

$$\omega_g \simeq 0.3 \frac{2\pi}{t_r} \quad (5.33)$$

Accordingly, the gain and phase shifts which the rest part of the force controller (expressed as  $G_{FC}^*(s)$ ) needs to provide to the system can be calculated in order to guarantee that the desired controller performance represented by the desired gain-crossover frequency and phase margin can be fulfilled.

As a consequence, the natural value of the gain shift which  $G_{FC}^*(s)$  needs to provide can be calculated by:

$$\Delta_{gn} = \left( \frac{K_{FC}K_P K_{dis} \sqrt{1 + \tau_e^2 \omega_g^2} \sqrt{1 + \tau_{Rz}^2 \omega_g^2}}{\omega_g \sqrt{1 + \omega_g^2 T_F^2} \sqrt{(1 - \omega_g^2 T_R^2)^2 + 4\xi_R^2 \omega_g^2 T_R^2} \sqrt{(1 - \omega_g^2 T_P^2)^2 + 4\xi_P^2 \omega_g^2 T_P^2}} \right)^{-1} \quad (5.34)$$

and its value in decibels can be given by:

$$\Delta_g = 20 \log_{10} \Delta_{gn} \quad (5.35)$$

Meanwhile, the phase shift which  $G_{FC}^*(s)$  needs to provide can be obtained by:

$$\begin{aligned} \Delta_\varphi = & (-180^\circ + m_\varphi) + 90^\circ - \frac{180^\circ}{\pi} [\arctan 2(-\omega_g T_F, 1) \\ & + \arctan 2(-2\xi_R \omega_g T_R, 1 - \omega_g^2 T_R^2) + \arctan 2(\omega_g \tau_{Rz}, 1) \\ & + \arctan 2(-2\xi_P \omega_g T_P, 1 - \omega_g^2 T_P^2) + \arctan 2(\omega_g \tau_e, 1)] \end{aligned} \quad (5.36)$$

where the contributions of the definition of the phase margin, the pole in the origin, the transfer function of the low-pass filter  $G_{filter}(s)$  and the transfer function of the rigid robot displacement model  $G_R(s)$  correspond to the first four terms in the above equation respectively. Besides, the transfer function  $G_{Pc}(s)$  corresponds to the last three terms. In some cases, the delay of the digital implementation of the controller needs to be considered for calculating the phase shift as well.

According to the signs of the gain shift expressed in decibels ( $\Delta_g$ ) and the phase shift ( $\Delta_\varphi$ ), four cases may occur and three kinds of controller structures can be applied in the corresponding cases. The Nichols chart providing the magnitude-phase plot of the open-loop transfer function (excluding the rest part of force controller to be designed  $G_{FC}^*(s)$ ) in the Nichols plane can be considered as a useful tool to understand the design method, as shown in Figure 5.13.

**Case 1:**  $\Delta_g < 0$ ,  $\Delta_\varphi < 0$

The first case when  $\Delta_g < 0$  and  $\Delta_\varphi < 0$  means that the controller needs to attenuate the gain and provide a negative phase shift at the frequency  $\omega_g$ . Thus, a phase-lag structure of the controller can be applied:

$$G_{FC}^*(s) = \frac{1 + sT_1}{1 + sT_2} \quad (5.37)$$

with  $T_1 < T_2$ .

**Case 2:**  $\Delta_g > 0$ ,  $\Delta_\varphi > 0$

The second case when  $\Delta_g > 0$  and  $\Delta_\varphi > 0$  indicates that the controller needs to provide an amplifying action on the gain and a positive phase shift at the frequency  $\omega_g$ . Hence, a phase-lead structure of the controller can be used:

$$G_{FC}^*(s) = \frac{1 + sT_1}{1 + sT_2} \quad (5.38)$$

with  $T_1 > T_2$ .

**Case 3:**  $\Delta_g < 0$ ,  $\Delta_\varphi > 0$ 

The third case when  $\Delta_g < 0$  and  $\Delta_\varphi > 0$  implies that the controller has to attenuate the gain and provide a positive phase shift at the frequency  $\omega_g$ , that is to say, both of the phase lead action and the phase lag action need to be applied. Therefore, the phase-lead-lag structure of the controller can be adopted:

$$G_{FC}^*(s) = \frac{1 + sT_1}{1 + sT_2} \frac{1 + sT_3}{1 + sT_4} \quad (5.39)$$

with  $T_1 < T_2$  in phase-lag stage and  $T_3 > T_4$  in phase-lead stage.

**Case 4:**  $\Delta_g > 0$ ,  $\Delta_\varphi < 0$ 

The last case when  $\Delta_g > 0$  and  $\Delta_\varphi < 0$  will not be taken into consideration since this case is extremely rare.

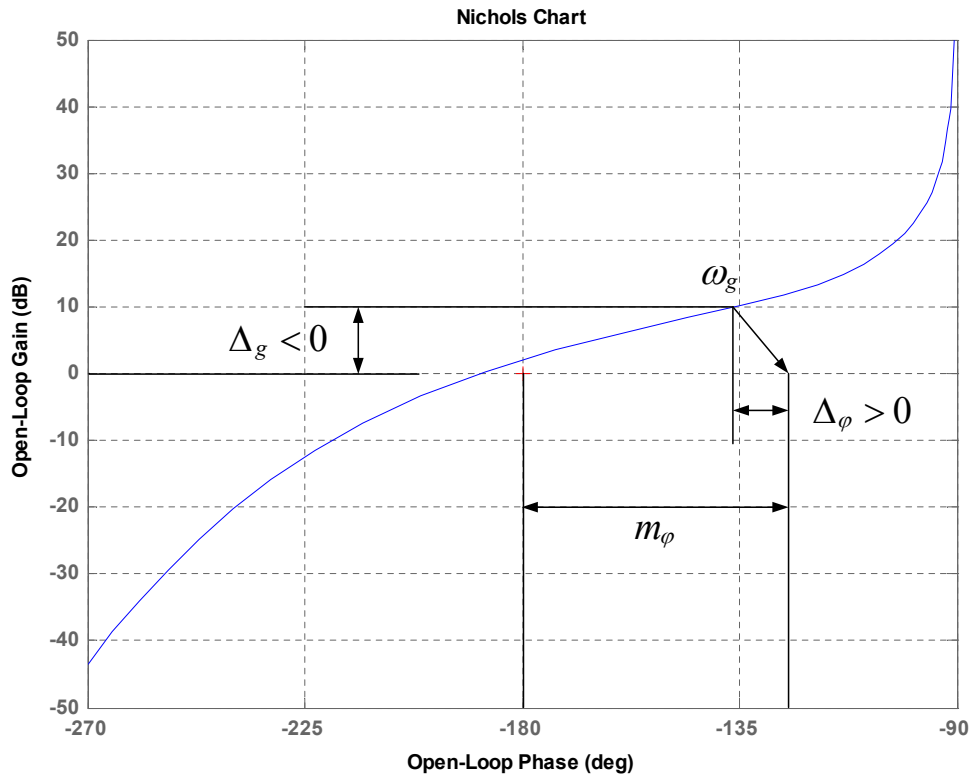


Figure 5.13: Nichols chart of the the open-loop transfer function excluding the rest part of force controller to be designed  $G_{FC}^*(s)$  (Phase-lead-lag case)

### 5.5.4 Calculating Parameters of the Controller

After choosing the appropriate structure of the controller, the values of corresponding parameters should be calculated in order to ensure the desired controller performance. According to the controller structure, the methods of computing controller

parameters can be divided into two categories. The former can be used for the controller with single-stage structure, such as the phase-lag type controller and the phase-lead type controller. The latter can be performed to obtain parameters of the controller with two-stage structure, like the phase-lead-lag type controller.

### 1) Calculation Method for Single-Stage Controller

As for the phase-lag controller or the phase-lead controller, the transfer function  $G_{FC}^*(s)$  at the frequency  $\omega_g$  can be given by:

$$G_{FC}^*(j\omega_g) = \frac{1 + j\omega_g T_1}{1 + j\omega_g T_2} \quad (5.40)$$

Since the gain shift  $\Delta_{gn}$  and phase shift  $\Delta_\varphi$  (at the frequency  $\omega_g$ ) which the transfer function  $G_{FC}^*(s)$  needs to provide to the system have already been calculated, the two unknown controller parameters  $T_1$  and  $T_2$  can be obtained by using the following two equations:

$$\left| \frac{1 + j\omega_g T_1}{1 + j\omega_g T_2} \right| = \Delta_{gn} \quad (5.41)$$

$$\angle \frac{1 + j\omega_g T_1}{1 + j\omega_g T_2} = \Delta_\varphi \quad (5.42)$$

By using the substitutions  $a = T_1/T_2$  and  $x = \omega_g T_2$ , the above two equations can be transformed to the following form:

$$\frac{1 + a^2 x^2}{1 + x^2} = \Delta_{gn}^2 \quad (5.43)$$

$$\frac{ax - x}{1 + ax^2} = \tan \Delta_\varphi \quad (5.44)$$

Then, the parameters  $a$  and  $x$  can be solved by:

$$x = \frac{1 - \Delta_{gn}^{-1} \sqrt{1 + \tan^2 \Delta_\varphi}}{\tan \Delta_\varphi} \quad (5.45)$$

$$a = \frac{x + \tan \Delta_\varphi}{x - x^2 \tan \Delta_\varphi} \quad (5.46)$$

Thus, the final solution for the two controller parameters can be obtained by:

$$T_2 = x/\omega_g \quad (5.47)$$

$$T_1 = aT_2 \quad (5.48)$$

In addition, the detailed computation procedures can be found in Appendix C.1. It should be noted that the hypothesis of a minimum-phase and stable controller should be presented and only the solutions with positive parameters  $x$  and  $a$  can be accepted, which is further discussed in Appendix C.1 as well.



## 2) Calculation Method for Two-Stage Controller

With regard to the two-stage controller with the phase-lead-lag structure, the controller design can be divided into two steps: the phase-lag stage design and the phase-lead stage design. The above two single-stage controllers can be obtained by introducing the gain offset  $O_g$  and the phase offset  $O_\varphi$  ( $O_g > 0, O_\varphi > 0$ ).

In the first step, the transfer function of the phase-lag stage is given by:

$$G_{plag}(s) = \frac{1 + sT_1}{1 + sT_2} \quad (T_1 < T_2) \quad (5.49)$$

and the stage can be designed based on the corrected gain and phase shift:

$$\Delta_{g1} = \Delta_g - O_g \quad (5.50)$$

$$\Delta_{\varphi1} = -O_\varphi \quad (5.51)$$

where the gain offset  $O_g$  and phase offset  $O_\varphi$  are used to make  $\Delta_{g1} < 0$  and  $\Delta_{\varphi1} < 0$ , which can satisfy the condition of designing the phase-lag stage controller. Meanwhile, the design of the second stage is simplified as well.

In the second step, the transfer function of the phase-lead stage is provided by:

$$G_{plead}(s) = \frac{1 + sT_3}{1 + sT_4} \quad (T_3 > T_4) \quad (5.52)$$

Thus, the stage can be designed based on the corrected gain and phase shift:

$$\Delta_{g2} = O_g \quad (5.53)$$

$$\Delta_{\varphi2} = \Delta_\varphi + O_\varphi \quad (5.54)$$

Obviously,  $\Delta_{g2} > 0$  and  $\Delta_{\varphi2} > 0$ , and the negative phase shift introduced by the first stage is recovered. The required phase shift and gain shift are provided by the two stages. The phase-lead action is quite similar to a derivative action and it can not only increase damping but also amplify the noise at high frequency. Hence, the gain offset  $O_g$  and the phase offset  $O_\varphi$  can be selected as a trade-off between transient performance and noise amplification [95].

### 5.5.5 Results of Force Controller Design

As mentioned above, by choosing the *Fast* performance type and *Soft* surface type in Table 5.4, the desired rise time  $t_r$  and phase margin  $m_\varphi$  can be chosen as  $3T_R$  and  $55^\circ$  respectively. As a result, the above design procedure leads to a phase-lead-lag force controller and the corresponding parameters in  $z$  direction of the Cartesian space are  $K_{FC} = 1.446 \times 10^{-5} \text{ m}/(Ns)$ ,  $T_1 = 0.2317 \text{ sec.}$ ,  $T_2 = 2.7191 \text{ sec.}$ ,  $T_3 = 0.5519 \text{ sec.}$ , and  $T_4 = 0.0989 \text{ sec.}$ . The transfer function of the force controller is:

$$G_{fast}(s) = \frac{1.446 \times 10^{-5}(1 + 0.2317s)(1 + 0.5519s)}{s(1 + 2.7191s)(1 + 0.0989s)} \quad (5.55)$$

where the units of the transfer function  $G$ , the coefficient  $K_{FC}$ , the operator  $s$  and the time constants are  $m/N$ ,  $m/(Ns)$ ,  $Hz$  and  $sec.$  respectively.

For comparison, the surface type can be kept as *Soft* and the performance type is changed to *Medium* and *Slow* respectively in Table 5.4, then, two phase-lag force controllers can be obtained as follows:

$$G_{med}(s) = \frac{1.446 \times 10^{-5}(1 + 1.1898s)}{s(1 + 6.3242s)} \quad (5.56)$$

$$G_{slow}(s) = \frac{1.446 \times 10^{-5}(1 + 0.9537s)}{s(1 + 7.3178s)} \quad (5.57)$$

where the units of the transfer function  $G$ , the coefficient  $K_{FC}$ , the operator  $s$  and the time constants are  $m/N$ ,  $m/(Ns)$ ,  $Hz$  and  $sec.$  respectively.

In addition, the bode diagrams of the above three force controllers and the corresponding open loop systems can be seen in Figures 5.14 and 5.15.

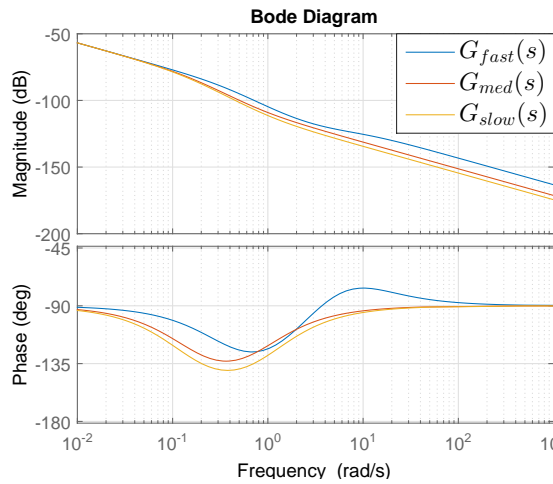


Figure 5.14: Bode diagram of three force controllers

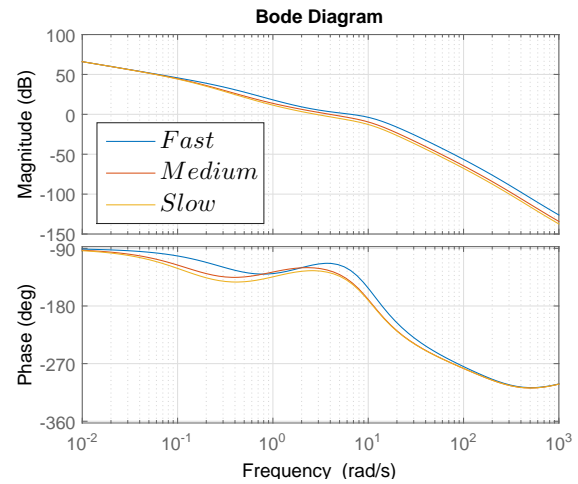


Figure 5.15: Bode diagram of the open loop systems with three force controllers

## 5.6 Simulation of Robotic FSW Process

In this section, a simulator of the robotic FSW process is established and simulation studies have been performed to test the performance of designed force controllers.

### 5.6.1 KUKA Robot Controller and Trajectory Generator

In order to establish a high-fidelity simulator for the robotic FSW process, it is necessary to learn about the control architecture of the industrial robot KUKA considered in this study. In reality, the control of Kuka robot is performed by a specially developed industrial computer with the Windows NT operating system. According to the information of configuration files provided by the industrial robot manufacturer, a schematic diagram of the control of KUKA robot for FSW process can be deduced, as shown in Figure 5.16. The entire control system includes the trajectory generator, robot joint motion controller, dynamic model of motor, dynamic model of flexible joint robot and FSW process model.

As a matter of fact, the trajectory generator of the industrial robot is realized by the user-written control programs which can generate the desired joint angular position, velocity and acceleration ( $q_d, \dot{q}_d, \ddot{q}_d$ ) in real time in the light of the reference trajectory data in Cartesian space ( $x, y, z, \gamma, \beta, \alpha, v$ ) defined by the user. As can be seen in Figure 5.16, the trajectory generation in real time can be divided into two successive steps: the generation of desired position, velocity and acceleration in Cartesian space ( $X_d, \dot{X}_d, \ddot{X}_d$ ) in the cycle of 12 ms, and the generation of desired joint angular position, velocity and acceleration ( $q_d, \dot{q}_d, \ddot{q}_d$ ) in the cycle of 2 ms. Besides, the trajectory generation relating to the movements of robot KUKA can be accomplished by three functions provided by the programming language of the robot controller: *ptp*, *lin* and *circ*.

In addition, the robot joint motion controller can be regarded as a combination of a joint position control loop with a sample time of 2 ms and a joint velocity control loop with a sample time of 0.5 ms based on the technical documents provided by the robot manufacturer. In the meantime, the servo control of the currents of the synchronous motors are implemented by the Siemens variable speed drives with a sample time of 0.2 ms, which is represented by a constant gain  $K_a$  in the schematic diagram.

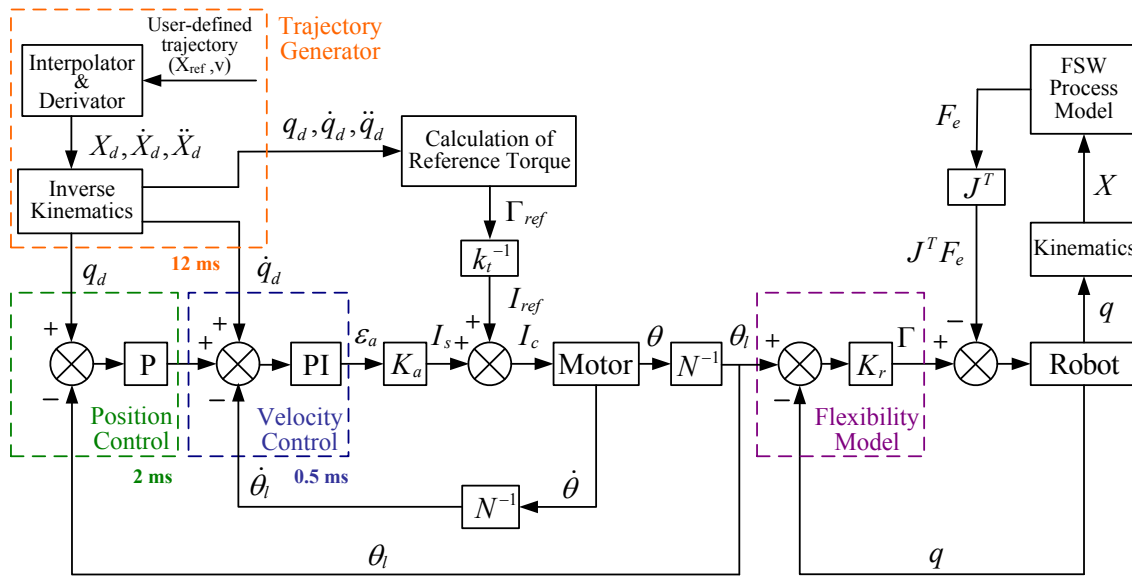


Figure 5.16: Schematic diagram of the position control of KUKA robot for FSW process (without the force control)

### 5.6.2 Dynamic Control of KUKA Robot

In the industrial robot controller, as shown in Figure 5.16, the linearization of robot dynamic equation can be carried out by using a decoupling method based on the rigid robot model and the calculation of reference motor torque in the light of the desired joint angular position, velocity and acceleration ( $q_d, \dot{q}_d, \ddot{q}_d$ ).

According to the motor dynamic equation (2.53), the gearbox output torque  $\Gamma$  can be written by:

$$\Gamma = N^T \Gamma_m - N^T I_a \ddot{\theta} - N^T F_{fm}(\dot{\theta}) \quad (5.58)$$

By substituting  $\Gamma$  in the robot dynamic equation (2.52) with the above equation, the following expression can be got:

$$M(q)\ddot{q} + H(q, \dot{q}) + F_{fr}(\dot{q}) - J^T(q)F_e = N^T \Gamma_m - N^T I_a \ddot{\theta} - N^T F_{fm}(\dot{\theta}) \quad (5.59)$$

In actual control of industrial robot KUKA, it can be supposed that the robot is completely rigid, that is to say, the deformation of the robot in joint space  $\tilde{q}$  can be considered as zero. Thus, equation (2.58) can be transformed as  $q = N^{-1}\theta$ , and the following formulations can be obtained:

$$\theta = Nq, \quad \dot{\theta} = N\dot{q}, \quad \ddot{\theta} = N\ddot{q}, \quad (5.60)$$

Then, equation (5.59) can be transformed to the following form:

$$M_e(q)\ddot{q} + H(q, \dot{q}) + F_{fe}(\dot{q}) - J^T(q)F_e = N^T \Gamma_m \quad (5.61)$$

with

$$M_e(q) = M(q) + N^T I_a N \quad (5.62)$$

$$F_{fe}(\dot{q}) = F_{fr}(\dot{q}) + N^T F_{fm}(N\dot{q}) \quad (5.63)$$

where  $M_e(q)$  represents the equivalent inertia matrix including the contribution of motor inertia transformed on the robot side,  $F_{fe}(\dot{q})$  denotes the equivalent friction including the contribution of motor friction transformed on the robot side.

As a consequence, the torque of the motor  $\Gamma_m$  can be calculated by:

$$\Gamma_m = N^{-T} [M_e(q)\ddot{q} + H(q, \dot{q}) + F_{fe}(\dot{q}) - J^T(q)F_e] \quad (5.64)$$

Since the reference torque of the motor is calculated based on the desired joint angular position, velocity and acceleration ( $q_d, \dot{q}_d, \ddot{q}_d$ ) and the hypothesis that the external force  $F_e$  is considered as zero, the reference torque can be obtained by:

$$\Gamma_{ref} = N^{-T} [M_e(q_d)\ddot{q}_d + H(q_d, \dot{q}_d) + F_{fe}(\dot{q}_d)] \quad (5.65)$$

As shown in Figure 5.16, the servo control of the torques of robot synchronous motors ( $\Gamma_m$ ) can be performed by the regulation of the corresponding commanded currents ( $I_c$ ), expressed by:

$$\Gamma_m = k_t I_c \quad (5.66)$$

where  $k_t$  is the diagonal matrix of torque constants of the synchronous motors which can be obtained by the identification.

Furthermore, the following control law can be applied in the regulation of the motor currents:

$$I_c = I_s + I_{ref} \quad (5.67)$$

with

$$I_s = K_a \varepsilon_a \quad (5.68)$$

$$I_{ref} = k_t^{-1} \Gamma_{ref} \quad (5.69)$$

where  $\varepsilon_a$  is the error of joint angular acceleration of the robot, which is also the output of robot motion controller,  $K_a$  is the gain of the motor current control loop,  $I_s$  is the current corresponding to the error of joint angular acceleration  $\varepsilon_a$ , and  $I_{ref}$  is the reference current corresponding to the reference torque  $\Gamma_{ref}$ .

Hence, it is quite obvious that the torque of the motor can be given by:

$$\Gamma_m = k_t I_s + \Gamma_{ref} \quad (5.70)$$

Based on equations (5.64) and (5.65), equation (5.70) can be transformed to:

$$\begin{aligned} M_e(q)\ddot{q} + H(q, \dot{q}) + F_{fe}(\dot{q}) - J^T(q)F_e \\ = N^T k_t I_s + M_e(q_d)\ddot{q}_d + H(q_d, \dot{q}_d) + F_{fe}(\dot{q}_d) \end{aligned} \quad (5.71)$$

By defining  $\Delta(q, \dot{q}) = H(q_d, \dot{q}_d) + F_{fe}(\dot{q}_d) - H(q, \dot{q}) - F_{fe}(\dot{q})$  and by assuming that  $M_e(q) \simeq M_e(q_d)$ , the above equation can be expressed as:

$$\begin{aligned} \ddot{q} - \ddot{q}_d &= M_e(q_d)^{-1} [N^T k_t I_s + \Delta(q, \dot{q}) + J^T(q)F_e] \\ &= M_e(q_d)^{-1} N^T k_t K_a \varepsilon_a + \Psi(q, \dot{q}, F_e) \end{aligned} \quad (5.72)$$

where  $\Delta(q, \dot{q})$  denotes the errors of the centrifugal, Coriolis and gravitational torques, and  $\Psi(q, \dot{q}, F_e) = M_e(q_d)^{-1} [\Delta(q, \dot{q}) + J^T(q)F_e]$ , which can be regarded as the term of perturbation based on the centrifugal, Coriolis and gravitational torque errors  $\Delta(q, \dot{q})$  and the external force  $F_e$ .

As a matter of fact, the influence of the above perturbation term  $\Psi(q, \dot{q}, F_e)$  can be neglected in the actual control of robot KUKA since the robot controller is designed based on the rigid robot model and the negligible external force.

Thus, the parameter  $\varepsilon_a$  is actually defined as the error of joint angular acceleration in the robot motion controller, as expressed by:

$$\varepsilon_a = \ddot{q} - \ddot{q}_d \quad (5.73)$$

As a result, the gain  $K_a$  of the motor current control in the schematic diagram of KUKA robot control (see Figure 5.16) can be calculated by:

$$\begin{aligned} K_a &= k_t^{-1} N^{-T} M_e(q_d) \\ &= k_t^{-1} N^{-T} [M(q_d) + N^T I_a N] \end{aligned} \quad (5.74)$$

Therefore, according to equations (5.65), (5.68), (5.70) and (5.74), the commanded currents of the motors can be given by:

$$I_c = k_t^{-1} N^{-T} [M_e(q_d)(\ddot{q}_d + \varepsilon_a) + H(q_d, \dot{q}_d) + F_{fe}(\dot{q}_d)] \quad (5.75)$$

### 5.6.3 Joint Motion Controller of the Simulator

As mentioned in the previous section, the actual joint motion control of the industrial robot KUKA is implemented by a P (Proportional) controller for the joint angular position control combined with a PI (Proportional and Integral) controller for the

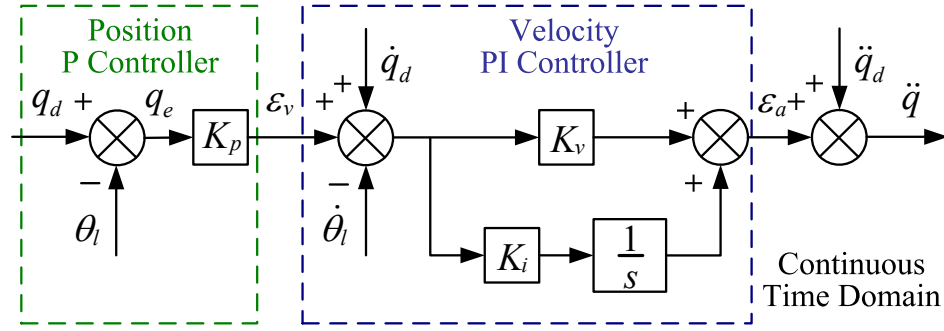


Figure 5.17: Diagram of the continuous-time joint motion controller of the simulator

joint angular velocity control. Consequently, the similar control structure can be performed in the corresponding simulator as well. Figure 5.17 shows the diagram of joint motion controller of the simulator in continuous time domain.

First of all, the P control law can be applied to the joint angular position of the robot, as expressed by:

$$\varepsilon_v = K_p(q_d - \theta_l) \quad (5.76)$$

where  $K_p$  is the vector of proportional gains of the P controller,  $\theta_l$  is the vector of measured angular positions of motor axes after the gear reduction.

Then, the PI control law of the joint angular velocity can be given by:

$$\varepsilon_a = K_v(\varepsilon_v + \dot{q}_d - \dot{\theta}_l) + K_i \int (\varepsilon_v + \dot{q}_d - \dot{\theta}_l) dt \quad (5.77)$$

where  $K_v$  is the vector of proportional gains of the PI controller, and  $K_i$  is the vector of integral gains of the PI controller.

According to the above P and PI control laws provided in Figure 5.17, the following equation can be obtained:

$$\begin{aligned} \varepsilon_a &= \ddot{q} - \ddot{q}_d \\ &= K_v[K_p(q_d - \theta_l) + \dot{q}_d - \dot{\theta}_l] + K_i \int [K_p(q_d - \theta_l) + \dot{q}_d - \dot{\theta}_l] dt \end{aligned} \quad (5.78)$$

Since the robot can be supposed as completely rigid in the design of joint motion controller, the vector of measured angular positions of motor axes after the gear reduction  $\theta_l$  can be considered as the vector of actual joint angular positions of the robot  $q$ , that is to say,  $\theta_l = q$ .

Let  $\varepsilon = q_d - q$ , equation (5.78) can be rewritten by:

$$\ddot{\varepsilon} + K_v K_p \varepsilon + K_v \dot{\varepsilon} + K_i \varepsilon + K_i K_p \int \varepsilon dt = 0 \quad (5.79)$$

By applying the Laplace transform to the above equation, the following characteristic equation yields:

$$s^3 + K_v s^2 + (K_i + K_v K_p) s + K_i K_p = 0 \quad (5.80)$$

Thus, the P and PI controller can be tuned by using pole placement method, and the control parameters  $K_p$ ,  $K_v$  and  $K_i$  can be determined by placing the three poles of the closed loop system on the complex plane.

In fact, the real-time control of the joint angular position and velocity are realized by the controller of robot KUKA with the sample times of 2 ms and 0.5 ms respectively. Therefore, the discrete-time joint motion controller of simulator can be established, as shown in Figure 5.18.

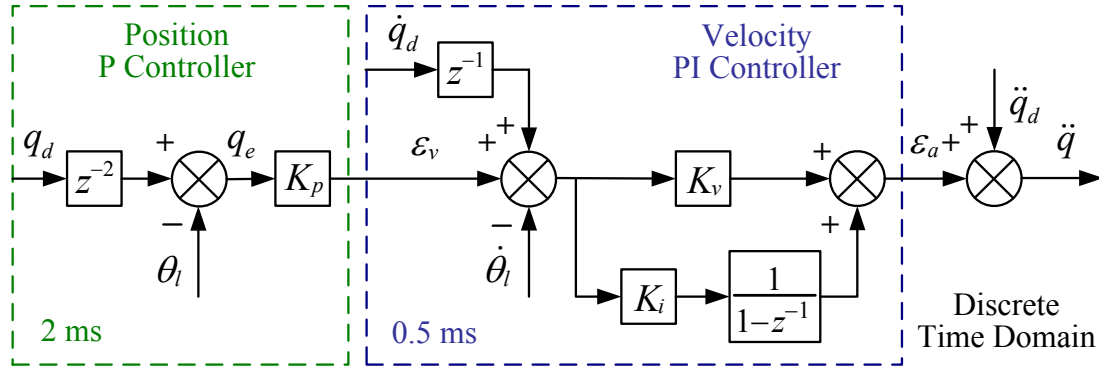


Figure 5.18: Diagram of the discrete-time joint motion controller of the simulator

As shown in the above diagram, the notation  $z^{-1}$  represents the operator of delay and  $z^{-n}$  denotes a delay of  $n$  sample periods for the input signal. As a consequence, the following equation can be obtained as similar with the continuous case:

$$\ddot{q} - \ddot{q}_d = [K_p(z^{-2}q_d - q) + z^{-1}\dot{q}_d - \dot{q}] \left( K_v + \frac{K_i}{1 - z^{-1}} \right) \quad (5.81)$$

Then, the error dynamics can be given by:

$$(1 - z^{-1})\ddot{\varepsilon} + (K_v + K_i - K_v z^{-1})\dot{\varepsilon} + K_p(K_v + K_i - K_v z^{-1})\varepsilon = 0 \quad (5.82)$$

Thus, the corresponding characteristic equation can be obtained by:

$$[(K_p + 1)(K_v + K_i) + 1]z^3 - (K_p K_v + 2K_v + K_i + 3)z^2 + (K_v + 3)z - 1 = 0 \quad (5.83)$$

Similarly, the control parameters  $K_p$ ,  $K_v$  and  $K_i$  of the discrete motion controller can be determined by means of pole placement as well. It should be noted that the actual values of these parameters of robot controller can be directly used in the simulator, which are listed in Table 5.5.

Table 5.5: Actual values of gains of robot joint motion controller

| Gains | Axis 1 | Axis 2 | Axis 3 | Axis 4 | Axis 5 | Axis 6 |
|-------|--------|--------|--------|--------|--------|--------|
| $K_p$ | 600    | 450    | 350    | 300    | 300    | 300    |
| $K_v$ | 100    | 80     | 80     | 45     | 50     | 25     |
| $K_i$ | 500    | 500    | 425    | 100    | 100    | 100    |

### 5.6.4 Establishment of Simulator for Robotic FSW Process

After building the model of the robot motion controller, the simulator of the robotic FSW process can be established in the Matlab/Simulink environment, and the schematic diagram of the simulator is shown in Figure 5.21.

In addition to the model of robot motion controller, the detailed presentations of other models built in the simulator are given as follows:

#### Model of Computing Commanded Motor Torques

According to equations (5.65), (5.68), (5.70) and (5.74), the model of computing the commanded motor torques ( $\Gamma_c$ ) can be provided by:

$$\begin{aligned}\Gamma_c &= k_t I_s + \Gamma_{ref} \\ &= N^{-T} [M_e(q_d)(\ddot{q}_d + \varepsilon_a) + H(q_d, \dot{q}_d) + F_{fe}(\dot{q}_d)]\end{aligned}\quad (5.84)$$

The above model can be obtained by respectively replacing the inertia matrix  $M(q)$  and the friction at robot side  $F_{fr}(\dot{q})$  with the equivalent inertia matrix  $M_e(q) = M(q) + N^T I_a N$  and the equivalent friction  $F_{fe}(\dot{q}) = F_{fr}(\dot{q}) + N^T F_{fm}(N\dot{q})$  in the inverse dynamic model of the robot. Besides, the torque of gravity compensator at the 2nd axis ( $\Gamma_{com2}$ ) is considered in the model as well.

#### Dynamic Model of Motors

According to equation (2.53) and the commanded motor torques  $\Gamma_c$ , the dynamic model of the motors are given by:

$$\ddot{\theta} = I_a^{-1} [\Gamma_c - N^{-T} \Gamma - F_{fm}(\dot{\theta})] \quad (5.85)$$

#### Flexibility Model of Joints

According to equation (2.56), the joint flexibility can be modeled by a linear torsional spring [20]. By using the diagonal matrix of the torsional stiffness  $K_r$ , the flexibility model can be obtained by:

$$\Gamma = K_r(\theta_l - q) \quad (5.86)$$

#### Direct Dynamic Model of the Robot

According to equation (2.52), the direct dynamic model of the robot can be provided by the following equation:

$$\ddot{q} = M(q)^{-1} [\Gamma + J^T(q) F_e - H(q, \dot{q}) - F_{fr}(\dot{q})] \quad (5.87)$$

The diagram of the above models are shown in Figure 5.19. Furthermore, since the dynamic modeling and identification of the axial force in the FSW process have already been carried out in Chapter 4, the model of the axial force ( $F_z$ ) during the FSW process can be directly used in the simulator.



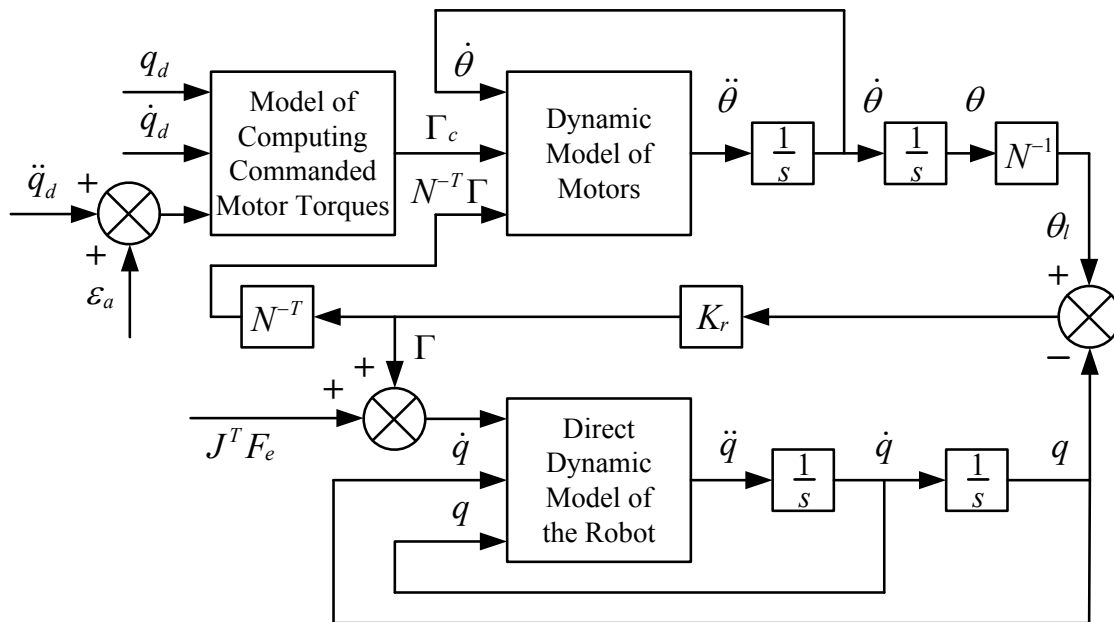


Figure 5.19: Diagram of the main models in the simulator

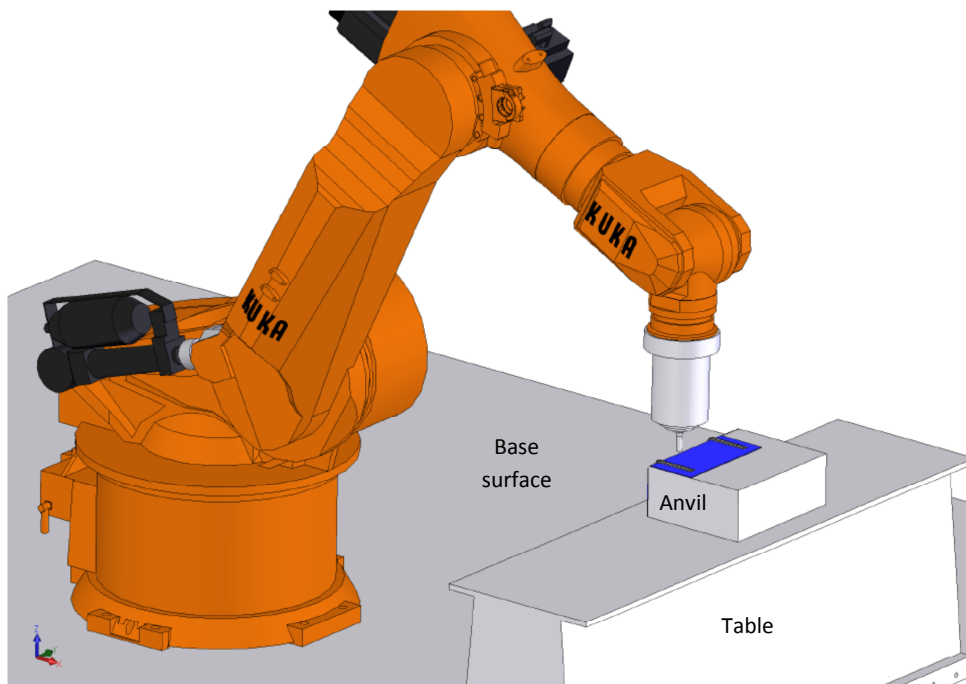


Figure 5.20: Three-dimensional model of robot KUKA in the FSW process

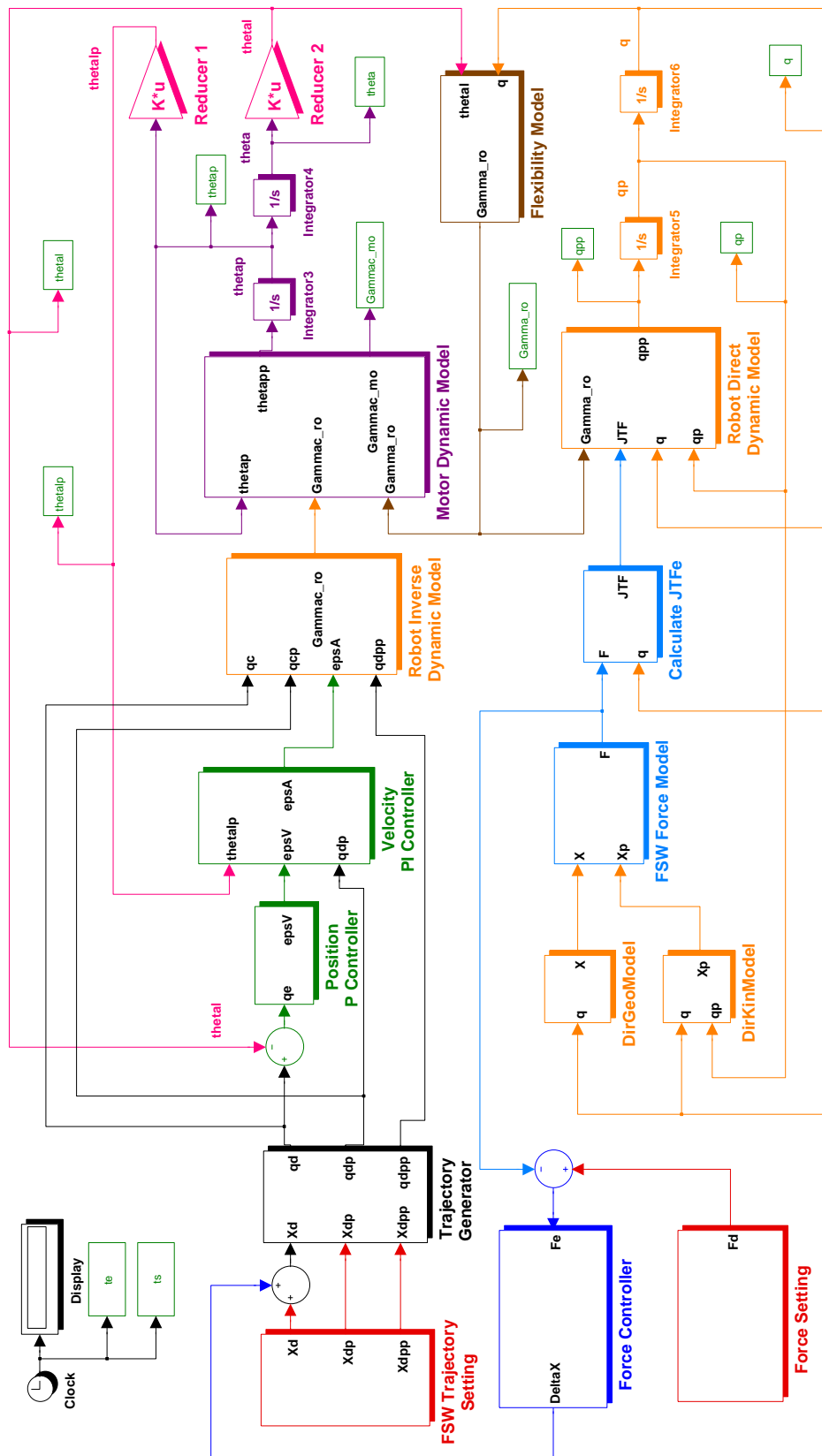


Figure 5.21: Schematic diagram of the simulator in Matlab/Simulink environment

### 5.6.5 Simulations and Results Analysis

Simulation studies have been implemented to test the rapidness and robustness of the force controller which is designed in section 5.5 as well as the performance of the simulator of the robotic FSW process.

As mentioned in section 5.5, by choosing different performance types (*Fast*, *Medium* and *Slow*) and *Soft* surface type in Table 5.4, three different force controllers can be obtained and their transfer functions can be given in equations (5.55), (5.56) and (5.57).

As a matter of fact, the axial force control of the robot should be implemented with the sample time of 12 *ms*. Consequently, the above continuous-time force controller needs to be discretized by using the MATLAB function *c2d* (continuous to discrete) and the corresponding discretization method can be specified as zero-pole matching method since the method can guarantee good matching in the frequency domain between the continuous-time and discrete-time models [163]. As a result, the transfer functions of the corresponding discrete-time force controllers can be provided by the following equations:

$$G_{fast}(z) = \frac{8.046 \times 10^{-8}(z^2 - 1.928z + 0.929)}{z^3 - 2.881z^2 + 2.763z - 0.8819} \quad (5.88)$$

$$G_{med}(z) = \frac{3.28 \times 10^{-8}(z - 0.9899)}{z^2 - 1.998z + 0.9981} \quad (5.89)$$

$$G_{slow}(z) = \frac{2.276 \times 10^{-8}(z - 0.9875)}{z^2 - 1.998z + 0.9984} \quad (5.90)$$

where the units of the transfer function  $G$ , the coefficient of the transfer function, the operator  $z^{-1}$  and the time constants are  $m/N$ ,  $m/(Ns)$ ,  $Hz$  and *sec.* respectively.

In the simulation of the robotic FSW process, the robot is controlled to weld along the  $x$  direction of the tool frame with constant travel speed ( $v = 400 \text{ mm/min}$ ) and rotation speed ( $\Omega = 1100 \text{ tr/min}$ ). Besides, the desired axial force is 9000  $N$  and the initial axial force is 2000  $N$ .

Table 5.6: Performance comparison of three force controllers in Simulation

| Performance Type                     | Fast           | Medium    | Slow      |
|--------------------------------------|----------------|-----------|-----------|
| Desired Rise Time ( $t_r$ )          | 285ms          | 476ms     | 667ms     |
| Surface Type                         | Soft           | Soft      | Soft      |
| Desired Phase Margin ( $m_\varphi$ ) | 55°            | 55°       | 55°       |
| Controller Type                      | Phase-lead-lag | Phase-lag | Phase-lag |
| Rise Time in Simulation              | 305ms          | 446ms     | 580ms     |
| Settling Time in Simulation          | 2.06s          | 2.81s     | 2.76s     |
| Peak Time in Simulation              | 456ms          | 684ms     | 1.14s     |
| Overshoot in Simulation              | 23.8%          | 19.1%     | 18.5%     |

Figure 5.22 shows the comparison of the axial forces in the simulations with three different force controllers. The simulation results demonstrate good performance of

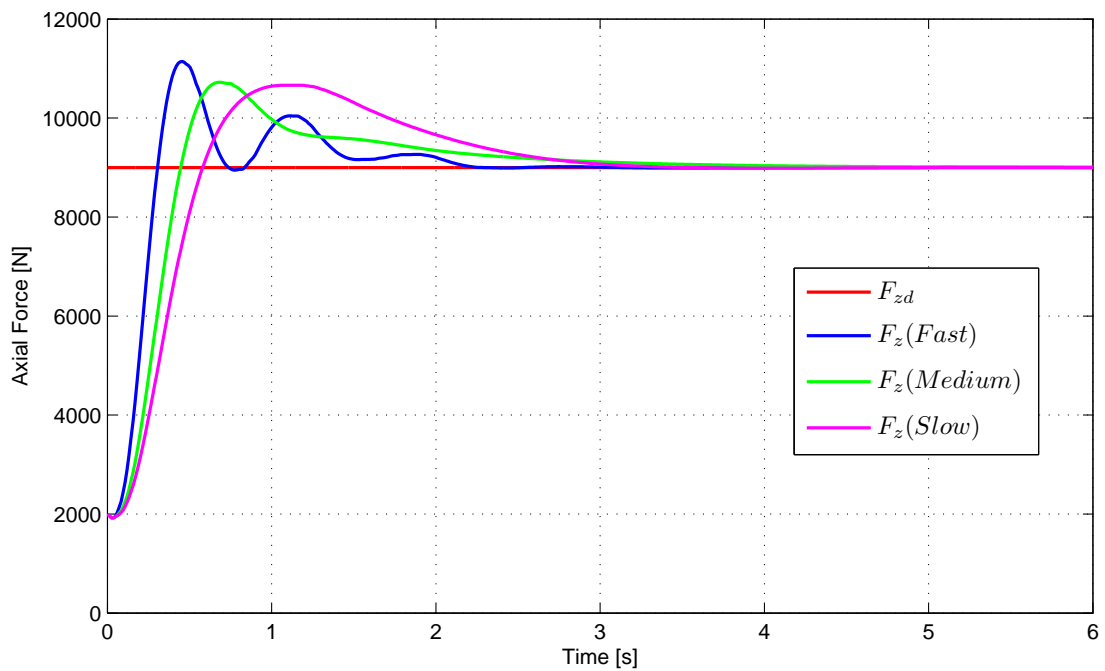


Figure 5.22: Comparison of the axial forces during the simulations with three different force controllers

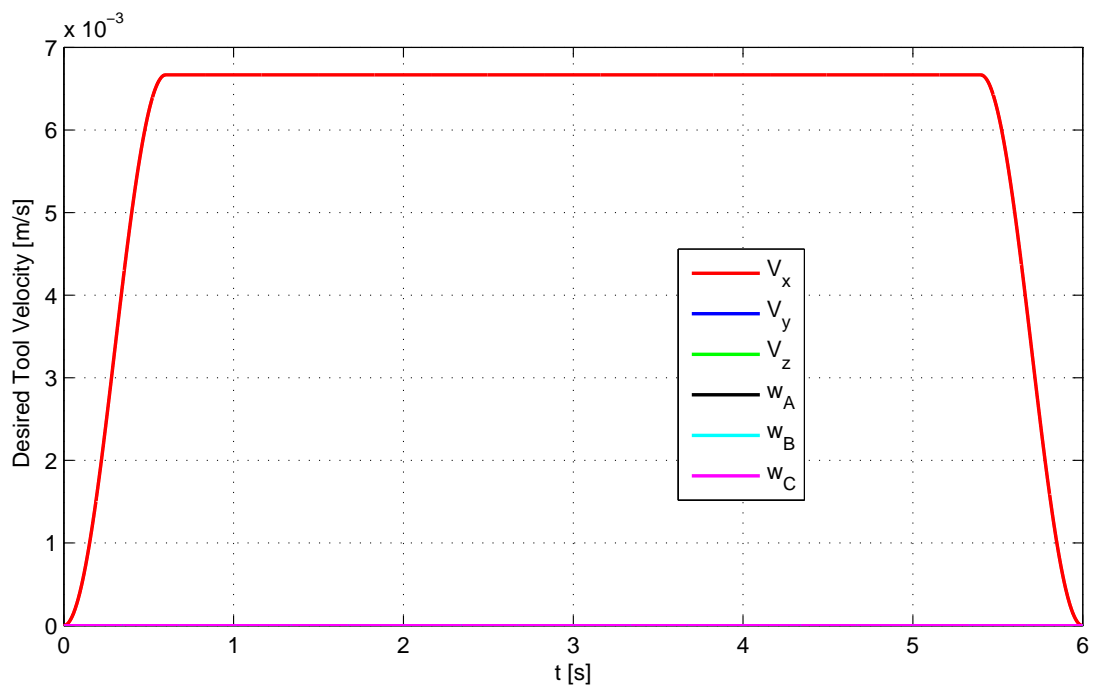


Figure 5.23: Desired tool velocity in Cartesian space used in the simulation

the force controllers, which are provided in Table 5.6. Since the hybrid position/force control scheme is applied, the force control can be implemented only in  $z$  direction.

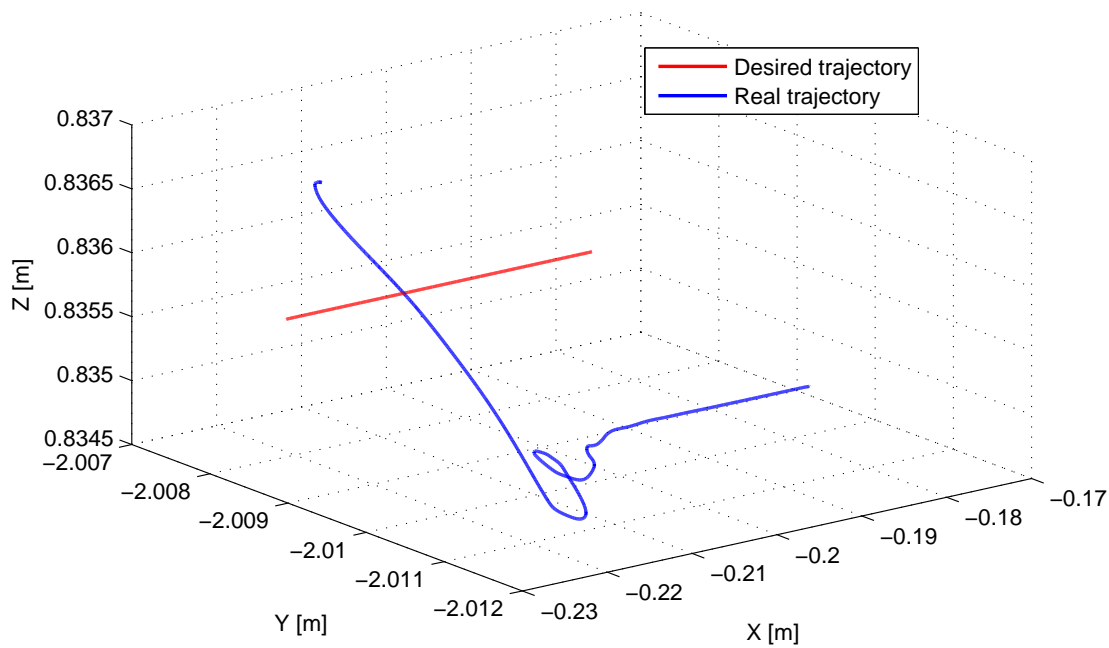


Figure 5.24: Comparison of the desired tool position and the actual tool position in 3D space during the simulation with only force control in  $z$  direction

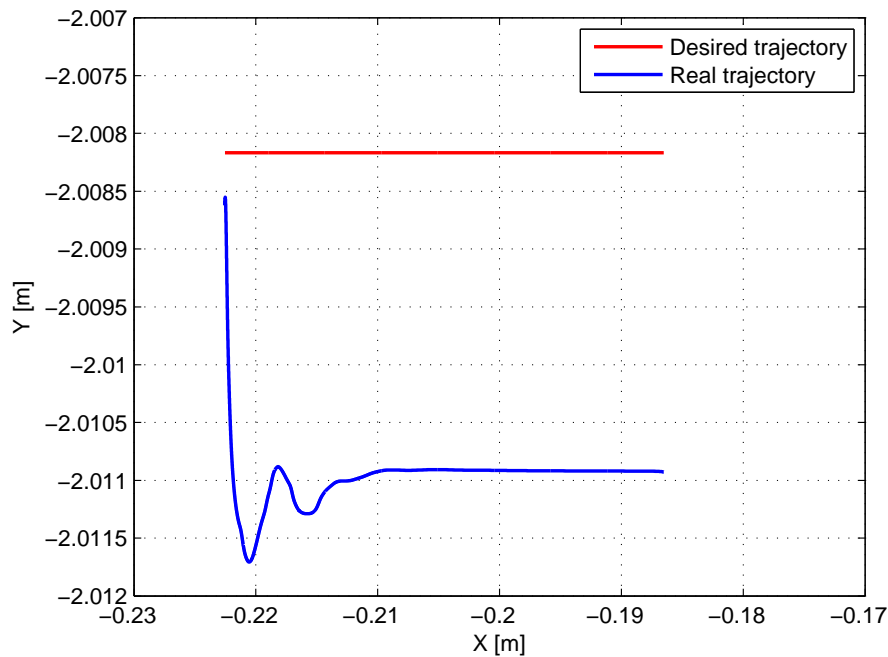


Figure 5.25: Comparison of the desired tool position and the actual tool position in X-Y plane during the simulation with only force control in  $z$  direction

The tool velocity is increased and decreased in the first 0.6 s and last 0.6 s of the simulation time, as shown in Figure 5.23.

Furthermore, the comparison of the desired tool position and the actual tool

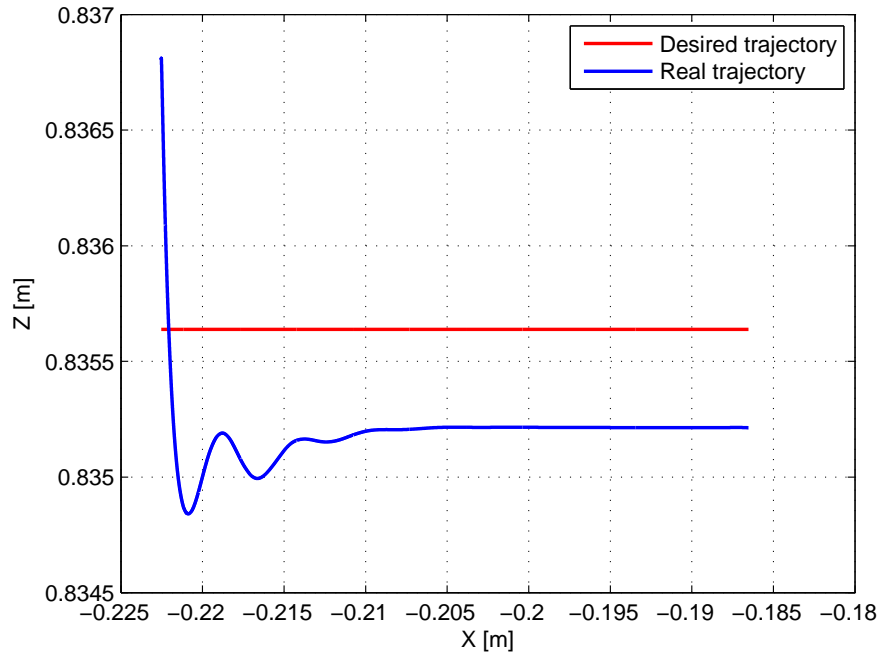


Figure 5.26: Comparison of the desired tool position and the actual tool position in X-Z plane during the simulation with only force control in  $z$  direction

position in 3D space, X-Y plane and X-Z plane during the simulation can be found in Figures 5.24, 5.25 and 5.26. As shown in these figures, the deviation of the tool position in  $y$  direction is about  $3.5\text{ mm}$ , and the corresponding deviation of the tool position in  $z$  direction is about  $0.5\text{ mm}$ .

## 5.7 Vibration Analysis of Axial Force

During the FSW process, a significant vibration can be observed in the measurements of the axial force  $F_{zm}$ , as shown in Figure 5.27. In this section, the oscillation of the axial force can be analyzed by identifying an experimental axial force model at first. Then, a disturbance model, which constantly changes the contact position of zero force in  $z$  direction (the vertical reference position  $Z_0$ ) is proposed and applied to the simulation of the robotic FSW process.

### 5.7.1 Identification of Experimental Axial Force Model

According to the research work of Zaeh and Gebhard [96], the FSW process forces and their impact on the machine structure were analyzed and an experimental process force model was proposed and validated by numerous welding experiments.

The proposed experimental process force model is composed of the static force components ( $F_{i0}$ ) and the superposition of harmonic functions with different amplitudes ( $A_{ij}$ ), phases ( $\varphi_{ij}$ ) and the frequency ( $\omega_r$ ) of the rotation speed ( $\Omega$ ) and its

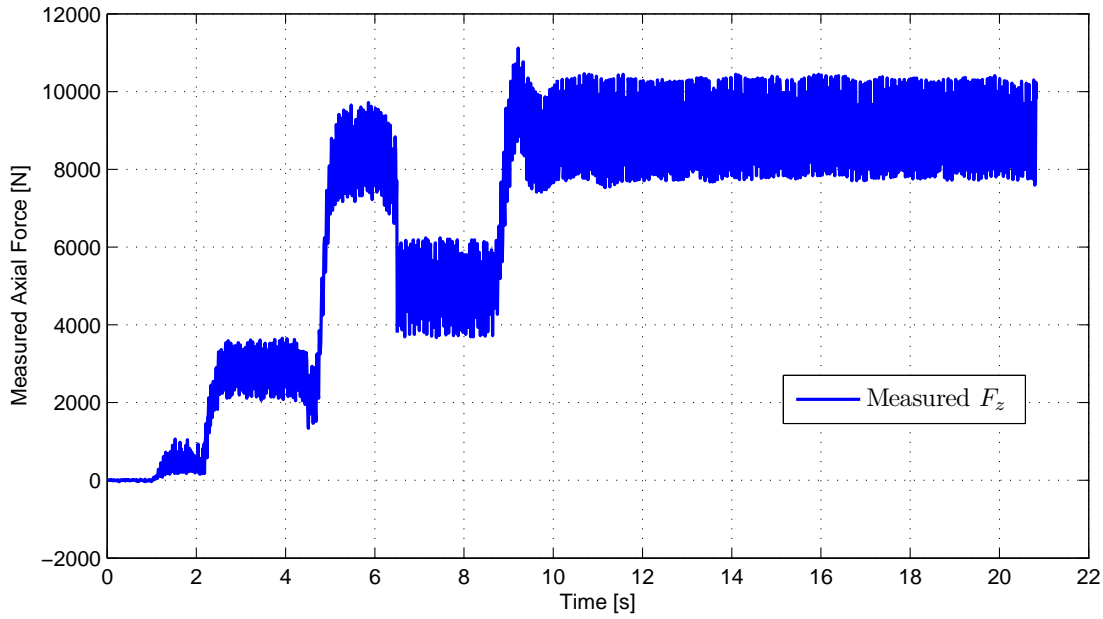


Figure 5.27: Measured axial force during friction stir welding process (test IS1709J:  $\Omega = 1100 \text{ tr/min}$ ,  $v = 400 \text{ mm/min}$ , welding depth 6 mm, material Al6082)

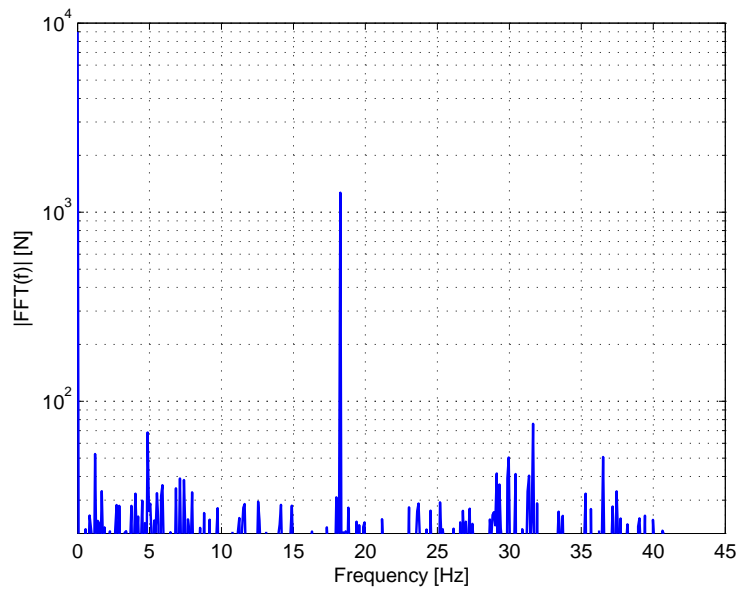


Figure 5.28: Spectrum of the axial force in FSW process (test IS1709J)

first harmonic, as expressed by the following equations [96]:

$$F_{x,fsw} = F_{x0} + A_{x1} \sin(\omega_r t) + A_{x2} \sin(2\omega_r t + \varphi_{x2}) \quad (5.91)$$

$$F_{y,fsw} = F_{y0} + A_{y1} \sin(\omega_r t + \varphi_y) + A_{y2} \sin(2\omega_r t + 2\varphi_y + \varphi_{y2}) \quad (5.92)$$

$$F_{z,fsw} = F_{z0} + A_z \sin(\omega_r t + \varphi_z) \quad (5.93)$$

where the units of  $F_{x,fsw}$ ,  $F_{y,fsw}$ ,  $F_{z,fsw}$ ,  $F_{x0}$ ,  $F_{y0}$ ,  $F_{z0}$ ,  $A_{x1}$ ,  $A_{x2}$ ,  $A_{y1}$ ,  $A_{y2}$  and  $A_z$  are  $N$ . The units of  $\varphi_{x2}$ ,  $\varphi_y$ ,  $\varphi_{y2}$  and  $\varphi_z$  are  $rad$  and the unit of  $\omega_r$  is  $rad/s$ .

Furthermore, the frequency of the above axial force equation can be verified by analyzing the spectrum of the axial force during the FSW process, as shown in Figure 5.28. According to the spectrum of the axial force in FSW process, the frequency of the wave equation in the axial force model can be obtained as  $18.26 Hz$ , which is extremely approximate to the frequency of the spindle rotation speed ( $f_r = 1100/60 \simeq 18.33 Hz$ ). Besides, the static axial force component  $F_{z0}$  and the amplitude  $A_z$  can be identified as  $9000 N$  and  $1250 N$  and the phase  $\varphi_z$  can be neglected based on the analysis of Figure 5.27 and 5.28.

Therefore, by considering the force vibration, the axial force  $F_z$  in the welding stage of FSW process can be expressed by:

$$F_z = 9000 + 1250 \sin(115.19t) \quad (5.94)$$

where the unit of  $F_z$  is  $N$  and  $F_{z0} = 9000 N$ ,  $A_z = 1250 N$  and  $\omega_r = 115.19 rad/s$ .

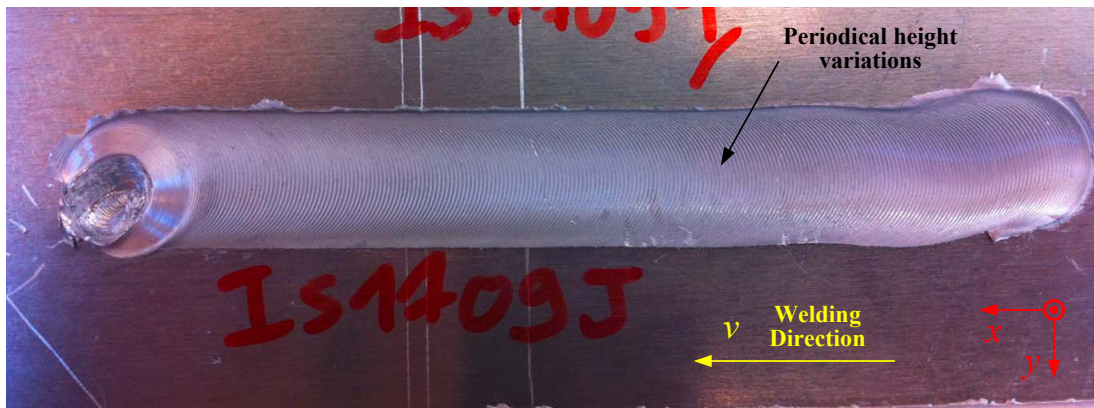


Figure 5.29: Welded workpiece after the FSW process (test IS1709J)

### 5.7.2 Disturbance Model for Force Vibration and Simulation

In the actual robotic FSW process, the periodical height variations of the surface of the welded workpiece can be clearly observed, as shown in Figure 5.29.

By means of a 2D laser scanner LLT2800-100 made by Micro-Epsilon, the values of  $x$  and  $z$  coordinates of the surface of the welded workpiece can be obtained in Figure 5.30, and the resolution of the 2D laser scanner is  $0.01 mm$ . The average value of  $z$  coordinates is  $123.69 mm$  and the standard deviation is  $0.0540 mm$ . In order to eliminate the influence of the noise and obtain more accurate height variations of the weld surface, a linear relationship between  $z$  and  $x$  coordinates can be obtained by using the least squares method, as expressed by:

$$Z_f = -0.0094x + 123.69 \quad (5.95)$$

where the units of  $Z_f$  and  $x$  are  $mm$ .

Accordingly, the height variations of the surface of the welded workpiece can be calculated by:

$$\Delta h = Z_m - Z_f \quad (5.96)$$



where  $Z_m$  and  $Z_f$  are measured and fitted value of  $z$  coordinates,  $\Delta h$  represents the height variations of the surface of the welded workpiece, which is provided in Figure 5.31. The average value of the height variations ( $\Delta h$ ) is approximate to zero ( $-9.4 \times 10^{-6} \text{ mm}$ ) and the standard deviation is  $0.0495 \text{ mm}$ .

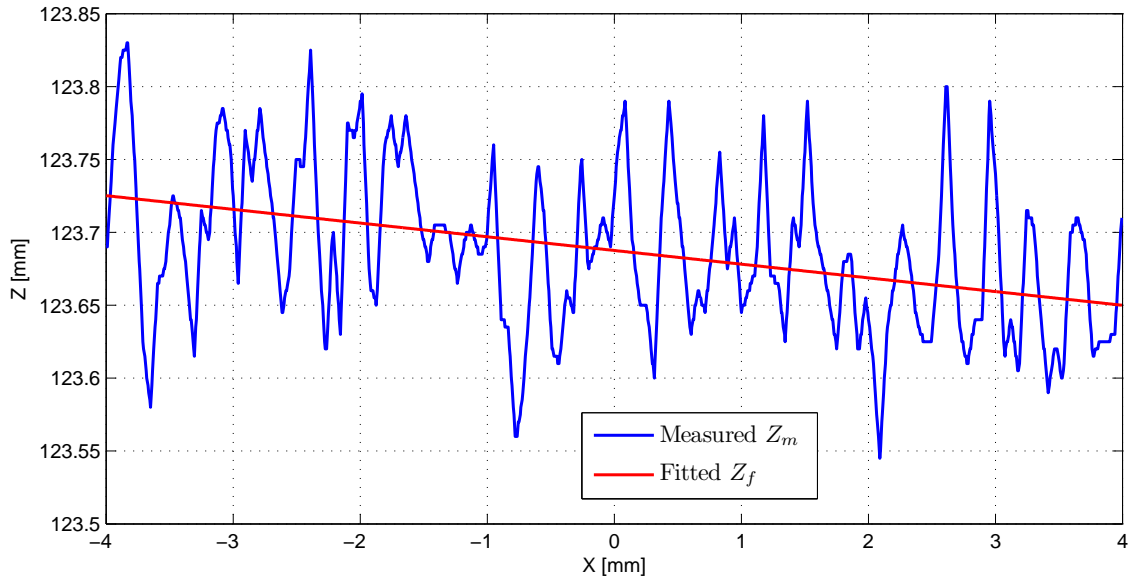


Figure 5.30: Measured and fitted  $z$  coordinates of the weld surface and corresponding  $x$  coordinates (test 1709J,  $z$  coordinates denotes the height of weld surface and  $x$  coordinated denotes the position along welding direction)

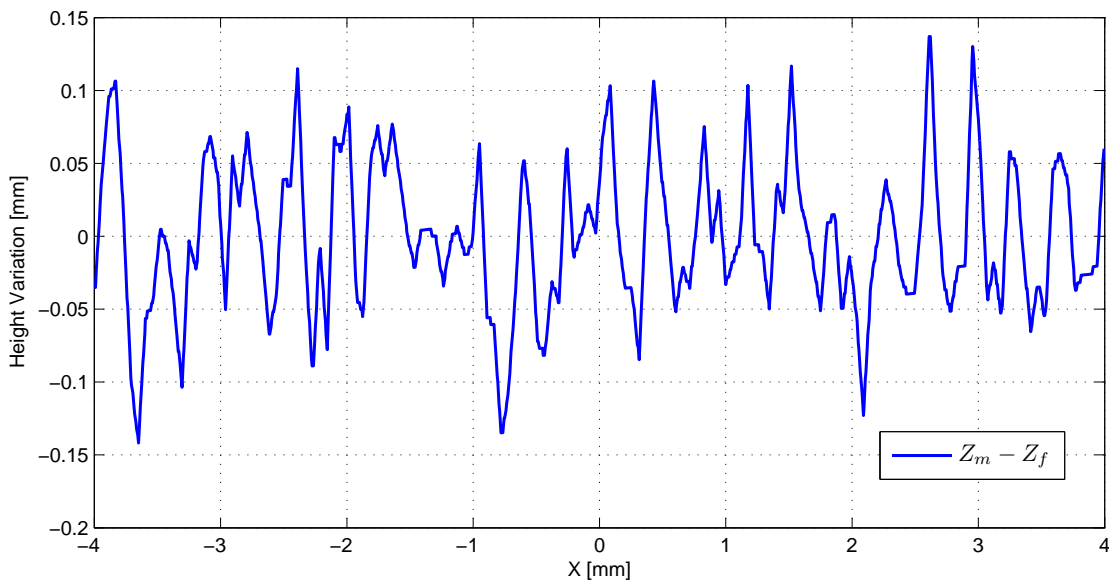


Figure 5.31: Height variations of the surface of welded workpiece  $\Delta h$  and corresponding  $x$  coordinates (test 1709J)

Since the measurements are implemented by the 2D laser scanner after the FSW

process, only  $x$  and  $z$  coordinates can be found. As the length of measurements in  $x$  direction is  $8 \text{ mm}$  and the robot is controlled to weld along the  $x$  direction of the tool frame with constant travel speed  $v = 400 \text{ mm/min}$ , the equivalent total time of the measurements can be calculated as  $1.2 \text{ s}$ , which can be used to obtain the equivalent sampling time of the measurements and calculate the Discrete Fourier Transform (DFT) by using the Fast Fourier Transform (FFT) algorithm provided by the MATLAB function `fft`. As a result, the spectrum of height variations of the weld surface in the FSW process can be given in Figure 5.32.

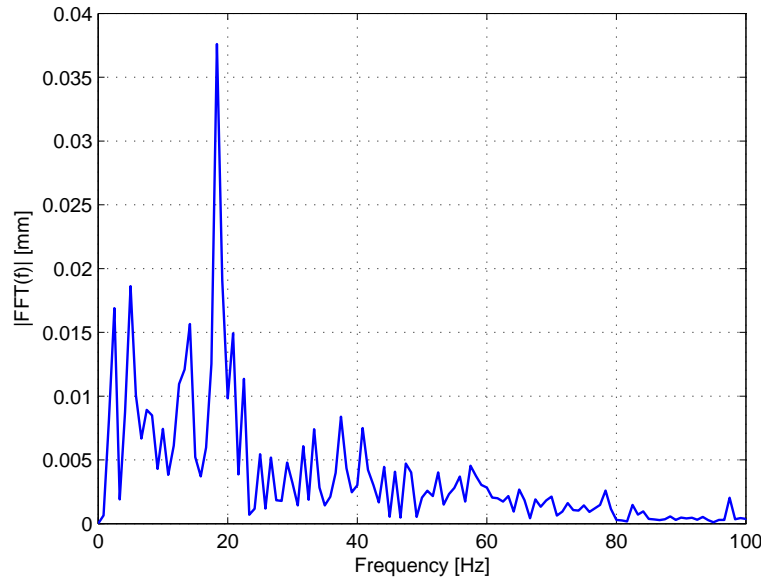


Figure 5.32: Spectrum of height variation of the weld surface in FSW process (test 1709J)

Consequently, a disturbance model which can continuously change the vertical tool position of zero force ( $Z_0$ ) can be proposed and defined as the following sinusoidal form (the phase is not considered here):

$$\Phi_{Z_0}(t) = d_\Phi \sin(\omega_\Phi t) \quad (5.97)$$

where  $d_\Phi$  and  $\omega_\Phi$  can be identified according to Figure 5.31 and 5.32. As a result, the frequency of the sinusoidal form disturbance can be obtained as  $18.33 \text{ Hz}$  (see Figure 5.32), which is nearly the same as the frequency of the spindle rotation speed ( $f_r = 1100/60 \simeq 18.33 \text{ Hz}$ ). Thus,  $d_\Phi = 0.00014 \text{ m}$  and  $\omega_\Phi = 115.19 \text{ rad/s}$ .

As shown in Figure 5.12 and equation (5.5), the vertical tool position of zero force  $Z_0$  can be regarded as the position disturbance in  $z$  direction in the axial force control system. Compared with the case without the proposed new disturbance model, the vertical position disturbance  $Z_0(t)$  can be changed based on the original constant value ( $Z_{con}$ ), as expressed by:

$$\begin{aligned} Z_0(t) &= Z_{con} + \Phi_{Z_0}(t) \\ &= Z_{con} + d_\Phi \sin(\omega_\Phi t) \end{aligned} \quad (5.98)$$

Since the introduction of above new vertical position disturbance model  $Z_0(t)$  can not change the open-loop transfer function of the axial force control system, the force controllers designed in Section 5.5 can be applied for the new case as well.

As for the FSW process model, the velocity disturbance caused by the position disturbance should be considered according to equation (5.97):

$$\frac{d\Phi_{Z0}}{dt} = d_{\Phi}\omega_{\Phi} \cos(\omega_{\Phi}t) \quad (5.99)$$

Thus, the axial force of the simulator can be calculated by:

$$F_z = K_e[Z - Z_{con} - d_{\Phi} \sin(\omega_{\Phi}t)] + f_e \left[ \frac{d}{dt}Z + d_{\Phi}\omega_{\Phi} \cos(\omega_{\Phi}t) \right] \quad (5.100)$$

The simulation of the robotic FSW process with the disturbance model can be carried out, and Figure 5.33 shows the time history of axial force in the simulation, demonstrating a good performance of the designed force controller. Besides, it can be observed that the oscillation amplitude of the simulated axial force is about 1300 N which corresponds to the oscillation amplitude of the measured axial force shown in Figure 5.27.

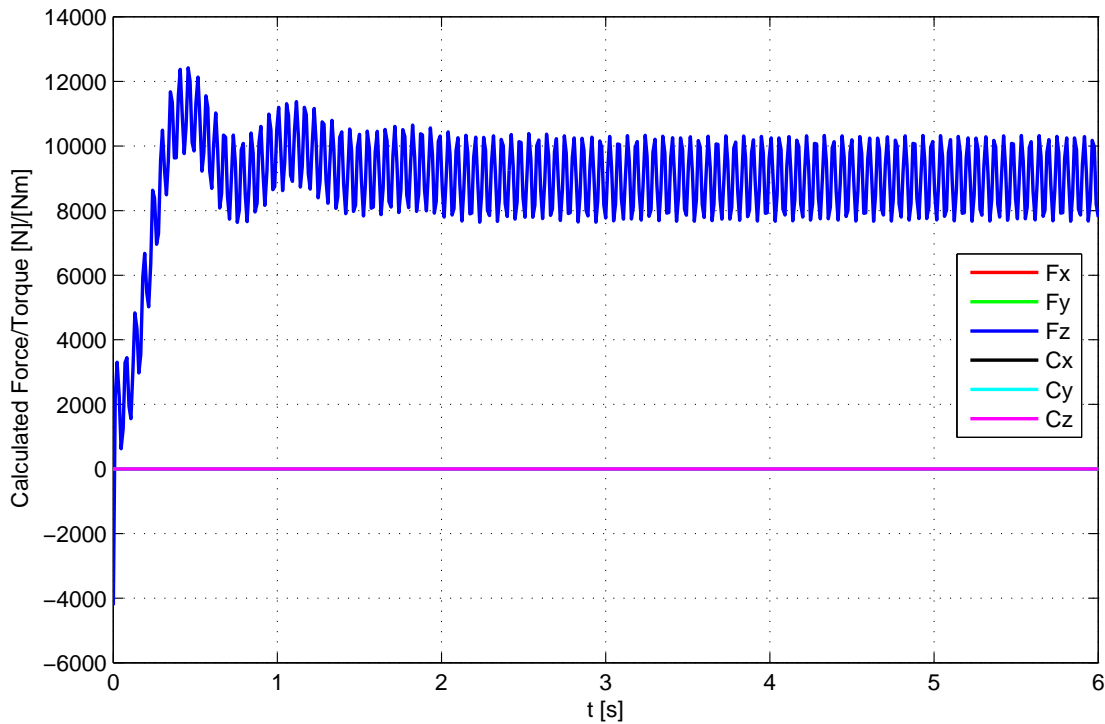


Figure 5.33: Simulation of robotic FSW process with the disturbance model (The case of using fast type force controller)

## 5.8 Conclusion

In this chapter, the design of robust force controllers for industrial robot manipulators and the development of a simulation system of robotic FSW process have been

presented. First of all, the inner position/outer force control strategy is selected for the robot system since the industrial robot KUKA KR500-2MT considered in this study can be controlled only by using its positional interface. Next, the system modeling of robotic FSW process is accomplished, simultaneously considering the complete system kinematics, the position response of the industrial robot controller (rigid robot displacement model), the joint flexibility of robot and the dynamic axial force model. The parameter identification of rigid robot displacement model is respectively performed by using the algorithms of transfer function estimation and process model estimation, which are two different algorithms of the System Identification Toolbox provided by the software MATLAB. With the help of a systematic design procedure [95] and the above system modeling, several force controllers are designed with high gain and sufficient phase margin for the closed-loop system, ensuring the robustness of force control system. Finally, the simulator of robotic FSW process is established and the corresponding simulation studies have been carried out. As a consequence, the simulation results demonstrate good performance of the designed force controllers. In addition, the vibration of the axial force during the robotic FSW process is analyzed by the identification of an experimental axial force model [96]. According to the height measurements of the weld surfaces, a disturbance model of zero force vertical position ( $Z_0(t)$ ) is proposed, which can lead to the oscillation of axial force. This disturbance model has been applied in the simulation studies and verified by observing the axial force vibration in the simulation results. Besides, the proposed force controllers can be used in the simulator as well.



# Chapter 6

## General Conclusions and Perspectives

### 6.1 General Conclusions

During the robotic FSW process, since the external force exerted on the robot is extremely large, the deformation of the robot can not be neglected and the flexibility of the robot needs to be considered, yielding some robot modeling and control problems. This thesis has presented the modeling of the flexible joint industrial robot manipulators for robust force control used in FSW process and the main contributions of this research can be listed as follows:

- a. **A state of the art related to the research was presented.** A state of the art related to flexible joint industrial robot manipulators was presented on the basis of the current published works, as well as a literature review on four aspects of closely related problems. First of all, the brief definition, history and current development of industrial robots were given and several kinds of famous industrial robots in the history were introduced. The sales of the worldwide industrial robots in the past ten years were provided, showing the rapid growth of the world robot market. Then, the modeling method of the flexible joint industrial robots were discussed and several different control methods for the flexible joint robots were reviewed, such as singular perturbation and integral manifold, feedback linearization, cascaded system and integrator backstepping, PD control, adaptive control, robust control, neural networks, fuzzy control and some other control methods. Besides, a literature review on force control strategies of robot was given in the fifth chapter as well.
- b. **Modeling of a 6-DOF flexible joint robot manipulator were presented.** Based on the fundamental knowledge, the modeling of a 6-DOF flexible joint robot manipulator was presented, including direct geometric model (DGM), inverse geometric model (IGM), direct kinematic model (DKM), inverse kinematic model (IKM), direct dynamic model (DDM) and inverse dynamic model (IDM). To begin with, the basic concepts of robotics were introduced, such as the spatial descriptions, the homogeneous transformations, the X-Y-Z fixed angles and the modified Denavit-Hartenberg convention, etc.

Then, the forward and inverse kinematics problems were solved by using homogeneous transformation matrices of all adjacent frames and Paul method respectively. Meanwhile, the forward and inverse instantaneous kinematics issues were addressed as well by introducing the velocity Jacobian matrix and its inverse matrix. Next, the dynamic model of the robot body and the motors can be obtained by means of the Lagrange method. It should be noted that the joint flexibility model, the friction models at both robot axes and motor axes were discussed and taken into the consideration of robot dynamic modeling, as well as the torque of gravity compensator at the second axis. All the above-mentioned models are extremely important and essential in the dynamic model simplification of the industrial robot, the simulation of robotic FSW process and the design of robust force controller of the robot.

- c. A novel simplification approach for the industrial robot dynamic model was proposed by means of interval method.** In order to reduce the computation time and memory occupation, a novel approach for the simplification of industrial robot dynamic model was proposed based on interval method. Seen from the symbolic dynamic model of the 6-DOF industrial manipulator generated by the software *SYMORO+*, all the expressions of components in inertia matrix  $M(q)$  and the vector of centrifugal, Coriolis and gravitational torques  $H(q, \dot{q})$  show complexities and strong nonlinearities, and a high coupling also exists between the control inputs. Thus, a generalized interval-based simplification algorithm was proposed to reduce the robot model complexity. The entire simplification process can be divided into three steps. After the simplification example, the results and error analysis of the simplified model for the whole robot workspace as well as corresponding further simplification were given, which can be applicable to arbitrary trajectories. As  $k_t = 5\%$  and  $k_p = 1\%$ , about 61% of terms in  $M(q)$  and 76% of terms in  $H(q, \dot{q})$  can be removed while the global RMS errors for the  $1 \times 10^5$  test points are less than 5%. Then, cases studies were carried out on three different test trajectories including the identification trajectory, the linear and the circular FSW trajectories. For the FSW trajectories, about 32% of terms can be reduced with a good precision of 3%. Comparisons between the simplified and original components of  $M(q)$ ,  $H(q, \dot{q})$  and  $T$  were implemented and the simulation results verified the effectiveness and good performance of the simplification method. Moreover, the torques in robot dynamic equation were analyzed in the case of linear FSW process as well. The proposed simplification algorithm can be generalized to other dynamic models (e.g., planes and helicopters), and the simplified model can be applied to many control methods so as to simplify the control structure and improve the real-time control performance, such as the computed torque control and the observer-based control.
- d. Negligible inertia parameters of the robot model were found by using the proposed simplification algorithm for the robot dynamic model.** Based on the proposed simplification algorithm for the robot dynamic model, the negligible inertia parameters of the robot model were found, which was extremely significant in trajectory optimization for the robot model identifica-

tion. The identified inertia parameters will be more precise when the negligible inertia parameters can be removed. Meanwhile, when adding a new axis to the existing industrial robot, the influence of the new axis parameters can be analyzed by using the simplification method, which will be extremely beneficial for designing the new robot.

- e. **Dynamic modeling and identification of robotic FSW process were realized without the sensors for measuring tool position directly.** The dynamic modeling and identification of the robotic FSW process were realized without the sensors which can directly measure the tool position of the robot. First of all, a brief description of the FSW process including its three main stages and process forces was given, followed by the presentation of two static nonlinear power models of the process forces. Then, the axial force of FSW process can be modeled by using the plunge depth and its derivative. However, the plunge depth can not be directly obtained by the robot measurements since the actual tool position is different from the commanded one during the FSW process due to the robot flexibility. With the help of the robotic FSW system, the experimental tests for the identification were implemented and three methods for calculating the plunge depth were proposed based on the joint flexibility model and the nonlinear observer presented by [4, 71, 72, 73] which can be used to estimate the actual joint angular position so as to obtain the required plunge depth. By means of data filtering, the parameters of linear and nonlinear dynamic axial force process models were identified according to the least squares method. Finally, systematic studies on the errors of identified models were carried out and the comparisons between the filtered and fitted axial forces were performed. As a result, acceptable accuracy of the identified models can be seen based on the RMS errors, the uncertainties of identified parameters and the normality tests, demonstrating good performance of the identification. Note that the identified models can be used for establishing the simulator of robotic FSW process, besides, it also contributes to the design of the robot force controller and the stability analysis of the control system.
- f. **Robust force controllers were designed for the robotic FSW process based on the frequency response approach.** On the basis of the modeling of robotic FSW process which simultaneously considers the complete kinematics, the position response of the industrial robot controller (rigid robot displacement model), the joint flexibility and the dynamic axial force process model, the robust force controllers were designed by using the systematic design procedure [95] based on the frequency response approach. To begin with, the inner position/outer force control strategy was selected for the robot system since the industrial robot KUKA KR500-2MT can be controlled only by using its positional interface. Then, the parameter identification of the rigid robot displacement model was performed based on the algorithms of transfer function estimation and process model estimation provided by MATLAB System Identification Toolbox. Finally, the force controllers were designed with high gains and sufficient phase margins for the closed-loop system, implying the robustness of the control system.



- g. Simulator of robotic FSW process was developed and a disturbance model concerning axial force vibration was proposed.** For the purpose of simulating the robotic FSW process, the corresponding simulator was developed by MATLAB/SIMULINK and simulation results proved good performances of the designed force controllers. In addition, the vibration of axial force during the robotic FSW process was analyzed by the identification of an experimental axial force model [96]. Based on the height measurements of the weld surfaces, a disturbance model of zero force vertical position which can lead to the oscillation of axial force was proposed and used in the simulation. Simulation results verified the effectiveness of the disturbance model and good performances of the force controllers.

## 6.2 Perspectives

On the basis of the research in this thesis, several problems concerning modeling and robust force control of industrial robot manipulators used in FSW process have been solved. However, several possible interesting research directions for the future work should be further carried out in the following aspects:

- The negligible inertia parameters from the dynamic model simplification can be used in trajectory optimization for the robot model identification. Besides, the simplification method can also be used to analyze the influence of the new axis parameters when adding a new axis to the existing industrial robot.
- The obtained dynamic axial force process models are not accurate since model parameters have been identified by using the calculated plunge depth based on the joint flexibility model and the nonlinear observer. To improve the identification accuracy, it is necessary to use the additional sensors which can directly measure the tool position of the industrial robot in Cartesian space.
- For obtaining good weld quality, although the contribution of the axial force  $F_z$  is more important than the travel force  $F_x$  and the lateral force  $F_y$ , the additional force sensors can be applied so as to obtain the models of process forces in other directions. As a consequence, the force control in other directions can be also realized other than the current axial force control.
- The designed force controllers have already been demonstrated by the simulations, and the corresponding experimental tests should be further carried out to validate the effectiveness and good performances of the force controllers.
- Toward the axial force vibration in FSW process, a new force control system can be designed to attenuate the force vibration, improving the weld quality.
- The experimental validations of the observer-based compensation have been implemented just for the linear and circular FSW trajectories in a plane and only the positioning error in the perpendicular direction to the welding trajectory has been corrected [4]. The experiments can be performed for more complicated welding trajectories in three-dimensional space and the corrections for other directions and orientations can be used as well.

## Part II

# Résumé Étendu en Français



# Chapter 7

## Résumé en Français: Modélisation d'un robot manipulateur en vue de la commande robuste en force utilisé en soudage FSW

### 7.1 Introduction générale et revue de littérature

#### Contexte et motivation

Le soudage par frottement malaxage ou Friction Stir Welding (FSW) en anglais a été de plus en plus appliqué dans une grande variété d'applications industrielles dans les années récentes, particulièrement dans l'industrie aéronautique et astronautique, ce type de soudage est massivement utilisée en vue de la réduction de poids. Cette technologie de soudage est comparativement émergente et innovante et elle a été inventée et développée par l'organisme TWI (The Welding Institute, Cambridge, le Royaume-Uni) en 1991 [1]. Durant le procédé de soudage FSW, la chaleur générée n'est toujours pas suffisante pour fondre le matériau. En conséquence, le point de fusion du matériau n'est pas atteint et le soudage est pratiquement effectué à l'état solide du matériau. Pour cette raison, les cordons de soudure produit par le soudage FSW sont communément de bonne qualité et ils n'ont pas les défauts qui sont constamment rencontrés dans les soudages traditionnels [2]. En outre, ce procédé peut être utilisée pour souder des alliages semblables ou dissemblables, des métaux et presque tous les types des alliages d'aluminium. Les autres avantages du soudage FSW comportent: l'efficacité énergétique, le respect de l'environnement, la polyvalence, l'excellente résistance à la soudure et la ductilité, les faibles contraintes résiduelles, pas besoin des soudures métalliques, la vitesse haute de soudage et la répétabilité [3].

Jusqu'à présent, les machines-outils dédiées sont couramment utilisées pour effectuer le procédé FSW. En général, les bonnes précisions peuvent être réalisées à cause de leurs axes extrêmement rigide. Cependant, ces machines ont besoin d'investissements importants parce qu'elles sont spécialement développées. Durant les années récentes, les robots manipulateurs industriels ont été largement appliqués

dans diverses applications industrielles en raison de leur excellente répétabilité, la flexibilité de la production et le faible coût, telles que l'assemblage, la peinture par pulvérisation, la manutention, les travaux de l'usinage et du soudage. Afin de réduire le coût et d'améliorer la productivité, la flexibilité et la qualité, les robots manipulateurs industriels sont utilisés en vue de remplacer ces machines-outils dédiées grâce à leur bon rapport coût-efficacité et leur grand espace de travail. Néanmoins, la robotisation du procédé FSW est encore une tâche extrêmement difficile jusqu'à maintenant parce que la capacité limitée de la force et les rigidités des robots industriels sont des obstacles pour l'expansion du soudage FSW robotisé.

Pendant le procédé de soudage FSW robotisé, parce que la force extérieure exercée sur le robot est extrêmement grande, la flexibilité du robot industriel doit être prise en considération. En conséquence, une déformation non-négligeable du robot est générée et certains problèmes concernant la modélisation et la commande du robot surgissent. Ainsi, un haut degré de précision et de performance en temps réel ne peuvent pas être atteints pour la plupart des robots industriels utilisés dans le procédé FSW. Afin d'assurer une meilleure performance de positionnement, de nombreuses recherches ont été effectuées sur la commande des robots industriels. Dans le travail de recherche présenté par Qin [4], la précision de positionnement peut être améliorée en utilisant le compensateur basé sur un observateur non-linéaire. En raison de la structure compliquée de la commande, beaucoup de calculs doivent être réalisés dans le contrôleur du robot, ce qui influence sa performance en temps réel. Par ailleurs, la commande robuste de la force axiale est également nécessaire pour une bonne qualité de la soudure.

Ce travail de recherche s'inscrit dans le cadre du projet COROUSSO du programme ARPEGE financé par l'Agence Nationale de la Recherche (ANR) en France. Ce projet vise à effectuer la robotisation de l'usinage de pièces de grande dimension en matériaux composites et le soudage FSW de pièces métalliques. Dans cette thèse, l'objectif est de réaliser le procédé FSW robotisé avec de bonnes performances de la précision de positionnement, la commande robuste de force et la commande en temps réel, ce qui peut être développé comme suit:

- 1) Proposition d'une nouvelle méthode pour la simplification des modèles dynamiques des robots en vue d'améliorer la performance en temps réel ainsi que des études de cas relatives;
- 2) Réalisation de la modélisation dynamique et l'identification du procédé de soudage FSW robotisé afin de concevoir le contrôleur de force du robot;
- 3) Conception d'un contrôleur robuste en force pour les robots industriels utilisés pendant le procédé FSW robotisé;
- 4) Développement d'un système de simulation du procédé FSW robotisé et la réalisation des simulations correspondantes dans le but d'analyser la performance du contrôleur de force;
- 5) Analyse des vibrations de la force axiale pendant le procédé FSW robotisé.

## Les robots manipulateurs industriels

Selon la définition officielle de l'organisme RIA (Robot Institute of America en anglais), un robot est un manipulateur reprogrammable et multifonctionnel qui est conçu en vue de déplacer des matériaux, des pièces, des outils ou des dispositifs spécialisés par des mouvements programmés variables pour la réalisation d'une variété de tâches [5]. En outre, le robot industriel est également défini par l'organisme ISO (International Organization for Standardization en anglais) comme un manipulateur commandé automatiquement, reprogrammable et polyvalent. Ce manipulateur est programmable en trois ou plusieurs axes, et il peut être soit fixé sur place soit mobile pour les applications d'automatisation industrielles [6]. Les applications typiques pour les robots industriels comportent le soudage, la peinture, l'assemblage, l'emballage, la palettisation, l'inspection des produits et l'essai, etc.

## Histoire des robots manipulateurs industriels

L'histoire moderne de la manipulation robotique remonte à la fin des années 1940. Afin de protéger les techniciens pour la manipulation de matériaux radioactifs, le premier téléopérateur électrique asservi a été développé en 1947 [7]. Presque en même temps, les scientifiques ont lancé la recherche sur une machine-outil à commande numérique [8]. Ces deux technologies importantes ont jeté une base solide pour les développements de robots manipulateurs industriels. Sur la base du rapport de l'organisme IFR (International Federation of Robotics en anglais), une histoire concise de la technologie de robot industriel est donnée ci-dessous [9].

En 1954, George Devol a demandé le premier brevet de robot appelé "Programmed Article Transfer" en anglais [9]. Sur la base de ce brevet, George Devol et Joe Engelberger ont établi la première société de robot connue sous le nom de Unimation en 1956, et ensuite ils ont développé le premier robot industriel au monde appelé Unimate en 1959. Deux ans plus tard, le robot à commande hydraulique Unimate a été installé dans une usine automobile de la société GM (General Motors en anglais) à New Jersey pour la première fois afin d'extraire des pièces coulées sous pression [8]. En 1962, le premier robot cylindrique Versatran a été utilisé dans une usine de la société Ford (Canton aux États-Unis) par l'organisme AMF (American Machine and Foundry en anglais). Le Stanford Arm a été inventé par Victor Scheinman en 1969 comme le premier bras robotique à six axes, électrique et commandé par l'ordinateur. Ce bras robotique est rapidement devenu une norme pour la conception de robots industriels à ce moment-là [10].

Les robots industriels ont été développés pour quelques nouvelles applications également: La société américaine GM a installé le premier robot de soudage par points dans une usine d'assemblage; l'entreprise norvégienne Trallfa a développé le premier robot commercial pour la peinture par pulvérisation pendant une pénurie de main-d'œuvre domestique; la société japonaise Hitachi a lancé le robot de boulonnage automatique qui était le premier robot industriel avec des capteurs de vision dynamiques dans le but de déplacer des objets; une autre société japonaise Kawasaki a développé le premier robot de soudage à l'arc électrique Hi-T-Hand. En 1973, l'entreprise allemande KUKA a développé son propre robot Famulus qui était le premier robot industriel à six axes électromécanique. La société Cincinnati Mi-

lacron a introduit le robot T3 sur le marché également, et il était le premier robot industriel commandé par le mini-ordinateur (voir la Figure 1.1).

En 1974, l'entreprise suédoise ASEA a développé le robot industriel IRB-6 qui est entièrement électrique et contrôlé par le microprocesseur (Intel 8 bit). Ce robot a été délivré à une société d'ingénierie mécanique en Suède pour polir des pipes. En Italie, le robot Olivetti SIGMA a premièrement été utilisé pour les opérations d'assemblage grâce à ses deux mains. En 1978, la société américaine Unimation/Vicarm a créé le robot PUMA (Programmable Universal Machine for Assembly en anglais) à l'aide de la société GM. Dans la même année, le robot SCARA (Selective Compliance Assembly Robot Arm en anglais) a été conçu par Hiroshi Makino à l'Université de Yamanashi au Japon. Le robot est légèrement flexible dans la direction X-Y mais rigide dans la direction Z à l'aide de sa disposition des articulations à axes parallèles. l'entreprise allemande Reis Robotics a présenté le premier robot à six axes avec son propre système de contrôle pour le chargement et le déchargement des pièces coulées sous pression dans des cisailles à molettes. En 1979, la société japonaise Nachi a développé le premier robot motorisé pour le soudage par points.

L'industrie de robot industriel a commencé sa croissance rapide à partir de 1980. Le premier bras robotique à entraînement direct au monde a été construit par Takeo Kanade à l'Université Carnegie Mellon en 1981. Les moteurs ont été installés directement dans les articulations du robot. En raison de la suppression des mécanismes de transmission entre les moteurs et les charges, le bras robotique à entraînement direct peut se déplacer facilement et librement. Par conséquent, il est plus rapide et beaucoup plus précis que les robots précédents. Dans la même année, l'entreprise Par Systems a lancé son premier robot industriel à portique qui peut fournir une plus grande amplitude de mouvement que les robots à piédestaux. En 1984, la société Adept Technology a réussi à commercialiser le premier robot SCARA à entraînement direct AdeptOne. Ce robot est non seulement extrêmement robuste pendant les procédés de fabrication, mais encore il peut marcher avec un degré élevé de précision grâce à un mécanisme simple (voir la Figure 1.2).

Dans les années 1980, le robot Delta (connu comme un type de robot parallèle) a été inventé par Reymond Clavel à l'Institut Fédéral de Technologie de Lausanne (EPFL, Suisse), afin de satisfaire les besoins industriels de manipuler quelques objets petits et légers à très grande vitesse [11]. Après l'achat d'une licence de fabrication de robot Delta, l'entreprise suisse Demarex a vendu son premier robot Delta dans le marché de l'emballage en 1992. Conformément au robot Delta et à la technologie d'image, la société ABB a développé le plus rapide robot de cueillette FlexPicker au monde en 1998. Le robot peut ramasser 120 objets par minute ou ramasser et libérer les objets à une vitesse de 10  $m/s$ . En 1999, l'entreprise Reis Robotics a lancé le robot à laser RV6L-CO2 en utilisant le guidage du faisceau laser intégré dans le bras robotique, et l'utilisateur du robot est autorisé à opérer le dispositif laser à haute dynamique.

Le système de commande du robot manipulateur industriel a également connu un grand développement. En 2004, la société japonaise Motoman a développé le système de commande amélioré NX100 pour ses robots commerciaux. Ce nouveau système de commande est capable de commander quatre robots de manière synchronisée (jusqu'à 38 axes). Deux ans plus tard, la société italienne Comau a lancé le

WiTP (Wireless Teach Pendant en anglais) pour la première fois, ce qui permet la communication des données du robot sans la connexion par câble.

En 2006, l'entreprise KUKA a premièrement présenté le robot LWR (Light Weight Robot en anglais). Le poids du robot est seulement de 16 *kg* mais sa capacité de charge utile est de 7 *kg*. Par ailleurs, ce robot portable écoénergétique permet d'effectuer une grande variété de tâches. Un nouveau robot LCR (Learning Control Robot en anglais) a été lancé par la société japonaise FANUC en 2010. Ce robot peut apprendre ses caractéristiques vibratoires avec l'aide de la méthode de commande LVC (Learning Vibration Control en anglais). Le robot peut également atténuer les vibrations dans le but de réduire le temps de cycle pour le mouvement du robot (voir la Figure 1.3).

### Développements actuels des robots manipulateurs industriels

Selon le rapport annuel de l'organisme IFR (appelé World Robotics en anglais) en 2014, le stock total des robots industriels opérationnels a atteint au moins 1332000 unités à la fin de 2013 dans le monde. Environ 52% des robots industriels sont en Asie, 29% en Europe, et 17% en Amérique du Nord. En outre, ce rapport estime que les installations mondiales des robots industriels vont atteindre 1946000 unités à la fin de 2017 [12].

Les installations annuelles mondiales des robots industriels de 2003 à 2013 sont résumées à la Figure 1.4. En raison de la situation économique mondiale améliorée, les ventes des robots en 2013 ont augmenté de 12% jusqu'à 178132 unités, ce qui est le plus grand nombre de robots industriels jamais vendu. En 2013, la plupart des ventes de robots (environ 70%) étaient dans cinq pays: la Chine, le Japon, les États-Unis, la Corée du Sud et l'Allemagne. Il convient de noter que la Chine est devenue le marché du robot qui est le plus grand et connaît la croissance la plus rapide dans le monde, avec une part de 20% de l'approvisionnement total. Les croissances fortes ont été réalisées pour les ventes de robots manipulateurs à presque toutes les industries, y compris l'industrie automobile, électronique, métallurgique, plastique et caoutchouc ainsi que l'industrie alimentaire.

La valeur du marché mondial du robot industriel a atteint 9.5 milliards de dollars US en 2013. Le chiffre d'affaires des systèmes de robot a été estimé 29 milliards de dollars US, y compris le coût des logiciels, des périphériques et de l'ingénierie des systèmes. La plupart des robots industriels dans le monde sont conçus et fabriqués par plusieurs sociétés robotiques célèbres et influentes, telles que ABB Robotics, FANUC Robotics, KUKA Robotics, Yaskawa Electric et Kawasaki Robotics, etc.

La Figure 1.5 montre les robots manipulateurs industriels lourds avec des capacités de charge élevées, et ils peuvent effectuer une grande variété des procédés de fabrication. Les informations sur les diverses robots manipulateurs industriels lancés par les quatre sociétés se trouvent dans [13, 14, 15, 16].

### Modélisation de robots industriels à articulations flexibles

Dans la plupart des cas, les robots manipulateurs lourds comme les robots utilisés dans cette étude sont souvent considérés comme rigides, parce que la force exercée sur



le robot pendant le procédé de fabrication n'est pas très grande. En conséquence, la déformation du robot à cause de sa flexibilité peut être négligée. Cependant, la flexibilité du robot doit être prise en compte lorsque le robot effectue certaines applications qui ont besoin d'une force de travail importante.

En ce qui concerne les manipulateurs industriels, on peut supposer que la flexibilité est concentrée au niveau des articulations ou distribuée le long des corps de différentes manières du point de vue de la modélisation [7]. En réalité, ces deux types de flexibilités existent en même temps. Conformément aux travaux de recherche de Dumas [17], les résultats montrent que les sources principales des flexibilités pour les robots industriels sériels sont les flexibilités localisées aux articulations (y compris les moteurs et les transmissions). Selon l'étude de Dumas, les corps des robots industriels utilisés dans cette recherche doivent être considérés comme rigides. Les problèmes sur la modélisation et la commande des robots manipulateurs industriels à articulations flexibles seront principalement concernés ci-après.

En effet, les flexibilités des articulations des robots industriels actuels viennent de certains éléments dans la transmission de mouvement ou dans la réduction, tels que les courroies, les câbles, les longues tiges, les réducteurs à entraînement harmonique et les trains d'engrenages épicycloïde. Ces composants sont utilisés en vue de placer les actionneurs au plus près de la base du robot, ou de garantir des taux de réduction élevés, ou d'améliorer l'efficacité dynamique. Toutefois, lorsqu'ils sont soumis à une force de travail importante pendant le procédé de fabrication robotisée, ces composants deviennent intrinsèquement flexibles [7]. Donc, de nombreuses recherches ont été faites sur la modélisation de l'articulation flexible du robot industriel.

Les études précédentes des problèmes causés par la présence de transmissions flexibles dans les robots manipulateurs peuvent être trouvées dans [18, 19], avec les premiers résultats expérimentaux sur le bras robotique appelé GE P-50. En 1987, Spong a proposé un modèle simplifié pour l'articulation flexible du robot industriel. Il considérait la flexibilité de l'articulation comme un ressort de torsion linéaire avec une constante de rigidité dans le modèle pour la première fois [20, 21]. Sur la base du modèle simplifié de Spong, Khorasani a établi un modèle à seul corps, rigide-flexible et couplé. Dans ce nouveau modèle, l'influence du terme d'amortissement et l'engrenage harmonique est également prise en compte [22].

Le modèle simplifié pour l'articulation flexible proposée par Spong est valable uniquement dans le cas de l'utilisation de grande démultiplication, parce que les effets d'inertie croisés pourraient être ignorés dans cette situation. En conséquence, Tomei a proposé un modèle complet pour l'articulation flexible en considérant les couplages entre les corps et les moteurs dans le robot [23].

La précision d'un modèle de robot peut être grandement influencée par les caractéristiques non-linéaires de la transmission à engrenages, ce qui contiennent principalement le frottement et la rigidité non-linéaire. Il faut noter que la rigidité non-linéaire peut également être comprise dans le modèle pour l'articulation flexible si on remplace la formulation de rigidité linéaire par une fonction non-linéaire qui représente un ressort non-linéaire. Plus de non-linéarités peuvent être ajoutées au modèle dynamique pour l'articulation flexible, telles que le jeu, l'hystérésis et l'amortissement non-linéaire [24]. Un modèle dynamique relativement précis pour le robot à articulation flexible a été proposé par Bridges, et la stabilité de ce nou-

veau modèle a également été prouvée. Dans le modèle, le frottement, la flexibilité non-linéaire, et les erreurs cinématiques sont pris en considération [25].

Conformément aux études précédentes, Moberg a proposé un modèle étendu pour l'articulation flexible. Dans ce modèle, l'élasticité est décrite par un certain nombre de paires ressort-amortisseur multidimensionnelles localisées [24]. Cette paire ressort-amortisseur est capable d'avoir jusqu'à six degrés de liberté et il comporte non seulement la déviation de translation mais aussi la déviation de rotation.

## Commande de robots manipulateurs à articulations flexibles

Généralement, les lois de commande le plus souvent utilisées sont: linéaires ou non-linéaires, statiques ou dynamiques, par rétroaction (feedback) ou par anticipation (feedforward), à temps discret ou à temps continu, robustes ou adaptatives, diagonales ou matrices complètes, etc.

Dans cette recherche, le robot manipulateur industriel à plusieurs axes doit être commandé en vue d'effectuer un procédé de fabrication précis, ce qui est une tâche difficile. Le robot manipulateur est un système multi-corps élastiques, à plusieurs variables fortement couplées, et sa dynamique non-linéaire change rapidement. Par ailleurs, pour les robots à articulations flexibles, les non-linéarités telles que le jeu, l'hystérésis, le frottement et l'élasticité non-linéaire devraient être pris en compte dans la commande également. Selon les études précédentes, les principaux concepts utilisés pour la synthèse de loi de commande de robots à articulations flexibles sont donnés ci-dessous.

Afin d'obtenir la bonne performance dans le domaine de la commande de robot, les premiers chercheurs ont concentré leurs efforts sur l'étude de l'influence de la dynamique du moteur. Cependant, en 1985, Sweet et Good ont souligné que la flexibilité de l'articulation du robot doit être prise en considération dans la modélisation et la commande du robot si une haute performance de suivi est demandée [18]. En 1987, Spong a proposé un modèle simplifié pour l'articulation flexible du robot [20]. Après cela, un grand nombre de recherches théoriques et expérimentales ont été effectuées sur la commande de robots à articulations flexibles: la perturbation singulière et la variété intégrale, la linéarisation par rétroaction, le système en cascade et la commande intégral backstepping, la commande PD, la commande adaptative, la commande robuste, la commande par réseaux de neurones, la commande floue et quelques autres méthodes de commande.

### Perturbation singulière et variété intégrale

Les robots manipulateurs industriels à articulations flexibles peuvent être considérés comme des systèmes à élasticité faible si leur rigidité articulaire est assez grande. Cette caractéristique permet que nous transformons le modèle de l'articulation flexible en un système à deux échelles de temps en utilisant la technique de perturbation singulière (ou singular perturbation technique pour SPT en anglais). Les deux échelles de temps comportent une échelle de temps rapide (ou fast time scale pour FTS en anglais) et une échelle de temps lente (ou slow time scale pour STS en anglais). Dans la dynamique du système, la technique SPT est largement appliquée

dans le système qui peut être divisé en un sous-système rapide et en un sous-système lent. Grâce à l'avantage de la réduction de modèle, cette technique est capable de décomposer un système d'ordre supérieur en deux systèmes d'ordre inférieur. L'idée essentielle de la technique SPT est basée sur l'ajout d'un terme de correction simple à la loi de commande pour des robots à corps rigides afin d'atténuer l'oscillation élastique à l'articulation du robot. Néanmoins, cette méthode peut être uniquement utilisée dans les systèmes robotiques avec les flexibilités articulaires faibles, où la dynamique des articulations flexibles est beaucoup plus rapide que la dynamique des corps rigides.

Le modèle de perturbation singulière pour le même système n'est pas unique. Spong a présenté une conception de commande qui modélise les forces élastiques d'articulation comme les variables rapides et les variables de corps comme les variables lentes [21]. Ge a proposé un nouveau contrôleur adaptatif basé sur la technique SPT, ce qui utilise seulement la rétroaction de la position et de la vitesse. Dans ce contrôleur, les erreurs de suivi du moteur est modélisées comme les variables rapides et les variables de corps est modélisées comme les variables lentes [26]. En conséquence, la dynamique de l'échelle de temps lente et les lois de commande résultantes sont également différentes de celles données par Spong.

Cependant, les systèmes dynamiques obtenus par la décomposition à l'aide de la technique SPT sont extrêmement compliqués à contrôler. Pour cette raison, il est inévitable de faire une réduction d'ordre et une simplification. Afin de résoudre ce problème, le concept de variété intégrale a été introduit par certains chercheurs. La variété intégrale est un outil utile pour la modélisation d'ordre réduit. Dans la recherche de Khorasani, un modèle lent de  $2n^{eme}$  ordre permet de représenter exactement un système de  $4n^{eme}$  ordre si les trajectoires du système sont sur la variété [27]. Ce fait sera utilisé pour construire des modèles corrigés d'ordre réduit. En utilisant cette idée pour représenter la dynamique du sous-système lent, Spong a obtenu un modèle d'ordre réduit pour le manipulateur. Ce modèle incorpore les effets des flexibilité des articulations sur le système [21].

Après avoir obtenu le système d'ordre réduit, plusieurs autres méthodes de commande peuvent être utilisées directement, telles que la commande robuste [28], la commande floue [29], etc. Ghorbel and Spong ont premièrement proposé une stratégie de commande composite adaptative pour les robots manipulateurs à articulations flexibles. Cette stratégie de commande est également appelé la stratégie de commande rapide/lente qui consiste en un contrôleur lent adaptatif pour le robot rigide ainsi que un contrôleur rapide pour atténuer l'oscillation élastique à l'articulation du robot [30]. La stabilité locale de cette stratégie de commande a également été analysé dans [31]. Par rapport à la méthode de linéarisation par rétroaction, cette méthode basée sur la variété intégrale ne nécessite pas la rétroaction d'accélération et de jerk du corps. Toutefois, il faut aussi noter que cette méthode manque de robustesse vis à vis de l'incertitude paramétrique. Ensuite ces deux chercheurs ont étendu la méthode de variété intégrale du cas de paramètre connu au cas adaptatif [32]. Les inconvénients évidents de cette variété intégrale adaptative est la complexité sur l'établissement de l'expression et le fardeau de calcul.

Certains chercheurs ont également étudié la question sur la robustesse de la méthode de commande basée sur la technique de perturbation singulière et de variété

intégrale. En prenant la dynamique non-modélisée et les vibrations des paramètres en compte, Khrasani a corrigé les équations perturbés, ensuite il a obtenu un modèle flexible d'ordre réduit sans l'incertitude paramétrique. En conséquence, la linéarisation par rétroaction et les stratégies de commande adaptatives robustes peuvent être appliquées à la commande des robots à articulations flexibles [33]. Al-Ashoor a développé un contrôleur adaptatif robuste qui est conçu afin de compenser les effets de la dynamique non-modélisée et les erreurs causé par les vibrations des paramètres, et de négliger les effets d'ordre supérieur de la flexibilité articulaire [34].

La technique SPT a l'avantage de la réduction de l'ordre et elle peut aussi utiliser l'algorithme de commande adaptative pour obtenir des paramètres plus précis. Néanmoins, il est difficile de trouver une loi de commande adaptative pour un système non-linéaire. Le fardeau de calcul est extrêmement élevé, en outre, cette méthode est limitée par la rigidité de l'articulation. Afin de résoudre ce problème, Liu a conçu un compensateur de flexibilité pour l'articulation du robot qui peut considérablement augmenter la rigidité articulaire équivalente. Par conséquent, cette méthodologie peut être étendue au système de robot avec la flexibilité articulaire normale [35]. Mais son universalité ne peut être prouvée parce qu'il a uniquement réalisé ce contrôleur sur une seule articulation.

### **Linéarisation par rétroaction**

La linéarisation par rétroaction est considérée comme une meilleure méthode pour commander un système non-linéaire. L'idée fondamentale de cette méthode est de transformer le système non-linéaire original en un système linéaire équivalent en utilisant la transformation non-linéaire appropriée de l'état et la loi de commande correcte, afin d'appliquer commodément les méthodes de commande pour le système linéaire à ce système linéaire équivalent. D'un point de vue théorique, cette méthode est considérablement attrayante parce qu'il est facile de démontrer la stabilité du système, c'est-à-dire, le système non-linéaire original est stable si la stabilité de son système linéaire équivalent est prouvée. Cette méthode est largement utilisée dans la commande des robots rigides, tels que la commande par couple calculé et la commande par dynamique inverse. C'est parce que les modèles des robots rigides peuvent toujours linéarisés par rétroaction. Cependant, certains modèles dynamiques des robots à articulations flexibles ne peuvent pas satisfaire les exigences de la linéarisation par rétroaction.

Spong a proposé un modèle simplifié pour l'articulation flexible où la flexibilité articulaire est considérée comme un ressort de torsion linéaire et l'énergie cinétique du rotor est principalement due à sa propre rotation. Il a également souligné que ce système de modèle simplifié peut être globalement linéarisé par la transformation non-linéaire de coordonnées et la rétroaction d'état statique [20]. Concernant le modèle simplifié pour l'articulation flexible, Grimm a effectué une analyse de robustesse sur les boucles de commande qui sont linéarisées par rétroaction ou découplées non-linéairement [36]. Sira-Ramirez et Spong ont proposé une commande à boucle externe basée sur la théorie des modes glissants des systèmes de structure variables pour le suivi robuste [37]. Ge a conçu un contrôleur de réseaux de neurones adaptatif robuste pour les robots à articulations flexibles en utilisant la technique de

linéarisation par rétroaction. Il convient de noter que ce contrôleur pourrait mieux gérer les changements des modèles dynamiques et les incertitudes des paramètres que les contrôleurs conventionnels de linéarisation en rétroaction [38]. En ce qui concerne le suivi de trajectoire pour les robots à articulations flexibles, Ider a proposé une loi de commande basée sur la résolution des équations dynamiques inverses au niveau de l'accélération. Ces équations sont singulières, et la méthode implicite d'intégration numérique a été utilisée en vue d'éviter les différenciations des équations de mouvement et de tâches [39].

Concernant le modèle dynamique complet des robots à articulations flexibles, il ne peut être linéarisé par seulement la rétroaction d'état statique. Toutefois, De Luca a prouvé que le modèle dynamique complet peut toujours être transformé entièrement en un système linéaire, découplés, contrôlable et d'entrée-sortie par la rétroaction d'état dynamique non-linéaire si le système est réversible sans la dynamique de zéro [40]. En outre, il a présenté un algorithme d'inversion général pour la synthèse d'une loi de rétroaction d'état dynamique qui donne le découplage d'entrée-sortie et la linéarisation d'état complet [41].

La loi de commande de linéarisation par rétroaction nécessite les termes différentiels d'ordre supérieur, tels que l'accélération et le jerk. Cela signifie que certains états doivent être estimés à partir des mesures disponibles, par exemple, le jerk. Par conséquent, on doit construire quelques observateurs appropriés dans le but d'effectuer l'estimation des états.

En conclusion, pour la commande de linéarisation par rétroaction, les performances de suivi dépendent fortement de la précision de modèle du système. En même temps, un observateur linéaire ou non-linéaire doit être construit parce que tous les états pour le calcul de rétroaction doivent être obtenus. Pour cette raison, le système devient plus complexe. En conséquence, une méthode est de combiner la commande de linéarisation par rétroaction avec la commande adaptative ou la commande par réseaux de neurones, afin de réduire l'exigence pour la précision de modèle. Une autre méthode est d'utiliser le contrôleur sans la demande pour la rétroaction de tous les états.

### **Système en cascade et la commande intégrale backstepping**

La méthode de système en cascade suppose premièrement que la dynamique relative aux manipulateurs rigides peut être commandée directement. En second lieu, elle décide la loi de commande pour l'actionneur en prenant les flexibilités des articulations en considération. En utilisant cette méthode, le modèle dynamique d'articulation flexible du robot est exprimé sous la forme de deux boucles en cascade: la boucle du manipulateur et la boucle de l'actionneur qui concerne la flexibilité de l'articulation. Une force désirée qui est introduite comme un signal synthétique d'entrée pour la boucle du manipulateur, est générée en raison de la flexibilité entre le moteur et le corps. Et elle doit être réalisée par la sortie de la boucle de l'actionneur avec l'aide de la conception d'un contrôleur approprié. Par rapport à la technique de perturbation singulière, cette méthode n'est pas soumise à des restrictions sur la grandeur de la flexibilité articulaire.

La méthode de système en cascade dépend d'un meilleur modèle du système,

par conséquent, il nécessite habituellement une combinaison de la loi de commande adaptative. Yuan a développé un contrôleur adaptatif composite pour les robots à articulations flexibles sans aucune restrictions sur les rigidités des articulations en utilisant cette méthode [42]. Colbaugh et Glass ont proposé plusieurs contrôleurs adaptatifs dans l'espace de tâche afin de résoudre les problèmes sur la commande de mouvement et la commande de flexibilité [43].

La technique de commande intégral backstepping a récemment été développée pour les systèmes en cascade. Cette méthode systématique est réalisée par la conception d'une fonction de Lyapunov pour le système dont le côté de l'entrée comporte une série d'intégrateurs. En conséquence, elle peut être considérée une méthode importante pour la conception théorique parce que la méthode peut fournir la démonstration de la stabilité de Lyapunov.

Concernant le système en cascade, les trois améliorations obtenues à partir de la technique de commande intégral backstepping sont: (1) la capacité pour construire des contrôleurs globalement stables en vue de compenser l'incertitude paramétrique, (2) l'élimination des restrictions sur la grandeur de la flexibilité articulaire, et (3) l'élimination de l'exigence des mesures des accélérations et des jerks pour les corps du robot. Cependant, toutes les améliorations ci-dessus ne sont pas pleinement exploités à la fois. Par exemple, Chen et Fu ont développé un contrôleur qui permet d'éliminer les restrictions sur la grandeur de la flexibilité articulaire, mais ce contrôleur permet de compenser seulement l'incertitude paramétrique dans la mesure de la démonstration de la stabilité locale. Par ailleurs, les mesures d'accélérations de corps sont nécessaires, mais elles sont impossibles pour la plupart des robots industriels [44]. Beneallegue et Sirdi ont utilisé une méthode de passivité afin de concevoir un contrôleur adaptatif globalement asymptotiquement stable, néanmoins, il faut les mesures pour à la fois les accélérations des corps et les jerks des corps [45].

En utilisant une méthode adaptative avec un suivi asymptotique de position comme objectif, les méthodes ci-dessus ont géré l'incertitude paramétrique. Toutefois, des difficultés ont été éprouvées parce que les mesures des accélérations et des jerks pour les corps sont exigées. Dawson a conçu un contrôleur robuste basé sur une méthode de backstepping afin de compenser les incertitudes paramétriques et les perturbations bornées inconnues. En outre, ce contrôleur permet de réaliser une erreur globalement uniformément finalement bornée pour le suivi de la position [46]. Nicosia et Tomei ont fait un pas de plus pour concevoir un contrôleur adaptatif qui peut satisfaire simultanément les trois améliorations ci-dessus, mais il est spécialement développé pour un robot à un seul corps à articulation flexible [47]. Ainsi, la commande des robots générales de  $n$ -corps à articulation flexible est encore un problème. Après cela, Lozano et Brogliato ont résolu ce problème et ils ont conçu un système adaptatif de commande. Ce système permet de réaliser le suivi de position globalement asymptotique, et il peut satisfaire également les trois améliorations [48]. Concernant leurs méthodes, la matrice d'inertie inverse est nécessaire pour éliminer les mesures de l'accélération de corps. Donc le temps de calcul et la complexité sont augmentés.

Kwan et Lewis ont proposé un contrôleur de backstepping robuste utilisant des réseaux de neurones [48]. En comparaison avec la commande de backstepping adaptative, la linéarité en paramètres inconnus n'est pas nécessaire. Par conséquent, le

problème principal sur la commande de backstepping est corrigé et les calculs fatigants de "matrices de régression" ne sont pas indispensables. Cependant, la plupart des réseaux de neurones exigent un temps long d'entraînement qui n'est pas pratique pour la commande en temps réel du robot. Avec l'aide d'un relativement petit réseau de neurones, Macnab et D'Eleuterio ont proposé une méthode de commande adaptative par réseaux de neurones [49]. Cette commande est plus pratique à effectuer en raison de la réduction du temps d'entraînement et de la charge de calcul en ligne.

Bien que la technologie intégral backstepping n'est pas soumise à des restrictions sur la grandeur de la flexibilité articulaire, elle a encore besoin de la recherche plus complète. La technologie contient généralement une loi de commande adaptative parce qu'elle est sensible aux paramètres du modèle. Un autre problème concernant la commande de backstepping est que l'équation construite de Lyapunov est limitée, et elle n'a pas un paramètre pour régler directement la performance du système. Le seul avantage de cette commande est la garantie de la stabilité du système. Plus d'éléments peuvent être trouvés dans un article de la méthode de backstepping pour les robots à articulations flexibles [50].

## **Commande PD**

Les contrôleurs PD (ou PID) traditionnels sont largement utilisés dans l'ingénierie pratique en raison de ses simplicités et ses significations physiques claires ainsi que ses fortes robustesses et hautes précisions de commande. En ingénierie, un contrôleur PD modifié avec une compensation de flexibilité articulaire peut être utilisé dans la commande de robots manipulateurs industriels.

Un contrôleur PID simple suffit pour stabiliser toutes sortes de robots rigides autour d'une position de référence si les forces gravitationnelles sont compensées. Bien que les robots rigides sont exactement linéarisable par la rétroaction de l'état statique, la stabilité du contrôleur PD ne compte pas sur cette propriété et elle dépend plutôt des propriétés de passivité intrinsèque. Tomei a proposé un contrôleur PD simple et il a démontré sa stabilité asymptotique globale [23]. Albu-Shaffer and Hirzinger ont développé un contrôleur PD de rétroaction de l'état articulaire avec les compensations pour la gravité et le frottement non-linéaire, ce qui peut être progressivement généralisé en vue de tenir compte de la dynamique complète du robot. La stabilité asymptotique globale a été prouvée sur la base de la première méthode de Lyapunov, et l'efficacité du contrôleur a été validée par des expériences sur un manipulateur léger à 7 degrés de liberté [51].

Bien que la commande PD est simple et utile, les flexibilités des articulations doivent être prise en compte dans la modélisation du robot afin d'améliorer la précision de commande.

## **Commande adaptative et commande robuste**

En plus des méthodes de commande mentionnées ci-dessus, la commande adaptative, la commande robuste, la commande par réseaux de neurones, la commande floue, la commande par modes glissants [37], etc., sont également appliqués à la commande de robots manipulateurs industriels.

La commande adaptative et la commande robuste sont deux stratégies fondamentales de commande pour résoudre les problèmes sur les incertitudes robotiques, telles que les incertitudes des paramètres de modèle, les charges inconnues ou variables, les flexibilités articulaires, les coefficients de frottement variables dans le temps, etc. Lorsque les paramètres du système contrôlé changent, la commande adaptative [26, 30, 32, 33, 34, 38, 42, 43, 45, 47, 49, 52, 53, 54] peut atteindre certaines performances par l'identification, l'apprentissage et le réglage de la loi de commande. Toutefois, elle ne peut pas garantir la stabilité du système quand l'exigence stricte en temps réel est proposée. En outre, la réalisation de cette commande est complexe, particulièrement lorsque l'incertitude non-paramétrique existe. Cependant, la commande robuste [28, 33, 34, 37, 38, 46, 48, 55, 56, 57] peut assurer la stabilité du système, et elle peut également maintenir une meilleure performance quand les incertitudes sont dans une certaine limite.

### **Commande par réseaux de neurones et commande floue**

Étant donné que le modèle précis du robot à articulation flexible ne peut pas être obtenu, la commande par réseaux de neurones [38, 48, 49, 58] est proposée en vue de résoudre ce problème avec l'aide de l'apprentissage des informations de modèle inconnu et de l'entraînement du réseau nerveux. Par conséquent, cette méthode de commande permet l'approximation de la dynamique du système.

Par contre, la commande floue [29] ne nécessite pas un modèle de système précis pour la conception du contrôleur de robot. Généralement, cette méthodologie a besoin de l'expérience de l'utilisateur ou de l'expert, les informations et les données opérationnelles, etc. Pour les avantages, il est commode d'obtenir la stratégie de commande pour les objets non-linéaires et incertains, de plus elle a une bonne performance en temps réel grâce à son algorithme simple.

### **Autres méthodes de commande**

Concernant les tâches robotiques typiques (par exemple, l'usinage et le procédé FSW) qui nécessitent une interaction avec l'environnement, les forces de contact doivent être proprement traitées par le contrôleur du robot manipulateur. Dans ces cas, un pur contrôleur de mouvement donne souvent de mauvaises performances et il peut même susciter l'instabilité du système. Chiaverini et al. ont proposé un système stable de commande en force/position pour les robots en contact avec une surface flexible et élastique, et il a été testé sur les robots manipulateurs industriels COMAU SMART 6.10R [59]. En ce qui concerne les robots manipulateurs en contact dynamique avec l'environnement, De Luca a également conçu un système de commande qui est hybride et robuste pour la commande de force et de vitesse [60].

En fait, une seule méthode ne peut pas obtenir une performance satisfaisante, et la combinaison des différentes méthodes de commande et l'utilisation des contrôleurs intelligents sont des domaines populaires de recherche. Une autre synthèse sur la commande de robots à articulations flexibles est décrite dans [61].



## Organisation de la thèse

Ce rapport de thèse est composé des six chapitres suivants:

Le **Chapitre 1** offre l'introduction générale de la thèse, y compris le contexte de la thèse, les problèmes et les objectifs de la recherche, une revue de littérature sur la base d'une étude profonde et étendue de la littérature et l'organisation de la thèse.

Le **Chapitre 2** présente la modélisation des robots manipulateurs à articulations flexibles. Plusieurs modèles essentiels d'un robot manipulateur à articulations flexibles à 6 degrés de liberté (ddl), y compris les modèles géométriques direct (DGM) et inverse (IGM), les modèles cinématiques direct (DKM) et inverse (IKM), les modèles dynamiques direct (DDM) et inverse (IDM) sont établies basés sur les connaissances fondamentales concernant la modélisation des robots, telles que les descriptions spatiales, les transformations homogènes, les angles fixes X-Y-Z et la convention Denavit-Hartenberg modifiée, etc. En outre, le modèle de flexibilité articulaire, le modèle de frottement et le couple du compensateur de gravité au deuxième axe sont également étudiés.

Le **Chapitre 3** propose une nouvelle méthode pour la simplification des modèles dynamiques des robots industriels en utilisant la méthode d'intervalle. Tout d'abord, les définitions et les opérations fondamentales d'analyse par intervalle, la fonction d'inclusion ainsi que la surestimation sont introduits. Ensuite, le modèle dynamique symbolique d'un manipulateur industriel KUKA KR500-2MT est généré. Par ailleurs, un algorithme généralisé de simplification est proposé et un exemple simple est fourni pour expliquer et décrire la méthode de simplification. En même temps, les résultats du modèle simplifié pour l'espace global de travail et l'analyse correspondante d'erreur ainsi que la simplification supplémentaire sont également donnés. Des études de cas ont été réalisées sur trois trajectoires différentes de test, y compris la trajectoire du procédé d'identification, la trajectoire linéaire et la trajectoire circulaire du procédé FSW, afin de démontrer l'efficacité et les bonnes performances de la méthode de simplification. En outre, les couples dans l'équation dynamique du robot a également été analysés dans le cas d'un procédé FSW linéaire. Finalement, certains usages de la méthode de simplification sont fournis.

Le **Chapitre 4** présente la modélisation dynamique et l'identification du procédé de soudage FSW robotisé. En premier lieu, une brève description du procédé FSW est donnée, y compris ses étapes principales et les forces du procédé ainsi que deux modèles de puissance non-linéaires statiques des forces de procédé. Ensuite, l'installation expérimentale pour l'identification des modèles est décrite. De plus, trois méthodes qui permettent de calculer la profondeur de plongée et sa dérivée sont proposées, suivi par le filtrage des données pour les mesures et les calculs. En outre, la modélisation dynamique linéaire et non-linéaire et l'identification de la force axiale sont effectuées à l'aide de la méthode des moindres carrés. Enfin, quelques études systématiques sur les erreurs des modèles identifiés ont été réalisées afin d'analyser les performances de l'identification des modèles.

Le **Chapitre 5** décrit la conception des contrôleurs robustes en force pour les manipulateurs de robots industriels et le développement d'un système de simulation pour le procédé de soudage FSW robotisé. Tout d'abord, une revue de littérature sur les méthodes de commande en force pour les robots est présentée, suivie par

la sélection de la stratégie de commande en force et la modélisation du procédé de soudage FSW robotisé. Ensuite, concernant le modèle de déplacement du robot rigide, l'identification des paramètres est effectuée en utilisant deux algorithmes différents. Selon la modélisation du procédé de soudage FSW robotisé qui considère simultanément la cinématique complète, le modèle de déplacement du robot rigide, les flexibilités des articulations ainsi que le modèle dynamique de la force axiale, les contrôleurs robustes de force ont été conçus à l'aide de la méthode de réponse en fréquence. Par ailleurs, le simulateur du procédé FSW robotisé est développé et les études de simulation ont été réalisées afin de tester les performances des contrôleurs de force. Finalement, l'oscillation de la force axiale dans le procédé FSW réalisé est analysée, et un modèle de perturbation de la position verticale de référence est proposé et utilisé dans les simulations afin de simuler l'oscillation de la force axiale.

Le **Chapitre 6** résume la thèse avec les principales contributions de la recherche et les perspectives pour les travaux futurs.

## 7.2 Modélisation des robots manipulateurs à articulations flexibles

### Introduction

Les travaux présentés dans ce chapitre concerne la modélisation des robots manipulateurs à articulations flexibles. Selon beaucoup de travail de recherche sur la modélisation du robot décrit dans les livres de Craig [62], Spong [8], et Siciliano [5], les connaissances fondamentales concernant la modélisation des robots sont données, y compris les descriptions spatiales, les transformations homogènes, les angles fixes X-Y-Z et la convention Denavit-Hartenberg modifiée, etc. Après cela, les études sont réalisées en vue d'obtenir les modèles différents, tels que les modèles géométriques, cinématiques, et dynamiques, ainsi que le modèle de flexibilité articulaire, le modèle de frottement, etc. Ces modèles sont extrêmement nécessaires et importants pour la simplification du modèle, la simulation de mouvement robotique et la conception du contrôleur dans les recherches suivantes.

### Descriptions spatiales et transformations homogènes

En générale, les concepts tels que les vecteurs de position, les plans et les systèmes de coordonnées seront utilisés pour décrire les relations entre des objets, y compris les pièces, les outils et le manipulateur lui-même. Comme montré sur la figure 2.1, un point arbitraire  $P$  dans le système de coordonnées cartésiennes (le repère  $\{A\}$ ) peut être représenté comme un vecteurs de position  ${}^A P$  dans l'équation (2.1). En plus de la position, comme illustré sur la figure 2.2, l'orientation du corps rigide (le repère  $\{B\}$ ) par rapport au repère  $\{A\}$  peut être exprimée par une matrice  ${}^A R_B$  dans l'équation (2.2). Donc, les rotations du repère  $\{B\}$  autour de l'axe  $X_A$ ,  $Y_A$  et  $Z_A$  avec un angle  $\theta$  peuvent être écrits comme les équations (2.3), (2.4) et (2.5). En outre, la description d'un repère en robotique peut être considérée comme un vecteur de position et une matrice de rotation dans l'équation (2.6).

En supposant que la coordonnée du point  $P$  dans le repère  $\{B\}$  est connue, les coordonnées de ce point dans le repère  $\{A\}$  peuvent être résolues par des transformations homogènes comme dans (2.7). Les transformations homogènes d'une rotation autour d'un axe et d'une translation le long d'un axe sont respectivement dénotées **Rot** et **Trans**, et les exemples sont donnés dans les équations (2.8) et (2.9).

Afin d'obtenir une représentation minimale, un opérateur de manipulateur préfère utiliser une représentation plus simple qui nécessite seulement trois numéros plutôt que de passer beaucoup de temps à entrer une matrice de neuf éléments avec les colonnes orthonormales. Ainsi, il y a plusieurs représentations qui peuvent être appliquées, telles que les angles fixes X-Y-Z, les angles d'Euler Z-Y-X, et les angles d'Euler Z-Y-Z, etc [5, 8, 62]. Selon les mesures de robot KUKA, le contrôleur du robot dans cette étude utilise les angles fixes X-Y-Z:

Considérons le repère fixe  $\{A\}$  et le repère tourné  $\{B\}$  qui sont montrés sur la figure 2.3, l'orientation du repère  $\{B\}$  par rapport au repère  $\{A\}$  peut être spécifiée par trois angles  $(\gamma, \beta, \alpha)$ , et obtenue par trois rotations successives comme suit:

- Rotation du repère fixe  $\{A\}$  autour de l'axe  $X_A$  avec un angle  $\gamma$ , en obtenant un nouveau repère  $\{1\}$ ;
- Rotation du repère  $\{1\}$  autour de l'axe  $Y_A$  avec un angle  $\beta$ , en obtenant un autre nouveau repère  $\{2\}$ ;
- Rotation du repère  $\{2\}$  autour de l'axe  $Z_A$  avec un angle  $\alpha$ , en obtenant le repère final  $\{B\}$ .

Tous les trois rotations sont effectuées autour d'un axe du repère fixe  $\{A\}$ , donc la matrice de transformation obtenue est donnée par l'équation (2.10). En réalité, cette matrice de transformation est exactement la matrice de rotation qui exprime l'orientation du repère outil par rapport au repère base. En conséquence, les trois angles  $\gamma$ ,  $\beta$ ,  $\alpha$  peuvent être directement obtenus à partir de la matrice de rotation, ce qui est extrêmement utile pour la cinématique du robot.

## Cinématique du robot

La cinématique du robot est une base importante pour l'étude des robots manipulateurs. La position, la vitesse, l'accélération et les dérivées d'ordre supérieur des variables de position sont décrites par la cinématique du robot. Cette section présente la convention de Denavit-Hartenberg modifiée ainsi que les modèles géométriques et cinématiques, ce qui sont essentiels pour commander le mouvement du robot.

Selon les travaux de recherche de Khalil et Dombre [63], la convention de Denavit-Hartenberg modifiée est souvent utilisée en vue de choisir le système de référence d'un corps en robotique. Une brève description de la convention est donnée sur la figure 2.4, et les paramètres relatif à la transformation de coordonnées sont également définis. Par ailleurs, la transformation homogène de coordonnées pour cette convention peut être représentée comme un produit de quatre transformations basiques comme dans l'équation (2.12). Le robot industriel considéré dans cette étude est un manipulateur série à chaîne ouverte avec 6 degrés de liberté KUKA KR500-2MT, qui est composé de six corps et six articulations rotatoires (voir la figure 2.5). Afin de simplifier la modélisation du robot, il est supposé que l'outil est fixé directement sur le corps 6 et l'axe de rotation de l'outil coïncide avec l'axe de l'articulation 6. Conformément à la convention de Denavit-Hartenberg modifiée, une description géométrique du robot manipulateur et les systèmes de référence des corps sont donnés sur la figure 2.6. Les paramètres géométriques correspondantes sont fournis dans le tableau 2.1.

Le problème de cinématique direct est de déterminer la position et l'orientation de l'outil par rapport au repère de base en fonction des variables articulaires, y compris les extensions des corps pour les articulations coulissantes et les angles entre les corps pour les articulations de rotation. Le repère fixé dans l'outil est généralement appelé le repère outil. Le modèle géométrique direct (DGM) permet de calculer la position et l'orientation du repère outil  $X$  en utilisant des déplacements angulaires  $q$  comme dans l'équation (2.13). Il faut noter que les trois angles  $(\gamma, \beta, \alpha)$  dans le vecteur  $X$  est exactement des angles fixes X-Y-Z qui représentent trois rotations autour d'un axe du repère de base. En générale, les déplacements angulaires sont connus pour la commande de robot. Selon le tableau 2.1 et l'équation (2.12), les matrices

des transformations homogènes pour tous les repères adjacents sont obtenus dans (2.15). Par conséquent, la matrice de transformation homogène ( ${}^0T_t$ ) qui définit le repère outil par rapport au repère de base est donnée dans les équations (2.16) et (2.17). Évidemment, le vecteur de position et la matrice de rotation sont en fonction des positions articulaires, et les résultats détaillés sont fournis en Annexe A.1. En fait, la matrice de rotation est exprimée en utilisant les angles fixes X-Y-Z dans le contrôleur du robot réel. En comparant les équations (2.10) et (2.17), les trois angles ( $\gamma, \beta, \alpha$ ) peuvent être calculés dans (2.19), (2.20) et (2.21).

Le problème de cinématique inverse est de trouver la valeur de positions angulaires en fonction de la position et l'orientation de l'outil. Les stratégies de solution pour ce problème peuvent être divisées en deux grandes catégories: les solutions fermées et les solutions numériques. Les solutions fermées, c'est-à-dire les solutions analytiques, sont couramment utilisées parce qu'ils soient généralement beaucoup plus rapides que les solutions numériques et ils permettent d'identifier toutes les solutions possibles facilement. Cependant, l'existence de solutions fermées dépendent de la structure du robot. Pieper a étudié les robots manipulateurs avec six degrés de liberté, et les conditions suffisantes pour obtenir des solutions fermées ont été données: trois axes des articulations consécutives s'entrecroisent sur un point commun comme dans un poignet sphérique, ou trois axes d'articulations consécutives sont parallèles [64]. La méthode de Paul peut être utilisée afin d'obtenir une solution explicite de forme fermée pour chaque variable articulaire [65].

Afin de résoudre ce problème, le modèle géométrique inverse (IGM) est établi dans l'équation (2.22). Ensuite, la matrice de transformation homogène ( ${}^0T_t$ ) qui définit le repère outil par rapport au repère de base est calculée dans (2.23). Si on remplace  ${}^0T_t$  par une nouvelle matrice  $U_0$  dans l'équation (2.16), une équation matricielle à résoudre peut être obtenue comme dans (2.24). En multipliant à gauche successivement les deux côtés de l'équation (2.24) par les matrices  $({}^{j-1}T_j)^{-1}$ , un nouvel ensemble d'équations peuvent être obtenues afin d'isoler et d'identifier les variables articulaires individuellement comme dans (2.27). Ces équations sont appelées les équations en avant (forward equations en anglais). Quand on multiplie à droite successivement les deux côtés de l'équation (2.24) par les mêmes matrices, les équations en arrière (backward equations en anglais) sont obtenues également dans (2.28). Ainsi, les positions angulaires des axes ( $q_1, q_2, q_3, q_4, q_5, q_6$ ) peuvent être calculées comme dans les équations de (2.29) à (2.42).

Le problème de cinématique instantanée inverse est de trouver les relations de vitesse entre les coordonnées opérationnelles et les coordonnées articulaires. Le modèle cinématique direct (DKM) permet d'exprimer les vitesses linéaires ( $v_x, v_y, v_z$ ) et les vitesses angulaires ( $\omega_x, \omega_y, \omega_z$ ) du repère outil en fonction des positions angulaires et les vitesses angulaires des axes ainsi que la matrice Jacobienne. Ce modèle peut être obtenu dans (2.43) par la différentiation par rapport au temps de l'équation du modèle géométrique direct. La matrice Jacobienne  $J(q)$  peut être décomposée en deux parties: la partie  $J_v(q)$  qui peut calculer le vecteur de vitesses linéaires et la partie  $J_\omega(q)$  qui permet de calculer le vecteur de vitesses angulaires, comme montré dans l'équation (2.44). Les éléments de la matrice Jacobienne peuvent être obtenues en différenciant le modèle géométrique directe par rapport aux positions angulaires des axes dans (2.45), et les expressions de la matrice Jacobienne sont détaillées en

Annexe A.2. Un autre l'intérêt de la matrice Jacobienne est la capacité d'étudier les singularités du robot. En outre, la transformation des couples dans le repère outil aux couples articulaires correspondants peut être également effectuée par la matrice Jacobienne, et la relation peut être obtenue par le principe des travaux virtuels comme dans (2.46). En dérivant l'équation (2.43), les accélérations linéaires et angulaires peut être obtenues par (2.47).

Afin de résoudre le problème de cinématique instantanée inverse, le modèle cinématique inverse (IKM) consiste à trouver les vitesses articulaires des axes ( $\dot{q}$ ) en utilisant les vitesses totales de l'outil ( $\dot{X}$ ) et les positions angulaires des axes ( $q$ ) comme montré dans (2.48). Il convient de noter que l'inverse de la matrice Jacobienne est utilisée pour le modèle cinématique inverse, les points singuliers doivent être évités pendant les calculs. De plus, les accélérations articulaires des axes peuvent être également obtenues dans (2.49).

## Dynamique du robot

Le modèle dynamique d'un robot manipulateur à articulations flexibles joue un rôle important dans la robotique pour la simulation du mouvement, l'analyse de la structure et la conception de la commande du robot. Les méthodes concernant la génération des modèles dynamiques pour différents robots manipulateur peuvent être trouvées dans plusieurs livres [5, 8, 62]. Ces modèles permettent de calculer les couples pendant le mouvement du robot et les forces extérieures exercées sur le robot lorsqu'il y a contact entre la pièce et l'outil. Cette section présente la modélisation dynamique d'un robot manipulateur à articulations flexibles couplée avec six axes, y compris les modèles dynamiques directes et inverses, le modèle de flexibilité, le modèle de frottement ainsi que les matrices de transmissions non diagonales. Il faut noter qu'un système de compensation de gravité sur le deuxième axe est également considéré dans la modélisation dynamique du robot.

Concernant la génération des modèles dynamiques pour les robots manipulateurs, les approches les plus couramment appliquées sont le formalisme d'Euler-Lagrange, le formalisme de Newton-Euler et la méthode de Kane. L'approche Euler-Lagrange a été proposée par Euler et Lagrange pendant leurs recherche de problème tautochrone dans les années 1750. Par rapport au formalisme de Newton-Euler, l'approche Euler-Lagrange est capable de considérer un robot à plusieurs axes dans son ensemble au lieu de traiter chaque axe de ce robot à tour de rôle. L'approche Euler-Lagrange est basée sur la description des fonctions de l'énergie scalaire du système, y compris l'énergie cinétique  $K(q, \dot{q})$  et l'énergie potentielle  $V(q)$ . Les deux fonctions d'énergie peuvent être exprimées en fonction des positions articulaires  $q$  et des vitesses articulaires  $\dot{q}$ . Donc la fonction de Lagrange  $L$  peut être définie comme dans (2.50) et l'équation de Lagrange peut être formalisée comme dans (2.51) avec l'aide du principe de moindre action. Par conséquent, le modèle dynamique peut être obtenu d'une manière directe.

En appliquant l'approche Euler-Lagrange au corps du robot et au moteur respectivement [8], le modèle dynamique pour le robot manipulateur à articulations flexibles avec six axes est donc obtenu par les équations (2.52) et (2.53).  $M(q)_{(6 \times 6)}$  est la matrice d'inertie généralisée du robot qui est systémique, définie uniformément

positive et bornée,  $F_{fr}(\dot{q})_{(6 \times 1)}$  est le vecteur de frottement au niveau des articulations,  $F_{e(6 \times 1)}$  est le vecteur des efforts appliqués sur l'environnement externe par le robot dans le repère outil,  $J^T(q)_{(6 \times 6)}$  est la matrice Jacobienne du repère outil projeté dans le repère de base,  $\Gamma_{(6 \times 1)}$  est le vecteur des couples en sortie des réducteurs,  $q$ ,  $\dot{q}$  et  $\ddot{q}$  représentent les vecteurs des positions, vitesses, et accélérations articulaires,  $I_{a(6 \times 6)}$  est la matrice d'inertie des arbres moteurs,  $\Gamma_{m(6 \times 1)}$  est le vecteur des couples moteurs,  $F_{fm}(\dot{q})_{(6 \times 1)}$  est le vecteur de frottement au niveau des axes moteurs,  $N$  est la matrice du rapport de transmission de vitesse,  $\theta$ ,  $\dot{\theta}$  et  $\ddot{\theta}$  représentent les vecteurs des positions, vitesses, et accélérations des axes des moteurs. De plus,  $H(q, \dot{q})_{(6 \times 1)}$  est le vecteur des couples centrifuges, de Coriolis et de gravité qui peuvent être exprimé sous la forme décrite dans (2.54).

Pour le robot KUKA KR500-2MT considéré dans cette étude, la matrice du rapport de transmission de vitesse  $N$  n'est pas diagonale, et il y a des couplages entre les axes 4, 5, et 6 comme montrés dans l'équation (2.55) [4]. Par ailleurs, il convient de noter qu'on tient également compte de l'effet du compensateur de gravité au niveau du deuxième axe dans notre modèle dynamique du robot, la présentation détaillée du compensateur de gravité se trouve dans [4].

Concernant les robots industriels, les deux sources principales des flexibilités sont considérées: les flexibilités concentrées au niveau des articulations et les flexibilités distribuées le long des corps [7]. Dans la thèse de Dumas [17], les sources de déformation pour les robots séries ont été analysées, et les résultats d'expériences montrent que la déformation de l'ensemble des corps du robot est responsable à hauteur de 25% du déplacement de l'effecteur. Par contre, les rotations localisées dans les articulations expliquent environ 75% du déplacement de l'effecteur. Ainsi, les sources principales des flexibilités viennent des articulations, y compris les moteurs et les transmissions. Selon l'étude de Dumas, les corps des robots industriels utilisés dans cette recherche doivent être considérés comme rigides, et on suppose que les flexibilités sont localisées seulement aux articulations. Comme montré sur la figure 2.7, la flexibilité de l'articulation est considérée comme un ressort de torsion linéaire avec une matrice de rigidité  $K_r$ , donc le modèle de flexibilité peut être donné par l'équation (2.56). En conséquence, la déformation du robot dans l'espace articulaire peut être obtenue comme dans (2.58).

Les frottements sont universellement présents pendant le mouvement du robot, et de nombreuses études sur la modélisation du frottement sont réalisées en vue d'analyser les frottements au niveau des articulations, des transmissions, des moteurs et des réducteurs. Les modèles de frottement avancés sont décrits dans [66]. Un modèle de frottement dynamique appelé le modèle LuGre a été présenté dans [67]. Ce modèle est trop compliqué bien qu'il soit capable de capturer la plupart des caractéristiques de frottement telles que l'effet Stribeck, les hystérésis, les caractéristiques de frottement statique, etc. Dans nos travaux de recherche, on a utilisé une approximation du frottement aux articulations par le modèle classique de frottement de Coulomb avec un terme visqueux. Ainsi les frottements au niveau des axes du robot et du moteur peuvent être exprimés dans les équations (2.59) et (2.60).

Dans notre modélisation du robot, le logiciel *SYMORO+* (SYmbolic MOdelling of Robots en anglais) a été utilisé en vue de générer les modèles symboliques des robots automatiquement [68, 69]. Il est basé sur le logiciel *Mathematica* qui a une

forte capacité de calculs symboliques. En utilisant ce logiciel, les divers modèles peuvent être générés pour la commande du robot et la simulation du mouvement du robot, tels que les modèles géométriques directs et inverses, les modèles cinématiques directs et inverses, les modèles dynamiques directs et inverses, la matrice Jacobienne, la matrice d'inertie, les couples centrifuges, de Coriolis et de gravité, etc. Ce logiciel utilise la convention de Denavit-Hartenberg modifiée. En entrant les paramètres nécessaires dans l'interface interactive, il est capable de fournir les modèles symboliques d'un robot rigide avec différents formats de fichier qui peuvent être directement utilisés dans les environnements de programmation tels que *Matlab*, *Mathematica* et *C*, etc. La figure 2.8 montre une vue de l'écran d'interface du logiciel *SYMORO+*.

## Conclusion

Dans ce chapitre, l'objectif est de présenter la modélisation d'un robot manipulateur à articulations flexibles avec 6 degrés de liberté. Tout d'abord, les connaissances fondamentales concernant la modélisation des robots sont données, y compris les descriptions spatiales, les transformations homogènes, les angles fixes X-Y-Z et la convention de Denavit-Hartenberg modifiée, etc. Ensuite, plusieurs modèles essentiels sont établies, tels que les modèles géométriques directs (DGM) et inverses (IGM), les modèles cinématiques directs (DKM) et inverses (IKM), les modèles dynamiques directs (DDM) et inverses (IDM), etc. Afin de obtenir les modèle complets précis pour la commande et simulation du robot, le modèle de flexibilité articulaire, le modèle de frottement et le couple du compensateur de gravité du deuxième axe sont également pris en compte. Ces modèles sont extrêmement importants dans la simplification du modèle de robot, la simulation du procédé FSW robotisé et la conception du contrôleur en force pour les robot. Finalement, un logiciel *SYMORO+* pour la génération des modèles symboliques des robots rigides est présenté.



## 7.3 Simplification des modèles dynamiques des robots en utilisant la méthode d'intervalle

### Introduction

Ce chapitre se concentre sur la proposition d'une approche de simplification pour les modèles dynamiques des robots manipulateurs industriels en utilisant la méthode d'intervalle [70]. Généralement, les modèles de faible complexité sont hautement préférés pour la conception de contrôleurs basée sur le modèle. Cependant, en raison de non-linéarités et de complexités fortes, certains composants du modèle dynamique du robot tels que la matrice d'inertie et le vecteur des couples centrifuge, de Coriolis et de gravité, sont extrêmement compliqués à calculer pour la commande en temps réel du robot. En appliquant le modèle simplifié obtenu au système de commande pratique, le temps de calcul et l'occupation de mémoire sera grandement réduits, particulièrement dans la commande basée sur un observateur non-linéaire qui est présentée par Qin et al. [4, 71, 72, 73]. Il convient de noter que cette méthode de simplification est appropriée pour les trajectoires arbitraires dans l'espace entière de travail du robot. Par ailleurs, cette simplification permet de contribuer à trouver des paramètres d'inertie négligeables, ce qui est extrêmement utile pour l'identification du modèle du robot.

Pour la plupart des robots industriels appliqués aux procédés de l'usinage ou du soudage par frottement malaxage (FSW), la haute précision et la bonne performance en temps réel ne peut pas être atteintes. De nombreuses études sur la commande de robots industriels ont été réalisées afin d'assurer une meilleure performance de suivi. Bien que la précision des procédés de fabrication peut être améliorée en utilisant les méthodes de commande basées sur l'observateur [72], il amènera inévitablement une structure de commande plus compliquée. Pour cette raison, beaucoup des calculs et des occupations de mémoire seront utilisés dans le contrôleur du robot, ce qui permet d'influencer la performance en temps réel. En conséquence, les méthodes concernant la réduction de modèle sont des outils essentiels et utiles.

La réduction de modèle pour les systèmes linéaires est un sujet de recherche matures, et les méthodes bien connues caractérisant des bornes d'erreur et la stabilité préservée sont disponibles. Cependant, des systèmes non-linéaires sont souvent rencontrés en réalité, et la réduction de modèle pour ce genre de modèle est un problème de recherche ouvert jusqu'à présent [74].

Les robots industriels sont des systèmes multi-corps élastiques, à plusieurs variables fortement couplées, et leurs dynamiques non-linéaires changent rapidement. Concernant les systèmes non-linéaires, les méthodes de réduction de modèle actuellement disponibles comprennent [74]: les méthodes heuristiques, la décomposition orthogonale propre, l'équilibrage en utilisant des fonctions de l'énergie, l'équilibrage de Gramian empirique [75], la linéarisation autour du point d'équilibre, l'approche linéaire par morceaux de trajectoires [76], la réduction de modèle par l'identification du système. Une méthode de réduction informatiquement efficace relative à la troncature équilibrée a également été proposée dans [77]. Toutes les méthodes permettent de réduire le nombre d'états et d'améliorer la vitesse de calcul, mais la limitation

est qu'ils s'appliquent uniquement aux trajectoires spécifiques ou dans une partie de l'espace de travail.

La méthode d'intervalle a été largement utilisée dans diverses applications, telles que la robotique, la commande robuste, l'estimation de paramètres et d'état, etc [78, 79]. Afin de résoudre les problèmes de suivi réels, un estimateur d'état non-linéaire récursif basé sur l'analyse par intervalle a été proposé par Kieffer et Jaulin [80]. Cette technique a été appliquée à la localisation autonome robuste de robots mobiles dans [81, 82]. La limitation principale est l'explosion de la complexité avec le nombre de variables d'état. Une méthode similaire robuste de navigation a été utilisée pour les robots de voilier, et une méthode basée sur l'intervalle pour la validation des règles fiables et robustes de navigation a été présentée [83]. Dans [84], le calcul d'intervalle et de la propagation de contrainte ont été combinés en vue de traiter des problèmes difficiles concernant la commande robuste et l'identification non-linéaire.

Dans le domaine des robots industriels, [85] a présenté une nouvelle approche basée sur l'intervalle afin de trouver la trajectoire globale minimum-jerk d'un robot manipulateur dans un espace articulaire en utilisant des splines cubiques. Le problème de conception géométrique des robots manipulateurs séries avec trois articulations rotatoires a été résolu pour la première fois grâce à l'analyse par intervalle [86]. Dans [87], une nouvelle approche basée sur la méthode d'intervalle a été utilisée en vue de traiter les problèmes de la détection dynamique d'auto-collision et la prévention pour un robot manipulateur à 2 degrés de liberté. Une méthode basée sur l'analyse par intervalle a été proposée à la détermination de l'espace de travail faisable des robots parallèles à  $n$  degrés de liberté entraînés par  $n$  ou plusieurs câbles [88]. Le plan cinématique direct d'un manipulateur série a été étendu à des intervalles en utilisant le produit de formulation exponentielle afin d'analyser les erreurs cinématiques [89].

Concernant la simplification de modèle basée sur l'intervalle, Martini a présenté une méthode pour simplifier le modèle d'un hélicoptère à 7 degrés de liberté [90]. Dans cette réduction de modèle, les termes relatifs aux composants de la matrice d'inertie et le vecteur des couples centrifuge, de Coriolis et de gravité ont premièrement été étudiés du point de vue de trajectoires spécifiques, y compris les trajectoires normales et hélicoïdales. Ensuite, tous les termes ont été analysés en utilisant la méthode d'intervalle, ce qui peut être généralisé à toute trajectoire. Une simulation sur les deux trajectoires a été réalisée afin de démontrer l'efficacité de la méthode de simplification. Néanmoins, cette méthode s'applique uniquement à certaines trajectoires spécifiques de vol.

## **Fondation de l'analyse par intervalle, la fonction d'inclusion et la surestimation**

L'analyse par intervalle, qui est née dans les années 1960, est un outil mathématique très utile pour des calculs rigoureux de bornes inférieures et supérieures de solutions. Cette méthode permet de modéliser les équations lorsque les intervalles expriment les paramètres d'entrée du modèle au lieu des valeurs ponctuelles. Avec l'aide de cette théorie, les équations peuvent être étendues au domaine de l'intervalle, ce qui

permet la propagation d'incertitude sans spécifier une distribution pour les intervalles [89]. Dans cette section, les définitions et les opérations des intervalles de base sont introduits, et plus de détails concernant les intervalles sont présentés dans les livres de Moore et al. [78] et Jaulin [79, 91].

Un intervalle réel (ou intervalle scalaire) est un ensemble fermé, borné et consécutif de nombres réels comme définie dans l'équation (3.1). Pour la simplicité, l'intervalle peut également être donnée dans (3.2). En outre, la largeur, le point milieu et la norme d'un intervalle peuvent être fournis dans les équations (3.3), (3.4) et (3.5). Les opérations arithmétiques d'intervalle y compris l'addition, la soustraction, la multiplication et la division sont définis dans les équations (3.6) à (3.11). De même, la notion d'intervalles réels et leurs opérations arithmétiques peut également être étendues à des vecteurs d'intervalle et des matrices d'intervalle. Les vecteurs d'intervalle, les matrices d'intervalle et les fonctions d'intervalle sont exprimées par des caractères gras. Les autres définitions concernant les vecteurs d'intervalle et les matrices d'intervalle sont données dans les équations (3.12) à (3.15).

En fait, l'essentiel de l'analyse par intervalle est l'extension des fonctions réelles à des intervalles qui permet de calculer des approximations extérieures de l'image des intervalles par des fonctions réelles [92]. Par conséquent, il est important de présenter les définitions de la fonction d'inclusion et de la fonction d'insertion minimale. Ainsi, il est important de présenter les définitions de la fonction d'inclusion  $\mathbf{f}$  et de la fonction d'inclusion minimale  $\mathbf{f}^*$ . Ces deux fonctions peuvent être définies par les équations (3.16) et (3.17) [93]. Par rapport à la fonction d'inclusion minimale, la fonction d'inclusion n'est pas unique. La figure 3.1 illustre la fonction  $\mathbf{f}$  et sa fonction d'inclusion ainsi que sa fonction d'insertion minimale dans le cas où  $\mathbb{X}$  et  $\mathbb{Y}$  correspondent à  $\mathbb{R}^2$ . Cette figure montre que la fonction d'insertion minimale est l'extension d'intervalle le plus précis que l'on souhaite construire.

Concernant les fonctions élémentaires  $f \in \{+, -, *, /, exp, sin, cos, \dots\}$ , leurs fonctions d'inclusion peuvent être obtenues en calculant directement les gammes exactes des fonctions. Cependant, le calcul de la fonction d'inclusion minimale d'une fonction polynomiale est prouvé comme un problème dur [92]. Pour la fonction arbitraire composé d'opérateurs élémentaires, une approche couramment utilisée pour calculer sa fonction d'inclusion est de remplacer chacun des opérateurs élémentaires par sa fonction d'inclusion minimale dans l'expression formelle. La fonction d'inclusion résultante est appelé comme l'extension d'intervalle naturelle [93]. Il convient de noter que la surestimation de la plage de la fonction peut se produire pendant le processus d'extension d'intervalle qui est due à la corrélation entre les variables et une exemple simple a été donnée. Afin de réduire la surévaluation, les autres extensions d'intervalle comme la forme centrée, la forme moyenne, la forme de la pente, la forme du test de monotonie peuvent être construites à part l'extension d'intervalle naturelle, et les algorithmes connexes sont décrits dans [78, 79]. Dans la première étape de l'algorithme de simplification, certaines transformations trigonométriques sont effectuées de manière à éliminer les corrélations entre les variables, ce qui permet d'éviter l'apparition d'une surestimation.

## Méthode de simplification et un exemple simple

La méthode de simplification doit être appliquée au modèle dynamique symbolique du robot. En utilisant l'approche Euler-Lagrange aux corps du robot et au moteur respectivement [8], le modèle dynamique pour le robot à articulations flexibles avec six axes est obtenu dans (3.24) et (3.25). Les expressions symboliques du modèle géométrique, cinématique et dynamique du robot peuvent être obtenues à l'aide du logiciel *SYMORO+* [68, 69]. Les valeurs des paramètres géométriques de robots utilisés pour la simplification sont fournies dans le tableau 3.1. Le schéma de la génération du modèle dynamique de robot symbolique est montré sur la figure 3.2.

Les expressions de certaines composantes dans l'équation (3.24) sont extrêmement compliquées, tels que la matrice d'inertie  $M(q)$  et le vecteur des couples centrifuge, de Coriolis et de gravité  $H(q, \dot{q})$ . Pour montrer la complexité, l'expression originale d'un composant  $M_{21}$  de la matrice d'inertie  $M(q)$  est donnée en annexe B.1. Toutes les expressions montrent les complexités et les fortes non-linéarités, et un couplage élevé entre les entrées de commande existe également. Dans ce cas, il est difficile de concevoir la loi de commande du robot. Ainsi, il est intéressant de simplifier ces expressions dans le but d'obtenir un modèle plus simple. Il existe diverses méthodes de réduction de modèles pour les systèmes non linéaires, toutefois, la plupart d'entre elles ne sont applicables qu'à des trajectoires spécifiques. Selon la spécification du robot KUKA KR500-2MT, les gammes de mouvement angulaire, de vitesse angulaire et d'accélération angulaire pour chaque axe sont données dans le tableau 3.2. Toutes les variables concernant les trajectoires dans l'espace de travail opérationnel sont des intervalles. En conséquence, la méthode d'intervalle sera un moyen plus facile et plus efficace pour simplifier le modèle dynamique du robot par rapport aux autres méthodes. De plus, cette méthode de simplification basée sur l'intervalle est appropriée pour les trajectoires arbitraires dans l'espace entière de travail du robot.

### Algorithm de la simplification

L'algorithme de simplification peut être appliqué aux expressions symboliques du modèle dynamique de robot, y compris la matrice d'inertie  $M(q)$  et le vecteur des couples centrifuges, de Coriolis et de gravité  $H(q, \dot{q})$ . Par ailleurs, la matrice des coefficients de couple de Coriolis  $B(q)_{(6 \times 15)}$ , la matrice des coefficients de couple centrifuge  $C(q)_{(6 \times 6)}$  et le vecteur du couple de gravité  $G(q)_{(6 \times 1)}$  peuvent également être simplifiés par l'algorithme. Pour chaque composant des matrices et des vecteurs ci-dessus dans les équation (3.24) et (3.26), son expression peut être décrite comme dans (3.27). Parce que toutes les variables  $sq$ ,  $cq$ ,  $\dot{q}$  et  $\ddot{q}$  sont des intervalles, l'objet cible de l'algorithme de simplification peut être généralisé comme dans (3.28). Comme montré sur la figure 3.3, l'algorithme de simplification proposé dans cette étude vise à obtenir le modèle dynamique du robot simplifié et à trouver les paramètres d'inertie négligeables, et le processus de simplification peut être divisée en 3 étapes:

**Étape 1. Compresser l'intervalle** Tout d'abord, certaines transformations trigonométriques peuvent être utilisées dans le but de réécrire les produits et les

puissances des fonctions sinusoïdal et cosinusoidal dans l'expression cible en utilisant les fonctions trigonométriques avec des arguments combinés, comme montré dans les équations (3.29) et (3.30). Les transformations trigonométriques ci-dessus seront à plusieurs reprises effectuées jusqu'à ce que l'expression ne change plus. Les transformations peuvent être réalisées en utilisant les fonctions "*TrigReduce*", "*TrigFactor*" et "*ReplaceRepeated*" du logiciel *Mathematica*. Après cela, les corrélations entre certains termes sont éliminées, ce qui réduit la surestimation dans une certaine mesure. On peut clairement voir que des limites plus strictes sont obtenues. Par conséquent, l'intervalle de l'expression originale sera fortement comprimée, ce qui est extrêmement nécessaire pour les étapes suivantes.

**Étape 2. Enlever les termes sans importance** La norme d'un intervalle (appelé comme la valeur absolue aussi) peut être obtenue par l'équation (3.5). Parce que la plupart des termes de l'expression sont des intervalles, il est possible de comparer les terme en calculant leurs normes. En définissant  $k_t$  comme un facteur proportionnel, aucune terme  $T_i$  dans l'expression cible peut être enlevée si elle satisfait l'expression (3.31). En conséquence, les termes sans importance dans l'expression cible peuvent être enlevés dans cette étape.

**Étape 3. Négliger les paramètres d'inertie sans importance** En fait, pour un terme donnée  $T_i$  dans l'expression cible, certains paramètres d'inertie dans  $T_i$  ont peu d'influence sur la norme de  $T_i$ . En définissant  $k_p$  comme un nouveau facteur proportionnel, aucun paramètre d'inertie  $P$  peut être négligé si il satisfait l'expression (3.32). Par conséquent, les paramètres d'inertie sans importance dans la terme  $T_i$  peuvent être trouvés dans cette étape. Finalement, l'expression cible peut être simplifiée en effectuant ces trois étapes, et l'organigramme de l'algorithme de simplification pour la programmation est donné sur la figure 3.4.

### Exemple de la simplification de composant $M_{44}$

Pour la simplicité, un composant simple  $M_{44}$  de la matrice d'inertie  $M(q)$  sera pris comme un exemple pour présenter comment mettre en œuvre l'algorithme de simplification ci-dessus. Le composant original  $M_{44}$  est formulé dans (3.33). En outre, le tableau 3.3 donne les valeurs des paramètres d'inertie utilisés dans la simplification, ce qui sont obtenus grâce à l'identification de modèle du robot [4, 94]. En premier lieu, en effectuant les transformations trigonométriques dans la première étape, le composant  $M_{44}$  peut être réécrit comme dans (3.34). Les résultats montrent que l'intervalle de  $M_{44}$  est comprimée fortement à partir de  $[-8.8580, 306.27]$  à  $[76.007, 195.34]$ , et le nombre total de termes dans l'expression de  $M_{44}$  est réduit de 16 à 10. Il convient de noter que les transformations trigonométriques dans cette étape sont mises en œuvre sans aucune approximation bien que le nombre total de termes est réduit. Ensuite, la deuxième étape de l'algorithme de simplification sera appliqué à  $M'_{44}$  dans (3.34) afin de supprimer des termes sans importance. Dans cette étude, la fonction de valeur maximum  $max$  est adoptée pour la fonction de mapping  $g$  dans (3.31). En conséquence, le ratio de la norme de chaque terme ( $|T_i|$ ) au celui maximum ( $|T_i|_{max}$ ) peut être calculé dans le tableau 3.4, sauf les terme de

constante. Pendant ce temps, les statistiques de l'histogramme de chaque section 5% peuvent être obtenues sur la figure 3.5. Selon le tableau 3.4 et la figure 3.5, le facteur proportionnel approprié  $k_t$  peut être choisi comme 5%. Ainsi, les termes  $T_8$  et  $T_9$  dans l'expression (3.34) peuvent être enlevés. L'intervalle du composant simplifié  $M''_{44}$  est [77.611, 193.74] qui est approximativement égale à l'intervalle du composant avant cette étape. Enfin, la troisième étape de l'algorithme est effectuée afin de trouver les paramètres d'inertie sans importance. Prenez le terme  $T_{10}$  dans (3.35) comme un exemple, les quatre paramètres  $XX_{5R}$ ,  $XY_5$ ,  $XX_{6R}$ , et  $ZZ_6$  peuvent être trouvés dans sa expression. Selon l'équation (3.32), les résultats pour cette étape sont donnés dans le tableau 3.5. Si le facteur proportionnel  $k_p$  est choisi comme 15%, le paramètre d'inertie  $XY_5$  peut être négligé. Finalement, le composant simplifié finale  $Ms_{44}$  est obtenu, et les normes de tous les composants générés au cours du processus de simplification sont fournies dans le tableau 3.6.

## Résultats du modèle simplifié et l'analyse d'erreurs pour l'espace de travail global ainsi que la simplification supplémentaire

Selon l'organigramme de l'algorithme de simplification (voir la figure 3.4), les programmes sont développés en utilisant le logiciel *Mathematica* qui est bien connu pour les calculs symboliques et d'intervalle. D'autre part, les facteurs proportionnels  $k_t$  et  $k_p$  ne dépendent pas des états initiaux du robot, et les résultats correspondants sont valables pour l'espace de travail global du robot. En choisissant  $k_t = 5\%$  et  $k_p = 5\%$ , les nombres de termes dans les composants simplifiés et originaux pour la matrice d'inertie  $M(q)$  et le vecteur  $H(q, \dot{q})$  sont fournis dans les tableaux 3.7 et 3.8. Afin de donner une explication plus intuitive pour la simplification ci-dessus, un composant  $M_{21}$  et sa forme simplifiée sont donnés en annexe B.1. On peut voir que l'expression originale du composant  $M_{21}$  est grandement compliquée et le tableau 3.7 montre qu'il a 112 termes. En fait, ce chiffre est encore plus grand (jusqu'à 162) avant les transformations trigonométriques. Par rapport à la composante originale, le composant simplifié  $Ms_{21}$  est extrêmement simple et il a seulement 4 termes. De plus, de nombreux paramètres connexes (ex.  $A_1$ ,  $A_2$ ,  $\varphi_1$  et  $\varphi_2$ ) sont des constantes qui peuvent être calculées hors ligne. Ainsi, le temps de calcul pratique sera grandement réduit. Par ailleurs, en utilisant les facteurs différents, certains paramètres d'inertie négligeables pour  $M(q)$  et  $H(q, \dot{q})$  peuvent également être obtenus dans le tableau 3.9. Par exemple, si  $k_t = 1\%$  et  $k_p = 3\%$ , les paramètres d'inertie de l'axe 3 ( $XY_{3R}$ ,  $XZ_3$ ) ont une influence négligeable sur le mouvement du robot KUKA.

Afin de prouver l'efficacité et la performance de la simplification ci-dessus, il est nécessaire d'analyser les erreurs des modèles simplifiés pour l'espace de travail global du robot. Ainsi, un grand nombre de points dans l'espace de travail opérationnel doivent être choisis pour évaluer la précision du modèle simplifié. Selon l'équilibre entre la précision et la simplicité, le nombre de termes après la simplification et les erreurs moyennes quadratiques (RMS) sont principalement discutés. Selon le tableau 3.2, les limites de mouvement angulaire pour chaque axe du robot KUKA sont données dans (3.36). Les valeurs sélectionnées des positions angulaires doivent être uniformément réparties dans leurs intervalles. En choisissant 10 valeurs pour la position angulaire de chaque axe sauf  $q_1$ , on peut obtenir un total de  $1 \times 10^5$  points

dans l'espace de travail opérationnel. En conséquence, les erreurs RMS des composants simplifiés de la matrice d'inertie peuvent être calculés comme dans (3.37). Quand on choisit les facteurs comme  $k_t = 1\%$  et  $k_p = 1\%$ , les nombres de termes dans les composants simplifiés et originaux ainsi que les erreurs RMS correspondantes sont donnés dans le tableau 3.10. Les résultats montrent une bonne précision de la matrice d'inertie simplifiée. Cependant, pour certains points, les erreurs maximales de certains composants sont extrêmement grandes. Par conséquent, ces points doivent être évités dans la planification de la trajectoire du robot.

En plus de la matrice d'inertie  $M_s(q)$  et le vecteur  $Hs(q, \dot{q})$ , le produit de  $M(q)$  et l'accélération angulaire  $\ddot{q}$  dans (3.24) peut également être simplifié en utilisant l'algorithme de simplification. Ce produit  $M(q)\ddot{q}$  est formulé par l'équation (3.38). De même, la deuxième étape de l'algorithme de simplification peut être appliquée à  $W_k$  en comparant les normes de ses termes. En définissant  $k_w$  comme un facteur proportionnel, aucune terme  $M_{ki}\ddot{q}_i$  dans l'expression cible peut être enlevée si elle satisfait l'expression (3.40). En conséquence, les termes sans importance dans l'expression de  $W_k$  peuvent être enlevés, et les résultats pour cette simplification supplémentaire sont fournis dans le tableau 3.12.

### Étude de cas: Simplification sur trois trajectoires différentes

Dans les applications pratiques, bien que le modèle simplifié obtenu dans la section précédente est valable pour l'ensemble de l'espace de travail du robot, la simplification peut également être effectuée basée sur les trajectoire données. Des études de cas ont été réalisées sur trois trajectoires différentes de test y compris une trajectoire du procédé d'identification, une trajectoire linéaire et une trajectoire circulaire du procédé FSW, afin d'analyser l'efficacité et la performance de la méthode de simplification. De plus, les couples dans l'équation dynamique du robot ont également été analysés dans le cas d'un procédé FSW linéaire. Tous les trajectoires sont obtenues à partir des essais du robot KUKA KR 500-2MT.

### Indices d'évaluation de la simplification

En générale, les facteurs proportionnels  $k_t$  et  $k_p$  peuvent être sélectionnés intuitivement. Le calcul des erreurs moyennes quadratiques (RMS) peut également être considéré comme une bonne méthode. Dans ce cas, plusieurs composants représentatifs sont choisis pour trouver des facteurs proportionnels les plus appropriés, tels que  $M_{11}$ ,  $M_{21}$ ,  $M_{22}$  et  $M_{41}$ . Selon l'équilibre entre la précision et la simplicité, les indices d'évaluation suivants de la simplification sont prises en compte:

- 1)  $N_t$ : le nombre de termes dans les composants du modèle simplifié;
- 2)  $e_{RMS}(\Psi s_i)$  ou  $e_{RMS}(\Psi s_{ij})$ : les erreurs RMS des composants simplifiés, exprimées en pourcentage des valeurs RMS des composants originaux;
- 3)  $e_{RMS}(\Psi s)_{global}$ : les erreurs RMS globales pour la matrice entière simplifiée ou le vecteur entier simplifié qui sera définies dans les équations suivantes;
- 4) la comparaison entre la plage de valeurs (ou la norme) du composant simplifié et celui du composant original.

En outre, les erreurs RMS des composants simplifiés  $M_{s_{ij}}$  et  $H_{s_i}$  peuvent être calculées par (3.41) et (3.42). Les erreurs RMS globales pour la matrice d'inertie  $M(q)$  et le vecteur  $H(q, \dot{q})$  sont définies dans les équations (3.43) et (3.44). De plus, le couple  $T = M(q)\ddot{q} + H(q, \dot{q}) + F_{fr}(\dot{q})$  dans (3.24) est souvent calculé dans la commande pratique en supposant que le robot fonctionne sans aucune force extérieure. Donc, le couple correspondant  $Ts$  peut être obtenu dans (3.45), et les méthodes de calcul pour les erreurs RMS des composants simplifiés  $Ts_i$  ainsi que l'erreur RMS globale du couple simplifié  $Ts$  sont similaires à  $Hs_i$  and  $Hs$  respectivement.

### Cas I: Simplification sur une trajectoire d'identification

Tout d'abord, la trajectoire d'identification obtenu à partir du test *IS0811A* est utilisée pour effectuer la simplification, comme montré sur la figure 3.6. Selon les données de trajectoire, les intervalles réels des états du robot y compris la position de l'outil, la position angulaire, la vitesse angulaire et l'accélération angulaire des articulations sont donnés dans le tableau 3.13. Il convient de noter que ces valeurs le long de la trajectoire d'identification ont relativement de grandes variations afin d'améliorer la précision de l'identification des paramètres.

Pour trouver des facteurs proportionnels les plus appropriés  $k_t$  et  $k_p$ , les composants  $M_{11}$ ,  $M_{21}$ ,  $M_{22}$  et  $M_{41}$  sont illustrés pour calculer les indices d'évaluation. En choisissant différents  $k_t$  et  $k_p$ , les nombres de termes dans les composants représentatifs simplifiés et originaux ainsi que les erreurs RMS correspondantes le long de la trajectoire d'identification sont donnés dans le tableau 3.14. En faisant un bon compromis entre la simplicité et la précision, les facteurs les plus appropriés peuvent être choisis comme  $k_t = 3\%$  et  $k_p = 1\%$ . Le nombre de termes pour chaque composant de la matrice d'inertie  $M(q)$  et sa forme simplifiée  $M_s(q)$  ainsi que les erreurs RMS correspondantes sont fournis dans le tableau 3.15.

Les résultats indiquent que la simplification induit de très petites erreurs RMS pour la plupart des composants simplifiés, en même temps, la quantité de termes dans ces composants sont considérablement diminuée. En prenant le composant  $M_{11}$  comme un exemple, environ 95% de termes sont éliminés après la simplification (de 238 à 15), toutefois, l'erreur RMS du composant simplifié  $M_{s_{11}}$  est de 1.08% seulement. En outre, le composant  $M_{41}$  a l'erreur RMS maximale qui est d'environ 10.5%. Les figures 3.7 et 3.8 montrent la comparaison entre le composant original de  $M(q)$  et le composant simplifié de  $M_s(q)$  le long de la trajectoire d'identification. Sur ces figures, on peut voir que les plages de valeurs de tous les composants simplifiés sont presque les mêmes que celles des composants originaux. Dans l'ensemble, les composants simplifiés sont en bon accord avec les composants originaux, même si certains composants ne correspondent pas exactement, par exemple  $M_{41}$ . Cependant, il faut noter que tous les paramètres d'inertie qui sont obtenus par l'identification du modèle ont un certain degré d'erreur [4]. Par rapport aux incertitudes des paramètres d'inertie, les erreurs provoquées par la simplification ci-dessus sont dans les limites acceptables, et l'erreur RMS globale de la matrice d'inertie  $M(q)$  est de 0.95%.

De même, en choisissant  $k_t = 3\%$  et  $k_p = 1\%$ , le nombre de termes pour chaque composant du vecteur  $H(q, \dot{q})$  et sa forme simplifiée  $H_s(q, \dot{q})$  ainsi que les erreurs RMS correspondantes sont données dans le tableau 3.16. La comparaison entre le



composant original de  $H(q, \dot{q})$  et le composant simplifié de  $Hs(q, \dot{q})$  le long de la trajectoire d'identification est montrée sur la figure 3.7, et l'erreur RMS globale du vecteur  $H(q, \dot{q})$  est de 4.41%. Après l'obtention de  $Ms(q)$  et  $Hs(q, \dot{q})$ , le couple  $T$  peut être également simplifié selon l'équation (3.45). La Figure 3.10 montre la comparaison entre le couple original  $T$  et le couple simplifié  $Ts$  le long de la trajectoire d'identification, et les erreurs RMS pour chaque axe sont de 0.89%, 3.05%, 1.58%, 2.52%, 1.57%, et 0.93% respectivement. En outre, l'erreur globale RMS du couple  $T$  est de 2.10%. Les résultats de simulation démontrent une bonne performance de simplification pour la trajectoire d'identification donnée.

### Cas II: Simplification sur une trajectoire linéaire du procédé FSW

En plus de la trajectoire d'identification, l'algorithme de simplification est également applicable à une trajectoire linéaire du procédé FSW. La trajectoire considérée ici est acquise à partir du test *IS1709J*, comme montré sur la figure 3.11. Les intervalles réels des états du robot y compris la position de l'outil, la position angulaire, la vitesse angulaire et l'accélération angulaire des articulations sont donnés dans le tableau 3.17. En comparaison avec le cas de la trajectoire d'identification, tous les états du robot pendant le procédé FSW linéaire ont beaucoup plus petites variations.

Comme montré dans le tableau B.1 en Annexe, les nombres de termes dans les composants représentatifs simplifiés et originaux ainsi que les erreurs RMS correspondantes le long de la trajectoire linéaire du procédé FSW sont fournis lorsque des facteurs différents sont choisis. Selon l'équilibre entre la précision et la simplicité, les facteurs pour  $M(q)$  et  $H(q, \dot{q})$  peuvent être choisis comme  $k_t = 1\%$  et  $k_p = 1\%$ . En conséquence, le nombre de termes pour chaque composant de  $M(q)$  et  $H(q, \dot{q})$  ainsi que les erreurs RMS correspondantes sont fournis dans les tableaux 3.18 et 3.19. Par rapport aux résultats pour la trajectoire d'identification, on peut observer que moins de termes sont simplifiés et plus de termes sont supprimés en utilisant un plus petit facteur  $k_t = 1\%$ , ce qui peut être expliqué par ce que les intervalles des états du robot sont différents pour les deux trajectoires. Bien que le nombre de termes soit considérablement réduit, la précision des composants simplifiés est assez bonne. Le composant  $M_{41}$  a l'erreur RMS maximale qui est d'environ 12.05%.

Les figures 3.13, 3.14, 3.15 et 3.16 fournissent la comparaison entre le composant original et le composant simplifié le long de la trajectoire linéaire du procédé FSW pour la matrice d'inertie  $M(q)$ , le vecteur  $H(q, \dot{q})$  et le couple  $T$ . En raison de petites variations des états du robot, les valeurs de tous les composants dans ces figures ont légèrement changé par rapport aux figures 3.7 à 3.10. Concernant le couple  $T$ , les erreurs RMS pour chaque axe sont de 0.03%, 3.39%, 2.76%, 0.01%, 0.001% et 0.005% respectivement. Dans l'ensemble, les composants simplifiés sont essentiellement identiques à ceux d'origine, et les erreurs globales RMS de  $M(q)$ ,  $H(q, \dot{q})$  et  $T$  sont de 0.56%, 3.16% et 3.21% respectivement. Les résultats montrent une bonne performance de simplification pour la trajectoire linéaire du procédé FSW.

### Cas III: Simplification sur une trajectoire circulaire du procédé FSW

Pendant le procédé FSW, les trajectoires circulaires sont aussi fréquemment utilisées pour les tests des robots industriels. Donc la trajectoire circulaire du procédé FSW

obtenu à partir du test *IS1006P* est utilisée pour effectuer la simplification, comme montré sur la figure 3.17. Sur cette figure, on peut observer que les états de l'axe 6 ont de plus grandes variations que d'autres axes. Les intervalles réels des états du robot y compris la position de l'outil, la position angulaire, la vitesse angulaire et l'accélération angulaire des articulations sont donnés dans le tableau 3.20.

En choisissant différents  $k_t$  et  $k_p$ , les nombres de termes dans les composants représentatifs simplifiés et originaux ainsi que les erreurs RMS correspondantes le long de la trajectoire circulaire du procédé FSW sont donnés dans le tableau B.2 en Annexe. En faisant un bon compromis entre la simplicité et la précision, les facteurs les plus appropriés peuvent être choisis comme  $k_t = 1\%$  et  $k_p = 1\%$ . En conséquence, le nombre de termes pour chaque composant de  $M(q)$  et  $H(q, \dot{q})$  ainsi que les erreurs RMS correspondantes sont fournis dans les tableaux 3.21 et 3.22. Les résultats indiquent que les termes supprimés dans la simplification pour la trajectoire circulaire sont moins nombreux que ceux enlevés dans la simplification pour la trajectoire linéaire lors de l'utilisation des mêmes facteurs proportionnels. En outre, la simplification pour la trajectoire circulaire produit de plus petites erreurs RMS pour presque tous les composants simplifiés. Comme dans les cas précédents, le composant  $M_{41}$  a l'erreur RMS maximale qui est d'environ 4.04%.

Les figures 3.18, 3.19, 3.20 et 3.21 montrent la comparaison entre le composant original et le composant simplifié le long de la trajectoire circulaire du procédé FSW pour la matrice d'inertie  $M(q)$ , le vecteur  $H(q, \dot{q})$  et le couple  $T$ . Pour le couple  $T$ , les erreurs RMS pour chaque axe sont de 0.01%, 0.01%, 0.003%, 0.005%, 0.01% et 0.001% respectivement. Globalement, les composants simplifiés coïncident avec ceux d'origine, et les erreurs globales RMS de  $M(q)$ ,  $H(q, \dot{q})$  et  $T$  sont de 0.54%, 0.004% et 0.005% respectivement. Les résultats de la simulation indiquent une bonne performance de simplification pour la trajectoire circulaire du procédé FSW.

#### Cas IV: Analyse des couples dans l'équation dynamique du robot

Au moyen de calculs d'intervalles, les couples de l'équation dynamique du robot ont été analysés en utilisant la trajectoire linéaire FSW (test *IS1709J*), comme montré sur la figure 3.11. Selon l'équation (3.24), les couples suivants peuvent être analysés:

- 1)  $T_{Racc}$  : le couple d'accélération angulaire  $M(q)\ddot{q}$  côté robot;
- 2)  $B(q)(\dot{q}\dot{q}) + C(q)(\dot{q}^2)$  : la somme du couple centrifuge et de Coriolis, qui peut être obtenu en mettant l'accélération de gravité égale à zéro dans  $H(q, \dot{q})$ ;
- 3)  $G(q)$ : le couple de gravité;
- 4)  $F_{fr}(\dot{q})$  : le couple de frottement au niveau des articulations du robot;
- 5)  $\Gamma$  : le couple en sortie de réducteur;

En outre, le couple externe ( $J^T(q)F_e$ ) appliqué par l'environnement sur le robot est très grand pendant le procédé FSW. Parce que le couple d'accélération angulaire du côté du moteur  $I_a\ddot{\theta}$  et le couple de frottement du côté du moteur  $F_{fm}(\theta)$  sont beaucoup plus petits que le couple moteur  $\Gamma_m$  dans l'équation (3.25), le couple en sortie de réducteur  $\Gamma$  peut être calculé approximativement par (3.46). La figure 3.22

fournit les courants mesurés de tous les moteurs pendant le procédé. Selon des intervalles réels des courants mesurés et les états du robot, les calculs d'intervalles des couples dans l'équation dynamique du robot peuvent être effectués, et les résultats sont donnés dans le tableau 3.23. Il peut être observé que le couple en sortie de réducteur  $\Gamma$  a les plus grandes variations entre les couples dans l'équation dynamique du robot. D'après la seconde étape de l'algorithme de simplification, tous les couples peuvent être analysés en comparant leurs normes. Selon l'équation (3.47), les résultats du ratio  $p_T$  pour tous les couples peuvent être trouvés dans le tableau 3.24.

Les résultats indiquent que tous les couples sauf celui externe de l'équation dynamique du robot sont beaucoup plus petits que le couple en sortie de réducteur  $\Gamma$  pendant le procédé FSW linéaire. Par ailleurs, le couple centrifuge ( $C(q)(\dot{q}^2)$ ), le couple de Coriolis ( $B(q)(\dot{q}\dot{q})$ ) et le couple d'accélération du robot ( $T_{Racc}$ ) sont négligeables. Les composants du couple  $T_{Racc}$  pour tous les axes sont plus petits que 1% du couple en sortie de réducteur  $\Gamma$ , et le résultat du ratio  $p_T$  pour le couple centrifuge et le couple de Coriolis atteignent même l'ordre de grandeur de  $1 \times 10^{-4}$ . De plus, le couple de gravité et le couple de frottement du robot ne peuvent pas être négligés, et certains éléments d'entre eux se situent dans la gamme de 10% à 15% du couple en sortie de réducteur  $\Gamma$ . Des résultats similaires peuvent également être obtenus pour l'équation dynamique du moteur.

## Discussion sur l'utilisation du modèle simplifié

Comme montré dans les résultats de la simulation ci-dessus, généralement, le modèle simplifié a non seulement des composants relativement simples, mais aussi une précision suffisante par rapport au modèle original. En conséquence, le modèle simplifié peut être appliqué à de nombreuses méthodes de commande de manière à simplifier la structure de commande et à améliorer la performance de commande en temps réel, telles que la commande de couple calculé et la commande basée sur l'observateur.

Dans le travail de recherche présenté par Qin et al. [4, 71, 72, 73], lorsque une grande force extérieure est exercée sur le robot pendant le procédé FSW, la rigidité naturelle du robot industriel est insuffisante. En conséquence, une déformation non-négligeable du robot est générée. Afin de compenser la déformation du robot, un observateur non-linéaire a été proposé en vue d'estimer les états du robot et la force extérieure. Dans la commande basée sur l'observateur pour le robot KUKA KR500-2MT, une position cible corrigée est donnée à l'interpolateur du contrôleur de robot toutes les 12 ms de manière à compenser l'erreur entre la position désirée et que l'on appelle la position réelle qui est estimée par un observateur discret dont le temps d'échantillonnage est de 1.2 ms [73]. Le schéma de la compensation externe de la déformation du robot KUKA est fourni sur la figure 3.23, ce qui peut expliquer la raison pour laquelle la simplification est nécessaire pour cette commande. La force extérieure estimée par l'observateur [72] peut être obtenue dans l'équation (3.48). Il est évident que la matrice d'inertie  $M(q)$  peut être remplacée par celle simplifiée  $M_s(q)$  dans cette équation.

En outre, les paramètres d'inertie négligeables trouvés par cette simplification sont extrêmement utiles pour l'identification du modèle du robot. Les paramètres d'inertie identifiés seront plus précis lorsque les paramètres d'inertie négligeables

peuvent être enlevés [94]. En utilisant la méthode de simplification du modèle dynamique du robot, les paramètres importants du robot peuvent être trouvés, ce qui sera extrêmement bienfaisant pour la conception du nouveau robot. Par exemple, on suppose qu'un nouvel axe est ajouté au robot industriel existant afin d'appuyer la pièce sur la plate-forme, l'influence des paramètres du nouvel axe peut être analysée en utilisant la méthode de simplification. De plus, l'algorithme de simplification peut également être appliqué aux autres modèles dynamiques tels que les avions et les hélicoptères.

## Conclusion

Dans ce chapitre, une nouvelle approche en utilisant la méthode d'intervalle pour la simplification du modèle dynamique de robot industriel est présentée. Concernant le modèle dynamique symbolique du manipulateur industriel à 6 degrés de liberté, toutes les expressions des composants dans la matrice d'inertie  $M(q)$  et le vecteur  $H(q, \dot{q})$  montrent les complexités et les fortes non-linéarités, par ailleurs, un couplage élevé entre les entrées de commande existe également. Ainsi, un algorithme de simplification basé sur l'intervalle est proposé afin de obtenir un modèle de robot beaucoup plus simple.

Tout d'abord, la théorie de l'analyse par intervalle est brièvement présentée, suivie par la génération du modèle dynamique symbolique du robot par le logiciel *SYMORO+*. Ensuite, un algorithme généralisé de simplification est proposé. Le processus de la simplification peut être divisé en trois étapes. Après l'exemple de simplification, les résultats et l'analyse des erreurs du modèle simplifié pour l'espace global de travail du robot ont été donnés, ce qui peut être applicable à des trajectoires arbitraires. L'expression originale d'un composant  $M_{21}$  dans la matrice d'inertie et de sa forme simplifiée  $M_{s21}$  ont été fournies afin d'expliquer intuitivement la méthode de simplification. En outre, la simplification supplémentaire correspondante a également été effectuée.

Ensuite, des études de cas ont été réalisées sur trois trajectoires différentes de test, y compris la trajectoire du procédé d'identification, la trajectoire linéaire et la trajectoire circulaire du procédé FSW. Les comparaisons entre les composants simplifiés et originaux de la matrice  $M(q)$ , le vecteur  $H(q, \dot{q})$  et le vecteur  $T$  ont été effectuées, et les résultats de simulation démontrent l'efficacité et les bonnes performances de la méthode de simplification. De plus, les couples dans l'équation dynamique du robot ont également été analysés dans le cas d'un procédé FSW linéaire.

L'algorithme de simplification proposé dans cette thèse peut être généralisé à d'autres modèles dynamiques tels que les avions et les hélicoptères. Le modèle simplifié peut être appliqué à de nombreuses méthodes de commande afin de simplifier la structure de commande et d'améliorer la performance de commande en temps réel, telles que la commande de couple calculé et la commande basée sur l'observateur. Par ailleurs, cette simplification permet de contribuer à trouver des paramètres d'inertie négligeables, ce qui est extrêmement utile pour l'identification du modèle du robot. Les paramètres d'inertie identifiés seront plus précis lorsque les paramètres d'inertie négligeables peuvent être enlevés.

## 7.4 Modélisation dynamique et identification du procédé de soudage FSW robotisé

### Introduction

Les travaux présentés dans ce chapitre se concentrent sur la modélisation dynamique et l'identification des forces totales qui agissent sur l'outil pendant le procédé FSW robotisé [97]. Ces modèles dynamiques peuvent être utilisés pour calculer les forces en utilisant les états du robot sans la réalisation de l'opération de soudage réel, ce qui contribue à la simulation du procédé FSW dans le simulateur du système entier robot-procédé. En outre, bien que les modèles statiques peuvent être décrits par les relations d'état en régime permanent entre les forces de procédé et les paramètres du procédé (y compris la profondeur de plongée, la vitesse d'avancement et la vitesse de rotation), les modèles dynamiques des forces de procédé sont très nécessaires pour la conception des stratégies de commande en soudage FSW robotisé, en particulier pour la conception du contrôleur en force du robot qui sera introduit dans le dernier chapitre. En raison de la rigidité naturelle insuffisante du robot industriel, les grandes forces externes exercées sur le robot aboutissent à une déformation non-négligeable qui peut être calculée par les modèles identifiés. Ainsi, les modèles peuvent être utilisés pour établir les lois de commande en vue de compenser les erreurs de positionnement du robot. En outre, les modèles dynamiques des forces de procédé peuvent être utilisés pour analyser la stabilité du contrôleur de force du robot.

Le soudage par frottement malaxage (ou Friction Stir Welding en anglais pour FSW) est une technologie de soudage relativement innovante et prometteuse qui a été brevetée et développée par l'organisme TWI (The Welding Institute, Cambridge, le Royaume-Uni) en 1991 [1]. En comparaison des technologies de soudage classiques, le soudage FSW est effectué à l'état solide et la température de fusion de matériau n'est pas atteinte pendant le procédé. Par conséquent, les cordons de soudure réalisés sont de haute qualité sans produire les défauts tels que la porosité, la redistribution des solutés, les criques de refroidissement ou les fissures de solidification [2]. Ce procédé permet l'assemblage de pièces en alliages d'aluminium difficilement soudable, tel que les alliages d'aluminium série 2XXX et 7XXX [98]. Par ailleurs, cette technique peut également être utilisée pour souder des alliages semblables ou dissemblables et des métaux tels que l'acier, le cuivre et les alliages de magnésium [99]. En plus de l'efficacité énergétique, le respect de l'environnement et la polyvalence, les avantages principaux du soudage FSW comportent: l'excellente résistance à la soudure et la ductilité, les faibles contraintes résiduelles, pas besoin de soudures métalliques, la vitesse haute de soudage et la répétabilité [3]. Grâce aux avantages ci-dessus, le soudage FSW a été largement appliqué dans une grande variété d'applications industrielles, telles que l'industrie de la construction navale, l'industrie des chemins de fer, l'industrie automobile et aérospatiale.

Au cours des dernières années, les machines dédiées qui sont couramment utilisées pour effectuer le procédé FSW ont été de plus en plus remplacées par les robots industriels en raison de leur excellente répétabilité, la flexibilité de la production et leur faible coût [100]. Cependant, la capacité limitée de la force et les rigidités

des robots industriels sont des obstacles pour l'expansion du soudage FSW robotisé. Smith et al. [101] ont développé un système FSW qui a été intégré dans un robot articulé (ABB IRB 7600), et la capacité pour réaliser le soudage de contours en trois dimensions dans différentes positions avec une excellente commande par le retour de force a été prouvée. Bres et al. [102] ont établi un cadre basé sur le modèle qui permet la simulation, l'analyse et l'optimisation du procédé FSW en utilisant des robots. De Backer et al. [103] ont appliqué des méthodes en ligne de détection et de correction de la trajectoire afin d'obtenir un soudage solide et sans défaut. Fehrenbacher et al. [104] ont prouvé un développement réussi et une évaluation d'un système de commande en boucle fermée pour le procédé FSW robotisé, et le système contrôle la force de plongée et la température d'interface de l'outil en changeant la vitesse de broche et la position verticale commandée de l'outil. De nombreuses recherches ont été réalisées pour résoudre les problèmes de commande en soudage FSW et la modélisation dynamique du procédé est extrêmement importante pour la conception des contrôleurs de force.

Dans le domaine de la modélisation et l'identification du procédé FSW, Ulysse [105] a développé un modèle viscoplastique en trois dimensions en utilisant la méthode des éléments finis, et l'auteur a également étudié l'influence de la vitesse d'avance et de la vitesse de rotation sur la force de soudage. Un modèle d'éléments finis thermomécanique a été présenté dans [106], et les calculs numériques de la force était en bon accord avec les résultats expérimentaux. Néanmoins, les applications de modèles ci-dessus sont limitées pour la commande en temps réel du procédé en raison de nombreux calculs. En conséquence, un modèle dynamique empirique a été développé dans [107], et ce modèle construit les relations entre les paramètres de procédé (ex., la profondeur de plongée, la vitesse d'avance et la vitesse de rotation) et les variables de procédé (ex., la force axial, la force d'avance et la force latérale). Des expériences ont également été effectuées afin d'identifier les caractéristiques dynamiques des variables de procédé utilisant la méthode récursive des moindres carrés. En analysant les forces du procédé et leur impact sur la structure de la machine, Zaeh and Gebhard [96] ont proposé un modèle expérimental des forces du procédé, et ce modèle est composé des composantes statiques des forces et la superposition des équations d'ondes avec des amplitudes différentes, des phases et la fréquence de la vitesse de rotation ainsi que sa première harmonique.

## **Description du procédé FSW et la modélisation statique des forces de procédé**

Le concept de base du procédé FSW peut être décrit comme suit: un outil non-consommable composé d'un pion spécialement conçu et un épaulement est inséré dans l'interface de deux pièces rigidement serrées à une vitesse spécifique de rotation et de plongée. Ensuite, l'outil est déplacé le long de la ligne de soudure lorsque l'épaulement est en contact avec les surfaces des pièces à souder. Pendant le procédé, l'outil contribue à deux fonctions principales: le chauffage des pièces à souder et le mouvement des matériaux pour produire le joint. Le chauffage peut être réalisé par la frottement entre l'outil et les pièces et par la déformation plastique sévère des pièces générée autour de l'outil, en même temps, les matériaux autour du pion

sont ramollis. Grâce à la combinaison de la rotation et la translation de l'outil, les matériaux ramollis sont remués de l'avant du pion à l'arrière du pion, ce qui joint les pièces ensemble [3]. Par ailleurs, il est intéressant de noter que la chaleur générée n'est toujours pas suffisante pour fondre du matériau pendant le procédé de soudage. En conséquence, le soudage est effectué à l'état solide et le point de fusion de matériau n'est pas atteint [99, 108]. L'outil est rétracté lorsque le procédé de soudage est terminé. Un schéma d'un procédé FSW linéaire lorsque deux pièces sont soudées est représenté sur la figure 4.1, et plusieurs paramètres du modèle des forces de procédé sont décrites dans cette figure, y compris la profondeur de plongée  $d_p$ , la vitesse d'avance  $v$  et la vitesse de rotation de l'outil  $\Omega$ . Concernant les forces de procédé, la force axiale  $F_z$  est générée par la pression agissant sur l'extrémité du pion et l'épaulement de l'outil, ce qui provoque une action de forgeage qui produit une bonne micro-structure des pièces soudées. En générale, la force axiale est la plus grande des trois forces du procédé et elle a une contribution importante à la qualité de la soudure. Donc, la force axiale doit être réglée comme une valeur constante désirée afin d'effectuer une bonne soudure. La force d'avance  $F_x$  est la force imposée sur l'outil le long de la direction de soudage. La force latérale  $F_y$  est la force imposée sur l'outil dans le plan de la pièce et perpendiculaire à la direction de soudage [107]. Le procédé FSW peut être divisé en trois phases principales: le plongement, le soudage et le retrait de l'outil, comme illustrés sur la figure 4.2. En outre, l'évolution des forces mesurées et des couples mesurés qui sont appliqués sur l'outil pendant le procédé FSW d'un alliage d'aluminium de la série 6000 d'épaisseur 6 mm [109], sont montrés sur la figure 4.3. Deux pics de forces pendant ce procédé sont analysés [110], de plus, la modélisation dynamique et l'identification du procédé FSW robotisé dans cette étude considèrent principalement la phase de plongement et la phase de soudage selon les travaux de recherche de Zimmer et al. [111].

Le procédé FSW peut être affecté par de nombreux paramètres et des facteurs tels que la profondeur de plongée, la vitesse de rotation, la vitesse d'avance, la géométrie de l'outil, les propriétés thermomécanique de matériaux, etc. Les paramètres du procédé pour la modélisation des forces comprennent principalement la profondeur de plongée, la vitesse de rotation et la vitesse d'avance de l'outil. Bien que les facteurs tels que les propriétés des matériaux de soudure, la géométrie de l'outil et les angles d'avance ont une influence significative sur les forces de procédé en plus des paramètres du procédé, ces facteurs peuvent être considérés comme constants pour la modélisation. Selon la recherche de [107], des modèles de puissance non-linéaires qui sont utilisés pour décrire les relations à l'état stable entre les forces de procédé et les paramètres de procédé pendant la phase de soudage peuvent être donnés par l'équation (4.1). Ces modèles non-linéaires peuvent être construits et évalués au moyen d'une analyse de régression linéaire, et les paramètres inconnus peuvent être estimés simplement en utilisant la méthode des moindres carrés (least-square method en anglais pour LS), comme montré dans (4.3). Il convient de noter que l'influence de la profondeur de plongée  $d_p$  sur la force d'avance  $F_x$  est négligeable par rapport à l'influence de la vitesse d'avance  $v$  et la vitesse de rotation  $\Omega$ . Selon les modèles statiques des forces de procédé, les modèles dynamiques empiriques pour le procédé FSW ont été proposés par [107] en utilisant la profondeur de plongée  $d_p$  comme entrée. Les paramètres des modèles dynamiques ont été estimés en utilisant

la méthode récursif des moindres carrés (recursive least-square method en anglais pour RSL) sur trois groupes d'essais. En conséquence, les fonctions de transfert des forces du procédé sont obtenues par (4.4).

En fait, la force axiale  $F_z$  est généralement mesurée par le capteur de force monté sur l'équipement de soudage inclus dans les robots industriels. Par conséquent, la force axiale  $F_z$  peut être considérée comme un paramètre d'entrée lors de l'identification des autres modèles des forces et des couples de procédé. En outre, les essais montrent qu'il n'y a pas de corrélation évidente entre les autres forces ( $F_x$ ,  $F_y$ ,  $C_z$ ) et la profondeur de plongée  $d_p$  [112]. Ainsi, les modèles statiques modifiés pour les autres forces peuvent être écrits comme dans (4.5). Ces modèles ont été identifiés par nos partenaires du projet et les modèles identifiés montrent bonnes précisions dans les validations expérimentales [2, 113].

## Montage expérimental

Le système de base de soudage utilisé pour réaliser le procédé FSW est composé du manipulateur industriel KUKA KR500-2MT, une broche à entraînement électrique pour le FSW, le système de commande du robot et les capteurs, comme montré sur la figure 4.4. Le soudage FSW de certaines plaques d'alliage d'aluminium (type Al6082) est mis en œuvre par le système de soudage avec un outil appelé COR-12-6 qui est spécialement développé par l'Institut de Soudure. Pendant le procédé FSW, la broche est commandé pour souder le long de la direction  $x$  dans le repère de l'outil avec la vitesse constante d'avance ( $v = 400 \text{ mm/min}$ ) et la vitesse de rotation ( $\Omega = 1100 \text{ tr/min}$ ). En outre, la force axiale désirée est de  $9000 \text{ N}$ . Toutes les conditions de soudage pour les essais expérimentaux sont données dans le tableau 4.1. La trajectoire désirée de soudure dans l'espace 3D, les positions angulaires, les vitesses angulaires et les accélérations angulaires des axes des moteurs (mesurées) après la réduction de vitesse sont montrées sur la figure 4.5. Concernant les capteurs, les positions angulaires des axes des moteurs après la réduction de vitesse peuvent être obtenues par les codeurs incrémentaux des moteurs, et les courants de tous les moteurs peuvent être acquis par les actionneurs (voir la figure 4.6). La force axiale peut être mesurée par le capteur de force (Schunk FTC) qui est monté sur l'axe 6 du robot. Le transfert des mesures peut être réalisé en connectant un ordinateur externe avec l'armoire de commande du robot KUKA via un câble Ethernet. Pendant les essais de soudage FSW, le contrôleur du robot KUKA envoie un fichier *XML* à l'ordinateur externe tous les  $12 \text{ ms}$ . Ce fichier contient toutes les mesures, y compris les positions angulaires mesurées des axes du robot après la réduction de la vitesse ( $\theta_l$ ), les courants des moteurs ( $I$ ) ainsi que la force dans la  $z$  direction du repère de l'outil ( $F_{zm}$ ). De plus, la position commandée peut également être donnée au contrôleur du robot KUKA par le fichier *XML*, et le schéma sur le flux du signal d'essai expérimental est présenté à la figure 4.7.

Dans cette étude, la modélisation dynamique et l'identification de la force en  $z$  (ci-après, appelée force axiale  $F_z$ ) sont principalement considérées. Il y a principalement deux raisons: tout d'abord, la force axiale est mesurée par la plupart des robots industriels en raison de la nécessité pour la commande de la force. En second lieu, comme mentionné dans la description du procédé FSW, la force axiale  $F_z$  est



beaucoup plus grande que la force d'avance  $F_x$  et la force latérale  $F_y$ . Afin d'obtenir une bonne qualité de soudure, la force axiale pendant la phase de soudage doit être contrôlée comme une valeur constante désirée. Pour la modélisation dynamique de la force d'avance et la force latérale, les modèles statiques modifiés dans (4.5) peuvent être utilisés. En plus de la force axiale, la position de l'outil dans l'espace cartésien est nécessaire pour l'identification. Mais les positions angulaires réelles des axes du robot ( $q$ ) sont différentes des positions angulaires mesurées des axes du robot ( $\theta_i$ ) en raison des flexibilités des articulations. En conséquence, la position réelle de l'outil qui est calculée en utilisant le modèle géométrique direct (DGM) et  $\theta_i$  est inexacte. Toutefois, la mesure directe de la position de l'outil est extrêmement compliqué et coûteux pour le procédé FSW robotisé. Ainsi, il est nécessaire d'utiliser les positions angulaires mesurées des axes des moteurs après la réduction de vitesse ( $\theta_i$ ) et les courants des moteurs ( $I$ ) pour estimer la position réelle de l'outil avec l'aide d'un observateur. Dans le travail de recherche présenté par [4, 72, 73], un observateur discret non-linéaire de gain élevé est proposé pour compenser les erreurs de positionnement en raison des flexibilités des articulations du robot. L'observateur proposé est utilisé pour estimer les positions et les vitesses réelles angulaires des axes du robot ( $q, \dot{q}$ ) ainsi que la force externe ( $F_e$ ) en utilisant uniquement les mesures du côté du moteur, et le schéma fonctionnel de l'observateur est représenté sur la figure 4.8. En outre, cet observateur a été validé expérimentalement sur un robot manipulateur industriel pendant le procédé FSW, et la compensation basée sur l'observateur a réussi à corriger les erreurs de positionnement dans la commande en temps réel [4, 73]. Par conséquent, les positions et les vitesses angulaires estimées des axes du robot ( $\hat{q}, \hat{\dot{q}}$ ) ainsi que la force externe estimée ( $\hat{F}_e$ ) peuvent être utilisées pour calculer la position réelle de l'outil. Le schéma d'acquisition et de traitement des données dans le test externe du robot KUKA basé sur l'observateur est donné sur la figure 4.9. L'excitation dans la direction  $z$  du repère de l'outil est ajouté pendant le procédé FSW afin d'obtenir des paramètres plus précis du modèle dans l'identification suivante.

## Calcul de la profondeur de plongée et filtrage des données

Comme mentionné dans le montage expérimental, la vitesse d'avance et la vitesse de rotation sont constants dans les essais. La modélisation dynamique et l'identification du procédé de soudage FSW robotisé sont réalisées en utilisant la profondeur de plongée  $d_p$  et sa dérivé ( $\frac{d}{dt}d_p$ ) qui ne peuvent pas être obtenues directement par les mesures expérimentales. En conséquence, le calcul de la profondeur de plongée sera une étape cruciale pour l'identification suivante. Comme montré sur la figure 4.10, la position initiale de l'outil du robot en  $z$  direction ( $Z_0$ ) est définie comme la position spécifique lorsque le fond de la broche est en contact avec la surface supérieure de la pièce. Dans ce cas, la profondeur de plongée  $d_p$  est juste le déplacement de l'outil en  $z$  direction ( $\Delta Z$ ), et elle peut être calculée par (4.7) et (4.8). Cependant, la position réelle de l'outil  $P$  est différente de la position commandée de l'outil  $P_c$  en raison des flexibilités des articulations du robot, donc elle n'est pas obtenue directement et la profondeur de plongée sera extrêmement compliquée. Selon la définition ci-dessus, trois méthodes en vue de calculer la profondeur de plongée  $d_p$  sont proposées ici.

La première méthode est basée sur les positions angulaires estimées des axes du robot qui sont obtenues à partir de l'observateur non-linéaire discret présenté par [4, 73], et la profondeur de plongée peut être calculée par les équations de (4.9) à (4.11). Bien que cette méthode basée sur l'observateur est extrêmement simple, les inconvénients sont évidents aussi: l'observateur non-linéaire discret utilisé dans les expériences produit un déphasage dû à l'utilisation de filtres en ligne, et le modèle de frottement utilisé dans l'observateur n'est pas précis. La deuxième méthode est basée principalement sur le modèle de flexibilité du robot à articulation flexible présenté dans la section 2.4. Selon la figure 4.11, la profondeur de plongée peut être calculée par les équations de (4.12) à (4.18). Dans cette méthode, il convient de noter que le vecteur du couple externe  $T_e$  peut être donné approximativement en supposant que tous les composants sont égales à zéro sauf la force axiale, de plus, les forces estimées par l'observateur peuvent également être utilisées. La troisième méthode est semblable à la deuxième, et elle est basée sur le modèle de flexibilité du robot aussi. La seule différence est le calcul du couple au niveau de la sortie de la transmission. Conformément à l'équation (2.53), la profondeur de plongée peut être obtenue par les équations de (4.21) à (4.25).

Ensuite, la dérivée de la profondeur de plongée peut être approchée par l'équation (4.26). En fait, les données obtenues à partir des mesures et des calculs contiennent généralement des bruits importants, par conséquent, le filtrage des données est essentiel pour obtenir une meilleure performance d'identification. Plusieurs filtres sont utilisés, y compris un filtre médian, un filtre coupe-bande, un filtre Butterworth passe-bas du cinquième ordre. La figure 4.12 montre la comparaison entre la force axiale mesurée et la force axiale filtrée. Par ailleurs, il convient de noter que la profondeur de plongée  $d_p$  peut être changée pendant la phase de soudage du procédé FSW en utilisant la commande de position dans  $z$  direction. Plus précisément, la force de référence change de 4000  $N$  à 9000  $N$ . Après 4.1  $s$ , une excitation (16.7  $mm/s$ ) dans la direction  $z$  du repère de l'outil est ajouté durant 72  $ms$  en vue d'obtenir des paramètres plus précis du modèle dans l'identification. En outre, en utilisant trois méthodes différentes pour calculer la profondeur de plongée  $d_p$  et le filtrage des données, la comparaison entre les données (la profondeur de plongée  $d_p$  et sa dérivée  $\frac{d}{dt}d_p$ ) calculées et filtrées est illustrée sur les figures 4.13, 4.14 et 4.15.

## Identification du modèle dynamique linéaire proposé de la force axiale et analyse d'erreur

Selon les travaux de recherche de [104], la relation entre la force axiale mesurée  $F_z$  et la position verticale réelle de l'outil  $Z$  pendant le procédé FSW robotisé peut être décrite à l'aide d'un modèle simple de masse localisée avec une masse ( $m$ ), deux ressorts avec des rigidités différentes ( $K_{rz}, K_e$ ) et un amortisseur avec un facteur d'amortissement ( $f_e$ ), comme montré sur la figure 4.16. Donc, les équations qui régissent la physique du modèle de masse localisée sont obtenues dans (4.29) et un modèle dynamique linéaire qui peut décrire la force axiale à l'aide de la profondeur de plongée et de sa dérivée peut être défini par (4.30).

La méthode des moindres carrés (LS) est une approche standard pour trouver la solution approximative, ce qui permet de minimiser la somme des carrés des erreurs

pour les problèmes d'identification du modèle. Cette méthode est largement utilisée dans l'analyse de régression et de l'estimation. Tout d'abord, le modèle dynamique linéaire de la force axiale dans l'équation (4.30) doit être transformé sous forme matricielle (4.31), et les paramètres peuvent être choisis par (4.32). Le paramètre  $\gamma$  représente le vecteur des paramètres inconnus comprenant  $K_e$ ,  $f_e$  and  $C$  qui doivent être identifiés. Selon la méthode des moindres carrés, la solution du vecteur  $\gamma$  peut être estimée par (4.33). Donc les paramètres inconnus  $K_e$ ,  $f_e$  et  $C$  dans le modèle dynamique linéaire de la force axiale peuvent être identifiés en utilisant les équations ci-dessus. En utilisant les données des mesures et des calculs, voir les figures de 4.12 à 4.15, les trois modèles linéaires identifiés peuvent être obtenus par les équations (4.41), (4.42) et (4.43).

Afin d'analyser les performances de l'identification du modèle et la précision du modèle identifié, les indices d'évaluation suivants sont pris en considération principalement: les erreurs du modèle identifié, y compris l'erreur RMS, l'erreur maximale et l'erreur moyenne; les incertitudes des paramètres identifiés; le test de normalité des erreurs. En générale, le calcul des erreurs moyennes quadratiques (RMS) peut être considéré comme une méthode efficace pour évaluer la précision du modèle identifié. Les erreurs du modèle identifié de la force axiale peuvent être définies par (4.34). Ensuite, les erreurs RMS du modèle identifiés de la force axiale peuvent être exprimées en pourcentage dans (4.35). Par ailleurs, les incertitudes des paramètres identifiés peuvent également être utilisés pour évaluer la précision de l'identification du modèle. Pour le modèle dynamique linéaire identifié, le vecteur de l'erreur d'identification  $\varepsilon$  peut être calculé par (4.36). Afin de calculer les incertitudes des paramètres identifiés, le carré de l'écart-type et la variance peuvent être calculés par (4.38) et (4.39). Ainsi, en utilisant la solution estimée ( $\hat{\gamma}$ ) du modèle identifié, le vecteur des incertitudes des paramètres identifiés peut être calculé par (4.40). De plus, le test de normalité des erreurs peut être réalisé à l'aide de la fonction *lillietest* du logiciel *Matlab* qui peut effectuer un test de Lilliefors pour les erreurs du modèle identifié, ce qui peut déterminer si les erreurs proviennent d'une distribution normale.

Des études systématiques sur les erreurs des modèles linéaires identifiés de la force axiale ont été réalisées afin d'analyser l'efficacité et la performance de l'identification du modèle. Les erreurs RMS des trois modèles linéaires sont calculées comme étant respectivement de 4.47%, 7.75% et 4.75%. Tous les autres résultats des calculs d'erreur, y compris l'écart-type, les erreurs maximales et les erreurs moyennes pour les trois modèles linéaires identifiés sont indiqués dans le tableau 4.2. Les valeurs des paramètres identifiés et leurs incertitudes pour les trois modèles linéaires sont données dans les tableaux 4.3, 4.4 et 4.5. Les figures 4.17, 4.18 et 4.19 montrent les comparaisons entre les forces axiales filtrées et les forces axiales d'ajustement ainsi que les erreurs générées pour les trois modèles linéaires identifiés. Comme montré sur la figure 4.17, dans l'ensemble, les force axiales d'ajustement sont en bon accord avec les forces axiales filtrées, et quelques parties ne correspondent pas exactement en raison des décalages de phase introduits par l'observateur non-linéaire discret. Toutefois, il convient de noter que les déphasages peuvent être éliminés si le filtre en ligne peut être appliqué. En revanche, les résultats ne sont pas satisfaisants sur les figures 4.18 et 4.19, ce qui sont peut-être du au modèle imprécis de frottement et

au calcul approximatif utilisé dans la seconde et la troisième méthode. Finalement, les tests de normalité des erreurs sont effectués et les histogrammes des erreurs et les gaussiennes estimées pour les trois modèles linéaires avec le filtrage des données sont montrées sur les figures 4.20, 4.21 et 4.22. Toutes les analyses et les résultats ci-dessus montrent les bonnes performances de l'identification du modèle.

## Identification du modèle dynamique non-linéaire proposé de la force axiale et analyse d'erreur

Selon l'équation (4.4), un modèle dynamique non-linéaire de la force axiale pendant le procédé FSW a été proposé par [107] en utilisant la profondeur de plongée  $d_p$  comme l'entrée. La fonction de transfert du modèle dynamique de second ordre peut être exprimée par (4.44). En outre, parce que les déformations élastiques et plastiques des matériaux sont inégalement réparties dans le plan  $xy$  pendant le procédé FSW, la force axiale  $F_z$  peut être considérée comme la résultante de toutes les composantes des forces dans la direction  $z$ . En conséquence, un modèle dynamique non-linéaire peut être défini par (4.45). En prenant le logarithme naturel de l'équation (4.45), le modèle dynamique non-linéaire peut premièrement être transformé en une forme linéaire comme dans (4.46) pour utiliser la méthode des moindres carrés pour l'identification des paramètres. Dans ce cas, l'identification des paramètres du modèle non-linéaire peut être divisée en deux étapes. Tout d'abord, une valeur fixe doit être accordée au paramètre de modèle  $\beta$ . Ensuite, cette équation peut être transformé en une forme matricielle dans (4.48). De même, les paramètres inconnus  $A$  et  $\alpha$  peuvent être calculés directement en utilisant (4.33), et le modèle non-linéaire correspondant de la force axiale peut être obtenu. En second lieu, en utilisant la méthode d'essai et d'erreur pour le paramètre du modèle  $\beta$ , beaucoup de solutions approximatives pour l'identification du modèle non-linéaire peuvent être obtenues de manière à trouver un meilleur paramètre  $\beta$  avec une précision suffisante, c'est-à-dire, le paramètre  $\beta$  peut être choisi en se basant sur les précisions du modèle non-linéaire de la force axiale et des autres paramètres du modèle. Finalement, le paramètre du modèle  $\beta$  peut être sélectionné comme 423.1, et le modèle dynamique non-linéaire de la force axiale peut être donné par (4.52).

Pour l'analyse d'erreur, la méthode de calcul de l'incertitude du modèle linéaire ne convient pas pour le modèle non-linéaire. Afin de résoudre ce problème, il est nécessaire de calculer les dérivées partielles de la force axiale  $F_z$  par rapport aux paramètres du modèle identifiés  $A$ ,  $\alpha$  et  $\beta$ , ce qui peut être désigné par la matrice  $J_a$  dans (4.49). Pour le calcul des incertitudes des paramètres identifiés du modèle non-linéaire, l'erreur de la force axiale et la matrice  $J_a$  doivent être utilisées pour remplacer respectivement le vecteur d'erreur d'identification  $\varepsilon$  et la matrice de mesure  $X$  dans les équations (4.38) et (4.39). Donc les nouvelles équations sont obtenues comme (4.50) et (4.51). Enfin, les incertitudes des paramètres identifiés du modèle non-linéaire peuvent être calculées en utilisant l'équation (4.40).

En utilisant l'équation (4.35), l'erreur RMS du modèle identifié non-linéaire est de 5.35%. Tous les autres résultats des calculs d'erreur, y compris l'écart-type, les erreurs maximales et les erreurs moyennes pour les trois modèles linéaires identifiés sont donnés dans le tableau 4.6. En outre, les valeurs des paramètres identifiés

et leurs incertitudes pour le modèle non-linéaire sont fournies dans le tableau 4.7. La figure 4.23 montre la comparaison entre la force axiale filtrée et la force axiale d'ajustement ainsi que les erreurs générées pour le modèle non-linéaire identifié. Dans l'ensemble, la force axiale d'ajustement est essentiellement identique à la force axiale filtrée. De plus, le test de normalité des erreurs est effectué et l'histogramme des erreurs et la gaussienne estimée pour le modèle non-linéaire avec le filtrage des données sont montrés sur la figure 4.24. Les résultats démontrent les bonnes performances de l'identification du modèle.

## Conclusion

Dans ce chapitre, la modélisation dynamique et l'identification du procédé de soudage FSW robotisé ont été présentées. La force axiale  $F_z$  est modélisée en utilisant la profondeur de plongée  $d_p$  et sa dérivée  $\frac{d}{dt}d_p$  dans cette étude. Pendant le procédé de soudage FSW robotisé, parce que la force extérieure exercée sur le robot est extrêmement grande, la déformation du robot ne peut pas être négligée et la flexibilité du robot doit être prise en compte. Ainsi, la position réelle de l'outil est différente de la position commandée de l'outil, et la profondeur de plongée ne peuvent pas être obtenue directement par les mesures du robot. Le modèle dynamique identifié peut être utilisé pour calculer les forces en utilisant les états du robot sans la réalisation de l'opération de soudage réel, ce qui a contribué à établir le simulateur pour le procédé FSW. Parce que les modèles dynamiques des forces du procédé sont grandement nécessaires pour la conception des stratégies de commande en robotique du procédé FSW, en particulier pour la conception du contrôleur de force du robot, les modèles peuvent être utilisés pour établir les lois de commande pour compenser les erreurs de positionnement du robot. En outre, les modèles dynamiques des forces de procédé peuvent être utilisés pour analyser la stabilité du contrôleur de force du robot. En premier lieu, une brève description du procédé FSW y compris ses trois étapes principales et les forces du procédé a été donnée, suivi par la présentation des deux modèles de puissance non-linéaires statiques des forces du procédé. Avec l'aide du système FSW robotisé, des expériences pour l'identification ont été effectuées et trois méthodes de calcul pour la profondeur de plongée ont été proposées sur la base du modèle de flexibilité articulaire et de l'observateur non-linéaire présenté par [4, 71, 72, 73], qui peut être utilisé pour estimer la position angulaire réelle de l'articulation dans les expériences afin d'obtenir la profondeur de plongée nécessaire. Par le moyen du filtrage des données, les paramètres des modèles dynamiques linéaires et non-linéaires pour la force axiale dans le procédé ont été identifiés selon la méthode des moindres carrés. Enfin, des études systématiques sur les erreurs des modèles identifiés ont été réalisées et les comparaisons entre les forces axiales filtrées et calculées ont été effectuées. En conséquence, des précisions acceptables des modèles identifiés peuvent être obtenues sur la base des erreurs RMS, des incertitudes des paramètres identifiés et des tests de normalité, ce qui démontre la bonne performance de l'identification.

## 7.5 Conception d'un contrôleur robuste en force pendant le procédé de soudage FSW robotisé

### Introduction

Ce chapitre se concentre sur la proposition de contrôleurs robustes de force pour les robots manipulateurs industriels et le développement d'un système de simulation pour le procédé de soudage FSW robotisé. Selon l'identification des paramètres pour le modèle de déplacement du robot rigide et le modèle dynamique de la force axiale pendant le procédé FSW, la modélisation du procédé FSW robotisé peut être accomplie et la conception des contrôleurs de force peut être effectuée au moyen de la procédure de conception systématique [95] basée sur la méthode de réponse en fréquence. En outre, le simulateur du procédé FSW robotisé est établi afin de simuler le procédé FSW robotisé et de tester les performances des contrôleurs de force. Par ailleurs, l'oscillation de la force axiale pendant le procédé FSW est étudiée expérimentalement. Selon les mesures de hauteur des surfaces soudées, un modèle de perturbation de la position verticale de référence ( $Z_0(t)$ ) est proposé et appliqué lors de simulations.

La commande de l'interaction physique entre le robot et l'environnement est cruciale afin de mettre en œuvre avec succès une variété d'applications industrielles où l'effecteur du robot est en contact avec l'environnement [7]. La quantité qui décrit l'état de l'interaction de manière efficace est la force de contact sur l'effecteur du robot [5]. Au cours des trois dernières décennies, un grand nombre de recherches théoriques et expérimentales ont été réalisées sur les stratégies de commande de la force du robot, y compris plusieurs revues de haute qualité de Whitney [114], Patarinski et Botev [115], Zeng et Hemami [116], De Schutter et al. [117], Chiaverini et al. [118] et Yoshikawa [119]. Une monographie dans ce domaine a été écrite par Siciliano et Villani [120]. De plus, les connaissances fondamentales des méthodes de commande de la force du robot peuvent également être trouvées dans les livres de Siciliano et al. [5], Sciciliano et Khatib [7], Craig [62], Khalil et Dombre [63].

Selon le livre de Siciliano et Villani [120], les stratégies de commande de la force peuvent être classées comme des méthodes indirectes de commande de la force et les méthodes directe de commande de la force. Dans la première catégorie, la commande de la force est accomplie en utilisant la commande de mouvement sans fermeture explicite d'une boucle de retour de force, comme la commande de rigidité et la commande d'impédance. Au contraire, la force de contact peut être commandée à une valeur désirée au moyen de la fermeture d'une boucle de retour de force, comme la commande hybride de position/force, la commande de position interne/force externe et la commande parallèle de force/position. En outre, les méthodes de commande de la force peuvent également être classées comme des méthodes de base de commande de la force et des méthodes avancées de commande de la force. Les méthodes de base de commande de la force comprennent la commande de rigidité (ou la commande de compliance) [116, 127, 128, 129, 130, 131], la commande d'impédance (ou la commande d'admittance) [132, 133, 134, 135, 136, 137, 138, 139, 140], la commande hybride de position/force [59, 141, 142, 143, 144, 145, 146], la commande hybride de vitesse/force [121], la commande hybride d'impédance [122], la com-

mande parallèle de force/position [123], la commande explicite de la force [124, 125], la commande implicite de la force [126], etc. Par ailleurs, les méthodes avancées de commande de la force sont basées sur les méthodes avancées de commande intégrées ou combinées avec les méthodes de base ci-dessus, y compris la commande adaptative de force [147, 148, 149, 150], la commande hybride robuste de position/force [151, 152], la commande robuste d'impédance [134, 135], la commande de force basée sur l'algorithme d'apprentissage [153, 154, 155, 156].

## Stratégie de commande de la force et la modélisation du système pour le procédé FSW robotisé

Pendant le procédé FSW robotisé, l'outil de la broche FSW installé sur le robot manipulateur est en contact avec les pièces, en outre, la force de contact doit être commandée pour garantir des performances appropriées de soudage. En particulier, la force axiale  $F_z$  devrait être contrôlée comme une valeur constante désirée, car elle a une contribution importante sur l'obtention d'une bonne qualité de la soudure. Afin d'assurer une régulation de la force de contact à une valeur désirée lors de l'interaction du robot industriel avec l'environnement, un grand nombre de stratégies de commande peuvent être prises en compte. En générale, dans la plupart des méthodes de commande, les erreurs mesurées de force sont directement converties en couples de l'actionneur qui seront appliqués au niveau des articulations du robot [117]. Cependant, ces méthodes de commande sont difficiles à réaliser sur les robots industriels avec une architecture fermée de contrôleur où seule la position de l'outil ou la vitesse de l'outil peut être contrôlée. En fait, la plupart des manipulateurs commerciaux ont des contrôleurs incorporés de position et un accès direct aux couples de moteur n'est pas autorisé. Par rapport aux autres méthodes de commande, dans la commande de mouvement interne/force externe, les erreurs mesurées de force sont converties en position ou en vitesse désirée, qui est exécutée par une boucle de commande de position ou de vitesse. Par conséquent, la plupart des contrôleurs de force réalisés sur les robots industriels sont basés sur la structure en cascade qui a une boucle externe de commande en force fermée autour d'une boucle interne de commande de mouvement [125, 126, 151, 157, 158, 159, 160]. Parce que le robot industriel KUKA KR500-2MT considéré dans cette étude peut être commandé uniquement en utilisant son interface de position, la stratégie de commande de la force pour le système du robot peut être choisie comme la commande de position interne/force externe, comme montré sur la figure 5.1. En même temps, le robot industriel sera commandé en position dans la direction  $x$  et  $y$ , et la régulation de la force sera effectuée dans la direction  $z$ , ce qui est également connue comme la commande hybride de position/force.

Il nous faut considérer aussi que la modélisation du système dans l'espace cartésien est nécessaire afin de concevoir le contrôleur de force pour le procédé FSW robotisé. Comme montré sur la figure 5.2, le système de l'interaction entre le robot manipulateur et l'environnement pendant le procédé FSW peut être décrit en utilisant un modèle simple de masse localisée avec les paramètres de masse, de rigidité et d'amortissement dans l'espace cartésien. Dans cette figure, le contrôleur de position du robot industriel comprend l'interpolateur, la cinématique inverse utilisée pour

transformer la trajectoire désirée de l'espace cartésien dans l'espace articulaire, le contrôleur de position articulaire du robot et la cinématique du robot.  $P_d$  est la position désirée de l'outil qui est l'entrée du contrôleur de position du robot.  $q_d$  et  $q_c$  sont les vecteurs des positions angulaires désirées et des positions angulaires commandées respectivement.  $P_c$  est la position commandée de l'outil qui est aussi la sortie du contrôleur de position du robot.  $P$  est la position réelle de l'outil et  $P_0$  est la position de référence (la position de contact de force zéro).  $K_{Rc}$  et  $f_{Rc}$  sont la rigidité articulaire et l'amortissement articulaire dans l'espace cartésien.  $M_{Rc}$  représente l'inertie et la gravité du robot dans l'espace cartésien.  $K_{ec}$  et  $f_{ec}$  sont la rigidité et l'amortissement de l'environnement dans l'espace cartésien.

En conséquence, le modèle du système peut être divisé en deux parties: la première partie est le modèle de déplacement du robot rigide qui peut transformer la position désirée de l'outil  $P_d$  à la position commandée de l'outil  $P_c$ , et la deuxième partie est le modèle d'interaction entre le robot et l'environnement qui peut être utilisé pour obtenir la position réelle de l'outil  $P$  à partir de la position commandée de l'outil  $P_c$ . Parce que la force axiale doit être contrôlée comme une valeur constante désirée pendant le procédé FSW, on considère seulement le modèle de masse localisée dans la direction  $z$ . En utilisant les paramètres correspondant dans la direction  $z$ , les équations qui régissent la physique du modèle de masse localisée sont obtenues dans (5.1) et (5.2). En appliquant une transformée de Laplace, la fonction de transfert qui établit un rapport entre la force axiale  $F_z$  et la position verticale commandée de l'outil  $Z_c$  et la position verticale de référence  $Z_0$  peut être donnée par les équations (5.3), (5.4), (5.5), (5.6) et (5.7).

Cette fonctions de transfert sont extrêmement importantes pour la conception du contrôleur de force, en particulier pour  $G_{Pc}(s)$ . Dans le système de commande de la force axiale, la position verticale de référence  $Z_0$  peut être considérée comme la perturbation de position dans la direction  $z$ . Les paramètres du modèle dynamique de force axiale  $K_e$  et  $f_e$  ont déjà été identifiés dans le chapitre précédent. Selon les travaux de recherche de Abele [161] et Dumas [162], les paramètres du modèle de masse localisée peuvent être calculés dans (5.8), (5.9) et (5.10). Il est à noter que des approximations existent dans les calculs des paramètres, par exemple, seul le frottement visqueux est prise en compte dans le calcul du facteur d'amortissement. En outre, ces paramètres changent continuellement pendant le procédé FSW en raison des configurations différentes du robot, comme montré sur la figure 5.3. Parce que les variations des paramètres du modèle sont relativement petites, ces paramètres peuvent être calculés en utilisant les valeurs initiales. De plus, les fonctions de transfert  $G_{Pc}(s)$  et  $G_{P0}(s)$  peuvent être transformées sous forme standard.

## Identification des paramètres du modèle de déplacement du robot rigide

Afin de concevoir le contrôleur de force robuste pour le robot industriel actuel KUKA, l'identification des paramètres du modèle de déplacement du robot rigide doit être effectuée tout d'abord. En ce qui concerne l'identification du modèle, les tests expérimentaux ont été effectués sans contact avec les pièces à usiner, c'est-à-dire, le robot industriel peut être considéré comme rigide dans les tests. Par con-



séquent, le modèle identifié sera le modèle de déplacement du robot rigide. L'entrée utilisée pour cette identification (aussi appelée déplacement désiré de l'outil) est un signal d'entrée en échelon de 2 mm dans la direction  $z$ ,  $y$  et  $x$  respectivement, ce qui est effectué par l'envoi de la correction de position à partir d'un ordinateur externe au robot KUKA tous les 12 ms. La sortie (aussi appelée déplacement commandé de l'outil) est une réponse à un échelon qui peut être calculée en utilisant le modèle géométrique direct (DGM) et la position angulaire mesurée du moteur après la réduction de vitesse. Dans cette section, l'identification des paramètres pour le modèle de déplacement du robot rigide sera effectuée en utilisant deux algorithmes fournis par la boîte à outils (System Identification Toolbox en anglais) dans le logiciel *Matlab*: l'estimation par fonction de transfert et l'estimation par modèle de procédé.

Concernant l'estimation par fonction de transfert, un modèle linéaire continu du second ordre a été proposé [125]. En outre, un temps de retard dans le modèle peut être considéré également [151]. Selon la recherche de Bigras [140], un modèle de troisième ordre par fonction de transfert qui constitue une relativement bonne approximation des données expérimentales pour le robot industriel KUKA KR270 est obtenu par (5.17). De même, pour le contrôleur de position du robot industriel KUKA KR500-2MT, le déplacement peut être modélisé par un système linéaire continu de  $n^{eme}$  ordre dans le domaine fréquentiel. Ensuite, l'identification des paramètres du modèle de déplacement du robot rigide est réalisée en utilisant l'estimation par fonction de transfert et les meilleurs modèles du troisième ordre par fonction de transfert dans la direction  $z$ ,  $y$  et  $x$  sont obtenus par les équations (5.19), (5.20) et (5.21). En outre, les incertitudes des paramètres du modèle, les valeurs de "Fit" et les erreurs RMS des modèles identifiés dans les trois directions peuvent être calculées dans le tableau 5.1. A la lumière de l'algorithme fourni par la boîte à outils, la valeur de "Fit" peut être calculée par (5.22). Cette valeur est calculée afin de trouver le meilleur modèle qui peut minimiser la somme des carrés de la différence entre les données de sortie mesurées pour la validation et les données de sortie simulées à l'aide du modèle identifié. Par ailleurs, les réponses des fonctions de transfert dans trois directions ainsi que la comparaison avec l'ensemble des données expérimentales sont fournies sur les figures 5.4, 5.5 et 5.6. Le diagramme de Bode de trois modèles de type fonction de transfert peut être trouvé dans la figure 5.7. Bien que les résultats montrent de bonnes précisions des modèles identifiés selon les valeurs "Fit" et les erreurs RMS, les incertitudes des paramètres des modèles ne sont pas satisfaisantes. Ainsi, les modèles par fonction de transfert ne peuvent être utilisés dans la conception du contrôleur de force et une nouvelle identification du modèle de déplacement du robot rigide sera effectuée.

Pour l'estimation par modèle du procédé, la relation entre la position désirée de l'outil et la position commandée de l'outil peut être approximée par le modèle de procédé du premier ordre basée sur [95]. Cependant, le modèle de déplacement du robot rigide dans cette étude sera approchée par un modèle de procédé du second ordre ou celui de troisième ordre, comme formulées par (5.24) et (5.25). Ensuite, en utilisant le même ensemble de données expérimentales, les paramètres des modèles de procédé du second ordre dans la direction  $z$ ,  $y$  et  $x$  peuvent être identifiés dans le tableau 5.2. Concernant les précisions des modèles identifiés, les incertitudes des

paramètres du modèle, les valeurs de "Fit" et les erreurs RMS des modèles identifiés dans les trois directions peuvent être calculées dans le tableau 5.3. Finalement, les réponses aux modèles par fonctions de transfert, les modèles de procédé du second ordre et les modèles de procédé du troisième ordre dans les trois directions ainsi que la comparaison avec l'ensemble des données expérimentales sont illustrées sur les figures 5.4, 5.5 et 5.6. Le diagramme de Bode correspondant des trois types de modèles dans la direction  $z$  sont donnés dans la figure 5.7. Les résultats ci-dessus démontrent les excellentes performances de l'identification du modèle, de plus, les modèles de procédé identifiés sont importants pour la conception d'un contrôleur de force robuste.

## Conception du contrôleur de force robuste

Après l'identification des paramètres du modèle de déplacement du robot rigide, la conception du contrôleur de force robuste peut être réalisée en trois étapes principales, qui sont basées sur la procédure de conception systématique des contrôleurs de force pour robots industriels présentée dans [95]. Dans la conception du contrôleur de force, la cinématique complète, la réponse de position du contrôleur de robot industriel (le modèle de déplacement du robot rigide), le modèle de flexibilité articulaire et le modèle dynamique de force axiale dans le procédé sont pris en compte. Les contrôleurs de force obtenus peuvent assurer des gains élevés et les marges suffisantes de phase pour le système en boucle fermée, ce qui indique la robustesse du système de commande.

Pour la description du système entier de commande, selon le schéma de la commande position intérieure/force extérieure illustrée à la figure 5.1 et les équations (5.5) à (5.7), un bloc-diagramme du système de commande de la force axiale est donné sur la figure 5.12. La fonction de transfert du contrôleur de force à concevoir peut être donnée par (5.30). Afin de concevoir le contrôleur de force, il est extrêmement utile de fournir la fonction de transfert du système à boucle ouverte, qui est exprimée par (5.31).

Concernant la procédure de calcul, la première étape consiste à spécifier les performances désirées du contrôleur de force, y compris la rapidité et la robustesse du système de commande. Tout d'abord, le principal paramètre caractérisant la rapidité du système de commande lors de la réponse à une entrée de référence est le temps de montée  $t_r$ . Les valeurs du temps de montée pour les types de performance différents pendant le procédé FSW peuvent être définies selon les travaux de recherche de Natale [95] dans le tableau 5.4. En plus du temps de montée  $t_r$ , les marges de phase  $m_\varphi$  pour différents types de surface de contact sont fournis dans le tableau 5.4. Si la rigidité de l'environnement est élevée, une marge de phase grande doit être utilisée pour la conception du contrôleur de la force. Dans la deuxième étape, une structure appropriée du contrôleur de la force doit être conçue afin d'obtenir un contrôleur de force approprié qui peut être utilisé pour atteindre les performances désirées, y compris le temps de montée désiré et la marge de phase désirée. En conséquence, la performance à l'état stable et la performance transitoire doivent être prises en compte. Donc, le gain statique  $K_{FC}$  peut être obtenu par (5.32), et la fréquence de coupure de gain souhaitée peut être calculée en utilisant le temps de montée désiré

et la relation approximative (5.33) [163]. Par conséquent, le décalage de gain et de déphasage fournis par la partie restante du contrôleur de force ( $G_{FC}^*(s)$ ) sont obtenus par les équations (5.34) et (5.36). Selon les signes du décalage de gain exprimé en décibels et le décalage de phase, quatre cas peuvent se produire et trois types de structures de contrôleurs peuvent être appliqués dans les cas correspondants. Le graphique de Nichols est donné sur la figure 5.13, ce qui peut fournir la magnitude et la phase de la fonction de transfert en boucle ouverte. En choisissant la structure appropriée du contrôleur, les valeurs des paramètres correspondants doivent être calculés afin de garantir les performances du contrôleur désirées. Selon la structure du contrôleur, les méthodes de calcul des paramètres du contrôleur peuvent être réparties en deux catégories. La première peut être utilisée pour la commande avec une structure à un seul étage, comme le contrôleur de type "phase-lag" et le contrôleur de type "phase-lead". Le dernier peut être effectuée pour obtenir les paramètres du contrôleur avec une structure à deux étages comme le contrôleur de type "phase-lead-lag". Les paramètres du contrôleur peuvent être obtenue par les équations dans ces deux types de méthodes de calcul. En conséquence, des contrôleurs robustes de force pour le procédé FSW robotisé ont été obtenus pour des performances différentes et les diagrammes de Bode des trois contrôleurs de force et des systèmes en boucle ouverte correspondant peuvent être vus sur les figures 5.14 et 5.15.

## Simulation du procédé FSW robotisé

Afin d'établir un simulateur de haute fidélité pour le procédé FSW robotisé, il est nécessaire d'en apprendre davantage sur l'architecture de commande du robot industriel KUKA considérée dans cette étude. En réalité, la commande du robot KUKA est exécutée par un ordinateur industriel spécialement développé avec le système opérationnel *WindowsNT*. D'après les informations des fichiers de configuration, un diagramme schématique de la commande du robot KUKA pour le procédé FSW peut être déduit, comme représenté sur la figure 5.16. Le système de commande complet comprend le générateur de trajectoire, le contrôleur de mouvement des articulations du robot, le modèle dynamique des moteurs, le modèle dynamique du robot à articulation flexible et le modèle de procédé FSW.

En effet, le générateur de trajectoire du robot industriel est réalisé par les programmes de commande écrits par l'utilisateur. Comme montré sur la figure 5.16, la génération de trajectoire en temps réel peut être divisée en deux étapes successives: la génération des états désirés dans l'espace cartésien ( $X_d, \dot{X}_d, \ddot{X}_d$ ) pour une période d'échantillonnage de 12 ms, et la génération de l'état articulaire désiré ( $q_d, \dot{q}_d, \ddot{q}_d$ ) pour une période d'échantillonnage de 2 ms. Par ailleurs, la génération de trajectoire se rapportant aux mouvements du robot KUKA peut être réalisée par trois fonctions fournies par le langage de programmation du contrôleur du robot: *ptp*, *lin* and *circ*. En outre, le contrôleur de mouvement articulaire du robot peut être considéré comme la combinaison d'une boucle d'asservissement de position articulaire avec un temps d'échantillonnage de 2 ms et une boucle de commande de vitesse articulaire avec un temps d'échantillonnage de 0.5 ms sur la base des documents techniques fournis par le fabricant de robot. Pendant ce temps, la commande d'asservissement des courants

des moteurs synchrones sont réalisées par les variateurs de vitesse (Siemens) avec un temps d'échantillon de  $0.2 \text{ ms}$ , ce qui est représenté par un gain constant  $K_a$  dans le schéma.

Dans le contrôleur du robot industriel comme représenté sur la figure 5.16, la linéarisation du robot peut être effectuée en utilisant une méthode de découplage basée sur le modèle de robot rigide et le calcul du couple moteur de référence selon les états angulaires articulaires désirés  $(q_d, \dot{q}_d, \ddot{q}_d)$ , comme montré dans les équations (5.58) à (5.75). En conséquence, le gain  $K_a$  de la commande du courant de moteur dans le schéma de commande du robot KUKA peut être calculé par (5.74), et les courants des moteurs commandés peuvent être donnés par (5.75).

Les robots concernés dans cette étude sont des robots commandés par un contrôleur KUKA qui utilise un correcteur P (proportionnel) pour l'asservissement de position et un correcteur PI (proportionnel-intégral) pour l'asservissement de la vitesse. Par conséquent, une structure de régulation similaire peut également être réalisée dans le simulateur correspondant. La figure 5.17 montre le schéma du contrôleur de mouvement articulaire du simulateur dans le domaine de temps continu. Ainsi, les régulateurs P et PI peuvent être réglés en utilisant la méthode de placement de pôles, et les paramètres de commande  $K_p$ ,  $K_v$  et  $K_i$  peuvent être déterminés en plaçant trois pôles du système en boucle fermée le plan complexe. En fait, la commande en temps réel de la position et la vitesse angulaire articulaire est réalisée par le contrôleur du robot KUKA avec les temps d'échantillonnage de  $2 \text{ ms}$  et  $0.5 \text{ ms}$  respectivement. Par conséquent, le contrôleur de mouvement articulaire du simulateur dans le domaine de temps discret peut être établi sur la figure 5.18. De même, les paramètres de commande  $K_p$ ,  $K_v$  et  $K_i$  peuvent être déterminés au moyen de placement de pôles aussi. Il convient de noter que les valeurs réelles de ces paramètres de commande de robot peuvent être utilisés directement dans le simulateur, comme montré dans le tableau 5.5.

Selon le modèle du contrôleur de mouvement du robot, le simulateur du procédé FSW robotisé peut être établi dans l'environnement *Matlab/Simulink*, et le schéma du simulateur est représentée sur la figure 5.21. En plus du modèle de contrôleur de mouvement du robot, les présentations détaillées des autres modèles construits dans le simulateur, y compris le modèle de calcul des couples moteurs commandés, le modèle dynamique du moteur, le modèle de flexibilité et le modèle dynamique direct du robot, sont donnés par les équation (5.84), (5.85), (5.86) et (5.87). Les schémas des modèles ci-dessus sont présentés sur la figure 5.19. De plus, le modèle de la force axiale ( $F_z$ ) pendant le procédé FSW peut être utilisé directement dans le simulateur. Des études en simulation ont été réalisées pour tester la rapidité et la robustesse du contrôleur de la force ainsi que la performance du simulateur du procédé FSW robotisé. En choisissant les types de performance différents et les types de surface dans le tableau 5.4, trois contrôleurs de forces différents peuvent être obtenus et leurs fonctions de transfert peuvent être données par les équations (5.55), (5.56) et (5.57). Par conséquent, les régulateurs de la force en temps discrets correspondant peuvent également être obtenus [163]. Dans la simulation du procédé FSW robotisé, le robot est commandé pour souder le long de la  $x$  direction dans le repère de l'outil avec la vitesse constante d'avance ( $v = 400 \text{ mm/min}$ ) et la vitesse de rotation ( $\Omega = 1100 \text{ tr/min}$ ). En outre, la force axiale souhaitée est de  $9000 \text{ N}$

et la force axiale initiale est de 2000  $N$ . La figure 5.22 montre la comparaison des forces axiales dans les simulations avec trois contrôleurs de force différents. Les résultats de simulation ont prouvés les bonnes performances des contrôleurs conçus de force, comme montées sur la figure 5.6. Par ailleurs, la vitesse de l'outil ainsi que la comparaison de la position de l'outil souhaitée et la position réelle de l'outil dans la simulation peuvent être trouvées dans les figures 5.23, 5.24, 5.25 and 5.26.

## Analyse des vibrations de la force axiale

Pendant le procédé FSW, une vibration importante peut être observée dans les mesures de la force axiale  $F_{zm}$ , comme montré sur la figure 5.27. Dans cette section, l'oscillation de la force axiale peut être analysée par l'identification d'un modèle de force axiale expérimentale. Ensuite, un modèle de perturbation de la position verticale de référence ( $Z_0(t)$ ) est proposé et appliqué à la simulation du procédé FSW robotique.

Selon les travaux de recherche de Zaeh et Gebhard [96], les forces de procédé FSW et leur impact sur la structure de la machine ont été analysées et un modèle de force de procédé expérimental a été proposé et validé par de nombreuses expériences de soudage. Le modèle de la force de procédé expérimental proposé est composé des composantes de force statique ( $F_{i0}$ ) et la superposition des équations d'ondes avec des amplitudes différentes ( $A_{ij}$ ), les phases ( $\varphi_{ij}$ ) et la pulsation ( $\omega_r$ ) de la vitesse de rotation ( $\Omega$ ) et son premier harmonique, comme exprimée par l'équation (5.91). En outre, la fréquence de l'équation de la force axiale peut être vérifiée en analysant le spectre de la force axiale pendant le procédé FSW, comme représenté sur la figure 5.28. En conséquence, la fréquence de l'équation d'onde dans le modèle de la force axiale peut être obtenu comme 18.26  $Hz$ , ce qui est approximativement la fréquence de la vitesse de rotation de la broche ( $f_r = 1100/60 \simeq 18.33 Hz$ ). De plus, la force axiale statique  $F_{z0}$  et l'amplitude  $A_z$  peuvent être identifiées comme 9000  $N$  et 1250  $N$ , et la phase  $\varphi_z$  peuvent être négligée sur la base de l'analyse de la figure 5.27 et 5.28. Ainsi, la force axiale  $F_z$  dans l'étape de soudage du procédé FSW peut être exprimée par (5.94).

Dans le procédé FSW robotisé réel, les variations périodiques de hauteur de la surface pour la pièce à souder peuvent être clairement observées, comme montré sur la figure 5.29. Au moyen d'un scanner laser, les valeurs des coordonnées  $x$  et  $z$  pour la surface de la pièce soudée peuvent être obtenues sur la figure 5.30. Afin d'éliminer l'influence du bruit sur la hauteur de la surface soudée, une relation linéaire entre les coordonnées  $z$  et  $x$  peut être obtenue en utilisant la méthode des moindres carrés dans (5.95). En conséquence, les variations de hauteur de la surface de la pièce à souder peuvent être calculées par (5.96), et le spectre de variations de hauteur de la surface soudée dans le procédé FSW peut être donné sur la figure 5.32. Ainsi, un modèle de perturbation de la position verticale de référence ( $Z_0(t)$ ) est proposé et identifié sous la forme sinusoïdale comme (5.97), et la fréquence de la perturbation sinusoïdale peut être obtenu comme 18.33  $Hz$  qui est presque la même que la fréquence de la vitesse de rotation de la broche ( $f_r = 1100/60 \simeq 18.33 Hz$ ). Comme montré sur la figure 5.12 et l'équation (5.5), la position verticale de référence ( $Z_0(t)$ ) peut être considérée comme la perturbation de position verticale dans le

système de commande de la force axiale dans (5.98). De plus, la perturbation de vitesse peut également être obtenue par (5.99), la force axiale dans le simulateur peut être calculée par (5.100). Enfin, la simulation du procédé FSW robotisé avec le modèle de perturbation peut être effectuée, et la figure 5.33 montre l'historique du temps de la force axiale dans la simulation, ce qui démontre les bonnes performances des contrôleurs de force proposés. Par ailleurs, il peut être observé que l'amplitude d'oscillation de la force axiale simulée est d'environ 1300 N, ce qui correspond à l'amplitude d'oscillation de la force axiale mesurée représentée sur la figure 5.27.

## Conclusion

Dans ce chapitre, la conception des contrôleurs robustes en force pour les manipulateurs de robots industriels et le développement d'un système de simulation pour le procédé de soudage FSW robotisé sont proposés. Tout d'abord, la stratégie de commande position intérieure/force extérieure a été sélectionnée pour le système robotique parce que le robot industriel KUKA KR 500-2MT peut être commandé seulement par le moyen de son interface de position. Ensuite, la modélisation du procédé FSW robotisé est accomplie, en considérant simultanément la cinématique complète, la réponse de position du contrôleur de robot industriel (le modèle de déplacement du robot rigide), le modèle de flexibilité articulaire et le modèle dynamique de force axiale dans le procédé. L'identification des paramètres pour le modèle de déplacement du robot rigide a été effectuée sur la base des deux algorithmes (y compris l'estimation par fonction de transfert et l'estimation par modèle de procédé) fournis par la boîte à outils (System Identification Toolbox en anglais) dans le logiciel *Matlab*. Avec l'aide de la procédure de conception systématique [95] basée sur la méthode de réponse en fréquence et la modélisation du procédé FSW robotisé, les contrôleurs de force ont été conçus avec des gains élevés et des marges suffisantes de phase pour le système en boucle fermée, ce qui assure la robustesse du système de commande. En outre, le simulateur du procédé FSW robotisé est développé et les études de simulation ont été réalisées. En conséquence, les résultats de simulation ont montrés de bonnes performances des contrôleurs de force. Finalement, la vibration de la force axiale pendant le procédé FSW a été analysée par l'identification d'un modèle expérimental de force axiale [96]. Selon les mesures de hauteur des surfaces soudées, un modèle de perturbation de la position verticale de référence ( $Z_0(t)$ ) est proposé. Ce modèle qui peut conduire à l'oscillation de la force axiale a été utilisé en simulation. Les résultats de simulation ont montrés l'efficacité du modèle de perturbation et les bonnes performances des contrôleurs de force.

## 7.6 Conclusions générales et perspectives

### Conclusions générales

Pendant le procédé de soudage FSW robotisé, parce que la force extérieure exercée sur le robot est extrêmement grande, la déformation du robot ne peut pas être négligée et la flexibilité du robot doit être prise en compte. Donc certains problèmes concernant la modélisation et la commande du robot surgissent. Cette thèse a présenté la modélisation des robots manipulateurs industriels à articulations flexibles en vue de la commande robuste en force utilisé en soudage FSW, et les contributions principales de cette recherche peuvent être données en détail comme suit:

- a. **Un état de l'art relatif à la recherche a été présenté.** Un état de l'art relatif à les robots manipulateurs industriels à articulations flexibles a été présenté sur la base des articles publiés actuels, ainsi que d'une revue de littérature sur quatre aspects des problèmes étroitement liés à cette recherche. Tout d'abord, la brève définition, l'histoire et le développement actuel des robots industriels ont été donnés et plusieurs types de robots industriels célèbres dans l'histoire ont été introduits. Les ventes mondiales des robots industriels dans les dix dernières années ont été fournies, montrant la croissance rapide du marché robotique mondial. Ensuite, les méthodes de la modélisation des robots manipulateurs industriels à articulations flexibles ont été discutées, et plusieurs méthodes différentes de commande pour les robots à articulations flexibles ont été revues, telles que la perturbation singulière et la variété intégrale, la linéarisation par rétroaction, le système en cascade et la commande intégrale backstepping, la commande PD, la commande adaptative, la commande robuste, la commande par réseaux de neurones, la commande floue et quelques autres méthodes de commande. En outre, une revue de littérature sur les méthodes de commande en force du robot a également été donnée dans le cinquième chapitre.
- b. **La modélisation d'un robot manipulateur à articulations flexibles avec 6 degrés de liberté a été présentée.** Conformément aux connaissances fondamentales, la modélisation des robots manipulateurs à articulations flexibles avec 6 degrés de liberté a été présentée, y compris les modèles géométriques directs (DGM) et inverses (IGM), les modèles cinématiques directs (DKM) et inverses (IKM), les modèles dynamiques directs (DDM) et inverses (IDM). En premier lieu, les concepts de base de la robotique ont été introduits, tels que les descriptions spatiales, les transformations homogènes, les angles fixes X-Y-Z et la convention Denavit-Hartenberg modifiée, etc. Ensuite, les problèmes de cinématique direct et inverse ont été résolus en utilisant respectivement les matrices de transformation homogène pour tous les repères adjacents et la méthode de Paul. Les problèmes de cinématique instantanée directe et inverse ont également été tranchés en introduisant la matrice Jacobienne de vitesse et sa matrice inverse. De plus, les équations dynamiques pour le corps du robot et les moteurs peuvent être obtenues avec l'aide de l'approche Euler-Lagrange. Il faut noter que le modèle de flexibilité articulaire, le modèle de frottement au niveau des axes du robot et du moteur, ainsi

que le couple du compensateur de gravité au deuxième axe ont été discutés et pris en considération pour la modélisation dynamique du robot. Tous les modèles mentionnés ci-dessus sont extrêmement importants et essentiels pour la simplification des modèles dynamiques des robots industriels, la simulation du procédé FSW robotisé et la conception du contrôleur robuste en force du robot.

- c. Une nouvelle approche de simplification pour les modèles dynamiques des robots industriels a été proposée par le moyen de la méthode d'intervalle.** Afin de réduire le temps de calcul et l'occupation de la mémoire, une nouvelle approche de simplification pour les modèles dynamiques des robots industriels a été proposée basée sur la méthode d'intervalle. Concernant le modèle dynamique symbolique du manipulateur industriel à 6 degrés de liberté généré par le logiciel *SYMORO+*, toutes les expressions de composants dans la matrice d'inertie  $M(q)$  et le vecteur des couples centrifuge, de Coriolis et de gravité  $H(q, \dot{q})$  montrent les complexités et les non-linéarités fortes, et un couplage élevé existe également entre les entrées de commande. Ainsi, un algorithme généralisé de simplification basé sur l'intervalle a été proposé en vue de réduire la complexité du modèle du robot. Le processus de la simplification peut être divisé en trois étapes. Après l'exemple de simplification, les résultats et l'analyse des erreurs du modèle simplifié pour l'espace global de travail du robot ainsi que la simplification supplémentaire correspondante ont été donnés, ce qui peut être applicable à des trajectoires arbitraires. Ensuite, des études de cas ont été réalisées sur trois trajectoires différentes de test, y compris la trajectoire du procédé d'identification, la trajectoire linéaire et la trajectoire circulaire du procédé FSW. Les comparaisons entre les composants simplifiés et originales de la matrice  $M(q)$ , le vecteur  $H(q, \dot{q})$  et le vecteur  $T$  ont été effectuées, et les résultats de simulation démontrent l'efficacité et les bonnes performances de la méthode de simplification. De plus, les couples dans l'équation dynamique du robot a également été analysés dans le cas d'un procédé FSW linéaire. L'algorithme de simplification proposé peut être généralisé à les autres modèles dynamiques (ex. les avions et les hélicoptères), et le modèle simplifié peut être appliqué à de nombreuses méthodes de commande afin de simplifier la structure de commande et d'améliorer la performance de commande en temps réel, telles que la commande de couple calculé et la commande basée sur l'observateur.
- d. Les paramètres d'inertie négligeables du modèle de robot ont été trouvés en utilisant l'algorithme de simplification proposé pour le modèle dynamique du robot.** Selon l'algorithme de simplification proposé pour le modèle dynamique du robot, les paramètres d'inertie négligeables du modèle de robot ont été trouvés, ce qui est extrêmement important pour l'optimisation de la trajectoire pour l'identification du modèle de robot. Les paramètres d'inertie identifiés seront plus précis lorsque les paramètres d'inertie négligeables peuvent être enlevés. Par ailleurs, lorsque un nouvel axe est ajouté au robot industriel existant, l'influence des paramètres du nouvel



axe peut être analysée en utilisant la méthode de simplification, ce qui sera extrêmement bénéfique pour la conception du nouveau robot.

- e. La modélisation dynamique et l'identification du procédé FSW robotisé ont été réalisées sans les capteurs pour mesurer directement la position de l'outil.** La modélisation dynamique et l'identification du procédé de soudage FSW robotisé ont été réalisées sans les capteurs qui peuvent mesurer directement la position de l'outil du robot. Tout d'abord, une brève description du procédé FSW y compris ses trois étapes principales et les forces de procédé ont été données, suivi par la présentation des deux modèles de puissance non-linéaires statiques des forces de procédé. Ensuite, la force axiale du procédé FSW peut être modélisée en utilisant la profondeur de plongée et sa dérivée. Toutefois, la profondeur de plongée ne peut pas être obtenue directement par les mesures du robot parce que la position réelle de l'outil est différente de la position commandée de l'outil pendant le procédé FSW en raison de la flexibilité du robot. Avec l'aide du système FSW robotisé, des expériences pour l'identification ont été effectuées et trois méthodes de calcul pour la profondeur de plongée ont été proposées sur la base du modèle de flexibilité articulaire et l'observateur non-linéaire présenté par [4, 71, 72, 73], qui peut être utilisé pour estimer la position angulaire réelle de l'articulation dans les expériences afin de obtenir la profondeur de plongée nécessaire. Par le moyen du filtrage des données, les paramètres des modèles dynamiques linéaires et non-linéaires pour la force axiale dans le procédé ont été identifiés selon la méthode des moindres carrés. Enfin, des études systématiques sur les erreurs des modèles identifiés ont été réalisées et les comparaisons entre les forces axiales filtrées et calculées ont été effectuées. En conséquence, des précisions acceptables des modèles identifiés peuvent être obtenues sur la base des erreurs RMS, les incertitudes de paramètres identifiés et les tests de normalité, ce qui démontre la bonne performance de l'identification. Il faut noter que les modèles identifiés peuvent être utilisés pour établir le simulateur du procédé FSW robotisé. De plus, ces modèles identifiés contribuent également à la conception du contrôleur de force du robot et à l'analyse de la stabilité du système de commande.
- f. Les contrôleurs robustes de force ont été conçus pour le procédé FSW robotisé par le moyen de la méthode de réponse en fréquence.** Conformément à la modélisation du procédé FSW robotisé qui considère simultanément la cinématique complète, la réponse de position du contrôleur de robot industriel (le modèle de déplacement du robot rigide), le modèle de flexibilité articulaire et le modèle dynamique de force axiale dans le procédé, les contrôleurs robustes de force ont été conçus en utilisant la procédure de conception systématique [95] basée sur la méthode de réponse en fréquence. En premier lieu, la stratégie de commande position intérieure/force extérieure a été sélectionnée pour le système robotique parce que le robot industriel KUKA KR500-2MT peut être commandé seulement par le moyen de son interface de position. Ensuite, l'identification des paramètres pour le modèle de déplacement du robot rigide a été effectuée sur la base des deux algorithmes (y compris l'estimation par fonction de transfert et l'estimation par modèle de procédé)

fournis par la boîte à outils (System Identification Toolbox en anglais) dans le logiciel *Matlab*. Finalement, les contrôleurs de force ont été conçus avec des gains élevés et des marges suffisantes de phase pour le système en boucle fermée, ce qui implique la robustesse du système de commande.

- g. Un simulateur du FSW robotisé a été développé et un modèle de perturbation concernant la vibration de la force axiale a été proposé.** Afin de simuler le procédé FSW robotisé, le simulateur correspondant a été développé sous *Matlab/Simulink* et les résultats de simulations ont montrés de bonnes performances des contrôleurs conçus pour l'asservissement de force. Ensuite, la vibration de la force axiale pendant le procédé FSW a été analysée par l'identification d'un modèle expérimental de force axiale [96]. En outre, un modèle de perturbation de la position verticale de référence est proposé selon les mesures de hauteur des surfaces soudées. Ce modèle qui peut conduire à l'oscillation de la force axiale a été utilisé dans la simulation. Les résultats de simulation ont vérifié l'efficacité du modèle de perturbation et les bonnes performances des contrôleurs de force.

## Perspectives

Selon la recherche dans cette thèse, plusieurs problèmes concernant la modélisation et la commande robuste de force pour les robots manipulateurs industriels utilisés dans le procédé FSW ont été résolus. Cependant, plusieurs directions de recherche intéressantes possibles pour les travaux futurs doivent être effectués davantage dans les aspects suivants:

- Les paramètres d'inertie négligeables obtenus par la simplification du modèle dynamique peuvent être utilisés dans l'optimisation de la trajectoire en vue de l'identification du modèle de robot. Par ailleurs, la méthode de simplification peut également être utilisée pour analyser l'influence des paramètres du nouvel axe lorsque un nouvel axe est ajouté au robot industriel existant.
- Les modèles dynamiques obtenus de force axiale dans le procédé ne sont pas précis, car les paramètres du modèle ont été identifiés en utilisant la profondeur calculée de plongée basée sur le modèle de flexibilité articulaire et l'observateur non-linéaire. Afin d'améliorer la précision de l'identification, il est nécessaire d'utiliser des capteurs supplémentaires qui peuvent mesurer directement la position de l'outil du robot industriel dans l'espace cartésien.
- Dans l'intention d'obtenir une bonne qualité de la soudure, bien que la contribution de la force axiale  $F_z$  est plus importante que la force d'avance  $F_x$  et la force latérale  $F_y$ , des capteurs supplémentaires de force peuvent être appliqués afin d'obtenir les modèles des forces du procédé dans les autres directions. Par conséquent, la commande des forces dans les autres directions peut être également réalisée en plus de la commande de la force axiale. actuelle.
- Les contrôleurs conçus de force ont déjà été validés par des simulations, et des expériences correspondantes doivent être encore réalisées en vue de valider l'efficacité et les bonnes performances des contrôleurs de force.

- Concernant la vibration de la force axiale dans le procédé FSW, un nouveau système de commande en force peut être conçu dans le but d'atténuer les vibrations de la force, ce qui améliorera la qualité de la soudure.
- Les validations expérimentales de la compensation basée sur l'observateur ont été effectués juste pour les trajectoires linéaires et circulaires du procédé FSW dans un plan. De plus, seul l'erreur de positionnement dans la direction perpendiculaire à la trajectoire de soudage a été corrigée [4]. Des expériences peuvent être réalisées pour des trajectoires de soudage plus complexes dans l'espace tridimensionnel et des corrections pour les autres directions et orientations peuvent également être utilisées.

# Bibliography

- [1] W. M. Thomas, E. D. Nicholas, J. C. Needham, M. G. Murch, P. Temple-Smith, and C. J. Dawes, “International patent application pct/ gb92/02203 and gb patent application,” The Welding Institute, London, UK, London, Tech. Rep. 9125978.8, December 1991.
- [2] S. Zimmer, “Contribution to the industrialization of friction stir welding,” Ph.D. dissertation, Arts et Métiers ParisTech (ENSAM), Metz, France, 2009, in French.
- [3] R. S. Mishra and Z. Y. Ma, “Friction stir welding and processing,” *Materials Science and Engineering, Reports*, vol. 50, no. 1, pp. 1–78, 2005.
- [4] J. Qin, “Robust hybrid position/force control of a manipulator used in machining and/or in friction stir welding,” Ph.D. dissertation, Arts et Métiers ParisTech (ENSAM), Metz, France, 2013, in French.
- [5] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, *Robotics: Modelling, Planning and Control*. Springer-Verlag London Limited, 2009.
- [6] ISO, “Industrial robot as defined by iso 8373,” International Organization for Standardization, Tech. Rep., 2012, <http://www.iso.org>.
- [7] B. Siciliano and O. Khatib, *Springer Handbook of Robotics*. Springer-Verlag Berlin Heidelberg, 2008.
- [8] M. W. Spong, S. Hutchinson, and M. Vidyasagar, *Robot Modeling and Control*. Wiley New York, 2006, vol. 3.
- [9] IFR, “History of industrial robots: Milestones of technology and commercialization,” International Federation of Robotics, Tech. Rep., 2012, <http://www.ifr.org/downloads>.
- [10] V. D. Scheinman, “Design of a computer controlled manipulator,” Ph.D. dissertation, Stanford University, Department of Computer Science, Stanford, 1968.
- [11] I. Bonev, “Delta parallel robot: the story of success,” *Newsletter*, available at <http://www.parallelmic.org>, 2001.

- [12] IFR, “Industrial robot statistics: World robotics 2014 industrial robots,” International Federation of Robotics, Tech. Rep., 2014, <http://www.ifr.org/industrial-robots/statistics>.
- [13] ABB-Robotics, <http://new.abb.com/products>.
- [14] KUKA-Robotics, <http://www.kuka-robotics.com/en/products>.
- [15] FANUC-Robotics, <http://robot.fanucamerica.com/products>.
- [16] Yaskawa-Robotics, <http://www.motoman.com/products>.
- [17] C. Dumas, “Development of methods for metal and composite parts trimming with a robot,” Ph.D. dissertation, Université de Nantes, Nantes, France, 2011, in French.
- [18] L. M. Sweet and M. C. Good, “Redefinition of the robot motion-control problem,” *IEEE Control Systems Magazine*, vol. 5, no. 3, pp. 18–25, 1985.
- [19] M. Good, L. Sweet, and K. Strobel, “Dynamic models for control system design of integrated robot and drive systems,” *Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME*, vol. 107, no. 1, pp. 53–59, 1985.
- [20] M. Spong, “Modeling and control of elastic joint robots,” *Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME*, vol. 109, no. 4, pp. 310–319, 1987.
- [21] M. W. Spong, K. Khorasani, and P. V. Kokotovic, “An integral manifold approach to the feedback control of flexible joint robots,” *IEEE Journal of Robotics and Automation*, vol. 3, no. 4, pp. 291–300, 1987.
- [22] K. Khorasani, “Nonlinear feedback control of flexible joint manipulators: A single link case study,” *IEEE Transactions on Automatic Control*, vol. 35, no. 10, pp. 1145–1149, 1990.
- [23] P. Tomei, “A simple pd controller for robots with elastic joints,” *IEEE Transactions on Automatic Control*, vol. 36, no. 10, pp. 1208–1213, 1991.
- [24] S. Moberg, “On modeling and control of flexible manipulators,” Ph.D. dissertation, Linköping University, Linköping, Sweden, 2007.
- [25] M. Bridges and D. Dawson, “Redesign of robust controllers for rigid-link flexible-joint robotic manipulators actuated with harmonic drive gearing,” *IEEE Proceedings: Control Theory and Applications*, vol. 142, no. 5, pp. 508–514, 1995.
- [26] S. Ge, “Adaptive controller design for flexible joint manipulators,” *Automatica*, vol. 32, no. 2, pp. 273–277, 1996.

- [27] K. Khorasani, "Adaptive control of flexible-joint robots," *IEEE Transactions on Robotics and Automation*, vol. 8, no. 2, pp. 250–267, 1992.
- [28] H. Taghiradt and M. Khosravi, "A robust linear controller for flexible joint manipulators," in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, vol. 3, Sendai, Japan, September 2004, pp. 2936–2941.
- [29] R. Amjadi, S. Khadem, and H. Khaloozadeh, "Position and velocity control of a flexible joint robot manipulator via a fuzzy controller based on singular perturbation analysis," in *Proceedings of IEEE International Conference on Fuzzy Systems*, vol. 1, Melbourne, Australia, December 2001, pp. 348–351.
- [30] F. Ghorbel, J. Y. Hung, and M. W. Spong, "Adaptive control of flexible-joint manipulators," *IEEE Control Systems Magazine*, vol. 9, no. 7, pp. 9–13, 1989.
- [31] F. Ghorbel and M. W. Spong, "Stability analysis of adaptively controlled flexible joint manipulators," in *Proceedings of the IEEE Conference on Decision and Control*, vol. 4, Honolulu, HI, USA, December 1990, pp. 2538–2544.
- [32] —, "Adaptive integral manifold control of flexible joint robot manipulators," in *Proceedings of IEEE International Conference on Robotics and Automation*, vol. 1, Nice, France, May 1992, pp. 707–714.
- [33] K. Khorasani, "Robust adaptive stabilization of flexible joint manipulators," in *Proceedings of IEEE International Conference on Robotics and Automation*, Scottsdale, AZ, USA, May 1989, pp. 1194–1199.
- [34] R. Al-Ashoor, R. Patel, and K. Khorasani, "Robust adaptive controller design and stability analysis for flexible-joint manipulators," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 23, no. 2, pp. 589–602, 1993.
- [35] Y.-C. Liu, M.-H. Jin, and H. Liu, "Singular perturbation control for flexible-joint manipulator based on flexibility compensation," *Jiqiren/Robot*, vol. 30, no. 5, pp. 460–466, 2008.
- [36] W. M. Grimm, "Robustness analysis of nonlinear decoupling for elastic-joint robots," *IEEE Transactions on Robotics and Automation*, vol. 6, no. 3, pp. 373–377, 1990.
- [37] H. Sira-Ramirez and M. W. Spong, "Variable structure control of flexible joint manipulators," *International Journal of Robotics and Automation*, vol. 3, no. 2, pp. 57–64, 1988.
- [38] S. Ge, T. Lee, and E. Tan, "Adaptive neural network control of flexible joint robots based on feedback linearization," *International Journal of Systems Science*, vol. 29, no. 6, pp. 623–635, 1998.
- [39] S. Ider and M. Özgören, "Trajectory tracking control of flexible-joint robots," *Computers and Structures*, vol. 76, no. 6, pp. 757–763, 2000.

- 
- [40] A. De Luca and L. Lanari, "Robots with elastic joints are linearizable via dynamic feedback," in *Proceedings of the IEEE Conference on Decision and Control*, vol. 4, New Orleans, LA, USA, December 1995, pp. 3895–3897.
- [41] A. De Luca and P. Lucibello, "General algorithm for dynamic feedback linearization of robots with elastic joints," in *Proceedings of IEEE International Conference on Robotics and Automation*, vol. 1, Leuven, Belgium, May 1998, pp. 504–510.
- [42] J. Yuan and Y. Stepanenko, "Composite adaptive control of flexible joint robots," *Automatica*, vol. 29, no. 3, pp. 609–619, 1993.
- [43] R. Colbaugh and K. Glass, "Adaptive task-space control of flexible-joint manipulators," *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 20, no. 2-4, pp. 225–249, 1997.
- [44] K. P. Chen and L. C. Fu, "Nonlinear adaptive motion control for a manipulator with flexible joints," in *Proceedings of IEEE International Conference on Robotics and Automation*, vol. 2, Scottsdale, AZ, USA, May 1989, pp. 1201–1206.
- [45] A. Beneallegue and N. Sirdi, "Passive control for robot manipulators with elastic joints," in *Proc. IMACS MCTS 91, Modeling and Control of Technological Systems*, vol. 1, Lille, France, 1991, pp. 481–487.
- [46] D. Dawson, Z. Qu, M. Bridges, and J. Carroll, "Robust tracking of rigid-link flexible-joint electrically-driven robots," in *Proceedings of the IEEE Conference on Decision and Control*, vol. 2, Brighton, UK, December 1991, pp. 1409–1412.
- [47] S. Nicosia and P. Tomel, "Design of global tracking controllers for flexible-joint robots," *Journal of Robotic Systems*, vol. 10, no. 6, pp. 835–846, 1993.
- [48] C. Kwan and F. Lewis, "Robust backstepping control of nonlinear systems using neural networks," *IEEE Transactions on Systems, Man, and Cybernetics Part A: Systems and Humans.*, vol. 30, no. 6, pp. 753–766, 2000.
- [49] C. Macnab and G. D'Eleuterio, "Neuroadaptive control of elastic-joint robots using robust performance enhancement," *Robotica*, vol. 19, no. 6, pp. 619–629, 2001.
- [50] M. Bridges, D. Dawson, and C. Abdallah, "Control of rigid-link, flexible-joint robots: a survey of backstepping approaches," *Journal of Robotic Systems*, vol. 12, no. 3, pp. 199–216, 1995.
- [51] A. Albu-Schäffer and G. Hirzinger, "A globally stable state feedback controller for flexible joint robots," *Advanced Robotics*, vol. 15, no. 8, pp. 799–814, 2001.
- [52] M.-C. Chien and A.-C. Huang, "Adaptive control for flexible-joint electrically driven robot with time-varying uncertainties," *IEEE Transactions on Industrial Electronics*, vol. 54, no. 2, pp. 1032–1038, 2007.

- [53] R. Lozano and B. Brogliato, "Adaptive control of robot manipulators with flexible joints," *IEEE Transactions on Automatic Control*, vol. 37, no. 2, pp. 174–181, 1992.
- [54] M.-C. Chien and A.-C. Huang, "Adaptive impedance controller design for flexible-joint electrically-driven robots without computation of the regressor matrix," *Robotica*, vol. 30, no. 1, pp. 133–144, 2012.
- [55] D. Kim and W. Oh, "Robust control design for flexible joint manipulators: Theory and experimental verification," *International Journal of Control, Automation and Systems*, vol. 4, no. 4, pp. 495–505, 2006.
- [56] J. Yeon and J. Park, "Practical robust control for flexible joint robot manipulators," in *Proceedings of IEEE International Conference on Robotics and Automation, ICRA 2008*, Pasadena, CA, United States, May 2008, pp. 3377–3382.
- [57] Y.-C. Chang and H.-M. Yen, "Design of a robust position feedback tracking controller for flexible-joint robots," *IET Control Theory and Applications*, vol. 5, no. 2, pp. 351–363, 2011.
- [58] F. Abdollahi, H. Talebi, and R. Patel, "A stable neural network-based observer with application to flexible-joint manipulators," *IEEE Transactions on Neural Networks*, vol. 17, no. 1, pp. 118–129, 2006.
- [59] S. Chiaverini, B. Siciliano, and L. Villani, "Force/position regulation of compliant robot manipulators," *IEEE Transactions on Automatic Control*, vol. 39, no. 3, pp. 647–652, 1994.
- [60] A. De Luca, C. Manes, and G. Ulivi, "Robust hybrid dynamic control of robot arms," in *Proceedings of the IEEE Conference on Decision and Control*, vol. 3, Tampa, FL, USA, December 1989, pp. 2641–2646.
- [61] S. Ozgoli and H. D. Taghirad, "A survey on the control of flexible joint robots," *Asian Journal of Control*, vol. 8, no. 4, pp. 332–344, 2006.
- [62] J. J. Craig, *Introduction to Robotics: Mechanics and Control*. Pearson Prentice Hall Upper Saddle River, 2005, vol. 3.
- [63] W. Khalil and E. Dombre, *Modeling, Identification and Control of Robots*, ser. Kogan Page Science paper edition. Oxford: Elsevier Science, 2004.
- [64] D. L. Peiper, "The kinematics of manipulators under computer control," Ph.D. dissertation, Stanford University, Stanford, 1968.
- [65] R. P. Paul, *Robot Manipulators: Mathematics, Programming, and Control*. MIT Press, 1981.
- [66] B. Armstrong-Helouvry, *Control of machines with friction*. Springer Science & Business Media, 2012, vol. 128.



- 
- [67] C. de Wit, H. Olsson, K. Astrom, and P. Lischinsky, “A new model for control of systems with friction,” *IEEE Transactions on Automatic Control*, vol. 40, no. 3, pp. 419–425, 1995.
- [68] K. Wisama and L. Philippe, *SYMORO+ Symbolic Modeling of Robots, User’s Guide*, IRCCyN, Nantes, France, October 2003.
- [69] W. Khalil and D. Creusot, “Symoro+: A system for the symbolic modelling of robots,” *Robotica*, vol. 15, no. 2, pp. 153–161, 1997.
- [70] K. Wang, F. Léonard, and G. Abba, “A novel approach for simplification of industrial robot dynamic model using interval method,” in *IEEE/ASME International Conference on Advanced Intelligent Mechatronics, AIM 2014*, Besancon, France, July 2014, pp. 1697–1703.
- [71] J. Qin, F. Léonard, and G. Abba, “Non-linear observer-based control of flexible-joint manipulators used in machine processing,” in *Proceedings of the ASME 11th Biennial Conference on Engineering Systems Design and Analysis, ESDA 2012*, vol. 2, Nantes, France, July 2012, pp. 251–260.
- [72] —, “Experimental external force estimation using a non-linear observer for 6 axes flexible-joint industrial manipulators,” in *The 9th Asian Control Conference, ASCC 2013*, Istanbul, Turkey, June 2013, pp. 701–706.
- [73] —, “Nonlinear discrete observer for flexibility compensation of industrial robots,” in *The 19th World Congress of the International Federation of Automatic Control*, Cape Town, South Africa, August 2014, pp. 5598–5604.
- [74] O. Nilsson, “On modeling and nonlinear model reduction in automotive systems,” Ph.D. dissertation, Lund University, Lund, Sweden, 2009.
- [75] S. Lall, J. Marsden, and S. Glavaški, “A subspace approach to balanced truncation for model reduction of nonlinear control systems,” *International Journal of Robust and Nonlinear Control*, vol. 12, no. 6, pp. 519–535, 2002.
- [76] D. Vasilyev, M. Rewieński, and J. White, “Macromodel generation for biomems components using a stabilized balanced truncation plus trajectory piecewise-linear approach,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 25, no. 2, pp. 285–293, 2006.
- [77] O. Nilsson and A. Rantzer, “A novel nonlinear model reduction method applied to automotive controller software,” in *Proceedings of the American Control Conference*, St. Louis, MO, USA, June 2009, pp. 4587–4592.
- [78] R. Moore, R. Kearfott, and M. Cloud, *Introduction to Interval Analysis*. Philadelphia, PA, USA: SIAM, 2009.
- [79] L. Jaulin, *Applied Interval Analysis: With Examples in Parameter and State Estimation, Robust Control and Robotics*. London: Springer, 2001.

- [80] M. Kieffer, L. Jaulin, and E. Walter, “Guaranteed recursive nonlinear state estimation using interval analysis,” in *Proceedings of the IEEE Conference on Decision and Control*, vol. 4, Tampa, FL, USA, December 1998, pp. 3966–3971.
- [81] M. Kieffer, L. Jaulin, E. Walter, D. Meizel, *et al.*, “Guaranteed mobile robot tracking using interval analysis,” in *Proceedings of MISC’99, Workshop on Application of Interval Analysis to System and Control*, Girona, Spain, February 1999, pp. 347–360.
- [82] M. Kieffer, L. Jaulin, . Walter, and D. Meizel, “Robust autonomous robot localization using interval analysis,” *Reliable Computing*, vol. 6, no. 3, pp. 337–362, 2000.
- [83] L. Jaulin and F. Le Bars, “An interval approach for stability analysis: Application to sailboat robotics,” *IEEE Transactions on Robotics*, vol. 29, no. 1, pp. 282–287, 2013.
- [84] L. Jaulin, I. Braems, and E. Walter, “Interval methods for nonlinear identification and robust control,” in *Proceedings of the IEEE Conference on Decision and Control*, vol. 4, Las Vegas, NV, USA, December 2002, pp. 4676–4681.
- [85] A. Piazzzi and A. Visioli, “Global minimum-jerk trajectory planning of robot manipulators,” *IEEE Transactions on Industrial Electronics*, vol. 47, no. 1, pp. 140–149, 2000.
- [86] E. Lee, C. Mavroidis, and J. Merlet, “Five precision point synthesis of spatial rrr manipulators using interval analysis,” *Journal of Mechanical Design, Transactions of the ASME*, vol. 126, no. 5, pp. 842–849, 2004.
- [87] H. Fang, J. Chen, and L. Dou, “Dynamic self-collision detection and prevention for 2-dof robot arms using interval-based analysis,” *Journal of Mechanical Science and Technology*, vol. 25, no. 8, pp. 2077–2087, 2011.
- [88] M. Gouttefarde, D. Daney, and J.-P. Merlet, “Interval-analysis-based determination of the wrench-feasible workspace of parallel cable-driven robots,” *IEEE Transactions on Robotics*, vol. 27, no. 1, pp. 1–13, 2011.
- [89] M. Pac and D. Popa, “Interval analysis of kinematic errors in serial manipulators using product of exponentials formula,” *IEEE Transactions on Automation Science and Engineering*, vol. 10, no. 3, pp. 525–535, 2013.
- [90] A. Martini, “Modeling and flight control of an autonomous helicopter under wind gust,” Ph.D. dissertation, Metz University, Metz, France, 2008, in French.
- [91] L. Jaulin, *Interval Robotics*, ENSTA-Bretagne, LabSTICC, Brest, France, February 2012.
- [92] A. Goldsztejn and L. Jaulin, “Inner approximation of the range of vector-valued functions,” *Reliable Computing*, vol. 14, pp. 1–23, 2010.

- 
- [93] L. Jaulin and E. Walter, "Set inversion via interval analysis for nonlinear bounded-error estimation," *Automatica*, vol. 29, no. 4, pp. 1053–1064, 1993.
- [94] M. Gautier, S. Briot, and G. Venture, "Identification of consistent standard dynamic parameters of industrial robots," in *IEEE/ASME International Conference on Advanced Intelligent Mechatronics: Mechatronics for Human Well-being, AIM 2013*, Wollongong, NSW, Australia, July 2013, pp. 1429–1435.
- [95] G. Natale, R. Koeppe, and G. Hirzinger, "A systematic design procedure of force controllers for industrial robots," *IEEE/ASME Transactions on Mechatronics*, vol. 5, no. 2, pp. 122–131, 2000.
- [96] M. Zaeh and P. Gebhard, "Dynamical behaviour of machine tools during friction stir welding," *Production Engineering*, vol. 4, no. 6, pp. 615–624, 2010.
- [97] K. Wang, F. Léonard, and G. Abba, "Dynamic model identification of axial force in robotic friction stir welding," in *the 15th IFAC/IEEE/IFIP/IFORS Symposium on Information Control Problems in Manufacturing, INCOM 2015*, Ottawa, Canada, May 2015, to be published.
- [98] R. Nandan, T. DebRoy, and H. Bhadeshia, "Recent advances in friction-stir welding - process, weldment structure and properties," *Progress in Materials Science*, vol. 53, no. 6, pp. 980–1023, 2008.
- [99] A. Fehrenbacher, N. Duffie, N. Ferrier, F. Pfefferkorn, and M. Zinn, "Effects of tool-workpiece interface temperature on weld quality and quality improvements through temperature control in friction stir welding," *International Journal of Advanced Manufacturing Technology*, vol. 71, no. 1-4, pp. 165–179, 2014.
- [100] G. Cook, R. Crawford, D. Clark, and A. Strauss, "Robotic friction stir welding," *Industrial Robot*, vol. 31, no. 1, pp. 55–63, 2004.
- [101] C. B. Smith, J. F. Hinrichs, and W. A. Crusan, "Robotic friction stir welding: the state of the art," in *4th Friction Stir Welding International Symposium*, Waukesha, WI, USA, May 2003.
- [102] A. Bres, B. Monsarrat, L. Dubourg, L. Birglen, C. Perron, M. Jahazi, and L. Baron, "Simulation of friction stir welding using industrial robots," *Industrial Robot: An International Journal*, vol. 37, no. 1, pp. 36–50, 2010.
- [103] J. De Backer, A.-K. Christiansson, J. Oqueka, and G. Bolmsjö, "Investigation of path compensation methods for robotic friction stir welding," *Industrial Robot*, vol. 39, no. 6, pp. 601–608, 2012.
- [104] A. Fehrenbacher, C. Smith, N. Duffie, N. Ferrier, F. Pfefferkorn, and M. Zinn, "Combined temperature and force control for robotic friction stir welding," *Journal of Manufacturing Science and Engineering, Transactions of the ASME*, vol. 136, no. 2, 2014.

- [105] P. Ulysse, “Three-dimensional modeling of the friction stir-welding process,” *International Journal of Machine Tools and Manufacture*, vol. 42, no. 14, pp. 1549–1557, 2002.
- [106] C. Chen and R. Kovacevic, “Thermomechanical modelling and force analysis of friction stir welding by the finite element method,” *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, vol. 218, no. 5, pp. 509–520, 2004.
- [107] X. Zhao, P. Kalya, R. Landers, and K. Krishnamurthy, “Empirical dynamic modeling of friction stir welding processes,” *Journal of Manufacturing Science and Engineering, Transactions of the ASME*, vol. 131, no. 2, pp. 1–9, 2009.
- [108] S. Zimmer, L. Langlois, J. Laye, and R. Bigot, “Experimental investigation of the influence of the fsw plunge processing parameters on the maximum generated force and torque,” *International Journal of Advanced Manufacturing Technology*, vol. 47, no. 1-4, pp. 201–215, 2010.
- [109] S. Zimmer, L. Langlois, J. Laye, J.-C. Goussain, P. Martin, and R. Bigot, “Using the plunging and welding process windows to determine a fsw means of production,” *Advanced Materials Research*, vol. 89-91, pp. 697–702, 2010.
- [110] A. Gerlich, P. Su, and T. North, “Tool penetration during friction stir spot welding of al and mg alloys,” *Journal of Materials Science*, vol. 40, no. 24, pp. 6473–6481, 2005.
- [111] S. Zimmer, L. Langlois, J. Laye, J.-C. Goussain, P. Martin, and R. Bigot, “Influence of processing parameters on the tool and workpiece mechanical interaction during friction stir welding,” *International Journal of Material Forming*, vol. 2, no. SUPPL. 1, pp. 299–302, 2009.
- [112] Z. Allam, S. Zimmer-Chevret, L. Langlois, G. Abba, and R. Bigot, “Statistical model of the tool/workpiece mechanical interactions in fsw,” in *International Conference on Friction Stir Welding and Processing (FSWP 2015)*. Saint Etienne, France: Ecole des Mines de Saint Etienne, 2012.
- [113] Z. Allam, “Modeling and control of mechanical interactions in friction stir welding,” Master’s thesis, Arts et Métiers ParisTech (ENSAM), Metz, France, 2011, in French.
- [114] D. E. Whitney, “Historical perspective and state of the art in robot force control.” *International Journal of Robotics Research*, vol. 6, no. 1, pp. 3–14, 1987.
- [115] S. P. Patarinski and R. G. Botev, “Robot force control: a review,” *Mechatronics*, vol. 3, no. 4, pp. 377–398, 1993.
- [116] G. Zeng and A. Hemami, “An overview of robot force control,” *Robotica*, vol. 15, no. 5, pp. 473–482, 1997.

- 
- [117] J. De Schutter, H. Bruyninckx, W.-H. Zhu, and M. Spong, “Force control: A bird’s eye view,” in *Control Problems in Robotics and Automation*, ser. Lecture Notes in Control and Information Sciences, B. Siciliano and K. Valavanis, Eds. Springer Berlin Heidelberg, 1998, vol. 230, pp. 1–17.
- [118] S. Chiaverini, B. Siciliano, and L. Villani, “A survey of robot interaction control schemes with experimental comparison,” *IEEE/ASME Transactions on Mechatronics*, vol. 4, no. 3, pp. 273–285, 1999.
- [119] T. Yoshikawa, “Force control of robot manipulators,” in *Proceedings of IEEE International Conference on Robotics and Automation, ICRA 2000*, vol. 1, San Francisco, CA, USA, April 2000, pp. 220–226.
- [120] B. Siciliano and L. Villani, *Robot force control*. Springer Science & Business Media, 2012, vol. 540.
- [121] F. Jatta, G. Legnani, A. Visioli, and G. Ziliani, “On the use of velocity feedback in hybrid force/velocity control of industrial manipulators,” *Control Engineering Practice*, vol. 14, no. 9, pp. 1045–1055, 2006.
- [122] R. Anderson and M. Spong, “Hybrid impedance control of robotic manipulators,” *IEEE Journal on Robotics and Automation*, vol. 4, no. 5, pp. 549–556, 1988.
- [123] S. Chiaverini and L. Sciavicco, “The parallel approach to force/position control of robotic manipulators,” *IEEE Transactions on Robotics and Automation*, vol. 9, no. 4, pp. 361–373, 1993.
- [124] R. Cortesão, R. Koeppel, U. Nunes, and G. Hirzinger, “Explicit force control for manipulators with active observers,” in *IEEE International Conference on Intelligent Robots and Systems*, vol. 2, Takamatsu, Japan, November 2000, pp. 1075–1080.
- [125] R. Volpe and P. Khosla, “Theoretical and experimental investigation of explicit force control strategies for manipulators,” *IEEE Transactions on Automatic Control*, vol. 38, no. 11, pp. 1634–1650, 1993.
- [126] P. Rocco, G. Ferretti, and G. Magnani, “Implicit force control for industrial robots in contact with stiff surfaces,” *Automatica*, vol. 33, no. 11, pp. 2041–2047, 1997.
- [127] J. Salisbury, “Active stiffness control of a manipulator in cartesian coordinates.” in *Proceedings of the IEEE Conference on Decision and Control, CDC 1980*, vol. 1, Albuquerque, NM, USA, December 1980, pp. 95–100.
- [128] J. Salisbury and J. J. Craig, “Articulated hands - force control and kinematic issues.” *International Journal of Robotics Research*, vol. 1, no. 1, pp. 4–17, 1982.
- [129] E. Degoulange and P. Dauchez, “External force control of an industrial puma 560 robot,” *Journal of Robotic Systems*, vol. 11, no. 6, pp. 523–540, 1994.

- [130] H. Kazerooni, B. Waibel, and S. Kim, "On the stability of robot compliant motion control. theory and experiments," *Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME*, vol. 112, no. 3, pp. 417–426, 1990.
- [131] H. Zhang, "Kinematic stability of robot manipulators under force control," in *Proceedings of the IEEE International Conference on Robotics and Automation, ICRA 1989*, Scottsdale, AZ, USA, May 1989, pp. 80–85.
- [132] N. Hogan, "Impedance control: An approach to manipulation," in *Proceedings of the IEEE International Conference on American Control Conference, ACC 1984*, San Diego, CA, USA, June 1984, pp. 304–313.
- [133] —, "Impedance control: An approach to manipulation: Part ii - implementation." *Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME*, vol. 107, no. 1, pp. 8–16, 1985.
- [134] H. Kazerooni, P. K. Houpt, and T. B. Sheridan, "Robust compliant motion for manipulators, part i: The fundamental concepts of compliant motion," *Robotics and Automation, IEEE Journal of*, vol. 2, no. 2, pp. 83–92, 1986.
- [135] —, "Robust compliant motion for manipulators, part ii: Design method," *Robotics and Automation, IEEE Journal of*, vol. 2, no. 2, pp. 93–105, 1986.
- [136] R. Colbaugh, H. Seraji, and K. Glass, "Impedance control for dexterous space manipulators," in *Proceedings of the 31st IEEE Conference on Decision and Control, CDC 1992*, vol. 2, Tucson, AZ, USA, December 1992, pp. 1881–1886.
- [137] W. McCormick and H. Schwartz, "An investigation of impedance control for robot manipulators," *The International journal of robotics research*, vol. 12, no. 5, pp. 473–489, 1993.
- [138] N. Hogan, "On the stability of manipulators performing contact tasks," *IEEE Journal on Robotics and Automation*, vol. 4, no. 6, pp. 677–686, 1988.
- [139] D. A. Lawrence, "Impedance control stability properties in common implementations." Philadelphia, PA, USA, April 1988, pp. 1185–1190.
- [140] P. Bigras, M. Lambert, and C. Perron, "New formulation for an industrial robot impedance controller: Real-time implementation on a kuka robot," Montreal, QC, Canada, October 2007, pp. 2782–2787.
- [141] M. Mason, "Compliance and force control for computer controlled manipulators," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 11, no. 6, pp. 418–432, 1981.
- [142] M. H. Raibert and J. J. Craig, "Hybrid position/force control of manipulators," *Journal of Dynamic Systems, Measurement, and Control*, vol. 103, no. 2, pp. 126–133, 1981.

- 
- [143] O. Khatib, "A unified approach for motion and force control of robot manipulators: The operational space formulation," *IEEE Journal on Robotics and Automation*, vol. 3, no. 1, pp. 43–53, 1987.
- [144] R. DeSantis, "Motion/force control of robotic manipulators," *Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME*, vol. 118, no. 2, pp. 386–389, 1996.
- [145] H. Faessler, "Manipulators constrained by stiff contact. dynamics, control, and experiments," *International Journal of Robotics Research*, vol. 9, no. 4, pp. 40–58, 1990.
- [146] C. Stefano, S. Bruno, and V. Luigi, "A stable force/position controller for robot manipulator," *The 31st Conf. on Decision and Control*, pp. 1869–1874, 1992.
- [147] H. Seraji, "Adaptive force and position control of manipulators," *J. of Robotic Systems*, vol. 4, no. 4, pp. 551–578, 1987.
- [148] R. Colbaugh, H. Seraji, and K. Glass, "Direct adaptive impedance control of robot manipulators," *Journal of Robotic Systems*, vol. 10, no. 2, pp. 217–248, 1993.
- [149] R. Colbaugh and A. Engelmann, "Adaptive compliant motion control of manipulators: theory and experiments," in *Proceedings of IEEE International Conference on Robotics and Automation, ICRA 1994*, no. pt 4, 1994, pp. 2719–2726.
- [150] S. Singh and D. Popa, "An analysis of some fundamental problems in adaptive control of force and impedance behavior: Theory and experiments," *IEEE Transactions on Robotics and Automation*, vol. 11, no. 6, pp. 912–921, 1995.
- [151] S. Lessard, L.-G. Durand, G. Cloutier, P. Bigras, G. Soulez, and J. De Guise, "Robust position/force controller design on an industrial robot for medical application using lmi optimization," in *Proceedings of IEEE International Conference on Systems, Man and Cybernetics, SMC 2004*, vol. 3, The Hague, Netherlands, October 2004, pp. 2913–2917.
- [152] P. Bigras, M. Lambert, and C. Perron, "Robust force controller for industrial robots: Optimal design and real-time implementation on a kuka robot," *IEEE Transactions on Control Systems Technology*, vol. 20, no. 2, pp. 473–479, 2012.
- [153] S. Arimoto, S. Kawamura, and F. Miyazaki, "Bettering operation of robots by learning," *Journal of Robotic Systems*, vol. 1, no. 2, pp. 123–140, 1984.
- [154] P. Lucibello, "Learning algorithm for hybrid force control of robot arms," in *Proceedings of IEEE International Conference on Robotics and Automation, ICRA 1993*, vol. 1, 1993, pp. 654–658.

- [155] D. Jeon and M. Tomizuka, "Learning hybrid force and position control of robot manipulators," *IEEE Transactions on Robotics and Automation*, vol. 9, no. 4, pp. 423–431, 1993.
- [156] S. Pandian and S. Kawamura, "Hybrid force/position control for robot manipulators based on a d-type learning law," *Robotica*, vol. 14, no. 1, pp. 51–58, 1996.
- [157] C. Natale, R. Koeppel, and G. Hirzinger, "Automatic procedure for force controller design," in *IEEE/ASME International Conference on Advanced Intelligent Mechatronics, AIM 1999*, Atlanta, GA, USA, September 1999, pp. 967–972.
- [158] L. Liu, B. Ulrich, and M. Elbestawi, "Robotic grinding force regulation: Design, implementation and benefits," in *Proceedings of IEEE International Conference on Robotics and Automation, ICRA 1990*, Cincinnati, OH, USA, May 1990, pp. 258–265.
- [159] J. A. Maples and J. J. Becker, "Experiments in force control of robotic manipulators," in *Proceedings of IEEE International Conference on Robotics and Automation, ICRA 1986*, San Francisco, CA, USA, April 1986, pp. 695–702.
- [160] P. Bigras, M. Lambert, and C. Perron, "New optimal formulation for an industrial robot force controller," *International Journal of Robotics and Automation*, vol. 23, no. 3, pp. 199–208, 2008.
- [161] E. Abele, M. Weigold, and S. Rothenbücher, "Modeling and identification of an industrial robot for machining applications," *CIRP Annals-Manufacturing Technology*, vol. 56, no. 1, pp. 387–390, 2007.
- [162] C. Dumas, S. Caro, S. Garnier, and B. Furet, "Joint stiffness identification of six-revolute industrial serial robots," *Robotics and Computer-Integrated Manufacturing*, vol. 27, no. 4, pp. 881–888, 2011.
- [163] G. F. Franklin, J. D. Powell, and M. L. Workman, *Digital control of dynamic systems*. Addison-wesley Menlo Park, 1998, vol. 3.





# Appendix A

## Supplementary Materials for Modeling of Robot

### A.1 Homogeneous Transformation Matrix ${}^0T_t$

According to Equation (2.16) and (2.17), the homogeneous transformation matrix  ${}^0T_t$  defining the tool frame with respect to the fixed frame is given as:

$$\begin{aligned} {}^0T_t &= {}^0T_1 {}^1T_2 {}^2T_3 {}^3T_4 {}^4T_5 {}^5T_6 {}^6T_t \\ &= \begin{bmatrix} {}^0R_t & {}^0P_t \\ 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} r_{11} & r_{12} & r_{13} & P_{tx} \\ r_{21} & r_{22} & r_{23} & P_{ty} \\ r_{31} & r_{32} & r_{33} & P_{tz} \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned}$$

Let  $L_{5tz} = L_5 + L_{tz}$ , the position vector  ${}^0P_t(q)$  and rotation matrix  ${}^0R_t(q)$ , which are functions of joint position  $q$ , can be expressed as:

$$\begin{aligned} P_{tx} &= L_{1x}c_1 + L_2c_1c_2 + L_{34}c_1c_{23} + L_{5tz}c_1c_{23}c_5 + D_4c_1s_{23} - L_{5tz}c_1c_4s_{23}s_5 - L_{5tz}s_1s_4s_5; \\ P_{ty} &= -L_{1x}s_1 - L_2c_2s_1 - L_{34}c_{23}s_1 - L_{5tz}c_{23}c_5s_1 - D_4s_1s_{23} + L_{5tz}c_4s_1s_{23}s_5 - L_{5tz}c_1s_4s_5; \\ P_{tz} &= L_{1z} + D_4c_{23} - L_2s_2 - L_{34}s_{23} - L_{5tz}c_5s_{23} - L_{5tz}c_{23}c_4s_5; \end{aligned}$$

$$\begin{aligned} r_{11} &= c_4c_6s_1 - c_1c_6s_{23}s_4 - c_1c_4c_5s_{23}s_6 - c_5s_1s_4s_6 - c_1c_{23}s_5s_6; \\ r_{21} &= c_1c_4c_6 + c_6s_1s_{23}s_4 + c_4c_5s_1s_{23}s_6 - c_1c_5s_4s_6 + c_{23}s_1s_5s_6; \\ r_{31} &= -c_{23}c_6s_4 - c_{23}c_4c_5s_6 + s_{23}s_5s_6; \\ r_{12} &= -c_1c_4c_5c_6s_{23} - c_5c_6s_1s_4 - c_1c_{23}c_6s_5 - c_4s_1s_6 + c_1s_{23}s_4s_6; \\ r_{22} &= c_4c_5c_6s_1s_{23} - c_1c_5c_6s_4 + c_{23}c_6s_1s_5 - c_1c_4s_6 - s_1s_{23}s_4s_6; \\ r_{32} &= -c_{23}c_4c_5c_6 + c_6s_{23}s_5 + c_{23}s_4s_6; \\ r_{13} &= -c_1c_{23}c_5 + c_1c_4s_{23}s_5 + s_1s_4s_5; \\ r_{23} &= c_{23}c_5s_1 - c_4s_1s_{23}s_5 + c_1s_4s_5; \\ r_{33} &= c_5s_{23} + c_{23}c_4s_5; \end{aligned}$$

## A.2 Jacobian Matrix

Let  $L_{5tz} = L_5 + L_{tz}$ , the Jacobian matrix is obtained as:

$$J(1, 1) = -L_{1x}s_1 - L_2c_2s_1 - L_{34}c_{23}s_1 - L_{5tz}c_{23}c_5s_1 - D_4s_1s_{23} + L_{5tz}c_4s_1s_{23}s_5 - L_{5tz}c_1s_4s_5;$$

$$J(2, 1) = -L_{1x}c_1 - L_2c_1c_2 - L_{34}c_1c_{23} - L_{5tz}c_1c_{23}c_5 - D_4c_1s_{23} + L_{5tz}c_1c_4s_{23}s_5 + L_{5tz}s_1s_4s_5;$$

$$J(3, 1) = 0;$$

$$J(4, 1) = 0;$$

$$J(5, 1) = 0;$$

$$J(6, 1) = -1;$$

$$J(1, 2) = D_4c_1c_{23} - L_2c_1s_2 - L_{34}c_1s_{23} - L_{5tz}c_1c_5s_{23} - L_{5tz}c_1c_{23}c_4s_5;$$

$$J(2, 2) = -D_4c_{23}s_1 + L_2s_1s_2 + L_{34}s_1s_{23} + L_{5tz}c_5s_1s_{23} + L_{5tz}c_{23}c_4s_1s_5;$$

$$J(3, 2) = -L_2c_2 - L_{34}c_{23} - L_{5tz}c_{23}c_5 - D_4s_{23} + L_{5tz}c_4s_{23}s_5;$$

$$J(4, 2) = s_1;$$

$$J(5, 2) = c_1;$$

$$J(6, 2) = 0;$$

$$J(1, 3) = D_4c_1c_{23} - L_{34}c_1s_{23} - L_{5tz}c_1c_5s_{23} - L_{5tz}c_1c_{23}c_4s_5;$$

$$J(2, 3) = -D_4c_{23}s_1 + L_{34}s_1s_{23} + L_{5tz}c_5s_1s_{23} + L_{5tz}c_{23}c_4s_1s_5;$$

$$J(3, 3) = -L_{34}c_{23} - L_{5tz}c_{23}c_5 - D_4s_{23} + L_{5tz}c_4s_{23}s_5;$$

$$J(4, 3) = s_1;$$

$$J(5, 3) = c_1;$$

$$J(6, 3) = 0;$$

$$J(1, 4) = -L_{5tz}c_4s_1s_5 + L_{5tz}c_1s_{23}s_4s_5;$$

$$J(2, 4) = -L_{5tz}c_1c_4s_5 - L_{5tz}s_1s_{23}s_4s_5;$$

$$J(3, 4) = L_{5tz}c_{23}s_4s_5;$$

$$J(4, 4) = -c_1c_{23};$$

$$J(5, 4) = c_{23}s_1;$$

$$J(6, 4) = s_{23};$$

$$J(1, 5) = -L_{5tz}c_1c_4c_5s_{23} - L_{5tz}c_5s_1s_4 - L_{5tz}c_1c_{23}s_5;$$

$$J(2, 5) = L_{5tz}c_4c_5s_1s_{23} - L_{5tz}c_1c_5s_4 + L_{5tz}c_{23}s_1s_5;$$

$$J(3, 5) = -L_{5tz}c_{23}c_4c_5 + L_{5tz}s_{23}s_5;$$

$$J(4, 5) = c_4s_1 - c_1s_{23}s_4;$$

$$J(5, 5) = c_1c_4 + s_1s_{23}s_4;$$

$$J(6, 5) = -c_{23}s_4;$$

$$J(1, 6) = 0;$$

$$J(2, 6) = 0;$$

$$J(3, 6) = 0;$$

$$J(4, 6) = -c_1 c_{23} c_5 + c_1 c_4 s_{23} s_5 + s_1 s_4 s_5;$$

$$J(5, 6) = c_{23} c_5 s_1 - c_4 s_1 s_{23} s_5 + c_1 s_4 s_5;$$

$$J(6, 6) = c_5 s_{23} + c_{23} c_4 s_5;$$



# Appendix B

## Supplementary Materials for Model Simplification of Robot

### B.1 Expression of Original Component $M_{21}$ and Simplified component $M_{s_{21}}$ of Inertia Matrix

The original expression of the component  $M_{21}$  is formulated as:

$$\begin{aligned} M_{21} = & YZ_2c_2 + XZ_3c_2c_3 + D_4MY_{4R}c_2c_3c_4 - XY_4c_2c_3c_4^2 + XZ_{2R}s_2 + XX_{6R}c_2c_5s_3s_4s_5s_6^2 \\ & + XZ_5c_2c_3c_4^2c_5 - D_4MX_6c_2c_3c_4c_6 + XY_6c_2c_3c_4^2c_5c_6^2 + YZ_4c_3c_4s_2 + XX_{6R}c_3c_5s_2s_4s_5s_6^2 \\ & + L_{34}MY_{4R}c_3c_4s_2 + L_2MY_{4R}c_4s_2 - YZ_5c_3c_4c_5s_2 - L_2MX_6c_4c_6s_2 + XY_6c_2c_4s_3s_5s_6^2 \\ & - XY_6c_4^2c_5c_6^2s_2s_3 - L_5MX_6c_3c_4c_5c_6s_2 - XZ_6c_3c_4c_5c_6s_2 - YZ_3c_2s_3 + XY_6c_3c_4s_2s_5s_6^2 \\ & + L_{34}MY_{4R}c_2c_4s_3 + YZ_4c_2c_4s_3 - YZ_5c_2c_4c_5s_3 - L_{34}MX_6c_2c_4c_6s_3 - XY_6c_5s_2s_3s_4^2s_6^2 \\ & + D_4MX_5c_2c_3c_5s_4 - XZ_6c_2c_4c_5c_6s_3 - XZ_3s_2s_3 - D_4MY_{4R}c_4s_2s_3 + XX_{6R}c_4c_5^2s_2s_3s_4s_6^2 \\ & - L_{34}MX_6c_3c_4c_6s_2 + XY_4c_4^2s_2s_3 - XZ_5c_4^2c_5s_2s_3 + D_4MX_6c_4c_6s_2s_3 + XY_6c_2c_3c_5s_4^2s_6^2 \\ & + D_4MX_4c_2c_3s_4 - XX_{4R}c_2c_3c_4s_4 + ZZ_{5R}c_2c_3c_4s_4 + XY_5c_3c_5^2s_2s_4 - XZ_6c_2s_3s_4s_5^2s_6 \\ & - XX_{5R}c_2c_3c_4c_5^2s_4 + XZ_4c_3s_2s_4 + XX_{6R}c_2c_3c_4c_6^2s_4 + L_2MX_4s_2s_4 - XZ_6c_3s_2s_4s_5^2s_6 \\ & - L_5MX_6c_2c_4c_5c_6s_3 + D_4MY_6c_2c_3c_5c_6s_4 + L_{34}MY_6c_3c_5c_6s_2s_4 - XX_{6R}c_2c_3c_4c_5^2s_4s_6^2 \\ & + L_{34}MX_4c_3s_2s_4 + L_{34}MX_5c_3c_5s_2s_4 + L_2MX_5c_5s_2s_4 + XZ_4c_2s_3s_4 + XY_6c_4^2c_5s_2s_3s_6^2 \\ & + YZ_6c_2c_5^2c_6s_3s_4 + XY_5c_2c_5^2s_3s_4 + L_{34}MX_4c_2s_3s_4 - ZZ_{5R}c_4s_2s_3s_4 - XY_6c_2c_3c_4^2c_5s_6^2 \\ & + L_{34}MX_5c_2c_5s_3s_4 + L_{34}MY_6c_2c_5c_6s_3s_4 + 2L_5MY_6c_2c_3c_4c_5c_6s_4s_5 - L_5MX_6c_2s_3s_4s_5^2s_6 \\ & + L_2MY_6c_5c_6s_2s_4 + YZ_6c_3c_5^2c_6s_2s_4 - D_4MX_4s_2s_3s_4 + XY_4c_2c_3s_4^2 - L_5MX_6c_3s_2s_4s_5^2s_6 \\ & + L_5MY_6c_3c_5^2c_6s_2s_4 - D_4MY_6c_5c_6s_2s_3s_4 + 2YZ_6c_2c_3c_4c_5c_6s_4s_5 + YZ_6s_2s_3s_4^2s_5s_6 \\ & - D_4MX_5c_5s_2s_3s_4 - XX_{6R}c_4c_6^2s_2s_3s_4 - XZ_5c_2c_3c_5s_4^2 - XY_4s_2s_3s_4^2 + L_5MY_6s_2s_3s_4^2s_5s_6 \\ & + XX_{4R}c_4s_2s_3s_4 - XY_6c_2c_3c_5c_6^2s_4^2 + XZ_5c_5s_2s_3s_4^2 - XZ_5c_3c_4s_2s_5 - L_5MY_6c_2c_3s_4^2s_5s_6 \\ & - YZ_5c_2c_3c_4^2s_5 - XY_6c_2c_4c_6^2s_3s_5 - XZ_6c_2c_3c_4^2c_6s_5 + YZ_5c_4^2s_2s_3s_5 - YZ_6c_2c_3s_4^2s_5s_6 \\ & - L_2MY_5R s_2s_4s_5 - XY_6c_3c_4c_6^2s_2s_5 - XZ_5c_2c_4s_3s_5 + YZ_5c_2c_3s_4^2s_5 - 2XZ_6c_4c_5s_2s_3s_4s_5s_6 \\ & + L_5MX_6c_4^2c_6s_2s_3s_5 - 2L_5MY_6c_4c_5c_6s_2s_3s_4s_5 + XX_{5R}c_3c_5s_2s_4s_5 + XZ_6c_3c_5^2s_2s_4s_6 \\ & + 2XY_5c_2c_3c_4c_5s_4s_5 - L_5MX_6c_2c_3c_4^2c_6s_5 - 2YZ_6c_4c_5c_6s_2s_3s_4s_5 + 2XY_6c_2c_5c_6s_3s_4s_5s_6 \end{aligned}$$

$$\begin{aligned}
 & -ZZ_6c_2c_5s_3s_4s_5 - XY_5c_3s_2s_4s_5^2 + XZ_6c_4^2c_6s_2s_3s_5 - YZ_5s_2s_3s_4^2s_5 + ZZ_6c_4s_2s_3s_4s_5^2 \\
 & - ZZ_6c_3c_5s_2s_4s_5 - YZ_6c_3c_6s_2s_4s_5^2 - XY_5c_2s_3s_4s_5^2 + L_2MY_6c_4s_2s_6 - YZ_6c_4^2s_2s_3s_5s_6 \\
 & + L_5MX_6c_2c_3c_6s_4^2s_5 - L_{34}MY_{5R}c_3s_2s_4s_5 - L_5MY_6c_2c_6s_3s_4s_5^2 + 2XY_6c_3c_5c_6s_2s_4s_5s_6 \\
 & - L_5MY_6c_3c_6s_2s_4s_5^2 - L_{34}MY_{5R}c_2s_3s_4s_5 + L_5MY_6c_2c_5^2c_6s_3s_4 + YZ_6c_2c_3c_4^2s_5s_6 \\
 & + D_4MY_{5R}c_2s_3s_4s_5 - 2XY_5c_4c_5s_2s_3s_4s_5 - L_5MX_6c_6s_2s_3s_4^2s_5 + 2L_5MX_6c_2c_3c_4c_5s_4s_5s_6 \\
 & + XX_{5R}c_4^2c_5^2s_2s_3s_4 + XX_{6R}c_2c_3c_4^2c_5c_6s_6 + XX_{5R}c_2c_5s_3s_4s_5 - XX_{6R}c_2c_4c_6s_3s_5s_6 \\
 & + L_5MY_6c_3c_4c_5s_2s_6 + L_5MY_6c_2c_4c_5s_3s_6 + L_{34}MY_6c_2c_4s_3s_6 - L_5MY_6c_4^2s_2s_3s_5s_6 \\
 & - YZ_6c_2c_6s_3s_4s_5^2 - XX_{6R}c_4^2c_5c_6s_2s_3s_6 + D_4MX_6c_2c_3c_5s_4s_6 + L_5MY_6c_2c_3c_4^2s_5s_6 \\
 & + L_{34}MX_6c_3c_5s_2s_4s_6 - D_4MY_{5R}c_2c_3s_4s_5 + 2XY_6c_4c_5^2c_6s_2s_3s_4s_6 + L_{34}MY_6c_3c_4s_2s_6 \\
 & + L_{34}MX_6c_2c_5s_3s_4s_6 + YZ_6c_3c_4c_5s_2s_6 - D_4MX_6c_5s_2s_3s_4s_6 - XX_{6R}c_3c_4c_6s_2s_5s_6 \\
 & - ZZ_6c_2c_3c_4s_4s_5^2 + D_4MY_6c_2c_3c_4s_6 + 2XY_6c_4c_6s_2s_3s_4s_6 + XX_{6R}c_5c_6s_2s_3s_4^2s_6 \\
 & - 2XY_6c_2c_3c_4c_6s_4s_6 - 2XY_6c_2c_3c_4c_5^2c_6s_4s_6 + L_2MX_6c_5s_2s_4s_6 + XY_6c_5c_6^2s_2s_3s_4^2 \\
 & + YZ_6c_2c_4c_5s_3s_6 - D_4MY_6c_4s_2s_3s_6 - 2L_5MX_6c_4c_5s_2s_3s_4s_5s_6 - XX_{6R}c_2c_3c_5c_6s_4^2s_6 \\
 & + XZ_6c_2c_3c_6s_4^2s_5 + L_5MX_6c_3c_5^2s_2s_4s_6 + 2XZ_6c_2c_3c_4c_5s_4s_5s_6 + XZ_6c_2c_5^2s_3s_4s_6 \\
 & - XZ_6c_6s_2s_3s_4^2s_5 + L_5MX_6c_2c_5^2s_3s_4s_6
 \end{aligned}$$

The simplified component  $M_{s_{21}}$  is obtained as:

$$\begin{aligned}
 M_{s_{21}} &= YZ_2 \cos(q_2) + XZ_{2R} \sin(q_2) \\
 &+ A_1 \cos(q_2 + q_3 + q_4 - \varphi_1) \\
 &+ A_2 \cos(q_2 + q_3 - q_4 - \varphi_2)
 \end{aligned}$$

with

$$\begin{aligned}
 A_1 &= \frac{1}{2} \sqrt{XZ_4^2 + YZ_4^2}, \\
 A_2 &= \frac{1}{2} \sqrt{XZ_4^2 + (L_{34}MY_{4R} + YZ_4)^2}, \\
 \varphi_1 &= \arctan(-XZ_4, YZ_4), \\
 \varphi_2 &= \arctan(XZ_4, L_{34}MY_{4R} + YZ_4).
 \end{aligned}$$

where  $s_i$ ,  $c_i$  denote  $\sin(q_i)$  and  $\cos(q_i)$  respectively,  $YZ_2$ ,  $XZ_{2R}$ ,  $XZ_4$ ,  $YZ_4$ ,  $MY_{4R}$ , etc. are 36 regrouped scalar inertia parameters used in robot modeling [4].

Since the simplified expressions are very simple and many related parameters (e.g.  $A_1$ ,  $A_2$ ,  $\varphi_1$  and  $\varphi_2$ ) can be computed offline, the practical computation time will be greatly reduced.

## B.2 Tables for choosing appropriate $k_t$ and $k_p$ for linear and circular FSW trajectories

In order to find the most appropriate proportional factors  $k_t$  and  $k_p$  for the linear and circular FSW trajectories, the components  $M_{11}$ ,  $M_{21}$ ,  $M_{22}$  and  $M_{41}$  are illustrated to calculate the evaluation indexes. By choosing different  $k_t$  and  $k_p$ , the number

of terms in the simplified components and the corresponding RMS errors along the linear and circular FSW trajectories are below.

Table B.1: Number of terms in the simplified components and corresponding RMS errors along the linear FSW trajectory when choosing different  $k_t$  and  $k_p$

| $k_t$ | $k_p$ | $M_{11}$ |               | $M_{21}$ |               | $M_{22}$ |               | $M_{41}$ |               |
|-------|-------|----------|---------------|----------|---------------|----------|---------------|----------|---------------|
|       |       | $N_t$    | $e_{RMS}$ (%) | $N_t$    | $e_{RMS}$ (%) | $N_t$    | $e_{RMS}$ (%) | $N_t$    | $e_{RMS}$ (%) |
| 0     | 0     | 238      | 0             | 112      | 0             | 73       | 0             | 89       | 0             |
| 0.5   | 0     | 124      | 0.14          | 43       | 0.33          | 57       | 0.01          | 75       | 5.10          |
|       | 0.5   | 119      | 0.37          | 43       | 0.32          | 53       | 0.12          | 75       | 5.53          |
|       | 1     | 119      | 0.37          | 43       | 0.29          | 52       | 0.53          | 75       | 5.58          |
|       | 3     | 117      | 3.92          | 43       | 0.10          | 52       | 0.53          | 75       | 2.24          |
|       | 5     | 117      | 3.65          | 43       | 0.22          | 52       | 1.17          | 75       | 7.81          |
| 1     | 0     | 74       | 0.30          | 24       | 2.88          | 50       | 0.01          | 66       | 13.15         |
|       | 0.5   | 69       | 0.53          | 24       | 2.89          | 46       | 0.12          | 66       | 13.58         |
|       | 1     | 69       | 0.53          | 24       | 2.93          | 45       | 0.53          | 66       | 12.05         |
|       | 3     | 67       | 4.08          | 24       | 3.08          | 45       | 0.53          | 66       | 9.54          |
|       | 5     | 67       | 3.82          | 24       | 3.42          | 45       | 1.17          | 66       | 19.77         |
| 3     | 0     | 25       | 0.30          | 10       | 3.11          | 44       | 0.01          | 45       | 68.76         |
|       | 0.5   | 20       | 0.54          | 10       | 3.12          | 40       | 0.10          | 45       | 69.19         |
|       | 1     | 20       | 0.54          | 10       | 3.15          | 39       | 0.54          | 45       | 67.84         |
|       | 3     | 18       | 4.08          | 10       | 3.31          | 39       | 0.54          | 45       | 65.34         |
|       | 5     | 18       | 4.16          | 10       | 3.64          | 39       | 1.18          | 45       | 75.56         |
| 5     | 0     | 19       | 0.81          | 3        | 2.02          | 39       | 0.03          | 25       | 23.04         |
|       | 0.5   | 14       | 1.04          | 3        | 2.02          | 35       | 0.14          | 25       | 23.34         |
|       | 1     | 14       | 1.04          | 3        | 2.02          | 34       | 0.51          | 25       | 21.78         |
|       | 3     | 12       | 4.58          | 3        | 2.10          | 34       | 0.51          | 25       | 18.87         |
|       | 5     | 12       | 4.65          | 3        | 2.27          | 34       | 1.15          | 25       | 29.48         |



Table B.2: Number of terms in the simplified components and corresponding RMS errors along the circular FSW trajectory when choosing different  $k_t$  and  $k_p$ 

| $k_t$ | $k_p$ | $M_{11}$ |               | $M_{21}$ |               | $M_{22}$ |               | $M_{41}$ |               |
|-------|-------|----------|---------------|----------|---------------|----------|---------------|----------|---------------|
|       |       | $N_t$    | $e_{RMS}$ (%) | $N_t$    | $e_{RMS}$ (%) | $N_t$    | $e_{RMS}$ (%) | $N_t$    | $e_{RMS}$ (%) |
| 0     | 0     | 238      | 0             | 112      | 0             | 73       | 0             | 89       | 0             |
| 0.5   | 0     | 149      | 0.10          | 51       | 0.39          | 62       | 0.004         | 77       | 3.51          |
|       | 0.5   | 144      | 0.32          | 51       | 0.45          | 58       | 0.11          | 77       | 2.58          |
|       | 1     | 144      | 0.33          | 51       | 0.58          | 57       | 0.54          | 77       | 4.04          |
|       | 3     | 142      | 3.86          | 51       | 0.69          | 57       | 0.54          | 77       | 11.23         |
|       | 5     | 142      | 3.66          | 51       | 1.21          | 57       | 0.41          | 77       | 21.58         |
| 1     | 0     | 92       | 0.35          | 29       | 0.28          | 58       | 0.02          | 77       | 3.51          |
|       | 0.5   | 87       | 0.53          | 29       | 0.27          | 54       | 0.10          | 77       | 2.58          |
|       | 1     | 87       | 0.54          | 29       | 0.22          | 53       | 0.54          | 77       | 4.04          |
|       | 3     | 85       | 4.00          | 29       | 0.23          | 53       | 0.54          | 77       | 11.23         |
|       | 5     | 85       | 3.90          | 29       | 0.74          | 53       | 0.42          | 77       | 21.58         |
| 3     | 0     | 25       | 1.03          | 11       | 9.46          | 48       | 0.04          | 67       | 19.19         |
|       | 0.5   | 20       | 1.12          | 11       | 9.43          | 44       | 0.15          | 67       | 19.29         |
|       | 1     | 20       | 1.12          | 11       | 9.48          | 43       | 0.50          | 67       | 18.63         |
|       | 3     | 18       | 4.22          | 11       | 9.64          | 43       | 0.50          | 67       | 15.01         |
|       | 5     | 18       | 4.25          | 11       | 9.78          | 43       | 0.37          | 67       | 13.12         |
| 5     | 0     | 20       | 1.16          | 3        | 5.05          | 47       | 0.02          | 32       | 59.08         |
|       | 0.5   | 15       | 1.29          | 4        | 5.05          | 43       | 0.12          | 32       | 59.03         |
|       | 1     | 15       | 1.29          | 4        | 5.05          | 42       | 0.53          | 32       | 59.03         |
|       | 3     | 13       | 4.50          | 4        | 5.14          | 42       | 0.53          | 32       | 58.93         |
|       | 5     | 13       | 4.53          | 4        | 5.32          | 42       | 0.40          | 32       | 58.59         |

# Appendix C

## Supplementary Materials for Design of Force Controller

### C.1 Detailed Computation Procedures for Single-Stage Controller

According to [95], the detailed computation procedures for the single-stage controllers can be presented. As for the phase-lag controller or the phase-lead controller, the transfer function  $G_{FC}^*(s)$  at the frequency  $\omega_g$  can be given by:

$$G_{FC}^*(j\omega_g) = \frac{1 + j\omega_g T_1}{1 + j\omega_g T_2} \quad (\text{C.1})$$

Since the gain shift  $\Delta_{gn}$  and phase shift  $\Delta_\varphi$  (at the frequency  $\omega_g$ ) which the transfer function  $G_{FC}^*(s)$  needs to provide to the system have already been calculated, the two unknown controller parameters  $T_1$  and  $T_2$  can be obtained by using the following two equations:

$$\left| \frac{1 + j\omega_g T_1}{1 + j\omega_g T_2} \right| = \Delta_{gn} \quad (\text{C.2})$$

$$\angle \frac{1 + j\omega_g T_1}{1 + j\omega_g T_2} = \Delta_\varphi \quad (\text{C.3})$$

By using the substitutions  $a = T_1/T_2$  and  $x = \omega_g T_2$ , the above two equations can be transformed to the following form:

$$\frac{1 + a^2 x^2}{1 + x^2} = \Delta_{gn}^2 \quad (\text{C.4})$$

$$\frac{ax - x}{1 + ax^2} = \tan \Delta_\varphi \quad (\text{C.5})$$

Thus, the objective is to find the values of  $a$  and  $x$  and the conditions which imply that both of them are positive.

Firstly, based on equation (C.5), the parameter  $a$  can be expressed by:

$$a = \frac{x + \tan \Delta_\varphi}{x - x^2 \tan \Delta_\varphi} \quad (\text{C.6})$$

According to Equation (C.5) and (C.6), we can obtain:

$$\Delta_{gn}^2(1+x^2) = 1 + \frac{(x + \tan \Delta_\varphi)^2}{(1 - x \tan \Delta_\varphi)^2} \quad (\text{C.7})$$

Supposing that  $1 - x \tan \Delta_\varphi \neq 0$ , the above equation can be expanded as:

$$\Delta_{gn}^2(1+x^2)(1 - x \tan \Delta_\varphi)^2 = (1+x^2)(1 + \tan^2 \Delta_\varphi) \quad (\text{C.8})$$

Due to the fact that  $\Delta_{gn} = 0$  corresponds to a non-realizable system, the above equation can be transformed as:

$$(1 - x \tan \Delta_\varphi)^2 = \frac{1 + \tan^2 \Delta_\varphi}{\Delta_{gn}^2} \quad (\text{C.9})$$

It should be noted that  $1 - x \tan \Delta_\varphi = 0$  only if  $\Delta_{gn} \rightarrow \infty$ , which is not possible and has no physical meaning, hence, the above assumption is verified.

Therefore, two solutions can be obtained for  $x$ , as given by:

$$x = \frac{1 \pm \Delta_{gn}^{-1} \sqrt{1 + \tan^2 \Delta_\varphi}}{\tan \Delta_\varphi} \quad (\text{C.10})$$

As a matter of fact, the solution for  $x$  exists only if  $\tan \Delta_\varphi \neq 0$ , that is to say, the single-stage controller cannot work without causing a phase shift. It can be assumed that  $\Delta_\varphi \in (-\pi/2, \pi/2)$ , which means only the same type single-stage controllers are considered hereafter.

When the phase-lag single-stage controller is chosen ( $\Delta_g < 0$ ,  $\Delta_\varphi < 0$ ), the following conditions hold:

$$0 < \Delta_{gn} < 1 \quad (\text{C.11})$$

$$\tan \Delta_\varphi < 0 \quad (\text{C.12})$$

Since the parameter  $x$  can not be negative, the solution in Equation C.10 corresponding to the plus sign can be neglected. By contrast, the solution with minus sign always leads to a positive value for  $x$ . Besides, the condition to have  $a > 0$  can be provided by:

$$a > 0 \Leftrightarrow \Delta_\varphi > -\arctan \sqrt{\Delta_{gn}^{-2} - 1} \quad (\text{C.13})$$

When the phase-lead single-stage controller is chosen ( $\Delta_g > 0$ ,  $\Delta_\varphi > 0$ ), the following conditions hold:

$$\Delta_{gn} > 1 \quad (\text{C.14})$$

$$\tan \Delta_\varphi > 0 \quad (\text{C.15})$$

Considering the solution in Equation C.10 corresponding to the minus sign, we can have:

$$x > 0 \Leftrightarrow \Delta_\varphi < \arctan \sqrt{\Delta_{gn}^2 - 1} \quad (\text{C.16})$$

which is sought restriction on the desired corrections.

If  $x > 0$ , the condition to have  $a > 0$  can be provided by:

$$a > 0 \Leftrightarrow 1 - \Delta_{gn}^{-1} \sqrt{1 + \tan^2 \Delta_\varphi} < 1 \quad (\text{C.17})$$

which is always true since  $\Delta_{gn} > 1$ .

If the solution with the plus sign is considered, then  $x$  will be always positive and the condition to have  $a > 0$  can be provided by:

$$a > 0 \Leftrightarrow x < \frac{1}{\tan \Delta_\varphi} \Leftrightarrow 1 + \Delta_{gn}^{-1} \sqrt{1 + \tan^2 \Delta_\varphi} < 1 \quad (\text{C.18})$$

which is always false since  $\Delta_{gn} > 1$ . Thus, the solution with the plus sign can be discarded as well.



# Appendix D

## Technical Documents of the Robot

### D.1 Technical Documents of the Robot

| Typ / Type:                                                                         | KR 500-2 MT                             | KR 500 L480-2 MT        |
|-------------------------------------------------------------------------------------|-----------------------------------------|-------------------------|
| Traglast / Payload:                                                                 | 500 kg                                  | 480 kg                  |
| Zusatzlast Arm/Schwinge/Karussell /<br>Suppl. load on arm/link arm/rotating column: | 50 kg / 100 kg / 400 kg                 |                         |
| Zusatzlast max. Arm und Schwinge /<br>Max. supplementary load on arm and link arm:  | 100 kg                                  |                         |
| Max. Gesamtlast / Total distributed load:                                           | 1000 kg                                 | 980 kg                  |
| Prozesskräfte, horizontal/vertikal / Process forces horizontal/vertical:            | 8000 N <sup>1)</sup>                    |                         |
| Anzahl der Achsen / Number of axes:                                                 | 6                                       |                         |
| Handvariante / Wrist variant:                                                       | Zentralhand 500 l / In-line wrist 500 l |                         |
| Handvariante / Wrist variant:                                                       | F (foundry), ≤ 180 °C (10 s/min)        |                         |
| Anbaufansch A6 / Mounting flange A6:                                                | DIN ISO 9409-1 - A200                   |                         |
| Einbaulage / Mounting position:                                                     | Boden/Floor                             |                         |
| Wiederholgenauigkeit (ISO 9283) / Repeatability (ISO 9283):                         | ± 0,08 mm                               |                         |
| Steuerung / Controller:                                                             | KR C2                                   |                         |
| Gewicht (ohne Steuerung) ca. / Weight (excl. controller) approx.:                   | 2400 kg                                 | 2425 kg                 |
| Arbeitsraumvolumen <sup>2)</sup> / Work envelope <sup>2)</sup> :                    | 68 m <sup>3</sup>                       | 118 m <sup>3</sup>      |
| Achsdaten / Axis data                                                               | Bereich (Software) / Range (software)   | Geschwindigkeit / Speed |
| Achse / Axis 1 (A1)                                                                 | ±185°                                   | 41 °/s                  |
| Achse / Axis 2 (A2)                                                                 | +20°/-130°                              | 41 °/s                  |
| Achse / Axis 3 (A3)                                                                 | +150°/-94°                              | 41 °/s                  |
| Achse / Axis 4 (A4)                                                                 | ±350°                                   | 76 °/s                  |
| Achse / Axis 5 (A5)                                                                 | ±118°                                   | 76 °/s                  |
| Achse / Axis 6 (A6)                                                                 | ±350°                                   | 120 °/s                 |

<sup>1)</sup> Auf Anfrage sind stellungsabhängig höhere Kräfte möglich. / Depending on the robot position, higher forces may be possible after consultation.

<sup>2)</sup> Bezogen auf Schnittpunkt Achse 4/5. / Referred to intersection of axes 4 and 5.

- Antriebssystem elektro-mech. mit burstenlosen AC-Servomotoren. / Drive system electromechanical, with brushless AC servomotors.
- Wegmessesystem digital-absolut. / Position sensing system digital-absolute.

Figure D.1: Technical data of robot KR500-2MT

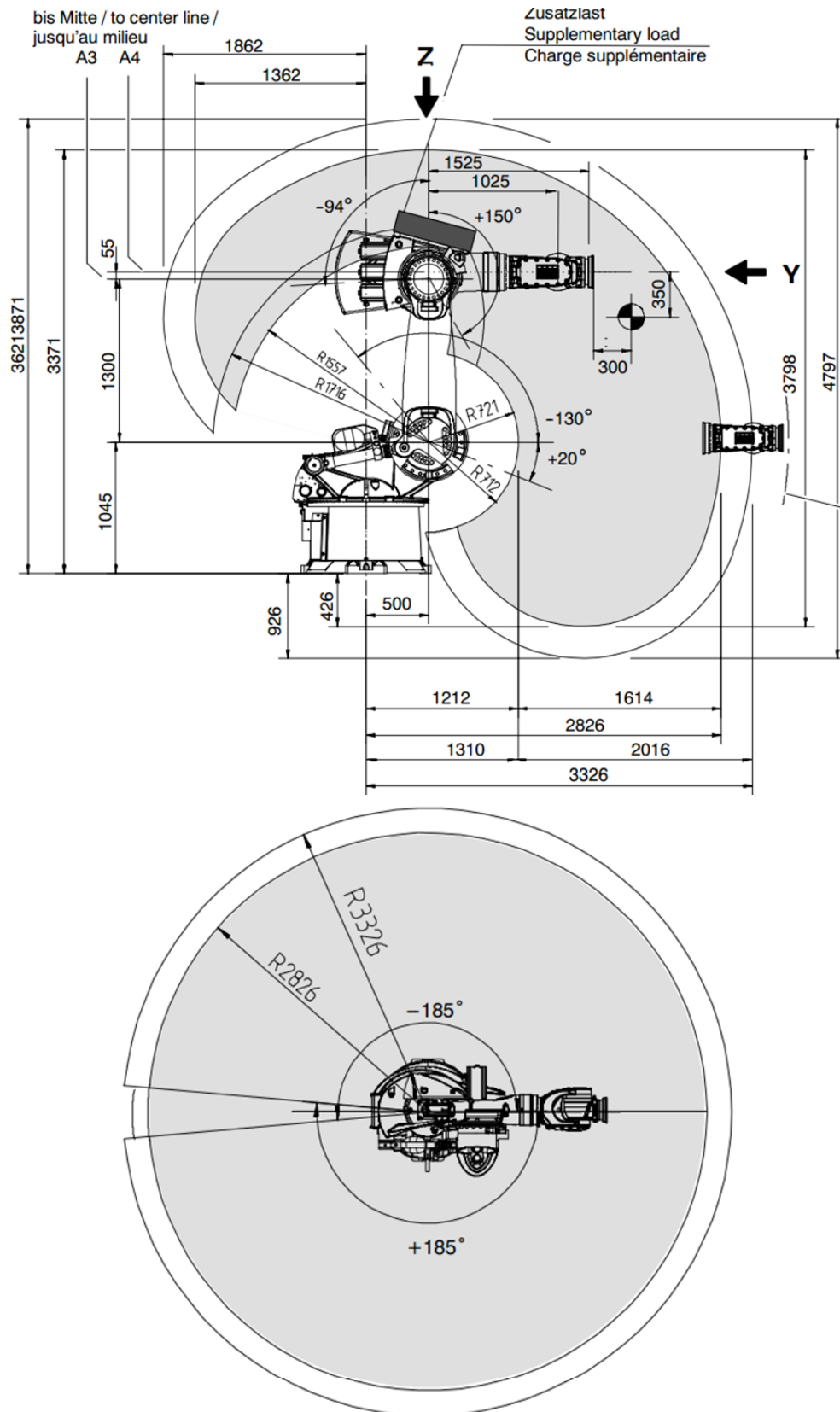


Figure D.2: Principle dimensions and working envelope (software values) for robot KR500-2MT





## MODELISATION D'UN ROBOT MANIPULATEUR EN VUE DE LA COMMANDE ROBUSTE EN FORCE UTILISE EN SOUDAGE FSW

**RESUME :** Le travail présenté dans cette thèse concerne la modélisation et la commande robuste en force de robots manipulateurs industriels à articulations flexibles utilisés pour le procédé FSW. Afin de réduire les temps de calcul et l'occupation de la mémoire, une approche basée sur la méthode par intervalle est proposée en vue de la simplification des modèles dynamiques des robots industriels, et contribue à identifier les paramètres d'inertie qui sont négligeables. Des études de cas sur trois types de trajectoires de test et l'analyse des couples moteurs ont démontré l'efficacité et les bonnes performances de la méthode de simplification. Ensuite, la modélisation dynamique et l'identification des paramètres du procédé FSW ont été effectuées. Les paramètres des modèles linéaires et non-linéaires de forces axiales sont identifiés. Sur la base de la modélisation du procédé FSW qui considère simultanément la cinématique du système complet, le modèle de déplacement du robot rigide, les flexibilités des articulations et le modèle dynamique de la force axiale, un contrôleur robuste en force est obtenu par la méthode de réglage fréquentielle. En outre, un simulateur du procédé FSW robotique est développé et les résultats de simulation montrent les bonnes performances du contrôleur en force. L'oscillation de la force axiale dans le procédé FSW peut être simulée en utilisant un modèle de perturbation de la position verticale de référence.

**Mots clés :** commande de robots industriels à articulations flexibles, commande robuste en force, identification du procédé de soudure par frottement-malaxage, simplification de modèles dynamiques, identification du modèle du robot dans l'espace cartésien.

## ROBOT MANIPULATOR MODELING FOR ROBUST FORCE CONTROL USED IN FRICTION STIR WELDING (FSW)

**ABSTRACT:** The work presented in this thesis focuses on the modeling and robust force control of flexible joints industrial robot manipulators used for FSW process. In order to reduce computation time and memory occupation, a novel interval-based approach for dynamic model simplification of industrial robots is proposed, which applies to arbitrary trajectories of whole robot workspace and contributes to obtaining negligible inertia parameters. Case studies have been carried out on three kinds of test trajectories and torques analysis of robot dynamic equation, demonstrating the effectiveness and good performance of the simplification method. Then, the dynamic modeling and identification of robotic FSW process is performed, and the parameters of linear and nonlinear dynamic axial force process models are identified by using the plunge depth and its derivative. On the basis of the modeling of robotic FSW process which simultaneously considers the complete kinematics, the rigid robot displacement model, the joint flexibility and the dynamic axial force process model, a robust force controller can be obtained by using the frequency response approach. Besides, a simulator of robotic FSW process is developed and simulation results show good performance of the force controller. The oscillation of axial force in FSW process can be simulated when a disturbance model of initial vertical reference position is proposed and used in the simulation.

**Keywords :** flexible joint industrial manipulator control, robust force control, friction stir welding process identification, dynamic model simplification, robot Cartesian model identification.