Linköping studies in science and technology. Dissertations.
No. 1349

Modeling and Control of Flexible Manipulators

Stig Moberg

Department of Electrical Engineering
Linköping University, SE–581 83 Linköping, Sweden

Linköping 2010

Cover illustration: A well-known path used by robot customers and robot manufacturers for evaluating the path accuracy of industrial robots (front). One element of the multivariable frequency response function magnitude for a modern industrial robot (back).

Linköping studies in science and technology. Dissertations.
No. 1349

Modeling and Control of Flexible Manipulators

Stig Moberg

stig@isy.liu.se
www.control.isy.liu.se
Division of Automatic Control
Department of Electrical Engineering
Linköping University
SE–581 83 Linköping
Sweden

ISBN 978-91-7393-289-9 ISSN 0345-7524

Printed by LiU-Tryck, Linköping, Sweden 2010

To Karin and John

Abstract

Industrial robot manipulators are general-purpose machines used for industrial automation in order to increase productivity, flexibility, and product quality. Other reasons for using industrial robots are cost saving, and elimination of hazardous and unpleasant work. Robot motion control is a key competence for robot manufacturers, and the current development is focused on increasing the robot performance, reducing the robot cost, improving safety, and introducing new functionalities. Therefore, there is a need to continuously improve the mathematical models and control methods in order to fulfil conflicting requirements, such as increased performance of a weight-reduced robot, with lower mechanical stiffness and more complicated vibration modes. One reason for this development of the robot mechanical structure is of course cost-reduction, but other benefits are also obtained, such as lower environmental impact, lower power consumption, improved dexterity, and higher safety.

This thesis deals with different aspects of modeling and control of flexible, i.e., elastic, manipulators. For an accurate description of a modern industrial manipulator, this thesis shows that the traditional flexible joint model, described in literature, is not sufficient. An improved model where the elasticity is described by a number of localized multidimensional spring-damper pairs is therefore proposed. This model is called the *extended flexible joint model*. The main contributions of this work are the design and analysis of identification methods, and of inverse dynamics control methods, for the extended flexible joint model.

The proposed identification method is a frequency-domain non-linear gray-box method, which is evaluated by the identification of a modern six-axes robot manipulator. The identified model gives a good description of the global behavior of this robot.

The inverse dynamics problem is discussed, and a solution methodology is proposed. This methodology is based on the solution of a differential algebraic equation (DAE). The inverse dynamics solution is then used for feedforward control of both a simulated manipulator and of a real robot manipulator.

The last part of this work concerns feedback control. First, a model-based nonlinear feedback control (feedback linearization) is evaluated and compared to a model-based feedforward control algorithm. Finally, two benchmark problems for robust feedback control of a flexible manipulator are presented and some proposed solutions are analyzed.

Populärvetenskaplig sammanfattning

Industrirobotar används inom många industrigrenar där bilindustrin är den största robotkunden. Andra industrier som har stora behov av robotar är halvledar- och elektronikindustrin, flygindustrin och livsmedelsindustrin. Exempel på nya tillämpningar för robotar är hantering av skärmar till platt-TV, tillverkning av solceller och solpaneler, samt sortering och paketering av läkemedel. Några typiska arbetsuppgifter för robotar i traditionell tillverkningsindustri är svetsning, målning, montering, laserskärning och materialbearbetning. Industrirobotar förekommer även inom nöjesindustrin, där de bland annat använts vid filminspelningar (Terminator Salvation), vid rock-konserter (Bon Jovi) och som avancerade nöjesfältsattraktioner (RoboCoaster). Det finns även exempel på att industrirobotar används inom sjukvården, t.ex. för rehabilitering av strokepatienter. Industrirobotar är med andra ord universalmaskiner och bara användarens fantasi begränsar möjligheterna! Totalt har mer än 2 miljoner industrirobotar levererats sedan introduktionen i slutet av 1960-talet och idag är mer än 1 miljon robotar i drift. Användning av robotar kan förstås minska antalet anställda inom industrin, men kan också skapa nya arbeten med ett bättre arbetsinnehåll och ta bort farliga, tunga och tråkiga arbetsuppgifter. Andra motiv för att använda robotar är ökad och jämn produktkvalitet och en effektivare produktion till en lägre kostnad. Eftersom robotar är omprogrameringsbara får man också, i jämförelse med fast automation, flexibilitet och möjligheter att ändra produktionen.

Robotens hjärna är styrsystemet, vars datorer står för robotens intelligens och bland annat styr robotens rörelser. Den del av robotstyrsystemet som styr robotens rörelser brukar kallas *rörelsestyrning*, och dess huvuduppgift är att styra vridmomenten i robotens elektriska motorer så att de rörelser som robotanvändaren programmerat utförs med största möjliga snabbhet och precision. Vridmomenten styrs i sin tur genom reglering av strömmarna till de motorer som är driver robotarmarna via växellådor. För att kunna beräkna det vridmoment som behövs för att röra t.ex. en svetspistol på önskat sätt används *matematiska modeller*. Denna styrprincip kallas modellbaserad *framkopplingsreglering*. Eftersom modeller aldrig kan vara helt perfekta, och eftersom vissa saker inte kan modelleras, t.ex. att någonting stöter till roboten, krävs det också kunskap om robotens verkliga rörelse. En vanligt sätt att få denna kunskap är att mäta motorvinklarna. Denna mätning används av styrsystemet för att korrigera de vridmoment som modellerna räknat fram. Denna styrprincip kallas *återkopplingsreglering*. Även återkopplingsregleringen kan använda sig av styrsystemets modeller. Rörelsestyrningen för en robot kan liknas vid den mänskliga hjärnans styrning av grov- och finmotoriken med hjälp av erfarenhet och instinkt (framkoppling) och sinnesintryck (återkoppling).

En trend inom industriell robotutveckling är att robotarna görs vigare och att dess vikt minskar relativt den last som roboten ska hantera. Detta ökar robotens användbarhet och minskar kostnader och energiförbrukning, men det gör också rörelsestyrningen betydligt svårare eftersom roboten totalt sett blir vekare. De

dominerande vekheterna (elasticiteterna) i en modern industrirobot finns hos växellådor, lager och robotarmar, men även underlaget som roboten är monterad på kan vara vekt. Problemet att styra en vek robot kan liknas vid att snabbt svinga ett metspö och att få toppen av metspöt att följa en viss bana och slutligen stanna på önskat ställe utan att svänga. En ökad intelligens i robotens rörelsestyrning är för dessa veka robotar helt avgörande för att kunna styra rörelserna med snabbhet och precision. Detta innebär att rörelsestyrningens intelligens också är avgörande för att kunna minska robotens tillverkningskostnad.

Denna avhandling handlar om modellering och reglering av moderna industrirobotar. Det som behandlas är hur man med hög noggrannhet matematiskt skall beskriva veka robotar, hur man skall utföra mätningar på en robot för att anpassa de matematiska modellerna till verkligheten och hur man använder modellerna för att med framkoppling och återkoppling kunna styra en robot. Den första delen av avhandlingen presenterar en matematisk modell som kan beskriva elasticiteten i en modern industrirobot, samt beskriver metoder för att ta fram modellparametrar. Att ta fram värden på okända modellparametrar kallas ofta *identifiering*. Metoden bygger på att man styr robotens rörelser så att de vridmoment och rörelser som registreras kan användas för att räkna ut de okända modellparametrarna. Enkelt uttryckt så skakar man på roboten så att den avslöjar värdena på modellens fjädrar och dämpare. De modeller och identifieringsmetoder som redovisas i avhandlingen beskriver robotens vekhet på ett unikt bra sätt.

Den andra delen av avhandlingen beskriver hur modellen kan användas i en modellbaserad framkopplingsreglering. Beräkningen av vridmoment, givet önskad robotrörelse, innebär lösandet av en *differential-algebraisk ekvation* (DAE). Denna ekvation är i detta fall mycket svår att lösa, och ett antal metoder för att lösa denna DAE föreslås och analyseras. De framtagna metoderna är alltför komplexa givet dagens datorkraft, men visar att problemet är lösbart. Metoderna kan också vara en utgångspunkt för att utveckla förenklade metoder som styrsystemets datorer har kapacitet att beräkna.

Avhandlingen avslutas med en beskrivning av två specialanpassade matematiska robotmodeller och tillhörande krav på styrsystemet. En viktig egenskap för dessa modeller är att endast motorernas vinklar mäts, vilket är det normala fallet för moderna industrirobotar. Dessa s.k. *benchmark-problem* är tänkta att användas för utvärdering och utveckling av alternativa metoder för återkopplingsreglering, och att ge forskare inom området realistiska modeller med tillhörande industriella krav. Det första benchmark-problemet presenterades 2004 under namnet *Svenska Mästerskapet i Robotreglering* och attraherade deltagare från tre världsdelar. Slutsatsen av denna utvärdering är att återkopplingsregleringen måste få tillgång till mer information för att väsentligen kunna höja prestanda över dagens nivå. Denna information kan erhållas genom mätningar av t.ex. robotarmarnas eller verktygets positioner och accelerationer. Hur flera mätningar kan användas för reglering är ett viktigt område för de kommande årens robotforskning.

Acknowledgments

First of all I would like to thank my supervisor Professor Svante Gunnarsson for helping me in my research, and for always finding time for a meeting in his busy schedule.

This work has been carried out in the Automatic Control Group at Linköping University and I am very thankful to Professor Lennart Ljung for letting me join the group. I thank everyone in the group for the inspiring and friendly atmosphere they are creating. I especially thank Professor Torkel Glad for sharing his extensive knowledge and for his excellent graduate courses, Ulla Salaneck and Åsa Karmelind for their help with many practical issues, and Dr. Johan Sjöberg for interesting and helpful discussions. I am also thankful to all the people in the the Automatic Control Robotics Group, for their support, and also for their first class research cooperation with ABB Robotics.

This work was supported by ABB Robotics, the Swedish Research Council (VR), and Vinnova, all gratefully acknowledged. At ABB Robotics, I would first of all like to thank the former head of controller development, Jesper Bergsjö, for supporting my research. The support from Niclas Sjöstrand, Henrik Jerregård, Wilhelm Jacobsson, and Staffan Elfving is also thankfully acknowledged. I am also greatly indepted to Dr. Torgny Brogård at ABB Robotics for the support and guidance he is giving me. Furthermore, I would like to thank all my other friends and colleagues at ABB Robotics for creating an atmosphere filled with great knowledge, but also with fun, two factors that provide a constant inspiration to my work. Among present and former colleagues, I would especially like to mention, in order of appearance, Ingvar Jonsson, Mats Myhr, Henrik Knobel, Lars Andersson, Sören Quick, Dr. Steve Murphy, Professor Mikael Norrlöf, Mats Isaksson, Professor Geir Hovland, Sven Hanssen, and Hans Andersson. I would also like to thank all master-thesis students whom I have had the privilege to supervise and learn from.

My co-authors are also greatly acknowledged, Sven Hanssen for his complete devotion to mechatronics and, as an expert in modeling, being absolutely invaluable for my work, Professor Svante Gunnarsson and Dr. Torgny Brogård for guiding me in my work and keeping me on the right track, Dr. Jonas Öhr who inspired me to take up my graduate studies, and for inspiring discussions about automatic control, and finally Dr. Erik Wernholt for equally inspiring discussions about identification. I am also thankful to Professor Per-Olof Gutman at Israel Institute of Technology, for teaching me QFT in an excellent graduate course, as well as being inspirational in many ways.

I am also very grateful to Dr. Torgny Brogårdh, Professor Svante Gunnarsson, Dr. Erik Wernholt, Professor Mikael Norrlöf, Dr. Jonas Öhr, Mats Isaksson, Sven Hanssen, Johanna Wallén, Dr. Tomas Olsson, Patrik Axelsson, André Carvalho Bittencourt, and Karin Moberg for reading different versions of this thesis, or parts thereof, and giving me valuable comments and suggestions.

Finally, I would like to thank my son John for programming of identification software, and for helping me with the cover design, and my wife Karin for, among many things, helping me with the English language, teaching me how to pronounce *manipulator*, *trajectory*, and *accuracy*, and for helping me structuring and shortening (?) my presentations. And to the both of you, as well as the rest of my immediate family, thanks for the love, patience, and support you are constantly giving me.

Stig Moberg

Linköping, December 2010

Contents

II Publications

Notation

The same symbol can sometimes be used for different purposes. The main uses of a symbol are listed here, any deviations are explained in the text. Vectors are in general column vectors.

Symbols and Operators

Notation	Meaning
$\mathbb{Z}$	The set of integers
$\mathbb{N}$	The set of natural numbers
$\mathbb{R}$	The set of real numbers
$\mathbb{C}$	The set of complex numbers
t	Time variable
s	Laplace transform variable
z	z-transform variable
q	Shift operator, $qu(t) = u(t + T_s)$
q^{-1}	Backwards shift operator, $q^{-1}u(t) = u(t - T_s)$
q	Vector of motor and joint angular positions
q_m	Vector of motor angular positions
q_a	Vector of joint angular positions
q_g	Vector of actuated joint angular positions
q_e	Vector of non-actuated joint angular positions
τ	Torque vector
τ_m	Motor torque vector
τ_a	Joint torque vector
τ_g	Gearbox torque vector (actuated joints)
τ_e	Constraint torque vector (non-actuated joints)
$\dot{p}$	Time derivative of a variable p
$p^{[n]}$	The nth time derivative of a variable p, $\frac{d^n p}{dt^n}$
X	Cartesian position and orientation of robot tool (TCP)
Z	Cartesian position and orientation of robot tool (TCP)

Symbols and Operators (contd.)

Notation	Meaning
$u(t)$	Vector of input signals at time t
$y(t)$	Vector of (measured) output signals at time t
$z(t)$	Vector of (controlled) output signals at time t
$x(t)$	State vector at time t
p_d	Reference (desired) value for a variable p
$\Gamma(q)$	Forward kinematics
$J(q)$	Velocity Jacobian $\frac{\partial\Gamma(q)}{\partial q}$
$M(q)$	Inertia matrix
$M_m(q)$	Inertia matrix of motors
$M_a(q)$	Inertia matrix of links (joints)
$c(q,\dot{q})$	Vector of Coriolis and centripetal torques
$g(q)$	Vector of gravity torques
m	Mass of a rigid body
ξ	Center of gravity (mass) of a rigid body
J	Inertia tensor of a rigid body
K	Stiffness matrix
D	Damping matrix
k	Spring constant, i.e., one element of K
d	Damping constant, i.e., one element of D
$f(q)$	Friction torque
η	Gear ratio matrix
Υ_a	Number of actuated joints
Υ_{na}	Number of non-actuated joints
T_s	Sample time
θ	Vector of model parameters
$\hat{\theta}$	Estimated vector of model parameters
$\hat{y}(t\|t-T_s;\theta)$	A model's prediction of $y(t)$ given θ and data up to time $t-T_s$
$\varepsilon(t,\theta)$	Prediction error $y(t)-\hat{y}(t\|t-T_s;\theta)$
$\arg\min_x f(x)$	The value of x that minimizes $f(x)$
Q	Number of operating points
N	Number of samples
M	Number of experiments (or blocks of experiments for MIMO systems)
P	Number of periods
N_f	Number of excited frequencies
N_p	Number of samples in one period
n_u	Number of inputs
n_y	Number of outputs
σ	Standard deviation

Symbols and Operators (contd.)

Notation	Meaning
$U(\omega_k)$	DFT of $u(t)$
$Y(\omega_k)$	DFT of $y(t)$
$\widehat{G}_k^{(i)}$	Non-parametric (estimated) FRF in operating point i at frequency k
$G_{k,\theta}^{(i)}$	Parametric (model) FRF in operating point i at frequency k using model parameters θ
$W_k^{(i)}$	Weighting matrix in operating point i at frequency k
vec(B)	Stacks the columns b_i of a matrix B into a column vector $= [b_1^T\, b_2^T \ldots b_n^T]^T$
$V_N(\theta)$	Cost function or prediction error criterion
$\hat{\theta}_N$	Parameter estimator
θ	Position of robot tool in inverse dynamics papers
ν	Differential index of DAE
diag(a)	A diagonal matrix with vector a on the diagonal
$\in$	Belongs to
$\|z\|$	Absolute value of complex variable z
arg(z)	Argument or phase of complex variable z
$\frac{\partial G(x)}{\partial x}$	Partial derivative of G with respect to x. Also denoted G_x.
det A	Determinant of matrix A
Tr A	Trace of matrix A
A^T	Transpose of matrix A
A^{-1}	Inverse of matrix A
A^H	Complex conjugate transpose of matrix A

Abbreviations

Abbreviation	Meaning
e.g.	for example
i.e.	in other words
BDF	Backwards Differentiation Formula
CAD	Computer Aided Design
DAE	Differential Algebraic Equation
DFT	Discrete Fourier Transform
DOF	Degrees Of Freedom
EE	Elastic Element

Abbreviations (contd.)

Abbreviation	Meaning
ETFE	Empirical Transfer Function Estimate
FDB	FeeDBack
FEM	Finite Element Model
FF	FeedForward
FFT	Fast Fourier Transform
FFW	FeedForWard
FL	Feedback Linearization
FRF	Frequency Response Function
IRB	Industrial RoBot, used in names for ABB robots
ILC	Iterative Learning Control
LLS	Logarithmic Least Squares
LQ	Linear Quadratic
MIMO	Multiple Input Multiple Output
MP	Minimum Phase
NLP	NonLinear Program
NLS	Nonlinear Least Squares
NMP	Non-Minimum Phase
ODE	Ordinary Differential Equation
PDE	Partial Differential Equation
PEM	Prediction Error Method
PD	Proportional and Derivative
PI	Proportional and Integral
PID	Proportional, Integral, and Derivative
QFT	Quantitative Feedback Theory
RB	Rigid Body
SISO	Single Input Single Output
TCP	Tool Center Point

1 Introduction

Models of robot manipulators are important components of a robot motion control system. The control algorithms and the trajectory generation algorithms are two equally important components. This thesis deals with some aspects of modeling and control of flexible manipulators. Here, flexible should be interpreted as elastic, and a manipulator should be interpreted as an industrial robot although the results to some extent can be applied to other types of manipulators and mechanical systems.

1.1 Motivation and Problem Statement

Robot motion control is a key competence for robot manufacturers, and current development is focused on increasing the robot performance, reducing the robot cost, improving safety, and introducing new functionalities as described in Brogårdh (2007, 2009). There is a need to continuously improve the models and control methods in order to fulfil all conflicting requirements, for example, increased performance for a robot with lower weight, and thus lower mechanical stiffness and more complicated vibration modes. One reason for this development of the robot mechanical structure is of course cost reduction, but other benefits are lower power consumption, as well as improved dexterity, safety issues, and lower environmental impact. The need of cost reduction can result in the use of optimized robot components, which usually have larger individual variation, e.g., variation of gearbox stiffness or in the parameters describing the mechanical arm. Cost reduction sometimes also results in a higher level of disturbances and nonlinearities in some of the components, e.g., in the actuators or in the sensors. The development of industrials robots is illustrated in Figure 1.1 which shows that the robot weight to payload ratio has been reduced 3 times since the 1980's.

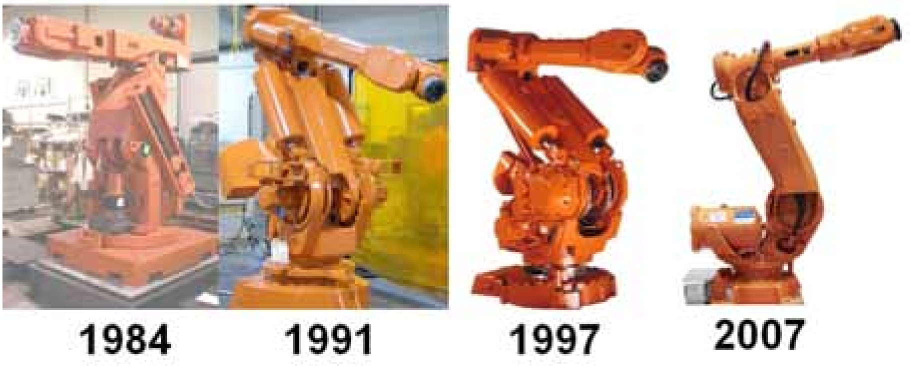

Figure 1.1: The development of large ABB robots. 1984: IRB90 1450 *kg and payload* 90 *kg (16:1), 1991: IRB6000* 1750 *kg and payload* 120 *kg (15:1), 1997: IRB6400R* 2100 *kg and payload* 200 *kg (10.5:1), 2007: IRB6640* 1300 *kg and payload* 235 *kg (5.5:1)*

An industrial robot is a general purpose machine for industrial automation, and even though the requirements of a certain application can be precisely formulated, there are no limits in what the robot users want with respect to the desirable performance and functionality of the motion control of a robot. The required motion control performance depends on the application. The better performance, the more applications can be subject to automation by a specific robot model. Some requirement examples are:

- High path accuracy in continuous applications (e.g., laser welding, laser cutting, dispensing, or water-jet cutting).
- High speed accuracy in continuous applications (e.g., painting or dispensing).
- Low cycle time (high speed and acceleration) in discrete applications (e.g., material handling).
- Small overshoots and a short settling time in discrete process applications (e.g., spot welding).
- High control stiffness in contact applications (e.g., machining).

For weight- and cost optimized industrial manipulators, the requirements above can only be handled by increased computational intelligence, i.e., improved motion control. Motion control of industrial robot manipulators is a challenging task, which has been studied by academic and industrial researchers for more than three decades. Some results from the academic research have been successfully implemented in real industrial applications, while other results are far away from being relevant to the industrial reality. To some extent, the development of motion control algorithms has followed two separate routes, one by academic researchers and one by robot manufacturers, unfortunately with only minor interaction.

The situation can partly be explained by the fact that the motion control algorithms used in the industry sometimes are regarded as trade secrets. Due to the tough competitive situation among robot manufacturers, the algorithms are seldom published. Another explanation is that the academic robot control researchers often apply advanced mathematics on a few selected aspects of relatively small systems, whereas the industrial robot researchers and developers must deal with all significant aspects of a complex system where the proposed advanced mathematics often cannot be applied. Furthermore, the problems that the robot industry sometimes present might include too much engineering aspects to be attractive for the academic community. Industrial robot research and development must balance short term against long term activities. Typical time constants from start of research to the final product, in the area of the motion control technology discussed in this work, can be between 5 and 10 years, and sometimes even longer. Thus, long-term research collaborations between industry and academia should be possible, given that the intellectual property aspects can be handled.

The problems of how to get industrial-relevant academic research results, and of how to obtain a close collaboration between universities and industry, are not unique for robotics motion control. The existence of a gap between the academic research and industrial practise in the area of automatic control is often discussed. One balanced description on the subject can be found in Bernstein (1999), where it is pointed out that the control practitioners must articulate their needs to the research community, and that motivating the researchers with problems from real applications "can have a significant impact on increasing the relevance of academic research to engineering practise". Another quote from Bernstein (1999) is "I personally believe that the gap on the whole is large and warrants serious introspection by the research community". The problem is somewhat provocatively described in Ridgely and McFarland (1999) as, freely quoted, "what the industry in most cases do not want is stability proofs, guarantees of convergence and other purely analytical developments based on idealized and unrealistic assumptions". Another view on the subject is that "the much debated theory-applications gap is a misleading term that overlooks the complex interplay between physics, invention and implementation, on the one side, and theoretical abstractions, models, and analytical designs, on the other side" (Kokotovic and Arcak, 2001). The need for a balance between theory and practise is expressed in Åström (1994), and finally a quote from Brogårdh (2007): "industrial robot development has for sure not reached its limits, and there is still a lot of work to be done to bridge the gap between the academic research and industrial development".

It is certainly true that, as in the science of physics, research on both "theoretical control" and "experimental control" is needed. The question is whether the proportions need adjustment. This subject is certainly an important one, as automatic control can have considerable impact on many industrial processes as well as on other areas, affecting both environmental and economical aspects. It is my hope that this work can help bridging the gap, as well as moving the frontiers somewhat in the area of robotics motion control.

This thesis concerns modeling and control of flexible manipulators. Most publications concerning flexible (elastic) robot manipulators only consider elasticity in the rotational direction. If the gear elasticity is considered we get the *flexible joint model*, and if link deformation restricted to a plane perpendicular to the preceding joint is included in the model we get the *flexible link model*. These restricted models simplify the control design but limit the attainable performance. Motivated by the trend of developing light-weight robots, a new model, here called the *extended flexible joint model*, is proposed for use in motion control systems as well as in robot design and performance simulation. The use of this extended model is the main theme of this thesis, and the following aspects are treated:

- Identification of the unknown elastic model parameters, applied to a real six-axes industrial robot.

- Inverse dynamics to enable high-accuracy path tracking by the use of feedforward control.

The remainder of this work deals with feedback control. For flexible joint models, feedback control for path tracking is investigated by comparing nonlinear feedback control to nonlinear feedforward control. Finally, two benchmark problems on robust feedback control are presented together with some suggested solutions.

1.2 Outline

Part I contains an overview of robotics, modeling, identification, and control. Part II consists of a collection of edited papers.

1.2.1 Outline of Part I

Chapter 2 gives an introduction to robotics in general, and the motion control problem in particular. Modeling of robot manipulators is described in Chapter 3, and some system identification methods that are relevant for this thesis are described in Chapter 4. A survey on control methods used in robotics can be found in Chapter 5. Finally, Chapter 6 provides a summary and some ideas for future research.

1.2.2 Outline of Part II

This part consists of a collection of edited papers, introduced below. Summary and background of each paper are given, together with the contribution of the author of this thesis.

Paper A: Modeling and Parameter Estimation of Robot Manipulators using Extended Flexible Joint Models

> S. Moberg, E. Wernholt, S. Hanssen, and T. Brogårdh. Modeling and parameter estimation of robot manipulators using extended flexible joint models. 2010. *Submitted to Journal of Dynamic Systems Measurement and Control, Transactions of the ASME.*

Summary: This paper considers the problem of dynamic modeling and identification of robot manipulators with respect to their elasticities. The so-called flexible joint model, modeling only the torsional gearbox elasticity, is shown to be insufficient for modeling a modern industrial manipulator accurately. The extended flexible joint model, where non-actuated joints are added to model the elasticity of the links and bearings, is used to improve the model accuracy. The unknown elasticity parameters are estimated using a frequency domain gray-box identification method. A detailed description of this method is provided in Paper B. Similar elasticity model parameters are obtained when using two different output variables for the identification, the motor position and the tool acceleration respectively. A brief time-domain model validation is also presented.

Background and contribution: The basic idea of extending the flexible joint model for a better description of modern light-weight manipulators is mainly due to the author and Sven Hanssen. These two also performed the first promising attempt to identify the model by use of frequency-domain methods. The research was continued by the authors of Öhr et al. (2006) where the use of the nonparametric frequency response function (FRF) for the estimation of the parametric robot model was first described. Erik Wernholt has continued to improve and analyze various aspects of the identification procedure as described in Wernholt (2007). The improvements of the identification procedure presented in this paper is due to the author and Erik Wernholt, the model equations are derived by Sven Hanssen, and the simulated and experimental evaluation were performed by the author. Torgny Brogårdh has throughout this whole work served as a most valuable discussion partner.

Paper B: Nonlinear Gray-Box Identification Using Local Models Applied to Industrial Robots

> E. Wernholt and S. Moberg. Nonlinear gray-box identification using local models applied to industrial robots. *Automatica*, 2010. Accepted for publication.

Summary: This paper studies the problem of estimating unknown parameters in nonlinear gray-box models that may be multivariable, nonlinear, unstable, and resonant at the same time. A straightforward use of time-domain predication-error methods for this type of problem easily ends up in a large and numerically stiff optimization problem. An identification procedure, that uses intermediate local models that allow for data compression and a less complex optimization problem, is therefore proposed. The procedure is illustrated by estimating elasticity parameters in a six-axes industrial robot. Different parameter estimators are compared and experimental results show the usefulness of the proposed identification procedure. A brief example of time-domain identification is also presented and compared to the suggested frequency-domain method.

Background and contribution: The basic idea of using the nonparametric FRF for the estimation of the parametric robot model is due to the author and Sven Hanssen as previously described. The research was continued by the authors of Öhr et al. (2006). Erik Wernholt has continued to improve and analyze various aspects of the identification procedure as described in Wernholt (2007) where the author has served as a discussion partner. The first theoretical part of this paper is mainly due to Erik Wernholt, whereas the second part with simulation and experimental results is mainly due to the author. The time-domain example is also performed by the author.

Paper C: Inverse Dynamics of Flexible Manipulators

> S. Moberg and S. Hanssen. Inverse dynamics of flexible manipulators. In *Multibody Dynamics 2009*, Warsaw, Poland, July 2009.

Summary: This paper investigates different methods for the inverse dynamics of the extended flexible joint model. The inverse dynamics solution is needed for feedforward control, which is often used for high-precision robot manipulator control. The inverse dynamics of the extended flexible joint model can be computed as the solution of a high-index differential algebraic equation (DAE) and different solvers are suggested and evaluated. The inverse dynamics can be solved as an initial-value problem if the zero dynamics of the system is stable, i.e., minimum phase. For unstable zero dynamics, an optimization approach based on the discretized DAE is suggested. An collocation method, using a continuous DAE formulation, is also suggested and evaluated. The solvers are illustrated by simulation, using a manipulator with two actuators and five degrees-of-freedom.

Background and contribution: The DAE formulation of the inverse dynamics problem was the result of a discussion between the author and Sven Hanssen. Sven Hanssen derived the simulation model, and the author has made the control research and implemented the DAE solvers that are used. Sven Hanssen has been a discussion partner for this part of the work and also derived and analyzed alternative solvers together with the author.

Paper D: Inverse Dynamics of Robot Manipulators Using Extended Flexible Joint Models

> S. Moberg and S. Hanssen. Inverse dynamics of robot manipulators using extended flexible joint models. 2010. *Submitted to IEEE Transactions on Robotics (under revision).*

Summary: This article is based on Paper C. The suggested concept for inverse dynamics is experimentally evaluated using an industrial robot manipulator. In this experimental evaluation, an identified model is used in the inverse dynamics computation. Simulations using the same identified model are in good agreement with the experimental results. The conclusion is that the extended flexible joint inverse dynamics method can improve the accuracy for manipulators with significant elasticities, that cannot be described by the flexible joint model.

Background and contribution: The background and contribution concerning the problem formulation and the suggested solvers are as described previously for Paper C. The experimental evaluation was performed by the author and Sven Hanssen.

Paper E: On Feedback Linearization for Robust Tracking Control of Flexible Joint Robots

> S. Moberg and S. Hanssen. On feedback linearization for robust tracking control of flexible joint robots. In *Proc. 17th IFAC World Congress*, Seoul, Korea, July 2008.

Summary: Feedback linearization is one of the major academic approaches for controlling flexible joint robots. This contribution investigates the discrete-time implementation of the feedback linearization approach on a realistic three-axis robot model. A simulation study of high speed tracking with model uncertainty is performed. The feedback linearization approach is compared to a feedforward approach.

Background and contribution: The feedback linearization for flexible joint robots was first presented in Spong (1987). After attending a workshop on nonlinear control of flexible joint robots, the author and Sven Hanssen discussed the lack of (published) evaluations of this control concept. This paper uses a realistic manipulator model and realistic requirements to evaluate the proposed concept in a simulation study. The author was responsible for the implementation of the control schemes, control analysis and performance evaluation. Sven Hanssen derived the simulation model and the model derivatives, needed for the implementation of the control algorithms.

Paper F: A Benchmark Problem for Robust Feedback Control of a Flexible Manipulator

> S. Moberg, J. Öhr, and S. Gunnarsson. A benchmark problem for robust feedback control of a flexible manipulator. *IEEE Transactions on Control Systems Technology*, 17(6):1398–1405, November 2009.

Summary: This paper describes a benchmark problem for robust feedback control of a flexible manipulator together with some proposed and tested solutions. The system to be controlled is a four-mass system subject to input saturation, nonlinear gear elasticity, model uncertainties, and load disturbances affecting both the motor and the arm. The system should be controlled by a discrete-time controller that optimizes the performance for given robustness requirements.

Background and contribution: The benchmark problem was first presented as *Swedish Open Championships in Robot Control* (Moberg and Öhr, 2004, 2005) where the author formulated the problem together with Jonas Öhr. The analysis of the solutions, as well as the experimental validation of the benchmark model, were performed mainly by the author. The final paper as presented in this thesis also includes Svante Gunnarsson as a valuable discussion partner and co-author.

Paper G: A Benchmark Problem for Robust Control of a Multivariable Nonlinear Flexible Manipulator

> S. Moberg, J. Öhr, and S. Gunnarsson. A benchmark problem for robust control of a multivariable nonlinear flexible manipulator. In *Proc. 17th IFAC World Congress*, Seoul, Korea, July 2008.

Summary: This paper describes a benchmark problem for robust feedback control of a two link manipulator with elastic gear transmissions. The gear transmission is described by nonlinear friction and elasticity. The system is uncertain according to a parametric uncertainty description and due to uncertain disturbances affecting both the motors and the tool. The system should be controlled by a discrete-time controller that optimizes performance for given robustness requirements. The proposed model is validated by experiments on a real industrial manipulator.

Background and contribution: This benchmark problem is a continuation of the problem presented in Paper F, using a more complex and realistic model, and with a more challenging design task. The author contributed to the majority of this work with Jonas Öhr and Svante Gunnarsson as valuable discussion partners.

Related Publications

Publications of related interest not included in this thesis, where the author of this thesis has contributed:

> T. Brogårdh, S. Moberg, S. Elfving, I. Jonsson, and F. Skantze. Method for supervision of the movement control of a manipulator. US Patent 6218801, April 2001. URL `http://www.patentstorm.us/patents/6218801`.
>
> T. Brogårdh and S. Moberg. Method for determining load parameters for a manipulator. US Patent 6343243, Januari 2002. URL `http://www.patentstorm.us/patents/6343243.html`.
>
> G.E. Hovland, S. Hanssen, E. Gallestey, S. Moberg, T. Brogårdh, S. Gunnarsson, and M. Isaksson. Nonlinear identification of backlash in robot transmissions. In *Proc. 33rd ISR (International Symposium on Robotics)*, Stockholm, Sweden, October 2002.
>
> S. Moberg and J. Öhr. Robust control of a flexible manipulator arm: A benchmark problem. Prague, Czech Republic, 2005. 16th IFAC World Congress.

S. Gunnarsson, M. Norlöf, G. Hovland, U. Carlsson, T. Brogårdh, T. Svensson, and S. Moberg. Pathcorrection for an industrial robot. US Patent 7130718, October 2006. URL `http://www.patentstorm.us/patents/7130718.html`.

J. Öhr, S. Moberg, E. Wernholt, S. Hanssen, J. Pettersson, S. Persson, and S. Sander-Tavallaey. Identification of flexibility parameters of 6-axis industrial manipulator models. In *Proc. ISMA2006 International Conference on Noise and Vibration Engineering*, pages 3305–3314, Leuven, Belgium, September 2006.

S. Moberg and S. Hanssen. A DAE approach to feedforward control of flexible manipulators. In *Proc. 2007 IEEE International Conference on Robotics and Automation*, pages 3439–3444, Roma, Italy, April 2007.

E. Wernholt and S. Moberg. Frequency-domain gray-box identification of industrial robots. In *17th IFAC World Congress*, pages 15372–15380, Seoul, Korea, July 2008a.

E. Wernholt and S. Moberg. Experimental comparison of methods for multivariable frequency response function estimation. In *17th IFAC World Congress*, pages 15359–15366, Seoul, Korea, July 2008b.

M. Björkman, T. Brogårdh, S. Hanssen, S.-E. Lindström, S. Moberg, and M. Norrlöf. A new concept for motion control of industrial robots. In *Proceedings of 17th IFAC World Congress, 2008*, Seoul, Korea, July 2008.

R. Henriksson, M. Norrlöf, S. Moberg, E. Wernholt, and T. Schön. Experimental comparison of observers for tool position estimation of industrial robots. In *Proceedings of 48th IEEE Conference on Decision and Control*, pages 8065–8070, Shanghai, China, December 2009.

1.3 Contributions

The main contributions of the thesis are:

- The extended flexible joint model presented in Öhr et al. (2006) and Moberg and Hanssen (2007) and validated in Paper A.
- The identification procedure introduced in Öhr et al. (2006) and further described in Papers A and B. The procedure has been successfully applied to experimental data from a six-axes industrial robot.
- The DAE formulation of the inverse dynamics problem for the extended flexible joint model as described in Paper C and D.
- The solution method for the inverse dynamics problem of the extended flexible joint model, and its application on a small but realistic robot model.

This is also described in Papers C and D.

- The experimental evaluation of the inverse dynamics method as described in Paper D.
- The evaluation of feedback linearization as described in Paper E.
- The formulation and evaluation of a relevant industrial benchmark problem as described in Paper F.
- The formulation of a second relevant industrial benchmark problem as described in Paper G. Solutions to the problem are planned to be published in the future. The model is available for download at Moberg (2007), and has also been used by other researchers for various purposes.

Part I

Overview

2 Robotics

Robotics involves many technical and scientific disciplines, for example, sensor- and vision technologies, computer architecture, drive systems and motor technologies, real time systems, automatic control, modeling, mechanical design, applied mathematics, man-machine interaction, system communication, and computer languages. This chapter gives a short introduction describing the parts that are relevant for this work, i.e., modeling and motion control.

2.1 Introduction

Throughout this work, the term robot is used to denote an industrial robot, i.e., a manipulator arm, mainly used for manufacturing in industry. Some examples of such robots are shown in Figure 2.1. The first industrial robots were installed in the 1960s and the number of operational industrial robots at the end of 2009 was in the range of $1,021,000$ to $1,300,000$ units (IFR, 2010). Robots are used for a variety of tasks, for example, welding, painting, cutting, dispensing, material handling, machine tending, machining, and assembly. There are many types of mechanical robot structures such as the parallel arm robot and the articulated robot, which can be of elbow or parallel linkage type. Examples of these three robot structures are shown in Figure 2.2. Further examples of mechanical robot structures (also called kinematic structures) can, e.g., be found in Spong et al. (2006). In the following, the presentation will be restricted to the, at present, most common type of industrial robot which is the six-axes articulated robot of elbow type. This robot, or manipulator arm, consists of seven serially mounted bodies connected by six revolute joints. The bodies are numbered from 0 to 6, and body 1 to 6 are also called links. The links and the joints are numbered from 1 to 6. The links are actuated by electrical motors via gear transmissions (gearboxes),

Figure 2.1: *ABB robot family and the IRC5 controller.*

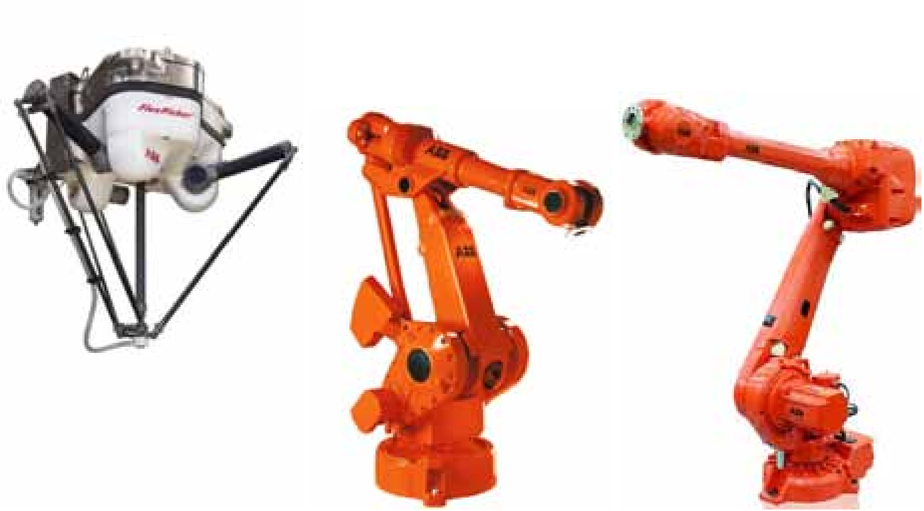

Figure 2.2: *Three examples of robot structures from ABB. The parallel arm robot IRB340 (left), the parallel linkage robot IRB4400 (middle), and the elbow robot IRB4600 (right).*

also named speed reducers. The motor positions are measured by sensors. The first body is connected to the base, and the last body (link) is connected to an end effector, i.e., a tool. With six actuated links, both the position and the orientation of the end effector can be controlled. The first three joints and links are often denoted main axes, and the last three are denoted wrist axes.

2.2 Models

A description of the most important models used in robotics can be found in, e.g., Craig (1989), Spong et al. (2006), Siciliano and Khatib (2008), and Siciliano et al. (2010). Here follows a short overview.

2.2.1 Kinematic Models

The kinematic models describe the robot motions without regard to the forces that cause the motions, i.e, all time-based and geometrical properties of the motion. The kinematics relate the joint angular position vector q to the position p and orientation ϕ of the tool frame attached to the tool and positioned in the tool center point (TCP). One example of a kinematic relation is the forward kinematics where the tool frame position and orientation are described as a function of the joint angular position vector as

$$X = \Gamma(q), \tag{2.1}$$

where X is the tool frame position and orientation, also named pose, defined as

$$X = \begin{bmatrix} p \\ \phi \end{bmatrix}, \tag{2.2}$$

and $\Gamma(\,\cdot\,)$ is a nonlinear function. The tool frame is described in a reference frame, i.e., a coordinate system, attached to the base of the robot, called base frame. Describing the manipulator pose by the joint angles is often denoted a joint space representation of the robot state while describing it by the tool position and orientation is denoted a task space representation which is usually implemented in Cartesian coordinates. The described frames and the joint positions are illustrated in Figure 2.3.

2.2.2 Dynamic Models

Dynamic robot models describe the relations between the motions of the robot and the forces that cause the motions. The models are most often formulated in joint space. One example of a dynamic model is the model of a rigid manipulator which can be expressed as

$$\tau = M(q)\ddot{q} + c(q, \dot{q}) + g(q) + f(\dot{q}), \tag{2.3}$$

where τ is the actuator torque vector, $M(q)$ is the inertia matrix, $c(q, \dot{q})$ is a vector of Coriolis and centripetal torques, $g(q)$ is the gravity torque vector, and $f(\dot{q})$ is the vector of, possibly nonlinear, joint friction torques. The rigid body inertial

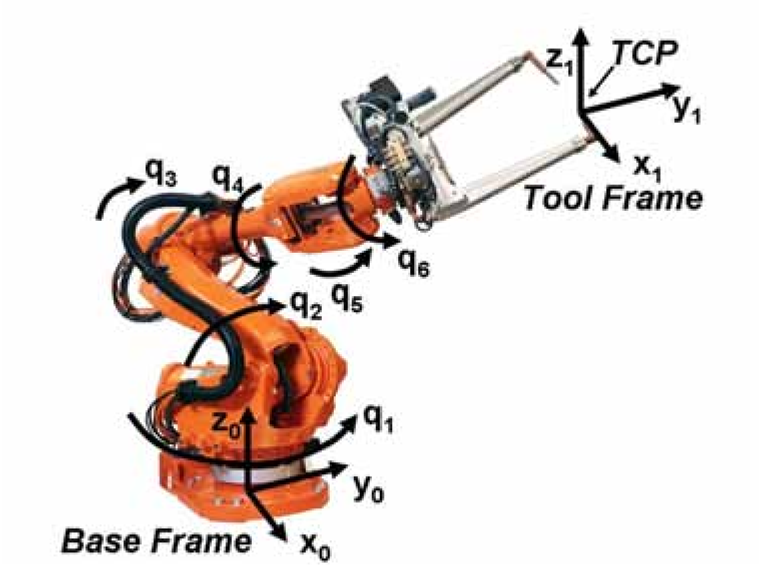

Figure 2.3: Base frame, tool frame, and joint positions illustrated on a robot equipped with a spotwelding gun.

parameters for each link are the mass, the center of mass, and the inertia. The actuator inertia and mass are added to the corresponding link parameters.

The inverse dynamics problem is useful for control, and consists of computing the required actuator torques as a function of the joint position vector q and its time derivatives, $\dot{q}$ and $\ddot{q}$. For the rigid model (2.3), this involves algebraic computations only. For simulation of the manipulator movement, the direct dynamics problem must be solved. The differential equation (2.3) is then solved with the actuator torques as input.

2.3 Motion Control

The motion control of a modern industrial manipulator is a complex task. A description of the current status of industrial robot motion control can be found in Brogårdh (2007, 2009), and references therein.

2.3.1 A General Motion Control System

A general robot motion control system, capable of synchronously controlling n robot manipulators, is illustrated in Figure 2.4. The system consists of the following components:

Robot 1, Robot 2, ..., Robot n The robot manipulators with actuators and sensors included. The manipulators can be in contact with the environment,

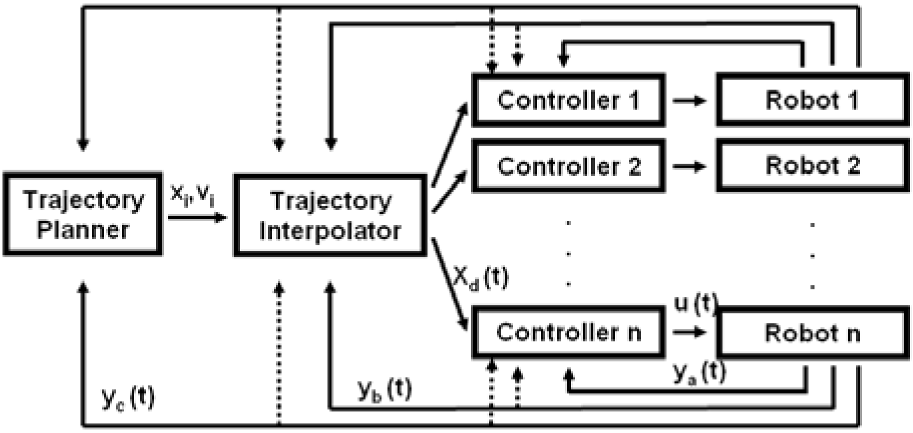

Figure 2.4: *A general robotics motion control system.*

e.g., in assembly tasks, or operate in free space without contact with the environment, e.g., in laser cutting. The sensors can be of different types. A first type of sensors generates sensor readings y_a which are used by the feedback controller. Examples of such sensors are encoders or resolvers measuring actuator positions, accelerometers measuring link and tool acceleration, force sensors[1] measuring the contact forces acting on the end effector, torque sensors measuring the joint torques on the link side of the gearbox, and joint encoders measuring the joint positions. The second type of sensor information, y_b, can be exemplified by conveyor positions measured by encoders, or the position of an arc-welding gun relative to the desired welding path as measured by a tracking sensor. The second type of sensors are used by the trajectory interpolator and the controller to adapt the robot motion to the measured path. A third type of sensors are primarily used by the trajectory planner, e.g., in the case of a vision system specifying the position of an object to be gripped by the robot. Some of the robots in Figure 2.4 could also be replaced by multi- or single-axis positioners used for, e.g., rotating the object in an arc-welding application.

Controller 1, Controller 2, ..., Controller n Generate control signals $u(t)$ for the actuators with the references $X_d(t)$ and the sensor readings $y_a(t)$ as inputs. The controller can operate in position control mode for point-to-point motion in, e.g., a spot-welding application, or continuous path tracking in, e.g., a dispensing application. When the robot is in contact with the environment, the controller can still be in position control mode, e.g., in premachining applications. Some contact applications require a compliant behavior of the robot due to uncertain geometry or process requirements. This

[1] The sensor is called force sensor for simplicity, even though both forces and torques are normally measured.

requires a controller in compliance control mode for some directions, and position control mode for other directions in task space. Examples of these applications, using compliance control, are assembly, machine tending, and product testing. In other applications as friction stir welding, grinding, and polishing, the contact force must be controlled in a specific direction while position or speed control is performed in the other directions. Compliance and force control can be accomplished with or without the use of force sensors, depending on the performance requirements.

Trajectory Interpolator The task of the trajectory interpolator is to compute controller references $X_d(t)$ that follow the programmed trajectory and which simultaneously are adapted to the dynamic performance of the robot. The input from the trajectory planner is the motion specification, e.g., motion commands specifying a series of end effector positions x_i along with desired end effector speeds v_i. Sensor readings $y_b(t)$ and $y_c(t)$ can also be used by the trajectory interpolator. The trajectory $X_d(t)$ contains positional information for all n robots, and can be expressed in Cartesian or joint space.

Trajectory Planner Specifies the desired motion of the robot end effector. This can be done manually by a robot programmer who specifies the motion in a robot programming language with a series of motion commands. The program can be taught by moving the robot to the desired positions and command the robot to read the actual positions. The motion commands can also be generated by an off-line programming system. The desired motion can also be expressed on a higher level by task programming as, e.g., by an instruction as picking all objects on a moving conveyor, and placing the objects in desired locations. The positions for picking the objects is then specified by a vision system.

Besides the basic motion control functionality as described in Figure 2.4, the robot motion control also includes, for example, the following functionality:

Absolute Accuracy Identification of the link parameters for an extended kinematic model in order to obtain high volumetric accuracy. This is useful for off-line programming and for fast robot replacement (Brogårdh, 2007).

ILC Iterative learning control for improved path accuracy, used, e.g., for high precision laser cutting (Norrlöf, 2000; Gunnarsson et al., 2006).

Load Identification Identification of user tool and load to improve the dynamic model accuracy for high control performance, and to avoid overload of the robot mechanics (Brogårdh and Moberg, 2002).

Collision Detection Model based collision detection to save equipment at accidental robot movements, for example during programming (Brogårdh et al., 2001).

Diagnosis Diagnosis of the status and behavior of, e.g., gearboxes and mechanical brakes.

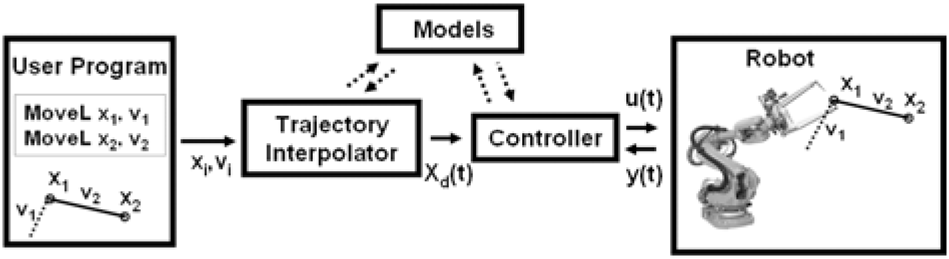

Figure 2.5: *A simplified robotics motion control system.*

Special Control Modes One example is emergency stop control modes where the mechanical brakes and the motors are used simultaneously to stop the robot. Another example from ABB is *WristMove*, where the robot accuracy in cutting applications is increased by moving the wrist axes only (Jerregård and Pihl, 2007).

Supervision Supervision of, e.g., position and speed for safety reasons, and for saving equipment on and close to the robot.

Calibration Calibration methods for, e.g., tool frame or actuators.

Autotune Methods Methods for adapting model parameters or controller tuning to a specific robot individual or installation.

2.3.2 A Model-Based Motion Control System for Position Control

A simplified outline of a motion control system is shown in Figure 2.5. The system controls one robot manipulator in position control mode. The trajectory interpolator and the controller make an extensive use of models, hence this type of control is denoted model-based control. The system has the following components:

Robot Manipulator The physical robot arm with actuators receiving the control signal $u(t)$ from the controller. The control signal can be, e.g., a torque reference to a torque controller, a velocity reference to a velocity controlled actuator or a three-phase current to an electrical motor. Throughout this work, the torque control is assumed to be ideal and a part of the actuator, which has been proven by experiments to be a reasonable assumption for most of the ABB robots. Hence, the control signal $u(t)$ will be a motor torque reference. The sensor readings $y(t)$ are normally the actuator positions only, but more sensors can be added as described in the previous section.

Models The models used by the motion control system, e.g., the kinematic and dynamic models described previously. The models and how to obtain the model parameters will be further described in Chapters 3 – 4.

Controller Generates control signals $u(t)$ for the actuators with the references

$X_d(t)$ and the sensor readings $y(t)$ as input. The controller can be split into a feedback controller and a feedforward controller, and will be further described in Chapter 5.

Trajectory Interpolator Creates a reference $X_d(t)$ for the controller with the user program as input. The first step of the trajectory interpolation could be to compute the continuous geometric path $X_d(\gamma)$ where γ is a scalar path parameter, e.g., the distance along the path. The second step of the interpolation is to associate a timing law to γ and obtain $\gamma(t)$. In this way, the path $X_d(\gamma)$ is transformed to a time determined trajectory $X_d(\gamma(t)) = X_d(t)$. Note that the speed and acceleration as well as higher order derivatives of the path, are completely determined once the trajectory is computed. The trajectory interpolator is responsible for limiting the speed and acceleration of the trajectory, to make it possible for the robot to dynamically follow the reference without actuator saturations. The requirement of path smoothness depends on the controller used and on the requested motion accuracy. One example of smoothness requirement is that the acceleration derivative, also called jerk, is limited.

User Program The desired motion of the robot end effector is specified by a series of motion commands in the *user program*. In the example program, the command `MoveL x2,v2` specifies a linear movement of the end effector to the Cartesian position `x2` with velocity `v2`. The movement starts in the end position of the previous instruction, i.e. `x1`. Generally, the movement can be specified in joint space or in Cartesian space. In joint space the positions are described by the joint angular positions, and in Cartesian space by the Cartesian position and orientation of the end effector. The movement in Cartesian space can typically be specified as linear or circular. In reality, more arguments must be attached to the motion command to specify, e.g., behavior when the position is reached (stop or make a smooth direction change to the next specified position), acceleration (do not exceed $5\,\mathrm{m/s^2}$), and events (set digital output 100 ms before the endpoint is reached).

The ultimate requirements on the described motion control system can be summarized as follows:

Optimal Time Requirement The user-specified speed must only be reduced if the robot movements are limited mechanically or electrically by the constraints from the robot components. Some examples of component constraints are maximum motor and gearbox torques, maximum motor speed, and maximum allowed forces and torques acting on the manipulator links. Note that the acceleration is always limited due to actuator constraints and that the user can add more constraints, for example, maximum allowed acceleration along the path.

Optimal Path Requirement The user-specified path must be followed with specified precision, also under the influence of different uncertainties. These uncertainties are disturbances acting on the robot and on the measurements, as well as uncertainties in the models used by the motion control system.

If these two requirements are fulfilled, the cycle-time performance of the robot depends entirely on the electromechanic components such as gearboxes, mechanical links, actuators, and power electronics. Generally, a given performance requirement can be fulfilled either by improving the electromechanics or by improving the computational intelligence of the software, i.e., improving the models, the control algorithms, and the trajectory optimization algorithms. The possibilities to use electromechanic or software solutions to fulfil performance requirements can be illustrated by two simple examples:

- **Requirement:** Path accuracy of 0.5 mm.
  - **Electromechanic solution:** Design the robot with a stiffer mechanical arm including bearings and gearboxes.
  - **Software solution:** Improve the models and control algorithms, i.e., a higher degree of fulfillment of the optimal path requirement.
- **Requirement:** The robot task must be accomplished in 10 s.
  - **Electromechanic solution:** Increase the power and torque of the drivetrain, i.e., the motors, gearboxes, and power electronics.
  - **Software solution:** Improve the trajectory optimization algorithms and the models used, i.e., a higher degree of fulfillment of the optimal time requirement.

A higher degree of fulfilment of the two requirements using software solutions means more complexity in the software and algorithms and, of course, more computational power. This means a more expensive computer in the controller, and initially, a longer development time. However, the trend of moving functionality from electromechanics to software will certainly continue due to the continuing development of low-cost computer hardware and efficient methods for developing more and more complex real-time software systems. However, in every product development project, there is an trade-off between electromechanics cost, i.e., the cost of gearboxes, mechanical arm, actuators, and power electronics, and the software cost[2], i.e., the cost of computers, memory, sensors, and motion control development.

To make cost optimization of robot systems, including robot electromechanics and controller software, for a spectrum of different applications, is a very difficult task. How this task is solved varies a lot from one robot manufacturer to another. However, moving as much as possible of the performance enhancement to the controller software must be regarded as the ultimate goal for any robot motion control system. Besides cost there is of course also a matter of usability which

[2] Software is not used in the normal sense here and the software cost could also be denoted the motion control cost. *Cost of brain* could also be used, and in that case, the electromechanics cost could be denoted the *cost of muscles*.

increases with the level of computational intelligence, e.g., easy programming and facilitation of advanced applications.

The *optimal time requirement* is mainly a requirement on the trajectory interpolator and the *optimal path requirement* is mainly a requirement on the controller. Accurate models are necessary for the fulfilment of both requirements. One small example of minimum-time trajectory generation can be found in Section 5.3.6.

The fulfilment of the second requirement, i.e., the optimal path requirement, is the subject of this work, and the next two chapters will treat the modeling aspects of robotics.

3 Modeling of Robot Manipulators

This chapter describes the models that are relevant for this work, namely the kinematic and dynamic models of a serial link robot of elbow type.

3.1 Kinematic Models

The kinematic models describe the motion without regard to the forces that cause it, i.e, all time-based and geometrical properties of the motion. The position, velocity, acceleration, and higher order derivatives are all described by the kinematics. A thorough description of kinematic models can be found in Craig (1989), Spong et al. (2006), Siciliano and Khatib (2008), and Siciliano et al. (2010).

3.1.1 Position Kinematics and Frame Transformations

The serial N-link robot has $N + 1$ serially mounted rigid bodies connected by N revolute joints. The rigid bodies are numbered from 0 to N, rigid body 0 is connected to the base, and rigid body N is connected to an end effector, i.e., a tool. Rigid body 1 to N are also called links and are actuated by electrical motors via gear transmissions. The joints are numbered from 1 to N and joint i connects link $i - 1$ to link i. The motor positions are measured by sensors[1]. With $N \geq 6$ actuated links, both the position and the orientation of the end effector can be fully controlled. The joint angular position vector, or joint angles, $q \in \mathbb{R}^N$, describe the configuration or position of the manipulator. The position of the tool is described by attaching a coordinate system, or frame, fixed to the tool. The tool

[1] The motor position sensor is usually an encoder or a resolver.

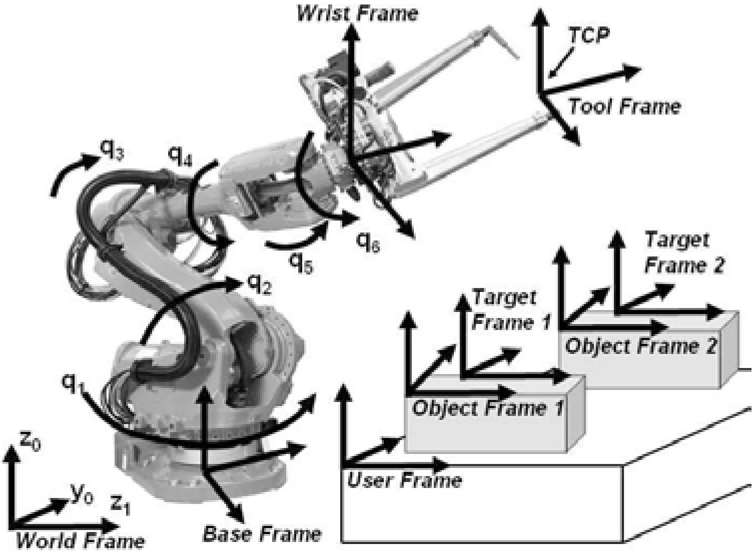

Figure 3.1: *A robot in a work-cell with the standard frames, as defined by ABB, used in cell modeling illustrated.*

pose is then described by the position $p \in \mathbb{R}^3$ and orientation[2] $\phi \in \mathbb{R}^3$ of this tool frame. The origin of the tool frame is known as the tool center point (TCP). The tool frame position is described relative to the base frame, attached to the base of the robot.

Describing the manipulator position by the joint angles q is often denoted a joint space description, while describing it by the tool position p and orientation ϕ is denoted a task space description[3]. The position kinematics relates the joint space to the task space. For a manipulator with gearboxes, the actuator space can also be defined. The actuator position $\tilde{q}$ is related to the joint position q by

$$\tilde{q} = \eta q, \tag{3.1}$$

where η is a (possibly non-diagonal) matrix of gear ratios. A robot with the described frames and the joint positions is illustrated in Figure 3.1. The other frames in the figure are typical frames used for work-cell modeling, to simplify the robot programming task[4]:

[2] There are several possible representations of orientation. Here, a minimal representation of orientation is assumed. One minimal representation is the use of three Euler angles (e.g., roll-pitch-yaw) although it suffers from mathematical singularities. These singularities can be avoided with a four component unit quaternion representation. The orientation can also be represented by a rotational matrix $R \in SO(3)$. Different representations of orientation is described in Siciliano and Khatib (2008).

[3] Task space is sometimes called operational space or Cartesian space. Joint space is also called configuration space.

[4] Different robot manufacturers have slightly different concepts and naming conventions for the frames used in cell modeling. In this text, the ABB frame concept and names are used.

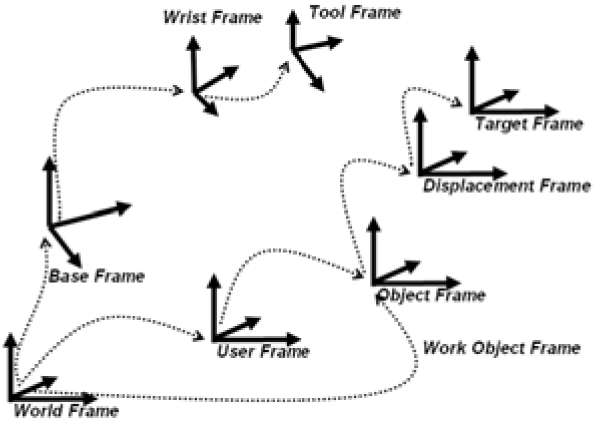

Figure 3.2: *Standard frames.*

- *World frame* is the common reference frame in a work cell, used by all robots, positioners, conveyors, and other equipment.
- *Base frame* describes the position and orientation of the base of a robot.
- *Wrist frame* is attached to the mounting flange of the robot.
- *Tool frame* describes the tool position and orientation.
- *User frame* describes a task relevant location.
- *Object frame* describes an object relative to the task-relevant location.
- *Work object frame* is the object frame as seen from the world frame, i.e., defined by user frame and object frame.
- *Displacement frame* describes locations inside an object.
- *Target frame* (or programmed position) is where the tool frame eventually should be positioned.

The position and orientation of frame B can be described relative to frame A by a homogenous transformation ${}^{A}_{B}T$ describing the translation and rotation required to move from A to B. Homogenous transformations are described in the general references given, e.g., Siciliano and Khatib (2008). Figure 3.2 shows the standard frames and the chains of transformations describing the frame positions. Note that, e.g., the user frame can be defined on a positioner, on another robot, or on a conveyor. This means that the transformation from world frame to user frame can be time-varying, defined by, e.g., the kinematics of the external positioner. The base frame can also be time-varying if, e.g., the robot base is moved by a track motion as illustrated in Figure 3.3. The desired position of the tool frame is described in the robot programs as a target frame. The target frame, which should be equal to the tool frame when the position is reached, can then be described

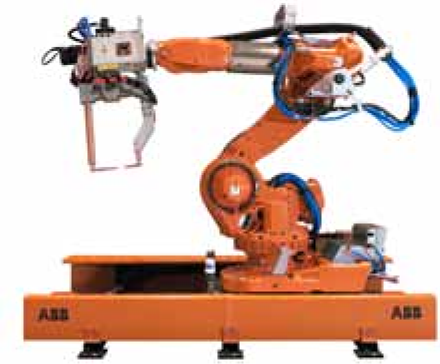

Figure 3.3: *A robot moved by a track motion.*

in the base frame using the chain of transformations. For some applications, a room-fixed TCP is convenient. Here, the tool is fixed and the work object frame is moved by the robot.

3.1.2 Forward Kinematics

The forward kinematics problem is, given the joint angles, to compute the position and orientation of the tool frame relative to the base frame, or

$$X = \Gamma(q, \theta_{kin}), \tag{3.2}$$

where X is the tool frame position and orientation defined as

$$X = \begin{bmatrix} p \\ \phi \end{bmatrix}, \tag{3.3}$$

and $\Gamma(\,\cdot\,)$ is a nonlinear function. θ_{kin} is a vector of fixed kinematic link parameters which, together with the joint positions, describe the relation between the base frame, frames attached to each link, and the tool frame. θ_{kin} consists of parameters[5] representing arm lengths and angles describing the rotation of each joint axis relative to the previous joint axis.

3.1.3 Inverse Kinematics

The inverse kinematics problem is, given the position and orientation of the tool frame, to compute the corresponding joint angles. The inverse kinematics problem is considerably harder than the forward kinematics problem[6], where a unique closed form solution always exists. Some features of the inverse kinematics problem are:

- A solution may not exist. The existence of a solution defines the workspace

[5] A well known parametrization of θ_{kin} are the so called Denavit-Hartenberg parameters.

[6] This is true for the serial link manipulator considered here. For a parallel arm robot, the situation is the opposite.

of a manipulator. The workspace is the volume which the end effector of the manipulator can reach[7].

- If a solution exists, it may not be unique. One example of multiple solutions is the elbow-up and elbow-down solutions for the elbow manipulator. There can even exist an infinite number of solutions, e.g., in the case of a kinematically redundant manipulator. In this case the number of link degrees-of-freedom (DOF)[8] is greater than the number of tool DOF, e.g., $N > 6$ for a tool which can be oriented and positioned arbitrarily. An example of a redundant system is the robot moved by a track motion in Figure 3.3, having totally 7 degrees-of-freedom. Methods for selecting one of many possible solutions are often needed, and each solution is then usually said to represent a particular robot configuration.

- The solution can be hard to obtain, even though it exists. Closed-form solutions are preferred, but for certain manipulator structures, only numerical iterative solutions are possible.

The inverse kinematics can be expressed as

$$q = \Gamma^{-1}(X, C, \theta_{kin}), \tag{3.4}$$

where C is some information used to select a feasible solution. One alternative is to let C be parameters describing the desired configuration. Another alternative is to let C be the previous solution, and to select the new solution as the, in some sense, closest solution.

3.1.4 Velocity Kinematics

The relation between the joint velocity $\dot{q}$ and the Cartesian velocity $\dot{X}$ is determined by the velocity Jacobian J of the forward kinematics relation

$$\dot{X} = \frac{\partial \Gamma(q)}{\partial q}\dot{q} = J(q)\dot{q}. \tag{3.5}$$

The relation between higher derivatives can be found by differentiation of the expression above, e.g., $\ddot{X} = J(q)\ddot{q} + \dot{J}(q)\dot{q}$. The velocity Jacobian, from now on called the Jacobian, is useful in many aspects of robotics. One example is the transformation of forces and torques, acting on the end effector, to the corresponding joint torques. This relation can be derived using the principle of virtual work, and is

$$\tau = J^T(q)F, \tag{3.6}$$

where $F \in \mathbb{R}^6$ is the vector of end effector forces and torques, and $\tau \in \mathbb{R}^N$ is a vector of joint torques.

[7] The volume which can be reached with arbitrary orientation is called dextrous workspace, and the volume that can be reached with at least one orientation is called reachable workspace.

[8] The number of independent coordinates necessary to specify the configuration of a certain system.

The Jacobian is also useful for studying singularities. A singularity is a configuration where the Jacobian looses rank. Some facts about singularities:

- A Cartesian movement close to a singularity results in high joint velocities. This can be seen from the relation $\dot{q} = J^{-1}(q)\dot{X}$.
- Most singularities occur on the workspace boundary but can also occur inside the workspace, e.g., when two or more joint axes are lined up.
- Close to a singularity there may be infinitely many solutions, or no solutions, to the inverse kinematics problem.
- The ability to move in a certain direction is reduced close to a singularity.

3.2 Dynamic Models

Dynamic models of the robot manipulator describe the relation between the motion of the robot, and the forces that cause the motion[9]. A dynamic model is useful for, e.g., simulation, control analysis, mechanical design, and real-time control. Some control algorithms require that the inverse dynamics problem is solved. This means that the required actuator torque is computed from the desired movement, including time derivatives. For simulation of the manipulator movement, the direct dynamic problem must be solved. This means that the dynamic model differential equations are solved with the actuator torques as input.

Depending on the intended use of the dynamic model, the manipulator can be modeled as rigid or elastic. A real flexible manipulator is a continuous nonlinear system, described by partial differential equations, PDEs, with infinite number of degrees-of-freedom. An infinite dimensional model is not realistic to use in real applications. Instead, finite dimensional models with the minimum number of parameters for the required accuracy level is preferred. The following three levels of elastic modeling are described in, e.g., Bascetta and Rocco (2002):

Finite Element Models These models are the most accurate models but normally not used for simulation and control due to their complexity. FEM models are widely used in the mechanical design of robot manipulators.

Assumed Modes Models These models are derived from the PDE formulation by modal truncation. Assumed modes models used for simulation and control design are frequently described in the literature.

Lumped Parameter Models The elasticity is modeled by discrete, localized springs. With this approach, a link can be divided into a number of rigid bodies connected by non-actuated joints. The gearbox elasticity can also be modeled with this approach.

[9] A somewhat different definition of dynamics is usually adopted in general multibody dynamics (Shabana, 1998), where kinetics deals with motion and the forces that produce it, and kinematics deals with the geometric aspects of motion regardless of the forces that cause it. Dynamics then includes both kinematics and kinetics.

In the following, both rigid and elastic dynamic models for robot manipulators will be described.

3.2.1 The Rigid Dynamic Model

There are several methods for obtaining a rigid dynamic model. The two most common approaches are the Lagrange formulation (Spong et al., 2006) and the Newton-Euler formulation (Craig, 1989). A third method is Kane's method (Kane and Levinson, 1983, 1985; Lesser, 2000). All methods are based on classical mechanics (Goldstein, 1980) and produce the same result even though the equations may differ in computational efficiency and structure. A detailed comparison of these methods is outside the scope of this work. In the following, only the Lagrange formulation will be described in some detail.

The Lagrangian method is based on describing scalar energy functions of the system, the kinetic energy $K(q,\dot{q})$ and the potential energy $V(q)$. These energy functions are expressed as functions of some suitable generalized coordinates, q, defined by

$$q = \begin{bmatrix} q_1 & q_2 & \dots & q_N \end{bmatrix}^T. \tag{3.7}$$

For the manipulator considered here, the generalized coordinates can be chosen as the joint angles. It can be shown that the kinetic energy can be expressed as

$$K(q,\dot{q}) = \frac{1}{2}\dot{q}^T M(q)\dot{q}, \tag{3.8}$$

where $M(\cdot)$ is the inertia matrix. The inertia matrix is positive definite and symmetric. More properties of Lagrangian dynamics are described in, e.g., Sciavicco and Siciliano (2000).

The next step is to compute the Lagrangian L as $L = K - V$. By applying the Lagrange equation

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{q}_j} - \frac{\partial L}{\partial q_j} = \tau_j, \tag{3.9}$$

the equations of motion, i.e., the dynamic model, can be derived. In the equation above, τ_j is called a generalized force, in our case the actuator torque.

In a rigid dynamic model, the links and gearboxes are assumed to be rigid. The mass and inertia of the actuators and gearboxes are added to the corresponding link parameters. The model consists of a serial kinematic chain of N links modeled as rigid bodies as illustrated in Figure 3.4. One rigid body rb^i is illustrated in Figure 3.5, and is described by its mass m^i, center of mass ξ^i, and inertia tensor with respect to center of mass J^i. Due to the symmetrical inertia tensor, only six components of J^i need to be defined. For simplicity, it is assumed that the structure of the serial manipulator, i.e., the orientations of the rotational joint axes, is given. The kinematics is then described by the length l^i. All parameters

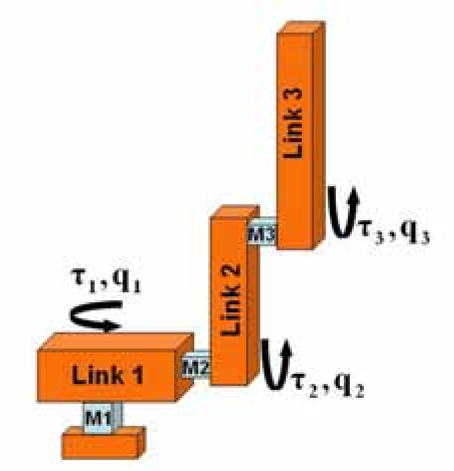

Figure 3.4: *A rigid dynamic model with 3 DOF.*

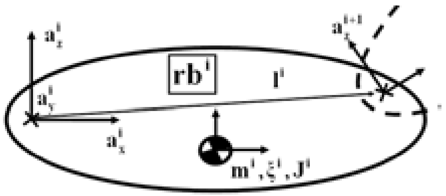

Figure 3.5: *A rigid body and its attributes.*

are described in a coordinate system a^i, fixed in rb^i, and are defined as follows

$$\xi^i = \begin{bmatrix} \xi_x^i & \xi_y^i & \xi_z^i \end{bmatrix}, \quad J^i = \begin{bmatrix} J_{xx}^i & J_{xy}^i & J_{xz}^i \\ J_{xy}^i & J_{yy}^i & J_{yz}^i \\ J_{xz}^i & J_{yz}^i & J_{zz}^i \end{bmatrix}, \quad l^i = \begin{bmatrix} l_x^i & l_y^i & l_z^i \end{bmatrix}.$$

The model can be derived by using, e.g., the Lagrange formulation which yields a system of second order ordinary differential equations or ODEs

$$M(q, \theta_{rb}, \theta_{kin})\ddot{q} + c(q, \dot{q}, \theta_{rb}, \theta_{kin}) + g(q, \theta_{rb}, \theta_{kin}) = \tau, \qquad (3.10)$$

where $\dot{x}$ denotes dx/dt and the time dependence is omitted in the expressions. $M(\,\cdot\,) \in \mathbb{R}^{N\times N}$ is the inertia matrix computed as $M(\,\cdot\,) = M_a(\,\cdot\,) + M_m$, where $M_a(\,\cdot\,)$ is the configuration dependent inertia matrix of the robot arm and, M_m is the inertia matrix of the rotating actuators expressed on the link side of the gearbox. The Coriolis, centrifugal, and gravity torques are described by $c(\,\cdot\,) \in \mathbb{R}^N$ and $g(\,\cdot\,) \in \mathbb{R}^N$, respectively. The vector of joint angles is denoted $q \in \mathbb{R}^N$, and the actuator torque vector is denoted $\tau \in \mathbb{R}^N$. Note that the equations are described on the link side, i.e., $\tau = \tau_m^a = \eta^T \tau_m^m$ and $M_m = M_m^a = \eta^T M_m^m \eta$ where η is the gear ratio matrix. The notation X_m^a should be interpreted[10] *quantity X for the motor expressed on the link side of the gearbox.* The rigid body and kinematic parameters described previously are gathered in θ_{rb} and θ_{kin} respectively, i.e., for each link i

$$\theta_{rb}^i = \begin{bmatrix} m_i & \xi_x^i & \xi_y^i & \xi_z^i & J_{xx}^i & J_{yy}^i & J_{zz}^i & J_{xy}^i & J_{xz}^i & J_{yz}^i \end{bmatrix}, \tag{3.11a}$$

$$\theta_{kin}^i = \begin{bmatrix} l_x^i & l_y^i & l_z^i \end{bmatrix}. \tag{3.11b}$$

For this model to be complete, the friction torque and the torque from gravity-compensating springs, if present, must be added to (3.10).

3.2.2 The Flexible Joint Dynamic Model

This model is an elastic, lumped parameter model. Consider the robot described in Section 3.2.1 with elastic gearboxes, i.e., elastic joints. This robot can be modeled by the so called flexible joint model which is illustrated in Figure 3.6. The rigid bodies are connected by torsional spring-damper pairs. If the inertial couplings between the motors and the rigid links are neglected we get the simplified flexible joint model[11]. If the gear ratio is high, this is a reasonable approximation as described in, e.g., Spong (1987). The motor mass and inertia are added to the corresponding rigid body. The total system has $2N$ DOF. The model equations of the simplified flexible joint model are

$$M_a(q_a)\ddot{q}_a + c(q_a, \dot{q}_a) + g(q_a) = \tau_a, \tag{3.12a}$$

$$\tau_a = K(q_m - q_a) + D(\dot{q}_m - \dot{q}_a), \tag{3.12b}$$

$$\tau_m - \tau_a = M_m\ddot{q}_m + f(\dot{q}_m), \tag{3.12c}$$

where joint and motor angular positions are denoted $q_a \in \mathbb{R}^N$ and $q_m \in \mathbb{R}^N$, respectively. τ_m is the motor torque and τ_a is the gearbox output torque. $K \in \mathbb{R}^{N\times N}$ is a stiffness matrix and $D \in \mathbb{R}^{N\times N}$ is the matrix of dampers. The dynamic and kinematic parameters are still described by θ_{rb} and θ_{kin} but are for simplicity omitted in the equations. A vector of friction torques is introduced for this model, and described by $f(\dot{q}_m) \in \mathbb{R}^N$. The friction torque is here approximated as acting on the motor side only. The convention of describing the equations on the link side is used, i.e., $q_m = q_m^a = \eta^{-1}q_m^m$ and $f(\cdot) = f_m^a(\cdot) = \eta^T f_m^m(\cdot)$

If the couplings between the links and the motors are included we get the complete flexible joint model (Tomei, 1991)

$$\tau_a = M_a(q_a)\ddot{q}_a + S(q_a)\ddot{q}_m + c_1(q_a, \dot{q}_a, \dot{q}_m) + g(q_a), \tag{3.13a}$$

$$\tau_m - \tau_a = M_m\ddot{q}_m + S^T(q_a)\ddot{q}_a + c_2(q_a, \dot{q}_a) + f(\dot{q}_m), \tag{3.13b}$$

$$\tau_a = K(q_m - q_a) + D(\dot{q}_m - \dot{q}_a), \tag{3.13c}$$

where $S \in \mathbb{R}^{N\times N}$ is a strictly upper triangular matrix of coupled inertia between links and motors. The structure of S depends on how the motors are positioned

[10] The a is explained by the fact that the link side can also be denoted the arm side and sometimes also the low-speed side. The motor side can also be denoted the high-speed side.

[11] Sometimes, the viscous damping is also neglected in the simplified model.

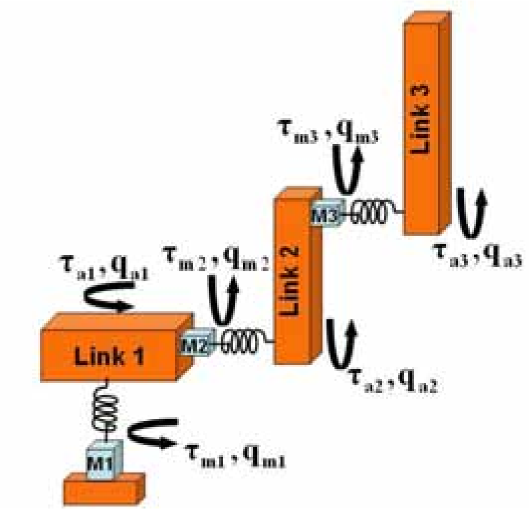

Figure 3.6: *A flexible joint dynamic model with 6 DOF.*

and oriented relative to the joint axis directions.

The flexible joint models can formally be derived in the same way as the rigid model, e.g., by the Lagrange equation. The potential energy of the springs must then be added to the potential energy expressions as

$$V_s(q_a, q_m) = \frac{1}{2}(q_a - q_m)^T K(q_a - q_m), \tag{3.14}$$

and the kinetic energy of the rotating actuators must be added as well.

3.2.3 Nonlinear Gear Transmissions

The nonlinear characteristics of the gear transmission can have a large impact on the behavior of a robot manipulator, for example at low speed when the friction parameters with nonlinear speed dependency are dominant. A classical friction model[12] includes the viscous and Coulomb friction, f_v and f_c respectively, and is given by

$$f(v) = f_v v + f_c \operatorname{sign}(v), \tag{3.15}$$

where $v = \dot{q}_m$. More advanced friction models are described in Armstrong-Hélouvry (1991). One model taking many phenomenon into account is the so called *LuGre Model*, described in Canudas de Wit et al. (1995) and Olsson (1996). This model is a nonlinear differential equation modeling both static and dynamic behavior

[12] All friction models described here are scalar models that can be used for each gear transmission.

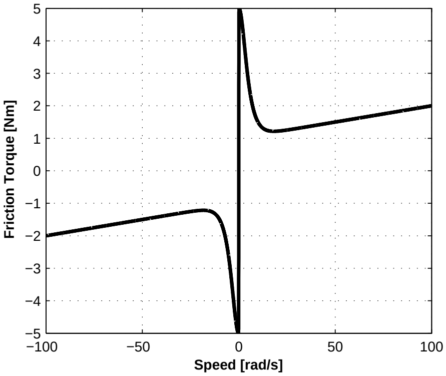

Figure 3.7: *Typical friction characteristics of a compact gearbox as described by* (3.17).

and is described by

$$\frac{dz}{dt} = v - \frac{|v|}{g(v)} z, \tag{3.16a}$$

$$\sigma_0 g(v) = f_c + (f_s - f_c) e^{-(v/v_s)^2}, \tag{3.16b}$$

$$\sigma_1(v) = \sigma_1 e^{-(v/v_d)^2}, \tag{3.16c}$$

$$f(v) = \sigma_0 z + \sigma_1 \frac{dz}{dt} + f_v v, \tag{3.16d}$$

which in its simplified form has $\sigma_1(v) = \sigma_1$. The Stribeck friction is modeled by f_s. Another dynamic friction model is the *Generalized Maxwell-Slip Model*, described in Al-Bender et al. (2005). A smooth static friction law is suggested in Feeny and Moon (1994). This model avoids discontinuities to simplify numerical integration and is given by

$$f(v) = f_v v + f_c(\mu_k + (1 - \mu_k) \cosh^{-1}(\beta v)) \tanh(\alpha v), \tag{3.17}$$

and is illustrated in Figure 3.7. Figure 3.8 shows the friction measured on one axis of an industrial robot under steady-state conditions, i.e., constant speed, in one configuration. In the same figure, the steady-state LuGre model and the Feeny-Moon model, fitted to the experimental data, are shown. Both models describe the static behavior in a good way. However, the friction of a real robot shows large variations for different configurations in the workspace, for different tool loads, and for different temperatures (Carvalho Bittencourt et al., 2010). This means that some kind of off-line or on-line adaption of the model is necessary. An open question is whether the existing friction models can capture the dynamic effects of the compact gear boxes, together with motor bearings, typically used in industrial robots of today.

Another important nonlinear gearbox characteristic is the nonlinear stiffness. The nonlinear stiffness can also be included in the flexible joint model by replacing

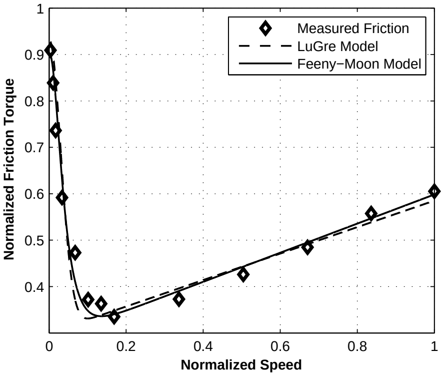

Figure 3.8: *One example of the friction characteristics for one axis of an industrial robot.*

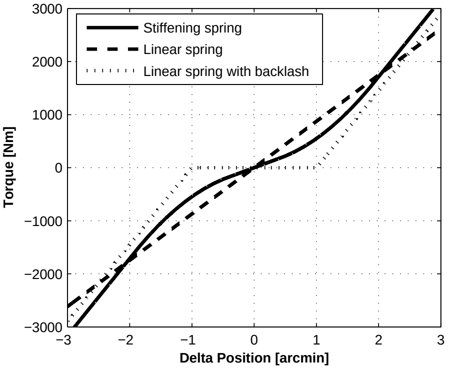

Figure 3.9: *Three typical gearbox stiffness characteristics.*

$K(q_m - q_a)$ in (3.12) by

$$\tau_s = \tau_s(q_m - q_a), \tag{3.18}$$

where $\tau_s(\cdot)$ is a nonlinear function describing a nonlinear spring. Three typical spring characteristics are shown in Figure 3.9. The first has a smooth nonlinear characteristics, often called a stiffening spring. This is typical for the compact gearboxes used by modern industrial robots. The second is an ideal linear spring and the third has a backlash behavior. Industrial gearboxes are designed for low backlash, i.e., a backlash that can be accepted with respect to the accuracy required. The backlash is, of course, not zero. More nonlinearities could be added to the model of the gear transmission. Modeling of hysteresis, backlash, and nonlinear stiffness/damping is, for example, described in Tuttle and Seering (1996), Dhaouadi et al. (2003), and Ruderman et al. (2009). A smooth nonlinear stiffness is used in the benchmark problems, described in Papers F and G.

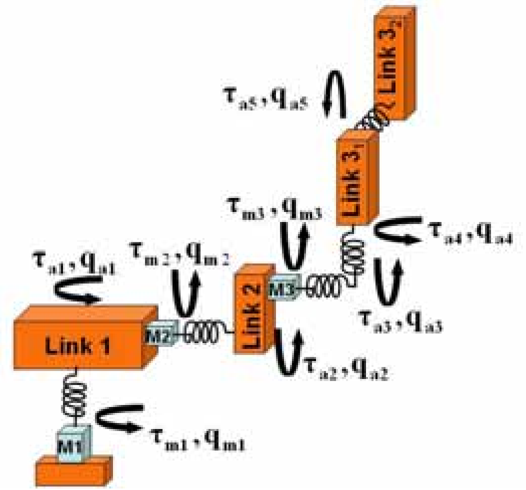

Figure 3.10: *An extended flexible joint dynamic model with 8 degrees-of-freedom.*

3.2.4 The Extended Flexible Joint Dynamic Model

Most publications concerning flexible robot manipulators only consider elastic deformation in the rotational direction. If only the gear elasticity is considered we get the flexible joint model, and if link deformation restricted to a plane perpendicular to the preceding joint is included in the model we get the flexible link model. One example of a flexible link model is described in De Luca (2000). In Paper A it is shown that the flexible joint model must be extended in order to describe a modern industrial robot in a realistic way. The added elasticity is the elasticity of bearings, foundation, tool, and load as well as bending and torsion of the links.

The extension of the flexible joint model is straightforward. First, replace the one-dimensional spring-damper pairs in the actuated joints with multidimensional spring-damper pairs. Second, if necessary, divide each link into two or more rigid bodies at proper locations, and connect the rigid bodies with multi-dimensional spring-damper pairs. In this way, non-actuated joints or pseudo-joints are added to the model. The principle is illustrated in Figure 3.10 where one extra torsional spring is added in the actuated third joint to model torsion of link three. Link three is then divided in two rigid bodies, and one more torsional spring is added to allow bending out of the plane of rotation. In this way two non-actuated joints are created. The model thus has 8 DOF, i.e., three motor DOF q_{m1}–q_{m3}, three actuated joint DOF q_{a1}–q_{a3}, and two non-actuated joint DOF q_{a4}–q_{a5}. The number of non-actuated joints and their locations are, of course, not obvious. This model structure selection is therefore a crucial part of the modeling and identification procedure.

Generally, if the number of added non-actuated joints is Υ_{na} and the number of actuated joints is Υ_a, the system has $2\Upsilon_a + \Upsilon_{na}$ degrees-of-freedom. The model equations can then be described as

$$M_a(q_a)\ddot{q}_a + c(q_a, \dot{q}_a) + g(q_a) = \tau_a, \tag{3.19a}$$

$$\tau_a = \begin{bmatrix} \tau_g \\ \tau_e \end{bmatrix}, \tag{3.19b}$$

$$q_a = \begin{bmatrix} q_g \\ q_e \end{bmatrix}, \tag{3.19c}$$

$$\tau_g = K_g(q_m - q_g) + D_g(\dot{q}_m - \dot{q}_g), \tag{3.19d}$$

$$\tau_e = -K_e q_e - D_e \dot{q}_e, \tag{3.19e}$$

$$\tau_m - \tau_g = M_m \ddot{q}_m + f(\dot{q}_m), \tag{3.19f}$$

where $q_g \in \mathbb{R}^{\Upsilon_a}$ is the actuated joint angular position, $q_e \in \mathbb{R}^{\Upsilon_{na}}$ is the non-actuated joint angular position, and $q_m \in \mathbb{R}^{\Upsilon_a}$ is the motor angular position. $M_m \in \mathbb{R}^{\Upsilon_a \times \Upsilon_a}$ is the inertia matrix of the motors and $M_a(q_a) \in \mathbb{R}^{(\Upsilon_a+\Upsilon_{na})\times(\Upsilon_a+\Upsilon_{na})}$ is the inertia matrix for the joints. The Coriolis and centrifugal torques are described by $c(q_a, \dot{q}_a) \in \mathbb{R}^{\Upsilon_a+\Upsilon_{na}}$, and $g(q_a) \in \mathbb{R}^{\Upsilon_a+\Upsilon_{na}}$ is the gravity torque. $\tau_m \in \mathbb{R}^{\Upsilon_a}$ is the actuator torque, $\tau_g \in \mathbb{R}^{\Upsilon_a}$ is the actuated joint torque, and $\tau_e \in \mathbb{R}^{\Upsilon_{na}}$ is the non-actuated joint torque, i.e., the constraint torque. K_g, K_e, D_g, and D_e are the stiffness- and damping matrices for the actuated and non-actuated directions, with obvious dimensions. This model is from now on called the *extended flexible joint model*. The nonlinearities of the gear transmission described previously can also be added to this model. The non-actuated joint stiffness can also be modeled as nonlinear if required.

For a complete model including the position and orientation of the tool, X, the forward kinematic model of the robot must be added. The kinematic model is a mapping of $q_a \in \mathbb{R}^{\Upsilon_a+\Upsilon_{na}}$ to $X \in \mathbb{R}^{\Upsilon_a}$. The complete model of the robot is then described by (3.19) and

$$X = \Gamma(q_a). \tag{3.20}$$

Note that no inverse kinematics exists. This is a fact that makes the control problem considerably harder.

The identification of the extended flexible joint model is the subject of Paper A and B, and the inverse dynamics problem when using this model is treated in Paper C and D.

3.2.5 Flexible Link Models

If the elasticity of the links cannot be neglected nor described by the joint elasticity approaches described in Sections 3.2.2 and 3.2.4, a distributed elasticity model can be used to increase the accuracy of the model. These models are described by partial differential equations with infinite dimension. One way to reduce the model to a finite-dimensional model is to use the assumed modes method. The link deflections are then described as an infinite series of separable

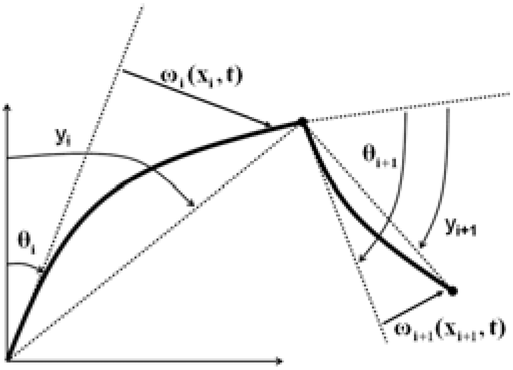

Figure 3.11: *The flexible link model.*

modes that is truncated to a finite number of modes.

A model of this type is described in De Luca et al. (1998) and illustrated in Figure 3.11. The model consists of a kinematic chain of N flexible links directly actuated by N motors. It is assumed that the link deformations are small and that each link only can bend in one direction, i.e., in a plane perpendicular to the previous joint axis. The link deflection $w_i(x_i, t)$ at a point x_i along link i of length l_i is described by

$$w_i(x_i, t) = \sum_{j=1}^{N_{ei}} \phi_{ij}(x_i)\delta_{ij}(t), \quad i = 1 \dots N, \tag{3.21}$$

where link i has N_{ei} assumed modes $\phi_{ij}(x_i)$, and $\delta_{ij}(t)$ are the generalized coordinates. By use of Lagrange equations the dynamic model for the system can be expressed as

$$\begin{bmatrix} B_{\theta\theta}(\theta) & B_{\theta\delta}(\theta) \\ B_{\theta\delta}^T(\theta) & B_{\delta\delta} \end{bmatrix} \begin{bmatrix} \ddot{\theta} \\ \ddot{\delta} \end{bmatrix} + \begin{bmatrix} c_\theta(\theta, \dot{\theta}, \dot{\delta}) \\ c_\delta(\theta, \dot{\theta}) \end{bmatrix} + \begin{bmatrix} 0 \\ D\dot{\delta} + K\delta \end{bmatrix} = \begin{bmatrix} \tau \\ 0 \end{bmatrix}, \tag{3.22}$$

where $\theta \in \mathbb{R}^N$ is a vector of joint angles, $\delta \in \mathbb{R}^{N_e}$ is a vector of link deformations, and $N_e = \sum_{i=1}^N N_{ei}$. B is a partitioned inertia matrix, c a vector of Coriolis and centripetal forces, and K and D are stiffness and damping matrices, respectively, all with obvious dimensions. The actuator torque is denoted τ. The reference frame where the deflection is described is clamped to the base of each link. The end effector position can be approximated by the angles y_i as illustrated in Figure 3.11 and described by

$$y_i = \theta_i + \Phi_{ei}\delta_i, \tag{3.23}$$

where

$$\Phi_{ei} = \left[\phi_{i,1}(l_i)/l_i, \dots, \phi_{i,N_{ei}}(l_i)/l_i\right], \tag{3.24}$$

and

$$\delta_i = \begin{bmatrix} \delta_{i,1} \\ \vdots \\ \delta_{i,N_{ei}} \end{bmatrix}. \tag{3.25}$$

Note that y can be regarded as output variable of this system, and that the same direct and inverse kinematic models as for the rigid or flexible joint can be used. This fact simplifies the inverse dynamics control problem. A fact that makes the inverse dynamics problem hard, however, is that the system from the actuator torque τ to the controlled variable y can have unstable zero dynamics[13], i.e., the system has non-minimum phase behavior (Isidori, 1995; Slotine and Li, 1991) which means that trajectory tracking is considerably harder to achieve.

The structure of the equations of motion and the assumed modes depend on the boundary conditions used, which is a critical choice for these models. Assumed mode models are further described in, e.g., Hastings and Book (1987), De Luca and Siciliano (1991), Book (1993), Book and Obergfell (2000), and Bascetta and Rocco (2002).

3.3 The Kinematics and Dynamics of a Two-Link Elbow Manipulator

In this section, the kinematic and dynamic models of a rigid two link elbow manipulator will be derived as a small but illustrative example of how the models can be derived. The manipulator is planar and constrained to movements in the x-y plane. By inspection of Figure 3.12, the forward kinematics is

$$\begin{aligned} X = \Gamma(q) = \begin{bmatrix} p_x \\ p_y \end{bmatrix} &= \begin{bmatrix} l_1 \sin(q_1) + l_2 \sin(\frac{\pi}{2} + q_1 + q_2) \\ l_1 \cos(q_1) + l_2 \cos(\frac{\pi}{2} + q_1 + q_2) \end{bmatrix} \\ &= \begin{bmatrix} l_1 \sin(q_1) + l_2 \cos(q_1 + q_2) \\ l_1 \cos(q_1) - l_2 \sin(q_1 + q_2) \end{bmatrix}. \end{aligned} \tag{3.26}$$

The inverse kinematics can here be derived in closed form by either algebraic or geometric methods. Using the geometric approach and the law of cosine

$$\cos(\gamma) = \frac{l_1^2 + l_2^2 - p_x^2 - p_y^2}{2 l_1 l_2} = \sin(q_2) \triangleq s_2, \tag{3.27}$$

and finally obtain the expression for q_2 by use of the atan2 function

$$q_2 = \operatorname{atan2}(s_2, \pm\sqrt{1 - s_2^2}), \tag{3.28}$$

[13] For uniform mass distributions, the system is always minimum-phase.

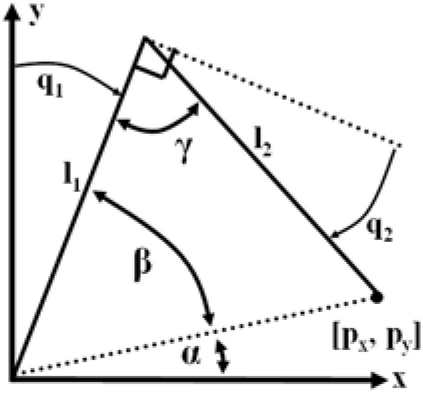

Figure 3.12: *Two link elbow manipulator kinematics.*

where the function atan2 is preferred for numerical reasons. Continuing with the solution for q_1, in the same way

$$\cos(\beta) = \frac{l_1^2 + p_x^2 + p_y^2 - l_2^2}{2l_1\sqrt{p_x^2 + p_y^2}} = c_\beta, \tag{3.29a}$$

$$\beta = \text{atan2}(\pm\sqrt{1 - c_\beta^2}, c_\beta), \tag{3.29b}$$

$$\alpha = \text{atan2}(p_y, p_x), \tag{3.29c}$$

$$q_1 = \frac{\pi}{2} - \alpha - \beta. \tag{3.29d}$$

The inverse kinematics is given by (3.29d) and (3.28). The alternative signs in (3.28) and (3.29b) should be chosen as both plus or both minus corresponding to the two solutions, elbow-up and elbow-down. The Jacobian of the velocity kinematics is obtained by differentiation of the forward kinematics (3.26), i.e.,

$$J(q) = \frac{\partial\Gamma(q)}{\partial q} = \begin{bmatrix} l_1c_1 - l_2s_{12} & -l_2s_{12} \\ -l_1s_1 - l_2c_{12} & -l_2c_{12} \end{bmatrix}, \tag{3.30}$$

where the notations $\sin(q_1) \triangleq s_1$, $\cos(q_1) \triangleq c_1$, and $\cos(q_1 + q_2) \triangleq c_{12}$ etc. are used.

The dynamics is computed based on the simplified model in Figure 3.13. The dynamic and kinematic parameters are defined according to Section 3.2.1. The actuator inertial parameters are here included in the link parameters. The first step is to derive the kinematics for each link center of mass, i.e.,

$$X_1 = \Gamma_1(q) = \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} \xi_1 s_1 \\ \xi_1 c_1 \end{bmatrix}, \tag{3.31a}$$

$$X_2 = \Gamma_2(q) = \begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = \begin{bmatrix} l_1s_1 + \xi_2c_{12} \\ l_1c_1 - \xi_2s_{12} \end{bmatrix}, \tag{3.31b}$$

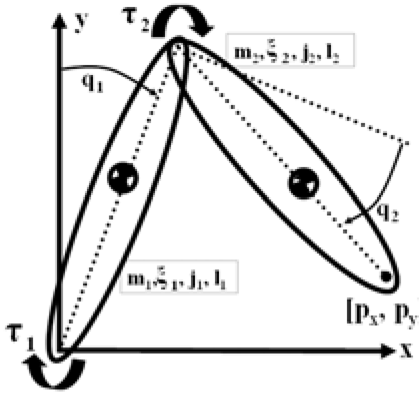

Figure 3.13: *Simplified two link elbow manipulator dynamic model.*

and the corresponding Jacobians

$$J_1(q) = \begin{bmatrix} \xi_1 c_1 & 0 \\ -\xi_1 s_1 & 0 \end{bmatrix}, \tag{3.32a}$$

$$J_2(q) = \begin{bmatrix} l_1 c_1 - \xi_2 s_{12} & -\xi_2 s_{12} \\ -l_1 s_1 - \xi_2 c_{12} & -\xi_2 c_{12} \end{bmatrix}. \tag{3.32b}$$

The rotational kinetic energy in this simple case is

$$K_{rot} = \frac{1}{2} j_1 \dot{q}_1^2 + \frac{1}{2} j_2 (\dot{q}_1 + \dot{q}_2)^2, \tag{3.33}$$

and the translational kinetic energy is generally

$$\begin{aligned} K_{trans} &= \frac{1}{2} \sum_{i=1}^{N} (m_i \dot{X}_i^T \dot{X}_i) \\ &= \frac{1}{2} \sum_{i=1}^{N} (m_i (J_i(q)\dot{q})^T J_i(q)\dot{q}) = \frac{1}{2} \dot{q}^T (\sum_{i=1}^{N} m_i J_i^T(q) J_i(q)) \dot{q}. \end{aligned} \tag{3.34}$$

Thus, the total kinetic energy is then

$$K = K_{trans} + K_{rot}. \tag{3.35}$$

The potential energy is given from the y-coordinate of each mass as

$$V = \sum_{i=1}^{N} (m_i g y_i), \tag{3.36}$$

where g is the gravitational constant. The Lagrange function is then

$$L = K - V. \tag{3.37}$$

Applying the Lagrange equation (3.9) and using some symbolic mathematical software, e.g., *Matlab Symbolic Toolbox*, gives the equations of motion as

$$\tau = M(q)\ddot{q} + C(q,\dot{q}) + G(q), \tag{3.38}$$

where

$$M(q) = \begin{bmatrix} J_{11}(q) & J_{12}(q) \\ J_{21}(q) & J_{22}(q) \end{bmatrix}, \tag{3.39a}$$

$$J_{11}(q) = j_1 + m_1\xi_1^2 + j_2 + m_2(l_1^2 + \xi_2^2 - 2l_1\xi_2 s_2), \tag{3.39b}$$

$$J_{12}(q) = J_{21}(q) = j_2 + m_2(\xi_2^2 - l_1\xi_2 s_2), \tag{3.39c}$$

$$J_{22}(q) = j_2 + m_2\xi_2^2, \tag{3.39d}$$

$$C(q,\dot{q}) = \begin{bmatrix} -m_2 l_1\xi_2 c_2(2\dot{q}_1\dot{q}_2 + \dot{q}_2^2) \\ m_2 l_1 \xi_2 c_2 \dot{q}_1^2 \end{bmatrix}, \tag{3.39e}$$

$$G(q) = \begin{bmatrix} -g(m_1\xi_1 s_1 + m_2(l_1 s_1 + \xi_2 c_{12})) \\ -m_2\xi_2 g c_{12} \end{bmatrix}. \tag{3.39f}$$

Note that the kinetic energy has the form

$$K = \frac{1}{2}\dot{q}^T M(q)\dot{q}, \tag{3.40}$$

as described in Section 3.2.1. If the rotational kinetic energy is included in (3.34), the inertia matrix of the system, $M(q)$, can be derived without applying the Lagrange equation. Deriving dynamic models is clearly a matter of formulating the kinematics, and then use a powerful tool for symbolic mathematics. Of course, the models can be much more complex than shown in this small example. Describing the problems and solutions for those cases is outside the scope of this work.

4

Identification of Robot Manipulators

This chapter gives a short introduction to system identification in general, and to the identification of robot manipulators in particular.

4.1 System Identification

System identification is the art of estimating a model of a dynamic system from measured data. For a thorough treatment of system identification, the reader is referred to Ljung (1999), Söderström and Stoica (1989), or Johansson (1993). An in-depth treatment of frequency-domain identification can be found in Pintelon and Schoukens (2001). In the area of automatic control, the estimated models are often used for controller design, on-line control, simulation, and prediction.

4.1.1 Introduction

The identification experiment can be performed in *open loop* or *closed loop*. Identification of a system not subject to feedback control, i.e, an open-loop system, is illustrated in Figure 4.1. This system has input u, output y, and is affected by a disturbance v. The disturbance can include measurement noise as well as external system inputs, not included in u. An identification experiment on a system subject to feedback control, i.e., a closed-loop system, is shown in Figure 4.2 where r is the reference signal for the system. A reason for performing a closed-loop experiment could be that the system is unstable, and must be controlled in order to remain stable. This is typically the case for a robot manipulator. Other reasons could be that safety or production restrictions do not allow open loop experiments. If we take a typical industrial robot manipulator as an example system, u is the motor torque, y is the motor position, and r is the position refer-

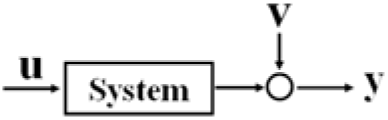

Figure 4.1: An open-loop system.

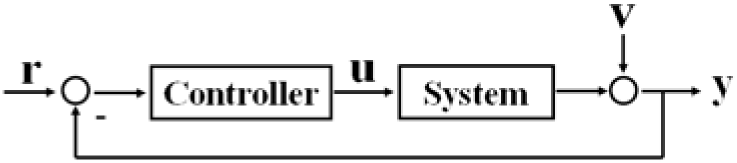

Figure 4.2: A closed-loop system.

ence. The disturbance v includes both measurement noise and internally generated disturbances, e.g., torque ripple generated by the motor[1].

Models can be divided into *nonlinear*- and *linear models*. A real world system is in general nonlinear. However, linear time-invariant approximations are often used for modeling the nonlinear reality.

Moreover, models can be described as *continuous-time models* or *discrete-time models* although the measurements, $u(t)$ and $y(t)$, are normally represented as sampled, discrete-time, data. It is assumed that the reader has a basic knowledge of linear system theory for continuous-time and discrete-time systems. Some books treating this subject in-depth are Kailath (1980) and Rugh (1996). Other recommended sources are Åström and Wittenmark (1996) and Ljung and Glad (2001).

The different types of models can be further divided into *nonparametric models* and *parametric models*. A nonparametric model is a vector of numbers or a graphical curve describing the model in the time-domain or frequency-domain. A parametric model is a model where the information obtained by the measurements has been condensed to a small number of parameters. The model can be described as a differential or difference equation or, in the case of a linear model, as a transfer function. In the next sections, these two model types are described.

[1] The torque ripple is not entering the system at the output which means that it must be filtered by some appropriate system dynamics before added to v.

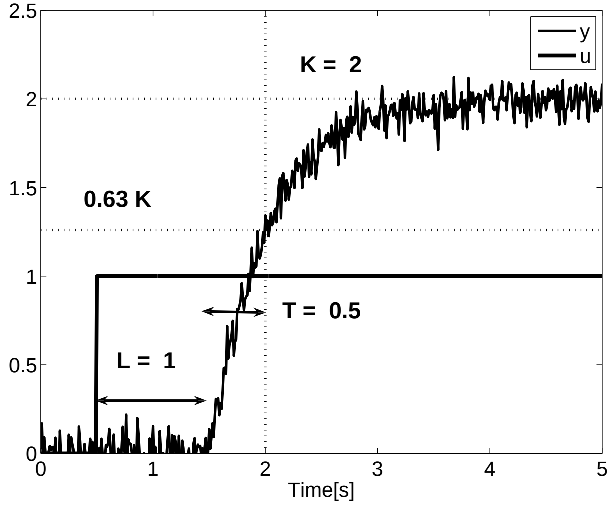

Figure 4.3: Step response of a first order process with delay.

4.1.2 Nonparametric Models

Examples of nonparametric models in the *time-domain* are impulse responses or step responses. Such models consist of vectors of system outputs and the corresponding time stamps. An example of a step response of a first-order system with a time-delay is shown in Figure 4.3. The measured output is affected by measurement noise. The nonparametric step response model can in this case be described by a parametric transfer function model

$$G(s) = \frac{K}{sT+1}e^{-Ls}. \tag{4.1}$$

This three-parameter model[2] is often used to describe systems in the process industry. The parametric model (4.1) can be identified by inspection of the step-response according to Figure 4.3. This model can then be used for tuning of a PI- or a PID controller using, e.g., lambda tuning. Identification and control of industrial processes are treated in Åström and Hägglund (2006). The described methodology of obtaining a parametric model from a nonparametric model is important, and the method considered in Papers A and B includes such a methodology, although based on a nonparametric frequency-domain model instead of a nonparametric time-domain model.

Methods for obtaining nonparametric *frequency-domain* models are, e.g., Fourier analysis, and spectral analysis. The obtained model is a frequency response function (FRF) consisting of a vector of complex numbers and the corresponding frequency vector. The complex numbers represent the transfer function from input to output at different frequencies. The FRF can be illustrated in a Bode diagram,

[2] Sometimes called the KLT model.

a Nichols diagram, or a Nyquist diagram, and can be used directly for controller design with frequency domain methods, such as loop-shaping using lead-lag compensation or QFT-design (Horowitz, 1991). The FRF can also be used for obtaining a parametric model as described in Papers A and B.

The method used in Papers A and B, for obtaining the FRF, is based on Fourier analysis and the discrete-time Fourier Transform. This transform, if the time-domain signal is $x(t)$, is defined by

$$X(e^{j\omega_k T_s}) = \sum_{n=-\infty}^{\infty} x(nT_s)e^{-j\omega_k nT_s}, \tag{4.2}$$

where T_s is the sample time and ω_k the frequency considered. As an approximation of this transform, when the measurement is a finite sequence of discrete-time data, sampled for $t = nT_s,\ n = 1,\ldots,N$, the discrete Fourier transform (DFT) is usually adopted. The DFT[3] is usually defined as

$$X_N(e^{j\omega_k T_s}) = \frac{1}{\sqrt{N}} \sum_{n=1}^{N} x(nT_s)e^{-j\omega_k nT_s}, \tag{4.3}$$

where

$$\omega_k = \frac{2\pi k}{NT_s},\ k = 1,\ldots,N. \tag{4.4}$$

For a thorough treatment of the discrete Fourier transforms, see Oppenheim and Schafer (1975).

The DFT may contain errors, known as leakage errors, due to the fact that it is computed for a data sequence with finite duration. This error can be reduced by applying windowing functions before computing the DFT. The leakage error can be eliminated if the signals are of finite duration, and if the signals are sampled until the system is at rest. This type of excitation is called burst excitation. Another way of eliminating the leakage error is to use a periodic excitation, and to sample the signals for an integer number of periods, when a steady state is reached. For further discussions on these issues see Pintelon and Schoukens (2001).

The FRF[4] for a SISO system with the transfer function G, at frequency ω_k, can be estimated as

$$\hat{G}_N(\omega_k) = \frac{Y_N(\omega_k)}{U_N(\omega_k)}, \tag{4.5}$$

where $Y_N(\cdot)$ and $U_N(\cdot)$ are the DFTs of $y(\cdot)$ and $u(\cdot)$, respectively. In the following, we assume that no leakage errors are present because burst or periodic excitation has been used in the experiments. We further assume that a proper anti-alias[5] filtering is performed so that no alias errors are present. For nota-

[3] The fast Fourier transform (FFT) is an efficient way of computing the DFT.

[4] This estimated FRF is sometimes called the empirical transfer function estimate (ETFE).

[5] Alias or frequency folding occurs when sampling a continuous-time signal with a frequency con-

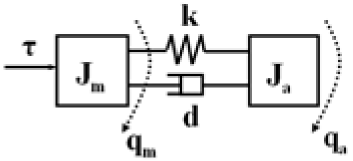

Figure 4.4: *A linear two-mass flexible joint model.*

tional simplicity, the arguments of $\hat{G}_N(\cdot)$, $Y_N(\cdot)$ and $U_N(\cdot)$ are written as ω_k, although $e^{j\omega_k T_s}$ would be more correct.

For a multiple-input multiple-output (MIMO) system described by the $n \times n$ transfer function matrix G with inputs u and outputs y, the following relation between the FRF of G, and the Fourier transforms of the input and output, U and Y, for frequency ω_k, holds

$$Y(\omega_k) = G(\omega_k)U(\omega_k). \tag{4.6}$$

If n independent experiments of length N are performed, the multivariable FRF can be estimated as

$$\hat{G}_N(\omega_k) = \bar{Y}_N(\omega_k)\bar{U}_N^{-1}(\omega_k), \tag{4.7}$$

where $\bar{U}_N(\omega_k)$ and $\bar{Y}_N(\omega_k)$ have n columns from the n experiments. If more than n experiments are performed, other FRF estimators can be applied, see, e.g., Pintelon and Schoukens (2001) or Wernholt and Moberg (2008b).

4.1.3 A Robot Example

A linear one-axis flexible joint model (see Section 3.2.2) will now be used as an example of how to obtain an FRF by closed-loop identification. The model, illustrated in Figure 4.4, can be described by the differential equations

$$0 = J_a\ddot{q}_a + k(q_a - q_m) + d(\dot{q}_a - \dot{q}_m), \tag{4.8a}$$

$$\tau = J_m\ddot{q}_m - k(q_a - q_m) - d(\dot{q}_a - \dot{q}_m), \tag{4.8b}$$

or by the transfer function

$$G(s) = \frac{s^2/\omega_z^2 + 2\zeta_z s/\omega_z + 1}{s^2(J_a + J_m)(s^2/\omega_p^2 + 2\zeta_p s/\omega_p + 1)}, \tag{4.9}$$

where

$$\omega_z = \sqrt{\frac{k}{J_a}},\ \omega_p = \sqrt{\frac{k(J_a + J_m)}{J_aJ_m}},\ \zeta_z = \frac{d}{2}\sqrt{\frac{1}{kJ_a}},\ \zeta_p = \frac{d}{2}\sqrt{\frac{J_a + J_m}{kJ_aJ_m}}. \tag{4.10}$$

The input u is the motor torque τ and the output y is the motor position q_m. J_a and J_m are the inertias of the arm and motor respectively, k is the joint stiffness,

tent above half the sample frequency. The alias effect is further described in Åström and Wittenmark (1996).

d is the joint viscous damping, and finally, q_a is the arm position. The measurement y is affected by measurement noise. The system is controlled by a speed controller of P-type with sample time $T_s = 1\,\text{ms}$. The motor speed is obtained by differentiation of the measured position, and the excitation is the the motor speed reference r. Two different excitations are evaluated. The first is a burst excitation in the form of a swept sinusoid (chirp)

$$r(t) = A \sin\Big(2\pi f_1(t - t_1) + \frac{\pi(f_2 - f_1)}{t_2 - t_1}(t - t_1)^2\Big), \tag{4.11}$$

with amplitude A, starting at $t_1 = 0.5\,\text{s}$ with frequency $f_1 = 50\,\text{Hz}$, and ending at $t_2 = 20\,\text{s}$ with frequency $f_2 = 0.5\,\text{Hz}$. The second excitation is a periodic multisine signal, i.e., a sum of sinusoids, computed as

$$r(t) = \sum_{k=1}^{N_f} \tilde{A} \cos(2\pi f_k t + \phi_k), \tag{4.12}$$

where N_f is the number of sinusoids, with frequencies f_k and random phases ϕ_k uniformly distributed between 0 and 2π. The frequency f_k is chosen from the grid $\{f_k = k/(T_s N_p)\ k = 1, \ldots, N_p/2\}$, where N_p is the number of samples in one period and T_s is the sampling time. The amplitude $\tilde{A}$ is computed, after the randomization of the phases, to fulfil input amplitude constraints. For optimizing the input power, given an amplitude constraint, the phases can be randomized repeatedly (or optimized). This means that the *crest factor* of the input is minimized (Ljung, 1999). The period time of the multisine is chosen to 20 s. The motor speed and motor torque are measured from $t = 0\,\text{s}$ to $t = 25\,\text{s}$. For the chirp excitation, the whole sequence is used when computing the FRF, in order to reduce the leakage errors (allowing all transients to decay during the last 5 s). For the multisine excitation, leakage is avoided by using only the last 20 s (allowing the initial transients to decay during the first 5 s). The motor torque, motor speed, and the excitation signals are shown in Figures 4.5–4.6. Clearly, the noise level is very high compared to the excitation.

Figures 4.7–4.8 show the magnitude of the true system FRF and the estimated FRFs (for chirp and multisine excitation), obtained by applying (4.5). The FRF is computed from the torque input to the differentiated output (motor speed). The errors in the estimated FRF, when using burst excitation, are quite high for frequencies above 20 Hz. These errors can be reduced by averaging over neighboring frequencies using a suitable window (smoothing). The errors in the FRF, when using periodic excitation, are much smaller. The reason is that the periodic excitation can be concentrated at a few selected frequencies, and thereby decreases the bias and the variance of the estimated FRF. This is further discussed in Paper B.

Figure 4.9 illustrates one potential problem due to closed loop identification. Here, the amplitude of the excitation signals is reduced by a factor of 100, and thus decreasing the signal-to-noise ratio. The errors in the estimated FRFs increase, and the estimated FRFs approach a constant gain, 0 dB. This is the inverse controller gain ($K = 1$) as expected (see Söderström and Stoica, 1989). Note that

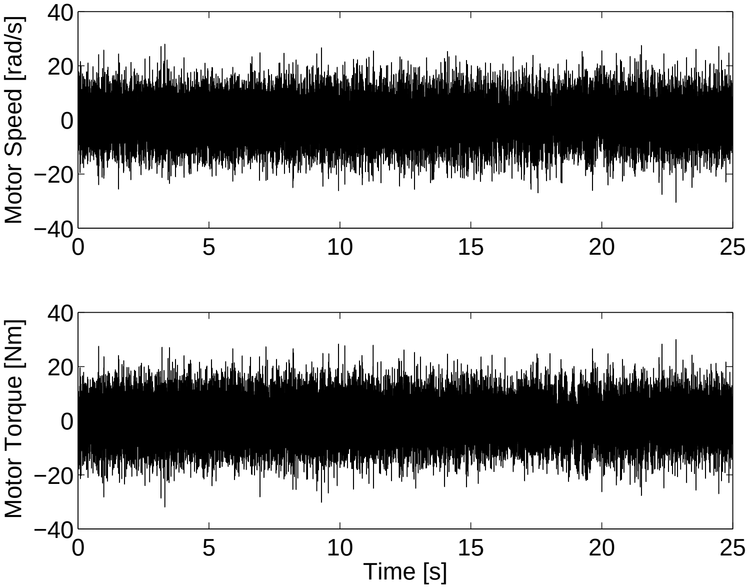

Figure 4.5: *Motor speed and torque during the identification experiment (chirp input).*

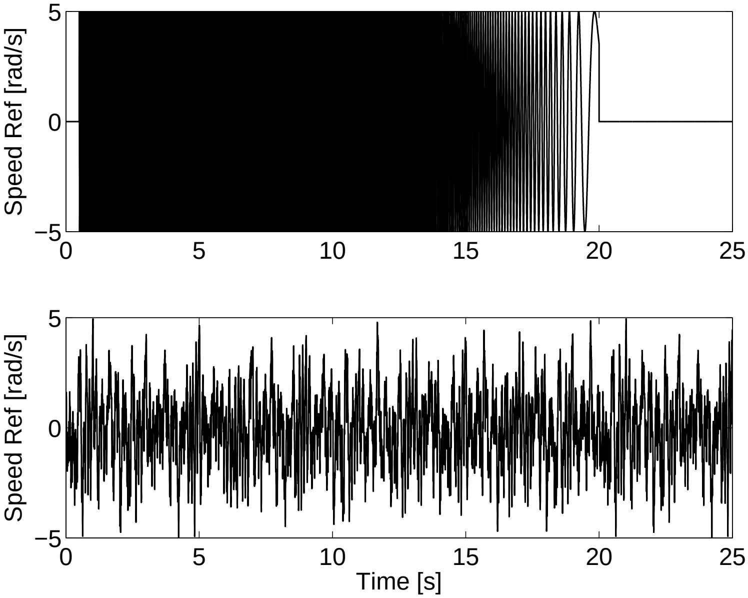

Figure 4.6: *Chirp excitation (upper) and multisine excitation (lower).*

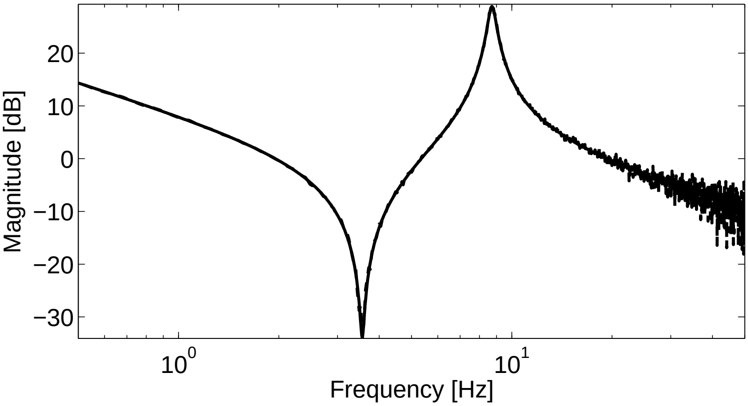

Figure 4.7: *Magnitude of true FRF (solid) and estimated FRF (dashed) using chirp excitation.*

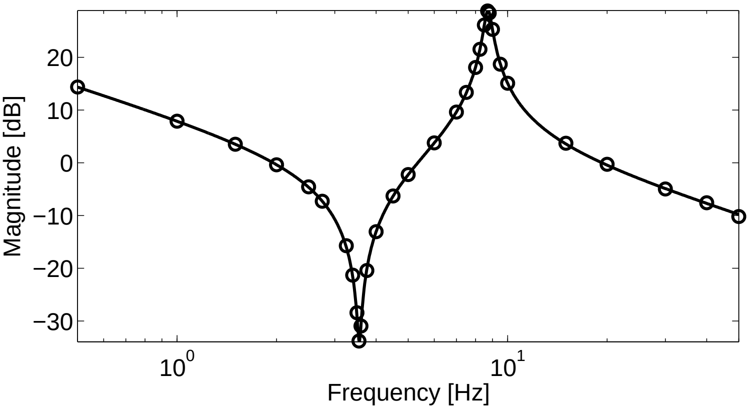

Figure 4.8: *Magnitude of true FRF (solid) and estimated FRF (circles) using multisine excitation.*

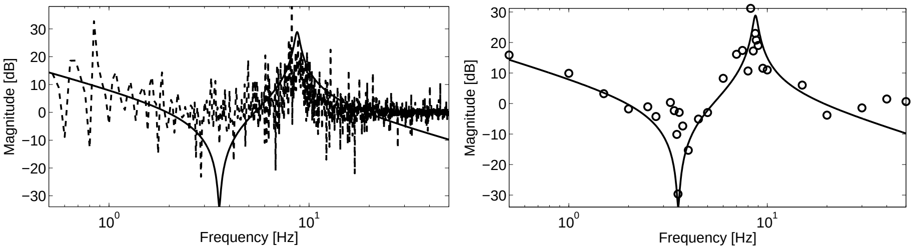

Figure 4.9: Low excitation amplitude: Magnitude of true FRF (solid), estimated FRF with chirp excitation (dashed, left), and estimated FRF with multisine excitation (circles, right).

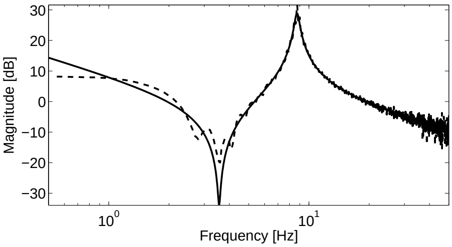

Figure 4.10: Magnitude of true FRF (solid), estimated FRF with large leakage errors (dashed).

multisine excitation improves the estimate somewhat.

The effect of frequency leakage is illustrated in Figure 4.10. Here, the measurement is stopped at $t = 20\,\mathrm{s}$, before the system has come to rest. Chirp excitation is used in this case, but leakage can also occur if multisine excitation is used, e.g., if data from the first transient period is used, or if the data is not exactly an integer number of multisine periods.

Finally, in Figure 4.11, a comparison is made between the model FRF of the continuous-time model, and of the discrete-time, zero-order hold sampled model. The FRFs are shown up to the Nyquist frequency 500 Hz. The FRFs are almost identical for lower frequencies. Some conclusions can be drawn from this example:

- FRFs can be estimated even if the system runs in closed loop. To reduce bias and variance errors, the excitation level must be reasonably high compared to the level of disturbances.
- A burst chirp excitation works if the system is at rest when measurements start and end. However, it might be necessary to smooth the estimated FRF if the disturbance level is high.
- A periodic multisine excitation improves the estimated FRF.

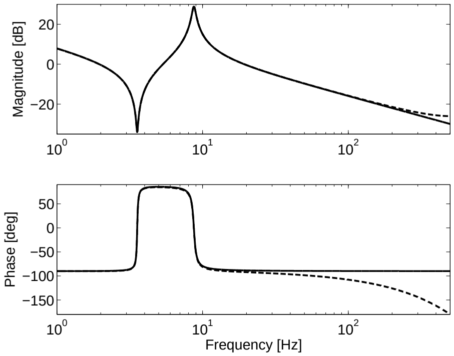

Figure 4.11: *FRF of the continuous (solid line) and discrete (dashed line) two-mass model.*

- If the sample frequency is well above the frequencies of the interesting process dynamics, continuous-time model FRFs can be directly compared to discrete-time estimated FRFs, e.g., if a parametric model is to be estimated from the nonparametric (estimated) FRF. If this is not the case, the discrete-time FRF of the model must be computed.

For further discussions on closed-loop versus open-loop identification, see, e.g, Ljung (1999). Periodic excitation is often recommended as an alternative way of obtaining a leakage-free FRF, see, e.g., Pintelon and Schoukens (2001). This is the adopted solution described in Paper B, where a multisine excitation is used. The multisine excitation has, among many things, the advantage that the excitation energy is concentrated at selected frequencies, thus improving the signal-to-noise ratio at the frequencies where the DFT is evaluated.

4.1.4 Parametric Models

A parametric model is a model described as, e.g., differential or difference equations. System identification is one route for obtaining a parametric model of a system. Another route is physical modeling, i.e., deriving a mathematical model from the basic laws of physics.

If the parameters of a physical model are known with sufficient accuracy, we get a *white-box model*, where both the model structure and the model parameters are known. An example of such a model is a rigid-body model, as described in Section 3.2.1, where the kinematic and inertial parameters are known from the CAD models.

A *gray-box model* is a physical model where the model structure is known but the physical parameters are unknown or only partly known. Identification of parameters in this case is called gray-box identification. A nonlinear gray-box model can be formulated in continuous state-space form, i.e.,

$$\dot{x}(t) = f(x(t), u(t), \theta), \tag{4.13a}$$

$$y(t) = h(x(t), u(t), \theta), \tag{4.13b}$$

with states x, control input u, and measurements y. The unknown parameters are contained in θ. One example of a gray-box model is a simplified flexible joint model (3.12) where the rigid-body parameters are known, and the elastic parameters, consisting of springs and dampers, are unknown. The simplified flexible joint model with measurement of motor positions can be expressed as

$$\dot{x} = \begin{bmatrix} x_3 \\ x_4 \\ M_a^{-1}(x_1)[-c(x_1, x_3) - g(x_1) + diag(\theta_k)(x_2 - x_1) + diag(\theta_d)(x_4 - x_3)] \\ M_m^{-1}[-f(x_4) + diag(\theta_k)(x_1 - x_2) + diag(\theta_d)(x_3 - x_4) + u] \end{bmatrix}, \tag{4.14a}$$

$$y = x_2, \tag{4.14b}$$

with states

$$x = \begin{bmatrix} x_1^T & x_2^T & x_3^T & x_4^T \end{bmatrix}^T = \begin{bmatrix} q_a^T & q_m^T & \dot{q}_a^T & \dot{q}_m^T \end{bmatrix}^T, \tag{4.15}$$

motor torque u, and the unknown elasticity parameters

$$\theta = \begin{bmatrix} \theta_k & \theta_d \end{bmatrix} = \begin{bmatrix} k_1 & \dots & k_N & d_1 & \dots & d_N \end{bmatrix}. \tag{4.16}$$

If (4.13) is discretized, we obtain the discrete-time gray-box model

$$x(t + T_s) = f_d(x(t), u(t), w(t), \theta), \tag{4.17a}$$

$$y(t) = h_d(x(t), u(t), v(t), \theta), \tag{4.17b}$$

with sample time T_s. Here, the process- and measurement disturbances w and v, assumed to be zero-mean white noise processes, are also included. The discretization of the continuous-time system can, for example, be carried out using the Euler forward formula.

A third type of parametric model is the so-called *black-box model*. In this case, the model structure is not known, and therefore, a model structure without any direct physical interpretation is used for the identification. The black-box model parameters can, e.g., be the coefficients of a linear difference equation. A general linear discrete-time black-box model can be expressed as

$$y(t) = G(q^{-1}, \theta)u(t) + H(q^{-1}, \theta)e(t), \tag{4.18}$$

where $y(t) \in \mathbb{R}^{n_y}$ is the output, $u(t) \in \mathbb{R}^{n_u}$ is the input, and $e(t) \in \mathbb{R}^{n_y}$ is a sequence of independent zero-mean random vectors with covariance matrix Λ. $G(\cdot)$ and $H(\cdot)$ are filters of obvious dimensions, rational functions of the backward shift operator q^{-1}, and of finite order. There are a number of standard black-box model structures, used for discrete-time identification. Examples of such standard structures are the output error structure and the ARX structure

(Ljung, 1999). The ARX structure for a single-input single-output system is

$$A(q^{-1})y(t) = B(q^{-1})q^{-n_k}u(t) + e(t), \tag{4.19a}$$

$$A(q^{-1}) = 1 + a_1 q^{-1} + \cdots + a_{n_a} q^{-n_a}, \tag{4.19b}$$

$$B(q^{-1}) = b_0 + b_1 q^{-1} + \cdots + b_{n_b} q^{-n_b}, \tag{4.19c}$$

with model parameters

$$\theta = \begin{bmatrix} a_1 & \dots & a_{n_a} & b_0 & \dots & b_{n_b} & n_k \end{bmatrix}. \tag{4.20}$$

4.1.5 Identification of Parametric Models

The gray-box and black-box models can be identified directly in the time domain. One way of estimating the parameters is by the use of a so-called *prediction error method* (Ljung, 1999). For linear systems described by (4.18), the optimal *one-step-ahead predictor* is

$$\hat{y}(t|t-T_s;\theta) = H^{-1}(q^{-1},\theta)G(q^{-1},\theta)u(t) + [I - H^{-1}(q^{-1},\theta)]y(t), \tag{4.21}$$

where the predictor $\hat{y}(t|t-T_s,\theta)$ depends on previous inputs and outputs. The prediction error is then

$$\varepsilon(t,\theta) = y(t) - \hat{y}(t|t-T_s;\theta) = e(t) = H^{-1}(q^{-1},\theta)[y(t) - G(q^{-1},\theta)u(t)]. \tag{4.22}$$

When using a prediction-error method, the error between the predicted model output and the measured output, i.e., $\varepsilon(t,\theta)$, is minimized according to some distance measure (norm). A common choice of norm is a quadratic criterion, and the estimator is then

$$\hat{\theta}_N = \arg\min_{\theta} V_N(\theta), \tag{4.23a}$$

$$V_N(\theta) = \frac{1}{N}\sum_{k=1}^{N} \varepsilon^2(kT_s,\theta), \tag{4.23b}$$

assuming N samples of data have been collected. For some model structures the parameters $\hat{\theta}_N$ can be estimated using linear regression, but for most structures a numerical search procedure is necessary.

For linear discrete-time state-space models, the Kalman filter is the optimal one-step-ahead predictor, and prediction-error methods can be used for identification. State-space models can also be expressed on the general form (4.18). Besides prediction-error methods, *subspace methods* are often used for identification of linear discrete-time state-space models (Ljung, 1999).

The gray-box model structure can also be identified in the time-domain. To design an optimal predictor for the nonlinear state-space model (4.17) is, in general, not possible. Approximate predictors, e.g, the *extended Kalman filter* (Anderson and Moore, 1979) can then be used. A simple predictor is the noise-free simulated system output of (4.17)

$$x(t + T_s) = f_d(x(t), u(t), 0, \theta), \tag{4.24a}$$

$$\hat{y}(t + T_s|t) = h_d(x(t + T_s), u(t + T_s), 0, \theta). \tag{4.24b}$$

However, this only works for stable systems, since a stable predictor is always required.

For closed loop identification, prediction-error methods that describe the true noise properties can be used for accurate estimates (small bias). However, some methods may fail, e.g., output error methods. For a thorough description of time-domain methods, see Ljung (1999) or Söderström and Stoica (1989).

The gray-box and black-box models can also be identified in the *frequency domain*. One way of doing this is to minimize the least squares error between the estimated nonparametric FRF and the parametric model FRF. The frequency-domain methods are described in, e.g., Pintelon and Schoukens (2001). In Paper B, it is shown that a time-domain prediction-error method for a nonlinear system can be approximated by a frequency-domain gray-box identification method.

Finally, an illustrative example of frequency-domain gray-box identification. The physical parameters of the the single-axis flexible joint model in Section 4.1.3 can be approximated by inspecting the estimated FRF from torque to acceleration, according to Figure 4.12, and using (4.9)–(4.10). The inertias can be estimated from the low- and high frequency gains, and the elasticity parameters can be estimated, e.g., from the frequency and gain of the anti-resonance. Note that the accuracy can be increased by fitting a parametric FRF over the whole frequency range. The resulting parameters are $J_a = 0.052$ ($J_a^{true} = 0.050$), $J_m = 0.009$ ($J_m^{true} = 0.010$), $k = 26$ ($k^{true} = 25$), and $d = 0.032$ ($d^{true} = 0.025$).

4.2 Identification of Robot Manipulators

This section briefly describes the identification of some models needed for the control and simulation of robot manipulators. The most important models for this work are the

- Kinematic model
- Rigid dynamic model
- Elastic dynamic model

Other important models subject to identification but not further described are, e.g., friction models, backlash and hysteresis models, thermal models, fatigue models, actuator models, and sensor models. For a more thorough survey of identification methods for robot manipulators, see, e.g., Kozlowski (1998) and Wernholt (2007).

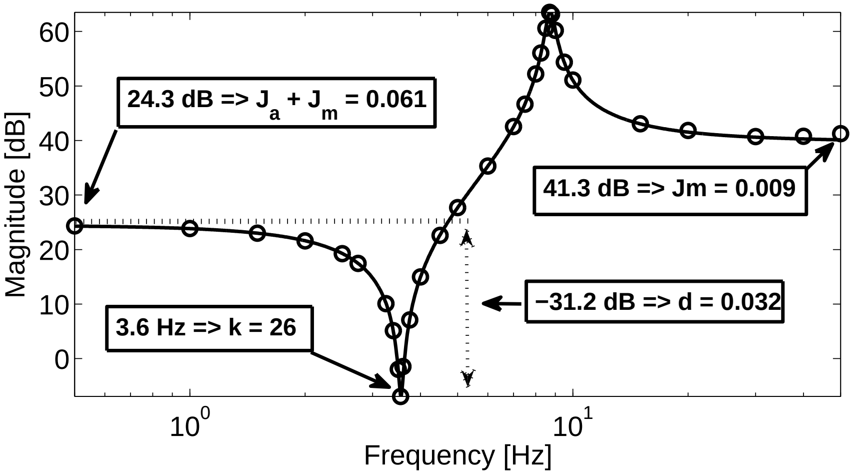

Figure 4.12: Approximate parameter estimation by inspecting the estimated FRF from motor torque to motor acceleration.

4.2.1 Identification of Kinematic Models and Rigid Dynamic Models

The described models depend on a number of parameters. The kinematic link parameters consist of lengths and angles. The dynamic model depends on the kinematic link parameters as well as the inertial link parameters. The accuracy of these models depends on the accuracy of these parameters. The parameter values could in some cases be obtained from the CAD models of the robot manipulator, and in other cases from measurements of the individual parts of the robot. If these methods are not accurate enough or simply not possible to perform, then identification of the unknown parameters can be used to obtain the unknown parameter values. This type of identification is usually denoted gray-box identification as described in Section 4.1.4.

The nominal kinematic model of a large industrial robot typically gives a volumetric accuracy of 2–15 mm due to the tolerances of components and variations in the assembly procedure. This does not fulfill the accuracy requirement for off-line programming of, e.g., a spot-welding application. By identification of the kinematic parameters of the individual manipulator, as well as the elastostatic model, used for compensating the deflection due to gravity, a volumetric accuracy of ±0.5 mm can be obtained.

It is also interesting to be able to identify the rigid dynamic model (2.3). This can be performed as a verification of the CAD model parameters when a new robot is developed or for direct use in the controller. The rigid dynamic model can be expressed as (Sciavicco and Siciliano, 2000)

$$\tau = H(q, \dot{q}, \ddot{q})\pi, \tag{4.25}$$

where π is a vector of the unknown dynamic parameters. The model thus has the property of linearity in the parameters. The parameters, called base parameters, are not the same as the original link inertial parameters which, in general, cannot all be identified. The parameters in π are a set of uniquely identifiable parameters, and consist of different combinations of the physical parameters described in Section 3.2.1, e.g., $J_{yy} + m(\xi_x^2 + \xi_z^2)$. If measurements of torques and positions are performed along an exciting trajectory, the identification problem can be formulated as a linear regression

$$\overline{\tau} = \begin{bmatrix} \tau(t_1) \\ \vdots \\ \tau(t_N) \end{bmatrix} = \begin{bmatrix} H(t_1) \\ \vdots \\ H(t_N) \end{bmatrix} \pi = \overline{H}\pi, \tag{4.26}$$

and the solution is obtained as the least-squares solution

$$\pi = (\overline{H}^T \overline{H})^{-1} \overline{H}^T \overline{\tau}. \tag{4.27}$$

Note that the speed and acceleration, if not measured, must be estimated from the measured positions. For the identification to be possible, the movements of the manipulator during the identification experiment must reveal information of all unknown parameters, i.e., the excitation must be rich enough. Furthermore, the movements should not excite the mechanical resonances of the manipulator. Identification of dynamic parameters is further described in, e.g., Swevers et al. (2007).

4.2.2 Identification of Elastic Dynamic Models

This section gives some examples of proposed identification methods for flexible robot manipulators. Most articles on flexible manipulator identification consider *local models*, i.e., models valid in one configuration, of black-box or gray-box type. Moreover, the standard assumption is that all parameters are unknown, and most suggested identification methods therefore estimate all parameters in one step. In Pham et al. (2001), all physical parameters of a single-input single-output (SISO) model are identified by linear regression, using a linear-in-parameter model structure. This is exemplified with measurements on both motor and joint side, but some parameters (stiffness and mass) can be identified with motor measurements only. In Pham et al. (2002), this method is extended with acceleration measurements on the joint side. A multiple-input multiple-output (MIMO) black-box model of two robot axes is identified from motor side measurements in Johansson et al. (2000) by use of a subspace method in combination with a friction estimation. A gray-box prediction error method is used in Östring et al. (2003) to identify all physical parameters of a SISO model, using motor measurements only. Inverse eigenvalue theory is used to identify a mass-spring model of any order in Berglund and Hovland (2000), also using motor measurements only. This method is extended to MIMO models in Hovland et al. (2001). Experimental modal analysis is used in Behi and Tesar (1991) to identify the local masses, springs, and dampers of an industrial manipulator. The system is then excited by an impact hammer and the response measured by accelerometers. An introduction to modal analysis can be found in Avitabile (2001).

Identification methods for *global flexible joint models* have also been suggested. Prior knowledge of the rigid body parameters is assumed in Hovland et al. (1999), where the stiffness and damping parameters are identified by using a frequency domain method where the frequency domain model is linear in the unknown parameters. This method works for MIMO flexible joint models with motor measurements only, and is exemplified on two axes of an industrial manipulator. A three-step identification method is suggested in Wernholt and Gunnarsson (2006). The first step identifies nonlinear friction and rigid body parameters using a separable least-squares approach, the second step is based on Berglund and Hovland (2000) and identifies approximate elasticity parameters and refines some rigid body parameters, and the last step further refines some parameters by time-domain nonlinear gray-box identification. The method is applied to one axis of an industrial manipulator using a three-mass model and a nonlinear transmission stiffness (stiffening spring). In Albu-Schäffer and Hirzinger (2001) the rigid body parameters are assumed to be known from CAD models, while the friction and elasticity parameters are identified in separate identification experiments. The elasticities of the transmissions are identified from impulse response experiments for each individual joint before the assembly of the robot, using joint torque sensors.

In Hardeman (2008), an identification method for an *extended flexible joint model* with transmission and bearing elasticity, is suggested. The method is based on the assumption that all DOFs (motor position, transmission and bearing deflection) are measurable. The unknown elasticity parameters are solved by linear regression. However, the measurement system used was not accurate enough for measuring the elastic deflections. An alternative method which further develops the ideas in Hovland et al. (2001) is then used for estimating the transmission stiffness. However, the bearing stiffness cannot be identified using this method, and the uncertainty of the estimated transmission stiffness is rather large, e.g., standard deviations larger than 30 %.

4.2.3 Identification of the Extended Flexible Joint Dynamic Model

In Papers A and B, and in Wernholt (2007), an identification procedure for the unknown elastic parameters of the extended flexible joint model (3.19), described in Section 3.2.4, is proposed. The model is global and nonlinear. The identification procedure is a frequency-domain gray-box method and can be summarized as:

1. Local nonparametric models are estimated in a number of configurations. The models are frequency response functions, FRFs.

2. The nonlinear parametric robot model is linearized in each of these configurations.
3. The parametric FRFs of these linearized models are obtained for a value of the unknown parameter vector.
4. The model FRFs and the estimated nonparametric FRFs are compared and an error computed. The parameter vector is adjusted to minimize the error.
5. Repeat from 3 until some criteria is fulfilled.

Paper B provides a detailed description of this method, and shows that it is an approximation of a time-domain prediction-error method. The method is exemplified, by estimating the elasticity parameters of a six-axes industrial robot. Some problems with applying a time-domain identification method to this problem is also discussed. Paper A show that the flexible joint model cannot describe a modern industrial robot accurately, and that the extended model increases the accuracy.

4.3 Summary

In system identification there are alternative choices concerning the experimental setup, excitation signals, model types, and identification methods. The following aspects have been discussed:

- Experimental setup
  - Open loop
  - Closed loop
- Excitation signal
  - Periodic (e.g., multisine)
  - Non-periodic (e.g., chirp or step)
- Model type
  - Nonparametric (e.g., step response or frequency response function)
  - Parametric (e.g., discrete-time linear black-box or continuous-time nonlinear gray-box)
- Identification method
  - Time-domain methods (e.g., prediction-error methods)
  - Frequency-domain methods

However, there are many aspects of system identification not mentioned in this brief description, e.g., validation, experimental design, noise models, and model quality measures like bias and variance. The interested reader is referred to the literature cited in this chapter.

Identification is often used for parameter estimation of robot manipulator models. The following cases have been discussed:

- Identification of kinematic parameters
- Identification of rigid dynamic models by linear regression
- Identification of local and global elastic dynamic models

Although, methods for identification of global elastic models have been suggested, the amount of published research in this area is surprisingly small compared to the amount of research that has been published on sophisticated control concepts for these assumed global models. To the authors knowledge, no global validation of the flexible joint model has been published for a multi-axes industrial-like robot manipulator. Moreover, no global validation or identification of a lumped parameter model (e.g., an extended flexible joint model) for a multi-axes industrial-like robot manipulator has been published. Paper A validates the global extended flexible joint model and Paper B (and its predecessors described in Section 1.2) describes such an identification procedure.

5

Control of Robot Manipulators

5.1 Introduction

Advanced motion control of robot manipulators has been studied by academic and industrial researchers since the beginning of the 1970's. A historical summary with many early references is given in Craig (1988).

This chapter is a survey of position control methods for articulated robot manipulators[1], suggested in the literature. Control during contact with the environment, as for example in the case of force control, will not be treated. The actuator is assumed to be an electrical motor, and the motor torque control will not be treated. It is assumed that the motor torque control is ideal, and that the torque reference generated by the position controller is equal to the real motor torque[2]. Furthermore, even though friction is the dominating source of error in some cases, e.g., at low speed, control methods specially designed for dealing with friction will not be covered. The emphasis will be on control methods for handling the elasticity of the manipulator, without influence of tool contact dynamics. This is the far most usual case for industrial robot applications today.

The focus will be on methods applicable to a typical industrial robot, i.e., an elastic manipulator with gear transmissions, where the only measured variables are the motor angular positions. Some of the described methods assume that more variables are measured but can in many cases be modified such that actuator position only is sufficient, e.g., by using state estimation, which is briefly

[1] Results for manipulators of parallel linkage type are also applicable to a large extent for serial link manipulators and vice versa.

[2] Experiments have shown that this is a reasonable approximation when the current- and torque-control servo has an appropriate tuning and uses feedforward.

described. The main approach for controlling an elastic manipulator is considered to be *linear feedback* in combination with nonlinear *feedforward*- or *feedback linearization* control. Therefore, these methods will be described in some detail. This is also motivated by the fact that Papers C–G treat these control methods. Some examples of methods not treated here, or only briefly mentioned, are iterative learning control, adaptive control, backstepping, sliding mode control, neural networks, singular perturbations, composite control, input shaping, passivity-based control, and robustification by Lyapunov's second method. Two survey articles with many references are Sage et al. (1999) and Benosman and Le Vey (2004).

The manipulator to be controlled is an elastic multibody system. The system is multivariable and strongly coupled, and its highly nonlinear dynamics changes rapidly as the manipulator moves within its working range. Moreover, for a robot with gear transmissions, the gears have nonlinearities such as hysteresis, backlash, friction, and nonlinear elasticity. The actuators have non-ideal characteristics with internally generated disturbances, e.g., torque ripple disturbances. For a typical industrial robot, the controlled variable (tool position and orientation), is not measured, and the only measured variable is the actuator position (motor angular position). This position measurement can be impaired with a high level of measurement noise as well as deterministic disturbances. For industrial manipulators, the dynamic position accuracy requirements for low speed applications (e.g., laser cutting), can be 0.1 mm at 20 mm/s. For high-speed applications, such as dispensing, the maximum allowed error can be 0.5 mm at 500 mm/s, without visible vibrations. Another requirement could be short cycle time, which means high acceleration. Thus, the control problem can, in general, not be solved by applying smooth trajectories to avoid exciting the mechanical resonances. The conclusion is that the flexible manipulator control problem is a challenging task.

The control of robot manipulators can be described and classified in many ways, according to, e.g.,

- type of drive system (direct drive or gear transmission).
- type of model used for the (model-based) control (rigid models, flexible joint models, or flexible link models).
- controlled variable (position, speed, compliance, or force).
- motion type considered (high-speed continuous path tracking, low-speed continuous path tracking, point-to-point movement, tracking in contact with the environment, or regulation control).
- type of control law (linear/nonlinear, feedback/feedforward, static/dynamic, robust/adaptive).
- type of measurements (actuator position, actuator speed, link position, link speed, link acceleration, link torque, tool position, tool speed, tool acceleration, tool force, or tool torque).

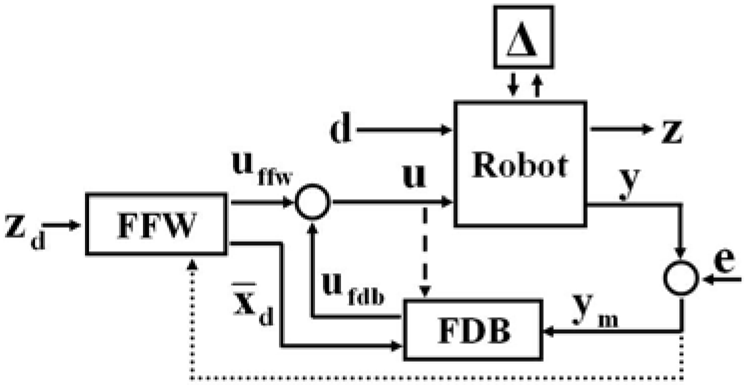

Figure 5.1: *General robot controller structure. Observer reconstruction is enabled by the dashed line, feedback linearization by the dotted line.*

A general controller structure is illustrated in Figure 5.1. The desired tool trajectory z_d is described in Cartesian coordinates, and z is the actual trajectory[3], also described in Cartesian coordinates. The reference- and feedforward generation block (FFW) computes the feedforward torque u_{ffw} and the state references $\bar{x}_d$ used by the feedback controller (FDB), with a correction torque, u_{fdb}, as output. The manipulator has uncertain parameters, illustrated as a feedback with unknown parameters Δ, and is exposed to disturbances d and measurement noise e. The measured signals are denoted y_m. Note that the dimension of $\bar{x}_d$ and y_m may differ if some states are reconstructed by FDB. The purpose of the FFW is to generate model-based references for perfect tracking (if possible). The purpose of the FDB is, under the influence of measurement noise, to stabilize the system, reject disturbances, and to compensate for errors in the FFW.

5.2 Control of Rigid Manipulators

Although this work is primarily focused on flexible manipulators, the control of rigid manipulators is a good starting point for this survey. A rigid manipulator can here, from a control point of view, be interpreted as a manipulator with the lowest mechanical resonances well above the bandwidth of the control. A direct-drive manipulator with stiff links is an example of a manipulator with resonances at high frequencies. In direct-drive manipulators, the motor axes are directly coupled to the links, and the negative effects of gear transmissions are eliminated, i.e., friction, backlash, and elasticity. The N-link rigid manipulator has N degrees-of-freedom, and the joint angle q, can be regarded as output variable since it can be computed from the Cartesian tool position and vice versa (by using the inverse and forward kinematics). Hence, the reference to the controller can be chosen as

[3] z is used instead of x to denote the Cartesian position and orientation to avoid confusion with the states of the system.

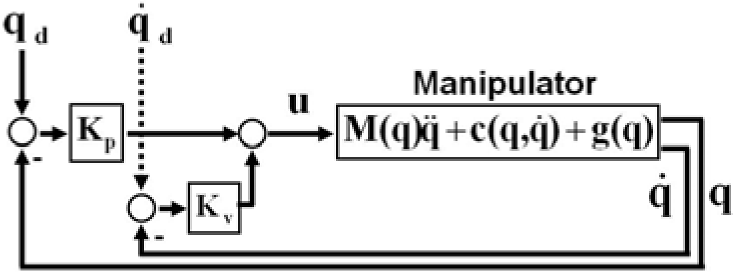

Figure 5.2: *Diagonal PD control (independent joint PD control).*

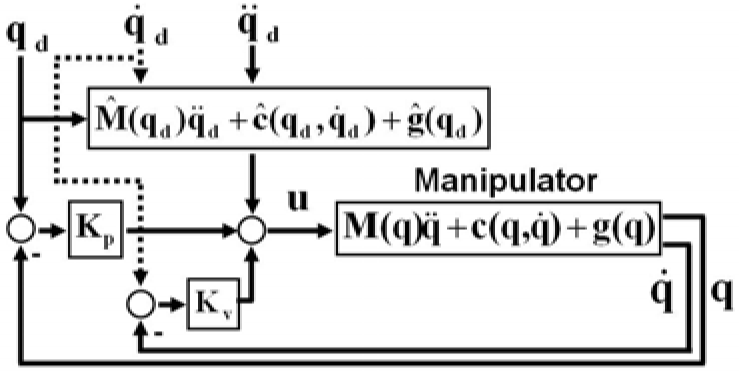
q_d
$\dot{q}_d$
$\ddot{q}_d$
$\hat{M}(q_d)\ddot{q}_d+\hat{c}(q_d,\dot{q}_d)+\hat{g}(q_d)$
Manipulator
u
K_p
K_v
$M(q)\ddot{q}+c(q,\dot{q})+g(q)$
$\dot{q}$
q
-
-

Figure 5.3: *Feedforward control combined with diagonal PD control.*

the desired joint angle $q_d(t)$.

5.2.1 Feedback Linearization and Feedforward Control

A summary of control methods and experimental results for rigid direct-drive robots can be found in An et al. (1988). In the model-based approaches, the rigid dynamic model (3.10) is used.

The first method described is called *independent joint PD control*, and is illustrated in Figure 5.2. The controller for the direct drive manipulator is given by

$$u = K_p(q_d - q) + K_v(\dot{q}_d - \dot{q}), \tag{5.1}$$

where the measured position and speed are q and $\dot{q}$, and the desired position and speed are q_d and $\dot{q}_d$. The control signal is the reference torque u. Finally, K_v and K_p are diagonal gain matrices with obvious dimensions. In this method, the multivariable couplings between the axes are regarded as unmodeled disturbances.

The second method shown in Figure 5.3 is called *feedforward control*, and is an

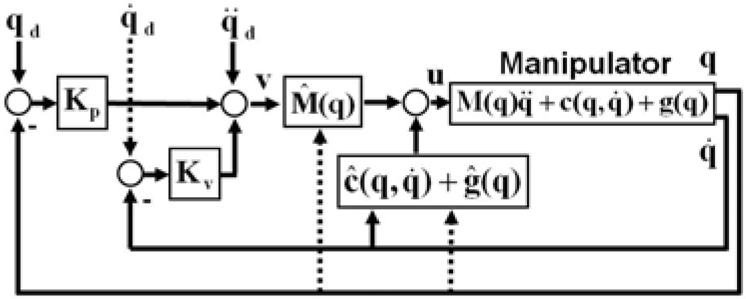

***Figure 5.4:** Feedback linearization (computed torque control) combined with outer-loop diagonal PD control.*

extension of the PD controller with a feedforward torque u_{ffw} according to

$$u = u_{ffw} + K_p(q_d - q) + K_v(\dot{q}_d - \dot{q}), \tag{5.2a}$$

$$u_{ffw} = \hat{M}(q_d)\ddot{q}_d + \hat{c}(q_d, \dot{q}_d) + \hat{g}(q_d). \tag{5.2b}$$

Note that the reference q_d must be at least twice differentiable. The ˆ signs indicates that a model, that always differs from the true system, is used by the controller.

The third method described is called *computed torque control*, and is illustrated in Figure 5.4. In this approach the system is first partly linearized by canceling the nonlinear dynamics $c(q_d, \dot{q}_d) + g(q_d)$ with a feedback term. The system is then linearized and decoupled by multiplying the controller output with the inverse system, i.e., the mass matrix. Ideally, the resulting system is a system of decoupled double integrators, i.e., $v = \ddot{q}$. The natural choice of controller, with output v, is again a PD controller. The first approaches of linearizing a nonlinear system by nonlinear feedback can be found in the robotics literature from the 1970's. The computed torque controller is

$$u = \hat{M}(q)v + \hat{c}(q, \dot{q}) + \hat{g}(q), \tag{5.3a}$$

$$v = \ddot{q}_d + K_p(q_d - q) + K_v(\dot{q}_d - \dot{q}). \tag{5.3b}$$

The terminology in this field is somewhat confusing. Computed torque can sometimes denote the feedforward control law, and sometimes the linearizing and decoupling control law. With standard control terminology, the control methods can also be described as *diagonal PD control*, *feedforward control*, and *feedback linearization control*. These terms will be used from now on. In both of these model-based methods it is necessary to solve the *inverse dynamics problem*, i.e., compute the torque from the desired trajectory. Also note that one part of the feedback linearization controller output in Figure 5.4 is computed from the feedforward acceleration $\ddot{q}_d$.

The bullets below summarize results concerning tracking errors according to An et al. (1988), for the diagonal PD, feedforward, and feedback linearization control methods, when applied to the main axes of a direct-drive manipulator (the MIT serial-link direct-drive arm). A smooth fifth order polynomial trajectory with high speed and acceleration (360 deg/s and 850 deg/s^2) was used in the experiments.

- For two axes, the model-based controllers reduced the tracking error to 2 deg compared to 4 deg with PD control. No significant difference between feedforward and feedback linearization was noticed.
- The third axis had the same tracking error, 4 deg, for all three methods. This was explained by unmodeled motor dynamics and bearing friction in combination with low inertia.
- The sampling time was critical for the model-based methods. It affected the discretization error but also the possible level of feedback gain.
- The feedforward control method was believed to be the best choice for free space movements. For cases with large disturbances, the feedback linearization controller was believed to yield better results.

One reflection concerning these results is that the PD controller performance was surprisingly good. It should also be noted that a feedforward controller has the advantage that it, in principle, can even be computed off-line in order to save on-line computer load. Even if the feedforward is computed on-line, the sample rate of the feedforward computations does not affect the stability of the system. Thus, it might be possible to save computer load by using a lower sample rate since it will not reduce the stability. On the other hand, the feedback linearization controller is a part of the feedback loop, and could cause instability if the model errors are large.

A more recent publication on the same topic is Santibanez and Kelly (2001). The conclusion in this and a number of other articles is the same as in An et al. (1988). Feedforward control gives the same tracking performance as feedback linearization and is the preferred choice.

Feedback linearization control and diagonal PD control is also evaluated with respect to tracking performance of a direct-drive manipulator in Khosla and Kanade (1989). The conclusion is that feedback linearization control gives better performance than diagonal PD control, and that it is important to include the centripetal and Coriolis terms in the linearization.

A final reflection on this topic is that the test cases studied in the referred articles are typically a few test trajectories in each article, and only with the best possible model. Feedback linearization and feedforward based on high-accuracy models should give the same tracking performance if the sample rate is chosen such that the discretization effects are negligible. It would be very interesting to see a comparative study concerning robustness to model errors and disturbance rejection, for the different published methods.

5.2.2 Other Control Methods for Rigid Manipulators

In this section, some examples of other control methods that have been considered for the control of rigid manipulators, are given.

Adaptive Control Adaptive control of direct-drive manipulators is studied in, e.g., Craig (1988). The model parameters of the feedback linearization controller described in the previous section are adapted on-line. An experimental study on two direct-drive axes of the Adept One robot is also included. It is shown that the constant gain default controller yields a better result than the adaptive controller. However, it is believed that more fine tuning and a new implementation of the adaptive controller concept could improve the result.

Nonlinear Robust Control A nonlinear outer-loop controller, replacing the PD controller, can be used to robustify the feedback linearization controller. One example of such a controller, based on *Lyapunov's Second Method*, is described in Spong et al. (2006), and shown to reduce the tracking errors when model errors are present. Another proposed controller is the *Sliding Mode Controller*. One example of this controller type used for robustification of a feedback linearization controller is described in Bellini et al. (1989), where the sliding mode controller was shown to decrease control errors caused by an approximative model. Both the Lyapunov- and the sliding mode controllers result in a discontinuous control signal (chattering) that can increase motor losses and excite unmodeled dynamics. Methods for decreasing the chattering exist for both of these controller types.

A summary of control methods for rigid manipulators is given in Spong (1996).

5.3 Control of Flexible Joint Manipulators

The flexible joint model, as described in Section 3.2.2, is a more realistic description of an industrial robot with gear transmissions. This model has elastic gear transmissions and rigid links. The N-link manipulator model has $2N$ degrees-of-freedom and, as in the case of the rigid manipulator, the Cartesian position z, or the joint angles q, can be regarded as the output variable since the links are rigid. Hence, the reference to the controller is assumed to be the vector of desired joint angles q_d.

5.3.1 Feedback Linearization and Feedforward Control

Feedback linearization and feedforward control can be regarded as the main approaches for the control of flexible joint manipulators, and will therefore be treated in some detail.

5.3.2 Simplified Flexible Joint Model

The model is here given by

$$M_a(q_a)\ddot{q}_a + c(q_a, \dot{q}_a) + g(q_a) + K(q_a - q_m) = 0, \tag{5.4a}$$

$$M_m\ddot{q}_m + K(q_m - q_a) = u, \tag{5.4b}$$

where the same notations as in Section 3.2.2 are used, except that the control signal, the motor torque, is denoted u. In Spong (1987) and Spong et al. (2006), control methods for the simplified flexible joint model are discussed. In the simplified model, the inertial couplings between the links and the motors are neglected. Furthermore, the viscous damping is also neglected in order to simplify the controller design. In Spong (1987) it is shown that a manipulator described by this model can be linearized and decoupled by static feedback linearization. As for the rigid model described in the previous section, the flexible joint model can be used for feedforward control or feedback linearization. The feedforward approach is described in, e.g, De Luca (2000).

The flexible joint manipulator is an example of a differentially flat system (Rouchon et al., 1993). Such a system can be defined as a system where all state variables and control inputs can be expressed as an algebraic function of the desired trajectory for a flat output, and its derivatives, up to a certain order. The flat output is the selected output variable of the system. Feedback linearization by static or dynamic state feedback is equivalent to differential flatness (Nieuwstadt and Murray, 1998). Solving (5.4a) for q_m, and differentiating twice, we get an expression for $\ddot{q}_m$, adding (5.4a) to (5.4b), and inserting the expression for $\ddot{q}_m$ yields

$$\begin{aligned} u = {} & (M_a(q_a) + M_m)\ddot{q}_a + c(q_a, \dot{q}_a) + g(q_a) + M_m K_g^{-1}[\ddot{M}_a(q_a, \dot{q}_a, \ddot{q}_a)\ddot{q}_a + \\ & 2\dot{M}_a(q_a, \dot{q}_a)q_a^{[3]} + M_a(q_a)q_a^{[4]} + \ddot{c}(q_a, \dot{q}_a, \ddot{q}_a, q_a^{[3]}) + \ddot{g}(q_a, \dot{q}_a, \ddot{q}_a)], \end{aligned} \tag{5.5}$$

where $x^{[i]}$ denotes $d^i x/dt^i$. This expression shows that the system is differentially flat with the flat output q_a and that the control signal can be expressed as

$$u = \tau_m(q_a, \dot{q}_a, \ddot{q}_a, q_a^{[3]}, q_a^{[4]}). \tag{5.6}$$

Using the states

$$x = \begin{bmatrix} q_a^T & \dot{q}_a^T & \ddot{q}_a^T & q_a^{[3]T} \end{bmatrix}^T, \tag{5.7}$$

the feedback linearization control law can be expressed as

$$u = \tau_m(x_{1m}, x_{2m}, x_{3m}, x_{4m}, v), \tag{5.8}$$

where x_{jm} are the measured states, and v is a new control signal for the linearized and decoupled system $q_a^{[4]} = v$ consisting of N independent chains of four integrators. For tracking control, v can be chosen as

$$v = q_{ad}^{[4]} + L_1(x_d - x_m), \tag{5.9}$$

where $L_1 \in \mathbb{R}^{N\times 4N}$ is a linear feedback gain matrix and x_d, x_m are the reference states and the measured states respectively. The fourth derivative of the reference trajectory is denoted $q_{ad}^{[4]}$. The derived control law is a combination of feedback and feedforward where the feedback part is dominating. The feedback gain matrix can be computed, e.g., by using LQ optimal control (Anderson and Moore, 1990). It is also possible to find a feedforward-dominant control law according to

$$u = \tau_m(x_{1d}, x_{2d}, x_{3d}, x_{4d}, q_{ad}^{[4]}) + L_2(x_d - x_m), \tag{5.10}$$

where, ideally in the case of a perfect model, all torques needed for the desired trajectory are computed by feedforward calculations, i.e., based on the reference states x_d. Note that the desired trajectory q_d must be at least four times differentiable for both the feedback linearization and the feedforward control laws. The feedback linearization control law (5.8) gives constant bandwidth of the feedback controller for all robot configurations. For constant bandwidth there would be no need for gain scheduling. Due to the varying manipulator dynamics, gain scheduling, or some other model-based feedback gain computation, is probably needed in the feedforward control law (5.10). A simulation study comparing these two control algorithms is presented in Paper E. Furthermore, feedforward control for an extended flexible joint model is investigated in Paper C and D.

5.3.3 Complete Flexible Joint Model

The complete flexible joint model (3.13) cannot be linearized by static state feedback. However, if the viscous damping is excluded, any flexible joint model can be linearized and decoupled by dynamic state feedback as shown in De Luca (1988) and De Luca and Lanari (1995). The static feedback controller described in the previous section has the form

$$u = \alpha(x) + \beta(x)v, \tag{5.11}$$

where x are the states, v are the new control signals for the linearized system, and $\alpha(\cdot)$, $\beta(\cdot)$ are nonlinear functions of appropriate dimensions. The dynamic feedback controller has the form (De Luca and Lanari, 1995)

$$\dot{\xi} = \alpha(x, \xi) + \beta(x, \xi)v, \tag{5.12a}$$

$$u = \gamma(x, \xi) + \delta(x, \xi)v, \tag{5.12b}$$

where ξ are the internal states of the controller ($\alpha(\cdot)$, $\beta(\cdot)$, $\gamma(\cdot)$, and $\delta(\cdot)$ are nonlinear functions of appropriate dimensions).

The proof that the complete model is feedback linearizable is based on the fact that the system is invertible with no zero dynamics. Given a desired trajectory and its derivatives up to a certain order, the motor states and the required torques can be computed recursively. The solution is based on the fact that S in (3.13) is upper triangular. This result can also be used for feedforward control based on the complete model (De Luca, 2000). Static or dynamic feedback linearization can also be performed when the damping term is included but in this case only input-output linearization is possible (De Luca et al., 2005). Input-output linearization can be used since the zero dynamics is stable.

The complete model increases the complexity of the linearization procedure considerably. The requirement on the smoothness of q_d is high using these control laws. If the manipulator has N links, q_d must be $2(N+1)$ times differentiable compared to four times for the simplified flexible joint model as described in Section 5.3.2.

5.3.4 State Estimation

All states must be available for feedback in order to use the control laws (5.8) and (5.10). However, the feedforward control law (5.10) can be combined with, e.g., a PD controller for the actuator position, which only requires the motor states to be available (De Luca, 2000). The motor position is available for all industrial manipulators considered, and the motor speed can be estimated by, e.g., differentiation of the motor position. This means that a simplified version of the feedforward control law is possible to evaluate on a standard industrial robot, if the required computational capacity is available.

The feedback linearization control law (5.8) requires the link position, speed, acceleration, and jerk for each link to be measured. This means that some of the states, e.g., the acceleration and jerk, must be estimated from the available measurements. If differentiation is used for estimating the higher order derivatives, the measurement noise is likely to reduce the performance. If link position, link speed, motor position, and motor speed are measured, the link acceleration and jerk can be computed by using the model (Siciliano and Khatib, 2008, Chapter 13). However, it is likely that the robustness and performance will be impaired, compared to full-state measurements, since the controller will depend on (uncertain) model parameters to a higher degree.

Estimation of states can be performed by observers. One well-known type of observer for linear systems is the Kalman filter (Anderson and Moore, 1979) where the observer gain is computed based on a stochastic description of the measurement- and system disturbances. For nonlinear systems, the extended Kalman filter, can be used. The gain of an observer with the same structure as the Kalman filter can also be determined by pole placement of the observer error dynamics. Another type of observer is the reduced observer, sometimes called Luenberger observer. Nonlinear observers are treated in, e.g., Isidori (1995) and Robertsson (1999). A state observer for the linear system

$$\begin{aligned}\dot{x} &= Ax + Bu,\\ y &= Cx\end{aligned}$$

can be expressed as

$$\dot{\hat{x}} = A\hat{x} + Bu + K(y - C\hat{x}),$$

where $\hat{x}$ is the estimated state vector and K is the observer gain.

A nonlinear observer for the motor position, motor speed, link position, and link speed is suggested in De Luca et al. (2007). The observer works for both the simplified and the complete flexible joint model, and requires measurements of motor position and link acceleration. The observer does not depend on the inertial link parameters but on the motor inertias, the joint spring-damper pairs, and the kinematic link parameters describing the accelerometer locations. For an N-link manipulator, M accelerometers are needed, with $M \geq N$. The observer is characterized by a linear and decoupled error dynamics. The observer is evaluated on the three main axes of an industrial robot, as well as in closed loop with the feedforward control law decribed in Section 5.3.2. The result is a significant improvement, with respect to damping and overshoot, compared to a control law using motor states only.

An experimental evaluation of observers for tool position estimation is described in Henriksson et al. (2009). Both the observer from De Luca et al. (2007) and the extended Kalman filter are evaluated with an emphasis on tuning and robustness. The Kalman filter showed better robustness, although a simple nonlinear deterministic observer is preferred for on-line control, due to the less expensive computations. More references on nonlinear observers for flexible joint robots can be found in De Luca et al. (2007).

5.3.5 Feedback Control

A feedback controller is needed to stabilize the system, reject disturbances, and to compensate for errors in the feedforward (if used). This section is a brief overview of feedback control for flexible joint robots. A general multivariable linear state feedback controller with integral action is

$$u = L(x_d - x_m) + L_i \int (x_d^i - x_m^i), \tag{5.13}$$

$L \in \mathbb{R}^{N \times 4N}$ is a linear feedback gain matrix, $x_d \in \mathbb{R}^{4N}$ are the reference states, and $x_m \in \mathbb{R}^{4N}$ are the measured or estimated states. $L_i \in \mathbb{R}^{N \times N}$ is a diagonal matrix of integral gains, and x^i are the states used by the integral term. Several state representations are possible, but here, it is assumed that the states are

$$x = \begin{bmatrix} q_m^T & q_a^T & \dot{q}_m^T & \dot{q}_a^T \end{bmatrix}^T. \tag{5.14}$$

For full-state multivariable control, the feedback gains L and L_i of (5.13) can be computed by LQ design (Anderson and Moore, 1990). If not all states are measured, a Kalman filter (Anderson and Moore, 1979) can be used for state estimation and combined with LQ design. The resulting controller is then called an LQG controller. LQ and LQG control are natural ways of approaching the flexible joint control problem as the design weights for the controlled variable (q_a) can be explicitly tuned. However, for uncertain and disturbed systems, the tuning of the state weights might not be intuitive, and the tuning of the Kalman filter covariances is even harder. The LQG approach for flexible joint robots is described in, e.g., Ferretti et al. (1998), Östring and Gunnarsson (1999) and Elmaraghy et al. (2002). The combination of LQG and disturbance observers is described in Ji et al. (2008). An alternative design method for state feedback and state estimation is pole placement (Kailath, 1980; Rugh, 1996; Kautsky et al., 1985). A similar multivariable controller can also be designed using H_∞ methods (Skogestad and Postlethwaite, 1996).

For a single axis flexible joint system, a general state feedback controller is

$$u = K_{pm}(\alpha_{pm}q_m^d - q_m) + K_{pa}(\alpha_{pa}q_a^d - q_a) + K_{vm}(\alpha_{vm}\dot{q}_m^d - \dot{q}_m) + \\ + K_{va}(\alpha_{va}\dot{q}_a^d - \dot{q}_a) + K_{im}\int(q_m^d - q_m) + K_{ia}\int(q_a^d - q_a), \tag{5.15}$$

where one of the integral gains must be zero to avoid saturation or oscillations due to the double integral parts. The gains α can be used for setpoint weighting (Åström and Hägglund, 2006). Note that, e.g., a PID controller for the motor variables often is implemented as a cascade controller, with an inner PI controller for speed control and an outer P controller for position control. If not all states are measured, the controller uses only partial state feedback. For the case of only motor side measurements, a PD controller for the motor position is suggested in Tomei (1991), i.e., all gains in (5.15) are zero except K_{pm}, α_{pm} and K_{vm}. In, Tomei (1991), it is also proved that the suggested controller globally stabilizes the manipulator around a reference position if gravity compensation is used. A practical controller for industrial systems also need an integral part on the motor side, i.e., $K_{im} \neq 0$. Another controller, suitable for relatively stiff systems, has all gains zero except K_{pa}, α_{pa}, K_{vm}, α_{vm}, and K_{ia}. Here, α_{vm} can also be zero if no speed reference is used. This controller combines high accuracy on the arm side with good damping. Using only the arm side measurements results in a low performance controller and the best control is of course obtained by using all states (if available) and with integral part on the arm side. More discussion on these issues can be found in Spong et al. (2006) and Siciliano and Khatib (2008, Chapter 13). To illustrate the effects of different feedback configurations, a single-link flexible joint robot (Figure 5.5) is used in the following configurations:

1. PD control with motor feedback, no speed reference (K_{pm}, α_{pm}, and K_{vm} used).

2. PD control with arm position- and motor speed feedback, no speed reference (K_{pa}, α_{pa}, and K_{vm} used).

3. PD control with arm feedback and lead filter on PD output, no speed reference (K_{pa}, α_{pa}, and K_{va} used).

4. Full state feedback, no speed references (K_{pa}, α_{pa}, K_{va}, K_{pm}, α_{pm}, K_{vm} used).

5. PD control with arm feedback, no speed reference (K_{pa}, α_{pa}, and K_{va} used).

The test consists of a filtered position reference step, followed by a pulse disturbance torque on the arm, and finally a pulse disturbance torque on the motor. The result is shown in Figures 5.6–5.8. PD control when measuring arm variables only, clearly results in a slow controller which is very sensitive to disturbances. However, with a more complex controller design, the performance can be improved. The use of a lead filter, that improves the phase by adding phase lead,

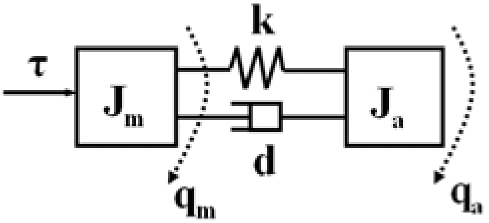

Figure 5.5: Linear model of a SISO flexible joint robot.

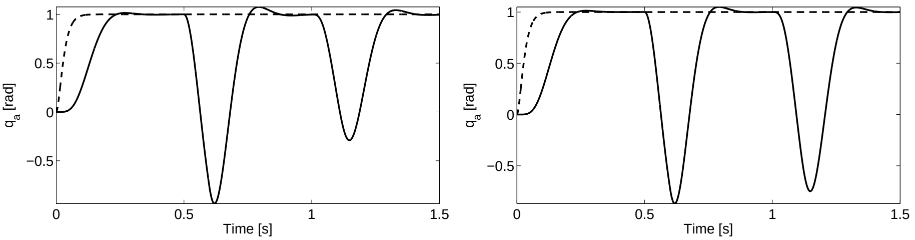

***Figure 5.6:** Arm position (solid) and position reference (dashed) for case* 1 *and* 2. 1*: Motor feedback (left).* 2*: Arm/Motor feedback (right). Arm disturbance at* 0.5 s*, motor disturbance at* 1.0 s*.*

is just an example to show this. Full state feedback gives the best result as expected. Note that the tunings are just examples (with an effort to be fair) for the different controller configurations. Effects of measurement noise and controller robustness have not been studied. However, from the tuning of the different controller configurations, it is clear that the configurations using arm feedback only (case 3 and 5), are not very robust. This can be explained by the concept of collocation.

A collocated system measures the response at the same location as the actuator input is applied, i.e., measuring the motor in our case, whereas a non-collocated system measures the response at another location, i.e., after the flexibility in our case. This can be illustrated by the benchmark model of Paper F (Figure 5.9). Figure 5.10 shows the frequency response from the control signal (motor torque) to the positions of the different masses. Clearly, the more flexibility between the torque input and the sensor, the more phase lag in the feedback loop.

Important aspects, often neglected in the robotics literature, are disturbance rejection and consequences of discrete-time implementation. This is the motivation for the two benchmark problems described in Papers F and G. The controllers presented up to now, must in practice most certainly be extended to deal with disturbances and stability issues. Loop shaping techniques, e.g., QFT (Horowitz, 1991), can be used for this part of the controller design.

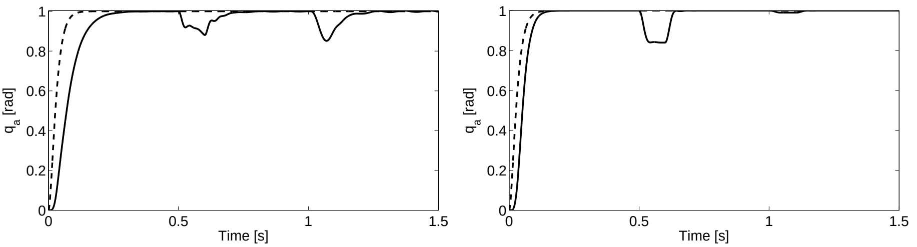

Figure 5.7: *Arm position (solid) and position reference (dashed) for case 3 and 4. 3: Arm feedback + lead filter (left). 4: Full state feedback (right). Arm disturbance at 0.5 s, motor disturbance at 1.0 s.*

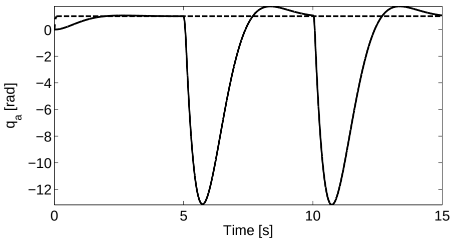

Figure 5.8: *Arm position (solid) and position reference (dashed) for case 5. Arm feedback. Arm disturbance at 5 s, motor disturbance at 10 s.* Note the scaling of the axes!

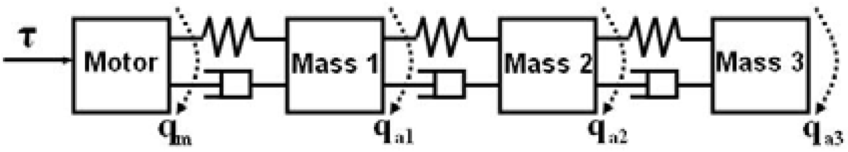

Figure 5.9: *The SISO benchmark model.*

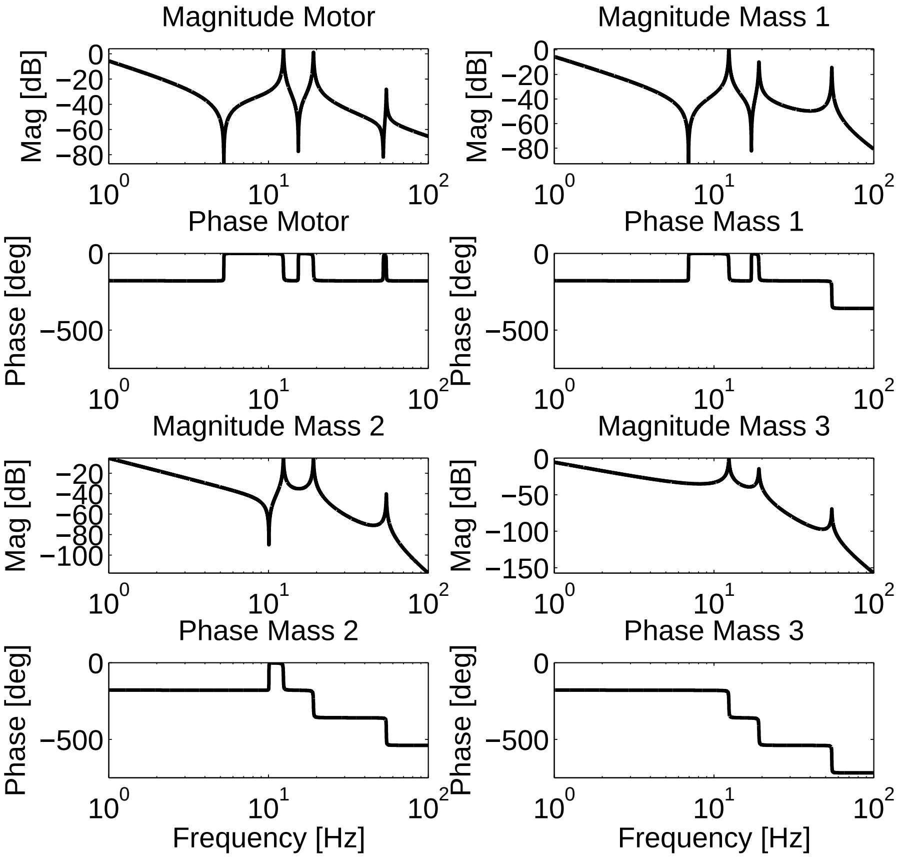

Figure 5.10: *Frequency response of the SISO benchmark model.*

5.3.6 Minimum-Time Control

Minimum-time control can also be used for generating the feedforward torque u_{ffw} and the references states $\bar{x}_d$ from Figure 5.1. The trajectory- and feedforward generation are here combined, as the trajectory reference z_d is also generated. In this section, minimum-time control is exemplified for a single-axis flexible joint manipulator.

Minimum-time control of robot manipulators can be divided into the minimum-time path-following problem and the minimum-time point-to-point problem. For the path-following problem, there are path constraints along the path and the robot is not allowed to deviate from the desired path. For the point-to-point problem, only the start and end positions are specified. There are two main approaches for solving minimum-time problems. Direct methods transcribe the problem to a nonlinear program (NLP) (Betts, 2001) and indirect methods solve the problem by solving the equations resulting from optimality conditions, e.g., the Pontryagin maximum principle (Bryson and Ho, 1975). The time-optimal control of rigid robot manipulators with path constraints is approached with direct methods in, e.g., Verscheure et al. (2009), and an indirect approach to the same problem is considered in, e.g, Shiller (1994). An article concerning the general problem of path-constrained trajectory optimization is Betts and Huffmann (1993). Dahl (1992) approaches the problem with direct and indirect methods and also shows some results for the flexible joint manipulator.

A linear model of a single-axis flexible joint robot arm (Figure 5.5) can be described by

$$J_m \ddot{q}_m = k(q_a - q_m) + d(\dot{q}_a - \dot{q}_m) + u, \tag{5.16a}$$

$$J_a \ddot{q}_a = -k(q_a - q_m) - d(\dot{q}_a - \dot{q}_m). \tag{5.16b}$$

The motor and arm inertia are described by J_m and J_a, respectively, the gear-box stiffness and damping are k and d, and the applied motor torque is u. The motor angular position is q_m and the arm angular position is q_a. The minimum-time problem for a point-to-point movement (constraint on control variable u only) can be solved using direct transcription. If the problem is discretized using M time-steps, sample-time h, states $x = \begin{bmatrix} q_m & q_a & \dot{q}_m & \dot{q}_a \end{bmatrix}^T$, movement distance Δq, and maximum allowed motor torque u_{max}, we get the following nonlinear program (NLP):

$$\begin{aligned} \min_{u(k),x(k),h} \sum_{k=1}^{M} h &= Mh, \\ \text{s.t.} \quad x(k+1) &= f(x(k), u(k), h), \\ x(1) &= \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}^T, \\ x(M) &= \begin{bmatrix} \Delta q & \Delta q & 0 & 0 \end{bmatrix}^T, \\ |u(k)| &\le u_{max}, \end{aligned}$$

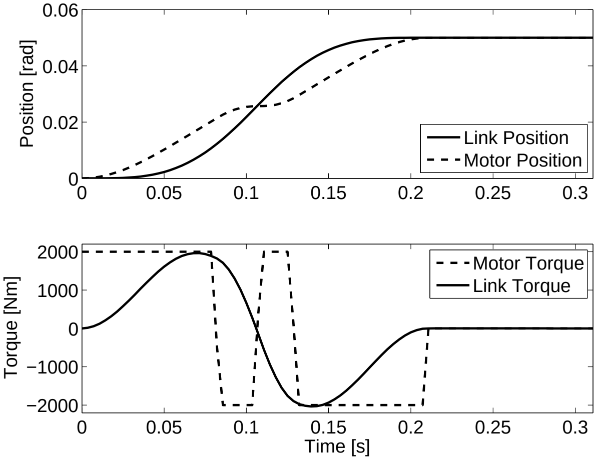

Figure 5.11: *Minimum-time control of single-link flexible joint manipulator with eigenfrequency* 5 *Hz and motor torque constraint* $|u(k)| \leq 2000$ *Nm. Movement time is* 0.21 *s.*

i.e., the problem is to minimize the sample time, which is a free parameter. The optimization parameters are $\begin{bmatrix} u(1), & \ldots, & u(M), & x(1), & \ldots, & x(M), & h \end{bmatrix}^T$. Figure 5.11–5.12 show the result for two different manipulator elasticities. The test case is a short movement of 0.05 rad which is discretized with $M = 60$ steps. The near minimum-time solution approaches bang-bang control with three switches. The manipulator eigenfrequency in the two cases are 5 Hz and 1 Hz, which result in movement times of 0.21 s and 0.45 s, respectively. The minimum time for the rigid system is 0.2 s. Clearly, the elasticity and the eigenfrequency should be taken into account for time-optimal trajectory generation as well as for robot design.

If a constraint on the link torque (i.e., gearbox torque $\tau_{gear} = k(q_a - q_m) + d(\dot{q}_a - \dot{q}_m)$) is added to the optimization problem, the solution approaches seven switches according to Figure 5.13. An approximative solution to the problem is to compute the minimum-time trajectory for the rigid system, i.e. bang-bang control with one switch (Bryson and Ho, 1975), and then use flexible joint feedforward control (5.10) with the rigid trajectory reference as input. However, this is not feasible since the feedforward torque depends on the second acceleration derivative, $q_{ad}^{[4]}$. One alternative is shown in Figure 5.14 where a rigid trajectory reference with a limited $q_{ad}^{[4]}$ is constructed, and combined with flexible joint feedforward control (Lambrechts et al., 2004). The movement time is somewhat larger for this approximative solution and the constraints cannot in general be guaranteed.

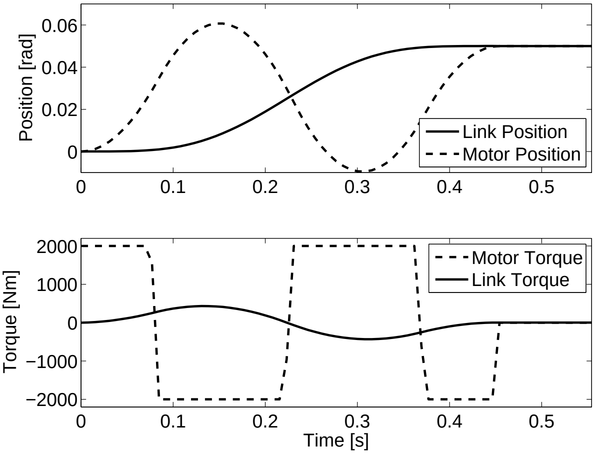

Figure 5.12: *Minimum-time control of single-link flexible joint manipulator with eigenfrequency* 1 *Hz and motor torque constraint* $|u(k)| \leq 2000$*. Movement time is* 0.45 *s.*

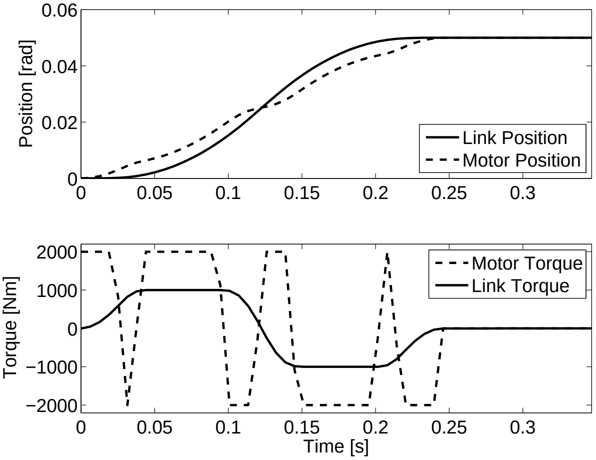

Figure 5.13: *Minimum-time control of single-link flexible joint manipulator with eigenfrequency* 5 *Hz, motor torque constraint* $|u(k)| \leq 2000$*, and gearbox torque constraint* $|\tau_{gear}(k)| \leq 1000$*. Movement time is* 0.25 *s.*

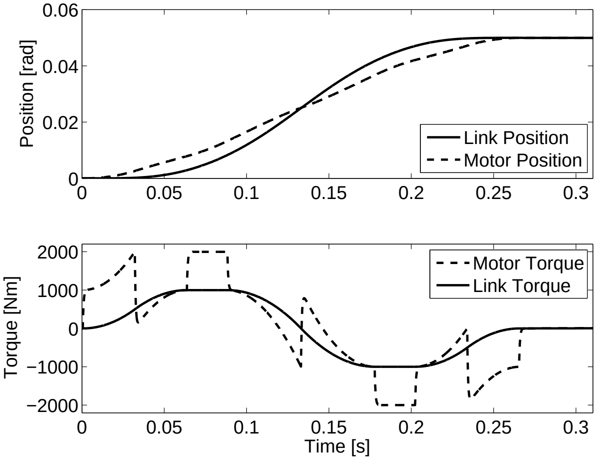

Figure 5.14: Approximate minimum-time control of single-link flexible joint manipulator with eigenfrequency 5 *Hz, motor torque constraint* $|u(k)| \leq 2000$*, and gearbox torque constraint* $|\tau_{gear}(k)| \leq 1000$*. Movement time is* 0.27 *s. The trajectory reference has a limited value of* $q_{ad}^{[4]}$ *and flexible joint feedforward is used.*

5.3.7 Experimental Evaluations

This section presents some reported experimental evaluations of control methods for flexible joint manipulators.

In Swevers et al. (1991), an industrial robot from KUKA is used for evaluation of an improved trajectory generation, and a model-based control concept. The robot was equipped with three extra encoders in order to measure the link positions of the main axes. The motor positions were measured by the standard controller. A state feedback controller combined with feedforward control based on a flexible joint model was evaluated and compared with the standard controller for this robot. The trajectory generation was also modified to yield a trajectory based on a 9th order polynomial. The performance of the standard- and model-based controllers were evaluated using some test cases[4] defined in the industrial robot test standard ISO 9283 (ISO, 1998). The tests showed significant improvements compared to the standard KUKA industrial controller implementation. The conclusion was that the smooth trajectory from the new trajectory generation contributed most, but also that the new flexible controller improved the performance at very high velocities and accelerations.

Jankowski and Van Brussel (1992b) mention some problems with feedback linearization, such as the complexity of the control laws, the need for measurement or estimation of link acceleration and jerk, and the need for high sampling fre-

[4] Settling time, overshoot, and path-following error according to the ISO standard were evaluated.

quencies. The suggested solution is a discrete-time formulation of the inverse dynamics which requires the solution of an index-3 differential algebraic equation. Some experimental results are presented in Jankowski and Van Brussel (1992a).

In Caccavale and Chiacchio (1994), an experimental evaluation on an industrial robot with gear transmission is reported. The robot is the SMART-3 6.12R robot from COMAU. A feedforward torque was added to the conventional PID controller output. Sample times were 1 ms and 10 ms for the PID controller and the feedforward computation, respectively. The feedforward torque calculations were based on a rigid dynamic model in identifiable form, as described in Section 4.2.1. The path error was decreased from 3.4 mm to 1.1 mm at a speed of 2 rad/s and an acceleration of 6 rad/s^2. It was concluded that the diagonal inertia terms are the most important terms for use in feedforward.

In Grotjahn and Heimann (2002), model-based control of a KUKA KR15 manipulator is described. It is concluded that the nonlinear multibody dynamics and the nonlinear friction are the dominating reasons for path deviation, and that the elasticity does not need to be considered. No torque interface to the controller was available. Instead a path correction interface is used together with a feedforward control method called nonlinear precorrection based on an identified rigid body model including friction. Improvement by learning control and training of the feedforward controller is discussed and evaluated. Learning control can compensate for all deviations whereas the feedforward training only compensates for the modeled effects. Thus, learning gives better performance but is very sensitive to path changes after the learning where the feedforward training is more roboust. The different algorithms improve the path-following in a path defined by the ISO 9283 standard.

A full-state feedback controller is presented in Albu-Schäffer and Hirzinger (2000). The motor position q_m and the joint output torque τ are measured for each joint. By numerical differentiation, the state vector $x = \begin{bmatrix} q_m^T & \dot{q}_m^T & \tau^T & \dot{\tau}^T \end{bmatrix}^T$ is obtained. The state feedback controller is diagonal, i.e., based on an independent joint approach that neglects the disturbance due to the strong coupling. Gravity and friction compensation is added to the controller, and gain scheduling based on the diagonal inertial terms is also suggested. A Lyapunov-based proof of global stability is given. The experimental evaluation is performed on a DLR light-weight robot and shows that the bandwidth of the proposed controller is twice the bandwidth of a motor PD controller, given the same damping requirement. This type of controller is further developed and analyzed in Le Tien et al. (2007) and Albu-Schäffer et al. (2007).

In Zhu (2007), a seven-axes robot with harmonic drives gearboxes is controlled by an adaptive joint-torque controller, where the friction parameters and the gearbox stiffness are continuously adapted. The outer position control loop is based on *virtual decomposition control*. Here, q_a, $\dot{q}_a$, $\ddot{q}_a$, q_m, $\dot{q}_m$, τ, and $\dot{\tau}$ are used by the controller. Only the positions and torques are measured, and the derivatives are estimated by numerical differentiation. The result was particulary good at ultralow speeds, due to the adaptive friction compensation. Virtual decomposition control is further described in Zhu (2010).

5.4 Control of Flexible Link Manipulators

This section describes some proposed control methods for the flexible link model described in Section 3.2.5. The choice of coordinates and reference frames is not unique and there is more than one possible approximate description of the same system (Book, 1993). If the actuator torque is applied at one point of the distributed structure, and the response is measured at another point, the system is non-collocated, and if the finite dimensional approximation of a beam-like system has a sufficiently large number of assumed modes, the system description will be non-minimum phase. This means that if the Cartesian end effector position is to be controlled, the system can be non-minimum phase, and the inverse dynamics is then hard to obtain. The control methods described in this section are in general feedforward control methods combined with a feedback controller of, e.g., PD-type.

If the desired Cartesian trajectory z_d is known for $t \in [0, T]$, the desired beam tip angles y_d, in Figure 3.11, can be computed by the inverse kinematics. In De Luca et al. (1998) it is shown that the state trajectories and the control torque can be obtained by solving an ordinary differential equation (ODE). The problem is to find a bounded solution to this ODE as the system is normally non-minimum phase, i.e., no bounded causal[5] solutions to the inverse dynamics exist. Three different methods for solving the problem are suggested, and one is experimentally evaluated. This method finds a non-causal solution by applying iterative learning control (ILC) for time $t \in [-\Delta, T + \Delta]$.

A different approach for the problem of point-to-point motion is described for a one-link manipulator in De Luca and Di Giovanni (2001a). Here, an auxiliary output is designed and used as output variable of the system. In this case the angle y_d points to a location where the system is minimum phase, but close to non-minimum phase. In De Luca and Di Giovanni (2001b) the same problem is solved for a two-link manipulator of which one link is flexible. A method based on dividing the inverse system in a causal and an anti-causal part is presented in Kwon and Book (1990). The method is limited to linear systems, i.e., one-link manipulators. In De Luca and Siciliano (1993) it is shown that a PD controller with gravity feedforward is globally asymptotically stable for the flexible link manipulator.

The robust control problem of a four-link flexible manipulator is treated in Wang et al. (2002). An H_∞ controller for regional pole-placement is designed for the uncertain linearized system. The controller is evaluated by simulation, and tests

[5] The output of a causal system depends only on past inputs, whereas the output of a non-causal system also depends on future inputs. Hence, a non-causal solution to the inverse dynamics, outputs a torque before the start of a movement.

on an experimental manipulator show that the proposed controller has better performance than an LQ controller.

To summarize, the control of the flexible link manipulator is complicated by the fact that the system can be non-minimum phase. There are several alternatives for solving the problem, e.g.,

- Find a stable non-causal solution of the inverse dynamics as described in the references of this section.
- Choose a new output so that the system becomes minimum phase, e.g., the joint angle for a flexible link robot. The new output should be a reasonable approximation of the original output.
- For a linear discrete-time system, the *zero phase tracking controller* (Torfs et al., 1991; Tung and Tomizuka, 1993) can be used.
- In Aguiar et al. (2005) a reformulation from tracking to path-following[6] is suggested. By parameterizing the geometrical path X by a path variable θ, and then selecting a timing law for θ, the inverse dynamics for the non-minimum phase system can be stabilized.

5.5 Industrial Robot Control

As described in Section 1.1, the control algorithms used by robot manufacturers are seldom published. This section gives a few examples of known facts about the control of commercial robots[7], taken from public information sources.

In Grotjahn and Heimann (2002) it is claimed that the feedback controller of a KUKA robot is a cascaded controller with an inner speed controller of PI type with sample time 0.5 ms. The outer loop is a position controller with 2 ms sample time. Only the position is measured, and the speed is estimated by differentiation and low-pass filtering. The controller can thus be described as a diagonal PID controller.

A sliding mode controller based on an elastic model of two-mass type is suggested in Nihei and Kato (1993) by Fanuc. The motivation is the reduction of vibrations in short point-to-point movements in, e.g., spotwelding applications. A patent by Fanuc (Nihei et al., 2007) describes a method for reduction of vibrations in robot manipulators. The method is based on observer estimation of the link position. The estimated variable is then used in a controller of internal model control (IMC) type. The controller is based on a linear elastic SISO model of two-mass type. On their web-site (Fanuc, 2007), the new Fanuc controller R-J3iC offers *enhanced vibration control* as a feature. Fanuc has recently applied for a patent

[6] The path-following error is in fact a more relevant performance measure for industrial applications as reflected by the ISO 9283 standard.

[7] The examples are restricted to the four major robot manufacturers, i.e., Fanuc, Motoman, ABB, and KUKA.

on arm position feedback, called secondary position feedback control (Tsai et al., 2010).

The motion control of ABB robots is described as model-based control, and the accuracy in continuous path tracking is claimed to be very high (Madesäter, 1995). The model-based controller is implemented in a controller functionality denoted *TrueMove*, and the time-optimal path generation is denoted *QuickMove*. An improved version of this functionality is called *The Second Generation of TrueMove and QuickMove* (ABB, 2007) and further described in Björkman et al. (2008).

Finally, Motoman has a control concept called *Advanced Robot Motion (ARM) control* for high-performance path accuracy and vibration control (Motoman, 2007).

Clearly, high performance motion control is important for the industrial robot manufacturers. The actual algorithms used are hard to reveal, and an article or a patent does not mean for sure that the described technology is actually used in the product.

It is also clear that the performance of industrial robots can be very high. This indicates that advanced concepts for motion control are used. Some examples of obtained performance for the "best in class" commercial robots during optimal-time movements:

- Path-following error of large robots with considerable link- and joint flexibilities and a payload of more than 200 kg can be in the order of 2 mm at 1.5 m/s according to the ISO 9283 standard, i.e., when the maximum path-following error is considered. The mean error is considerably smaller, about 0.5 mm.

- The corresponding errors for a medium-sized robot with a payload of 20 kg could be maximum 0.5 mm and mean 0.1 mm.

- A positioning time of 0.25 s is obtained for a 50 mm point-to-point movements for the large 200 kg payload robot described above. Time is measured from start of movement until the tool is inside an error band of 0.3 mm. This means that there are almost no vibrations or overshoots when reaching the final position.

- Dynamic position accuracy better than 0.15 mm considering maximum error, and with a mean value of 0.05 mm at 50 mm/s in laser-cutting applications when using iterative learning control (ILC).

5.6 Conclusion

Control of flexible joint and flexible link manipulators is a large research area with numerous publications, and almost every possible control method ever invented has been suggested for dealing with these systems. This survey has described only the main approaches.

It is clear that the theoretical foundation of the control methods, and the ability to prove stability, is in focus for many academic robot control researchers. Evaluation of nominal and robust performance of the proposed methods, is often neglected. From the reported simulation studies and experimental evaluations, it is in fact quite hard to judge what the attainable performance is for the different methods. It is true that it can be hard to use commercial robots for control evaluations, and that some methods require a larger computational capacity than currently available in present robot controllers. However, simulation studies are always possible to perform. One further comment is that an integral term is most certainly needed in order to handle model errors and disturbances in a real application. The reason for avoiding the integral term in many publications is probably motivated mainly by the need to prove stability.

The following facts regarding the tracking and point-to-point performance are at least indicated in the experimental results presented in this survey:

1. Model-based control improves the performance of an industrial-type robot.
2. A rigid model can improve the performance although the robot has elastic gear transmissions.
3. A flexible joint model improves the performance even more. It is not clear whether the nonlinear model should be used in the feedback loop, i.e., feedback linearization, or in the feedforward part of the controller.
4. More measurements improve the result, e.g., measurement of link position or acceleration.

6 Conclusion

This first part of the thesis has served as an introduction to modeling and control of robot manipulators. The aim has been to show how the included papers in Part II relate to the existing methods, and to motivate the need for the research presented. In Section 6.1 a summary of the results is given. Suggestions for future research in this area are discussed in Section 6.2.

6.1 Summary

This thesis has investigated aspects on modeling and control of elastic manipulators. The work is motivated by the industrial trend to develop weight- and cost-minimized robot manipulators. A large amount of applied research is needed in order to maintain and improve the motion performance of new generations of robot manipulators. The approach adopted in this thesis is to improve the model-based control by developing and validating more accurate elastic models. A model, called the *extended flexible joint model*, is suggested for use in motion control systems, as well as for robot design and performance simulation. Different aspects of this model, illustrated in Figure 6.1, are treated in this thesis.

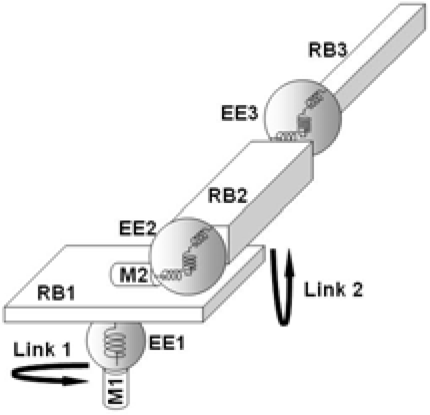

Figure 6.1: *A 9 DOF extended flexible joint model with 2 links, 2 motors (M), 3 elastic elements (EE) and 3 rigid bodies (RB).*

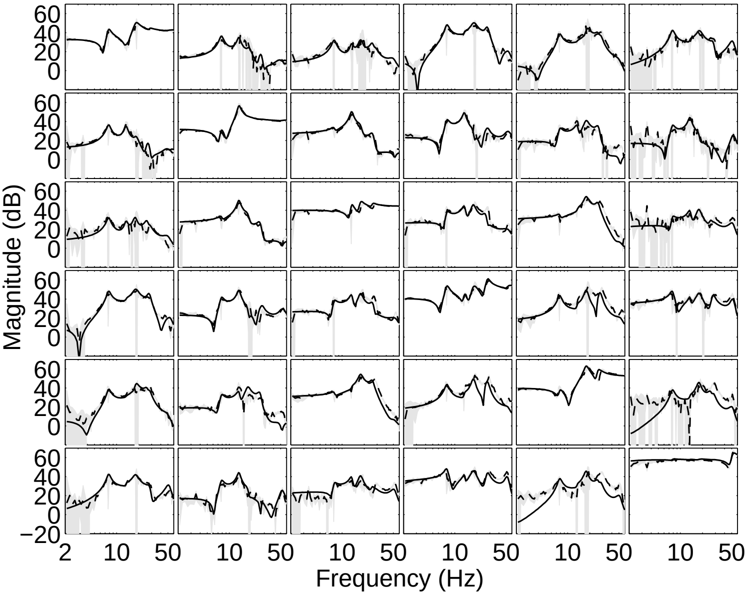

Figure 6.2: *The magnitude of a 6 × 6 frequency response function (FRF), describing the dynamics between applied motor torques and motor accelerations. Nonparametric FRF obtained from measurements (dashed) and parametric model FRF (solid).*

A procedure for multivariable identification of the unknown elastic parameters of the extended model is proposed and described in Papers A and B. The procedure is applied for identifying these parameters of a real six-axes industrial robot. Paper B gives a detailed description of the proposed frequency-domain gray-box method, and shows that the proposed method is an approximation of a time-domain prediction-error method. Paper A shows that the flexible joint model is insufficient for modeling a modern industrial manipulator accurately, and that the extended model significantly improves the model accuracy. An example of measured- and modeled frequency response functions is shown in Figure 6.2.

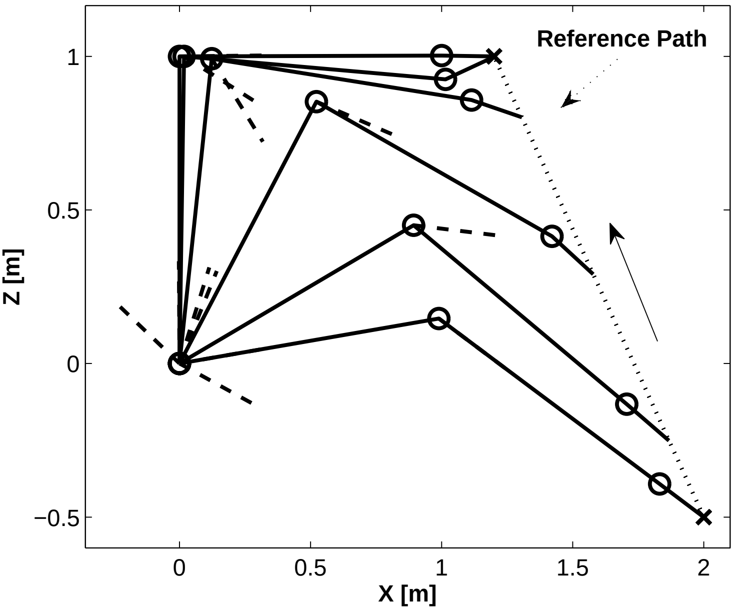

Figure 6.3: *Snapshots from an animation of a movement resulting from control using the inverse extended flexible joint model. The manipulator is described by a* 5 *DOF extended flexible joint model with* 2 *actuated joints and* 1 *non-actuated joint (the outer joint). The reference path is shown as a dotted line with the direction indicated by the solid arrow. The motor angular positions (transformed to the arm side) are indicated by the dashed lines, and the joint angular positions are indicated by the solid lines.*

In Papers C and D, the inverse dynamics of the extended model is derived and studied. The inverse dynamics solution can be used for feedforward control. It is shown that the inverse dynamics can be computed as the solution of a high-index differential algebraic equation (DAE). Different DAE solvers are suggested and evaluated, both for minimum phase and non-minimum phase systems. A simulation study is presented in Paper C, and in Paper D, the suggested concept for inverse dynamics is experimentally evaluated using an industrial robot manipulator. Figure 6.3 illustrates the inverse dynamics solution. The conclusion is that the extended flexible joint inverse dynamics method can improve the accuracy for manipulators with significant elasticities, that cannot be described by the flexible joint model.

Paper E investigates the discrete-time implementation of the feedback linearization approach for a realistic three-axes flexible joint robot model and compares the result with a feedforward approach. An example trajectory from the evaluation is shown in Figure 6.4. The conclusion is that feedforward control gives better performance and reduces the requirements on the controller- and sensor hardware.

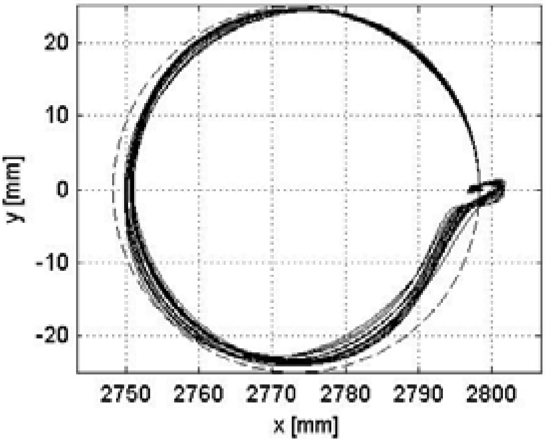

Figure 6.4: 20 *Monte Carlo simulations, where the uncertain robot executes a circular trajectory, using a feedback linearization controller.*

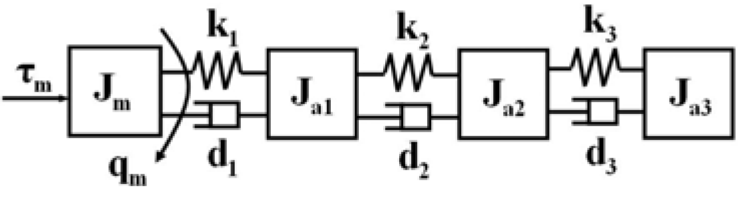

Figure 6.5: *The SISO Benchmark Robot Model.*

Robust feedback control of a one-axis four-mass model, illustrated in Figure 6.5, is studied in Paper F. The proposed SISO benchmark problem concerns disturbance rejection for the uncertain robot manipulator using a discrete-time controller. The benchmark model is validated by experiments on a real industrial manipulator. Several proposed solutions are presented and analyzed. The conclusion is that, for the uncertain SISO system using motor measurements only, it is hard to improve the result of a PID-controller. One proposed solution not included in Paper F is Varso et al. (2005).

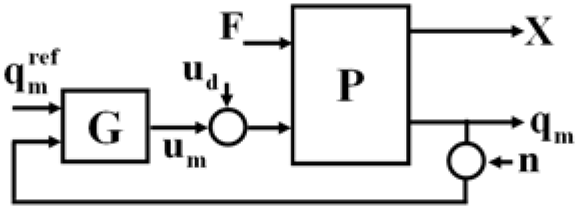

Figure 6.6: *The MIMO Benchmark System.*

Finally, Paper G presents a MIMO benchmark problem for a nonlinear two-link manipulator with elastic nonlinear gear transmissions, illustrated in Figure 6.6. This problem also concerns disturbance rejection for the uncertain robot manipulator using a discrete-time controller. The benchmark model is validated by experiments on a real industrial manipulator. Two solutions have been received for this benchmark problem:

A Solution A (unpublished) has a controller that is based on a nominal model with robot configuration dependence. The (SISO) controllers uses polynomial pole placement with additional closed loop poles to increase the robustness. The states are estimated by a Kalman filter. The resulting performance, using the proposed controller, is worse than the performance of the default PID controller included in the benchmark problem.

B Solution B is based on a robust frequency-domain method called QFD, which is an extension of quantitative feedback theory (QFT), see, Horowitz (1991). The solution is described in Yaniv and Pila (2009), and the resulting (SISO) controllers are filtered PD controllers in series with two complex lead filters. The controller has a slightly worse performance than the default PID controller.

To summarize, no solutions with better performance than the default PID controller has yet been received. The conclusion is that, for the uncertain MIMO system using motor measurements only, it is hard to improve the result of a diagonal PID-controller. However, no multivariable controller has yet been proposed.

6.2 Future Research

Based on the experience obtained from the work described in this thesis, several research-related questions have come up. Thus, it would have been interesting to proceed working on the following topics:

- **Modeling and Identification**
  - Identification methods with minimum energy, i.e., minimum experiment time and minimum amplitude.
  - Identification of transmission nonlinearities.
  - Automatic frequency-weight selection.
  - Automatic gray-box model structure selection.
  - Identification of elasticity models with more DOF than described in this thesis.
  - Identification based on both motor position and arm side measurements. Some possibilities are position, speed, or acceleration for the joints/links and/or the tool. This will include investigations of the optimal accelerometer/gyro placement, in combination with optimal configurations for identification.
- **Feedforward Control**
  - Experimental evaluation of feedforward control based on the extended flexible joint model, using a more complicated and realistic robot, e.g., a six-axes industrial robot.
  - Extended theoretical analysis of the inverse dynamics problem (e.g., solvability, controllability and uniqueness).
  - Development of more efficient DAE solvers.
  - Development of alternative solvers for non-minimum phase dynamics.
  - Development of approximate control algorithms for on-line computation.
- **Feedback Control**
  - Design of a MIMO controller for the MIMO benchmark problem, including experimental evaluation.
  - Further development of robust feedback control using additional arm sensors, e.g., arm side encoders, accelerometers, or gyros.

Bibliography

ABB. IRB 6640 - ABB announce a new stronger robot - the next generation. `http://www.abb.com/cawp/seitp202/ce7d8060f91e36e4c125736a0024afb5.aspx`, 2007. 2007-12-03.

A.P. Aguiar, J.P. Hespanha, and P.V. Kokotivic. Path-following for nonminimum phase systems removes performance limitations. *IEEE Transactions on Automatic Control*, 50(2):234–239, 2005.

F. Al-Bender, V. Lampaert, , and J. Swevers. The generalized maxwell-slip model: A novel model for friction simulation and compensation. *IEEE Transactions on Automatic Control*, 50(11):1883–1887, 2005.

A. Albu-Schäffer and G. Hirzinger. Parameter identification and passivity based joint control for a 7DOF torque controlled light weight robot. In *Proc. 2001 IEEE International Conference on Robotics and Automation*, pages 2852–2858, Seoul, Korea, May 2001.

A. Albu-Schäffer and G. Hirzinger. State feedback controller for flexible joint robots: A globally stable approach implemented on DLR's light-weight robots. In *Proceedings of the 2000 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1087–1093, Takamatsu, Japan, October 2000.

A. Albu-Schäffer, C. Ott, and G. Hirzinger. A unified passivity-based control framework for position, torque and impedence control of flexible joint robots. *The International Journal of Robotics Research*, 26(1):23–39, 2007.

C.H. An, C.G Atkeson, and J.M. Hollerbach. *Model-Based Control of a Robot Manipulator*. The MIT press, Cambridge, Massachusetts, 1988.

B. Anderson and J.B. Moore. *Optimal Filtering*. Dover Publications, Inc, Mineola, New York, 1979.

B. Anderson and J.B. Moore. *Optimal Control: Linear Quadratic Methods*. Prentice-Hall, Englewood Cliffs, NJ, USA, 1990.

B. Armstrong-Hélouvry. *Control of Machines with Friction*. Kluwer Academic Publishers, Norwell, Massachusetts, USA, 1991.

K.J. Åström. The future of control. *Modeling, Identification and Control*, 15(3): 127–134, 1994.

K.J. Åström and T. Hägglund. *Advanced PID Control*. ISA - The Instrumentation, Systems and Automation Society, Research Triangle Park, NC, USA, 2006. ISBN 1-55617-942-1.

K.J. Åström and B. Wittenmark. *Computer-Controlled Systems: Theory and Design*. Prentice Hall, Englewood Cliffs, New Jersey, USA, 1996.

P. Avitabile. Experimental modal analysis - a simple non-mathematical presentation. *Sound and vibration*, 35(1):20–31, January 2001.

L. Bascetta and P. Rocco. Modelling flexible manipulators with motors at the joints. *Mathematical and Computer Modelling of Dynamical Systems*, 8(2): 157–183, 2002.

F. Behi and D. Tesar. Parametric identification for industrial manipulators using experimental modal analysis. *IEEE Transactions on Robotics and Automation*, 7(5):642–52, 1991.

A. Bellini, G. Figalli, and G. Ulivi. Sliding mode control of a direct drive robot. In *Conference Record of the IEEE Industry Application Society Annual Meeting, vol.2*, pages 1685–1692, San Diego, CA, USA, 1989.

M. Benosman and G. Le Vey. Control of flexible manipulators: A survey. *Robotica*, 22:533–545, 2004.

E. Berglund and G. E. Hovland. Automatic elasticity tuning of industrial robot manipulators. In *39th IEEE Conference on Decision and Control*, pages 5091–5096, Sydney, Australia, December 2000.

D.S. Bernstein. On bridging the theory/practise gap. *IEEE Control Systems Magazine*, 19(6):64–70, 1999.

J. T. Betts. *Practical Methods for Optimal Control Using Nonlinear Programming*. Society for Industrial and Applied Mathematics, Philadelphia, 2001.

J. T. Betts and W. P. Huffmann. Path-constrained trajectory optimization using sparse sequential quadratic programming. *Journal of Guidance, Control, and Dynamics*, 16(1):59–68, 1993.

M. Björkman, T. Brogårdh, S. Hanssen, S.-E. Lindström, S. Moberg, and M. Norrlöf. A new concept for motion control of industrial robots. In *Proceedings of 17th IFAC World Congress, 2008*, Seoul, Korea, July 2008.

W. Book and K. Obergfell. Practical models for practical flexible arms. In *Proc. 2000 IEEE International Conference on Robotics and Automation*, pages 835–842, San Fransisco, CA, 2000.

W.J. Book. Controlled motion in an elastic world. *Journal of Dynamic Systems Measurement and Control, Transactions of the ASME*, 115:252–261, 1993.

T. Brogårdh. Present and future robot control development—an industrial perspective. *Annual Reviews in Control*, 31(1):69–79, 2007.

T. Brogårdh. Robot control overview: An industrial perspective. *Modeling, Identification and Control MIC*, 30(3):167–180, 2009.

T. Brogårdh and S. Moberg. Method for determining load parameters for a manipulator. US Patent 6343243, Januari 2002. URL `http://www.patentstorm.us/patents/6343243.html`.

T. Brogårdh, S. Moberg, S. Elfving, I. Jonsson, and F. Skantze. Method for supervision of the movement control of a manipulator. US Patent 6218801, April 2001. URL `http://www.patentstorm.us/patents/6218801`.

A. E. Bryson and Y.-C. Ho. *Applied Optimal Control: Optimization, Estimation, and Control.* Taylor and Francis, 1975.

F. Caccavale and P. Chiacchio. Identification of dynamic parameters and feedforward control for a conventional industrial manipulator. *Control Eng. Practice*, 2(6):1039–1050, 1994.

C. Canudas de Wit, H. Olsson, K.J. Åström, and P. Lischinsky. A new model for control of systems with friction. *IEEE Transactions on Automatic Control*, 40 (3):419–425, 1995.

A. Carvalho Bittencourt, E. Wernholt, S. Sander-Tavallaey, and T. Brogårdh. An extended friction model to capture load and temperature effects in robot joints. In *Proceedings of the 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Taipei, Taiwan, October 2010.

J.J. Craig. *Adaptive Control of Mechanical Manipulators.* Addison-Wesley Publishing Company, Menlo Park, California, USA, 1988.

J.J. Craig. *Introduction to Robotics Mechanics and Control.* Addison Wesley, Menlo Park, California, USA, 1989.

O. Dahl. *Path Constrained Robot Control.* PhD thesis, Lund Institue of Technology, SE-221 00 Lund, Sweden, 1992.

A. De Luca. Feedforward/feedback laws for the control of flexible robots. In *Proceedings of the 2000 IEEE International Conference on Robotics and Automation*, pages 233–240, San Francisco, CA, April 2000.

A. De Luca. Dynamic control of robots with joint elasticity. In *Proceedings of the 1988 IEEE International Conference on Robotics and Automation*, pages 152–158, Philadelphia, PA, 1988.

A. De Luca and G. Di Giovanni. Rest-to-rest motion of a one-link flexible arm. In *2001 IEEE/ASME International Conference on Advanced Intelligent Mechatronics Proceedings*, pages 923–928, Como, Italy, 2001a.

A. De Luca and G. Di Giovanni. Rest-to-rest motion of a two-link robot with a flexible forearm. In *2001 IEEE/ASME International Conference on Advanced Intelligent Mechatronics Proceedings*, pages 929–935, Como, Italy, 2001b.

A. De Luca and L. Lanari. Robots with elastic joints are linearizable via dynamic feedback. In *34th IEEE Conference on Decision and Control*, pages 3895–3897, New Orleans, LA, 1995.

A. De Luca and B. Siciliano. Closed-form dynamic model of planar multilink lightweight robots. *IEEE Transactions on Systems, Man, and Cybernetics*, 21 (4):826–839, 1991.

A. De Luca and B. Siciliano. Regulation of flexible arms under gravity. *IEEE Transactions on Robotics and Automation*, 9(4):463–467, 1993.

A. De Luca, S. Panzieri, and G. Ulivi. Stable inversion control for flexible link manipulators. In *Proc. 1998 IEEE International Conference on Robotics and Automation*, pages 799–804, Leuven, Belgium, May 1998.

A. De Luca, R. Farina, and P. Lucibello. On the control of robots with visco-elastic joints. In *Proc. 2005 IEEE International Conference on Robotics and Automation*, pages 4297–4302, Barcelona, Spain, 2005.

A. De Luca, D. Schröder, and M. Thummel. An acceleration-based state observer for robot manipulators with elastic joints. In *Proc. 2007 IEEE International Conference on Robotics and Automation*, pages 3817–3823, Roma, Italy, April 2007.

R. Dhaouadi, F. H. Ghorbel, and P. S. Gandhi. A new dynamic model of hysteresis in harmonic drives. *IEEE Transactions on Industrial Electronics*, 50(6):1165–1171, 2003.

H.A. Elmaraghy, T. Lahdhiri, and F. Ciuca. Robust linear control of flexible joint robot systems. *Journal of Intelligent and Robotic Systems: Theory and Applications*, 34(4):335–356, 2002.

Fanuc. R-30iA - Product detail. `http://www.fanucrobotics.cz/products/robots/controller.asp?idp=317&id=13`, 2007. 2007-12-03.

B. Feeny and F.C. Moon. Chaos in a forced dry-friction oscillator: Experiments and numerical modelling. *Journal of Sound and Vibration*, 170(3):303–323, 1994.

G. Ferretti, G. Magnani, and P. Rocco. LQG control of elastic servomechanisms based on motor position measurements. In *AMC'98 - Coimbra. 1998 5th International Workshop on Advanced Motion Control. Proceedings*, pages 617–622, Coimbra, Portugal, 1998.

H. Goldstein. *Classical Mechanics*. Addison-Wesley Publishing Company, Reading, Massachusetts, USA, 1980.

M. Grotjahn and B. Heimann. Model-based feedforward control in industrial robots. *The International Journal of Robotics Research*, 21(1):45–60, 2002.

S. Gunnarsson, M. Norlöf, G. Hovland, U. Carlsson, T. Brogårdh, T. Svensson, and S. Moberg. Pathcorrection for an industrial robot. US Patent 7130718, October 2006. URL `http://www.patentstorm.us/patents/7130718.html`.

T. Hardeman. *Modelling and Identification of Industrial Robots including Drive and Joint Flexibilities*. Phd thesis, University of Twente, The Netherlands, February 2008.

G.G. Hastings and W.J. Book. A linear dynamic model for flexible robotic manipulators. *IEEE Control Systems Magazine*, 7(1):61–64, 1987.

R. Henriksson, M. Norrlöf, S. Moberg, E. Wernholt, and T. Schön. Experimental comparison of observers for tool position estimation of industrial robots. In *Proceedings of 48th IEEE Conference on Decision and Control*, pages 8065–8070, Shanghai, China, December 2009.

I. Horowitz. Survey of quantitative feedback theory (QFT). *International Journal of Control*, 53(2):255–291, 1991.

G. E. Hovland, E. Berglund, and S. Hanssen. Identification of coupled elastic dynamics using inverse eigenvalue theory. In *32nd International Symposium on Robotics (ISR)*, pages 1392–1397, Seoul, Korea, April 2001.

G.E. Hovland, E. Berglund, and O.J. Sørdalen. Identification of joint elasticity of industrial robots. In *Proceedings of the 6th International Symposium on Experimental Robotics*, pages 455–464, Sydney, Australia, March 1999.

G.E. Hovland, S. Hanssen, E. Gallestey, S. Moberg, T. Brogårdh, S. Gunnarsson, and M. Isaksson. Nonlinear identification of backlash in robot transmissions. In *Proc. 33rd ISR (International Symposium on Robotics)*, Stockholm, Sweden, October 2002.

IFR. International federation of robotics - statistics 2010. `http://www.ifr.org/uploads/media/2010_Executive_Summary_rev.pdf`, 2010.

A. Isidori. *Nonlinear Control Systems*. Springer-Verlag, London, Great Britain, 1995.

ISO. ISO 9283:1998, manipulating industrial robots - performance criteria and related test methods. `http://www.iso.org`, 1998.

K.P. Jankowski and H. Van Brussel. An approach to discrete inverse dynamics control of flexible-joint robots. *IEEE Transactions on Robotics and Automation*, 8(5):651–658, October 1992a.

K.P. Jankowski and H. Van Brussel. An approach to discrete inverse dynamics control of flexible-joint robots. *Journal of Dynamic Systems, Measurement, and Control*, 114:229–233, June 1992b.

H. Jerregård and N. Pihl. Method for controlling a robot. US Patent 7209802, April 2007. URL `http://www.patentstorm.us/patents/7209802.html`.

C.-Y. Ji, T.-C. Chen, and Y.-L. Lee. Joint control for flexible-joint robot with input-estimation approach and LQG method. *Optimal Control Applications and Methods*, (29):101–125, 2008.

R. Johansson. *System Modeling and Identification*. Prentice Hall, 1993.

R. Johansson, A. Robertsson, K. Nilsson, and M. Verhaegen. State-space system identification of robot manipulator dynamics. *Mechatronics*, 10(3):403–418, 2000.

T. Kailath. *Linear Systems*. Prentice Hall, Englewood Cliffs, New Jersey, USA, 1980.

T. R. Kane and D. A. Levinson. The use of Kane's dynamical equations in robotics. *International Journal of Robotics Research*, 2(3), 1983.

T. R. Kane and D. A. Levinson. *Dynamics: Theory and Applications*. McGraw-Hill Publishing Company, 1985.

J. Kautsky, N.K. Nichols, and P. Van Dooren. Robust pole assignment in linear state feedback. *International Journal of Control*, 41(5):1129–1155, 1985.

P.K. Khosla and T. Kanade. Real-time implementation and evaluation of computed-torque scheme. *IEEE Transactions on Robotics and Automation*, 5 (2):245–253, 1989.

P. Kokotovic and M. Arcak. Constructive nonlinear control: a historical perspective. *Automatica*, 37:637–662, 2001.

K. Kozlowski. *Modelling and identification in robotics*. Advances in Industrial Control. Springer, London, 1998.

D.-S. Kwon and W.J. Book. An inverse dynamic method yielding flexible manipulator state trajectories. In *Proceedings of the 1990 American Control Conference, vol 1*, pages 186–193, San Diego, CA, USA, 1990.

P. Lambrechts, M. Boerlage, and M. Steinbuch. Trajectory planning and feedforward design for high performance motion systems. In *Proceedings of the 2004 American Control Conference*, Boston, Massachusetts, 2004.

L. Le Tien, A. Albu-Schäffer, and G. Hirzinger. Mimo state feedback controller for a flexible joint robot with strong joint coupling. In *Proc. 2007 IEEE International Conference on Robotics and Automation*, pages 3824–3830, Roma, Italy, April 2007.

M. Lesser. *The Analysis of Complex Nonlinear Mechanical Systems: A Computer Algebra assisted approach*. World Scientific Publishing Co Pte Ltd, Singapore, 2000.

L. Ljung. *System Identification: Theory for the User*. Prentice Hall, Upper Saddle River, New Jersey, USA, 2nd edition, 1999.

L. Ljung and T. Glad. *Control Theory*. CRC Press, 2001. ISBN 0748408789.

Å. Madesäter. Faster and more accurate industrial robots. *Industrial Robot*, 22 (2):14–15, 1995.

S. Moberg. Robust control of a multivariable nonlinear flexible manipulator - a benchmark problem. `www.robustcontrol.org`, 2007.

S. Moberg and S. Hanssen. A DAE approach to feedforward control of flexible manipulators. In *Proc. 2007 IEEE International Conference on Robotics and Automation*, pages 3439–3444, Roma, Italy, April 2007.

S. Moberg and S. Hanssen. On feedback linearization for robust tracking control of flexible joint robots. In *Proc. 17th IFAC World Congress*, Seoul, Korea, July 2008.

S. Moberg and S. Hanssen. Inverse dynamics of flexible manipulators. In *Multibody Dynamics 2009*, Warsaw, Poland, July 2009.

S. Moberg and S. Hanssen. Inverse dynamics of robot manipulators using extended flexible joint models. 2010. *Submitted to IEEE Transactions on Robotics (under revision).*

S. Moberg and J. Öhr. Svenskt mästerskap i robotreglering. In Swedish Control Meeting 2004, Göteborg, Sweden*, May 26-27 2004.*

S. Moberg and J. Öhr. Robust control of a flexible manipulator arm: A benchmark problem. Prague, Czech Republic, 2005. 16th IFAC World Congress.

S. Moberg, J. Öhr, and S. Gunnarsson. A benchmark problem for robust control of a multivariable nonlinear flexible manipulator. In Proc. 17th IFAC World Congress*, Seoul, Korea, July 2008.*

S. Moberg, J. Öhr, and S. Gunnarsson. A benchmark problem for robust feedback control of a flexible manipulator. IEEE Transactions on Control Systems Technology*, 17(6):1398–1405, November 2009.*

S. Moberg, E. Wernholt, S. Hanssen, and T. Brogårdh. Modeling and parameter estimation of robot manipulators using extended flexible joint models. 2010. Submitted to Journal of Dynamic Systems Measurement and Control, Transactions of the ASME.

Motoman. Introducing NX100 - the Next Generation of Robot Controller. `http://www.motoman.co.uk/NX100.htm`, 2007. 2007-12-03.

M.J. Van Nieuwstadt and R.M. Murray. Real-time trajectory generation for differentially flat systems. *International Journal of Robust and Nonlinear Control*, 8 (11):995–1020, 1998.

R. Nihei and T. Kato. Servo control for robot. In *24th International Symposium on Industrial Robots (ISIR), Tokyo*, Tokyo, Japan, 1993.

R. Nihei, T. Kato, and S Arita. Vibration control device. US Patent 7181294, February 2007. URL `http://www.patentstorm.us/patents/7181294.html`.

M. Norrlöf. *Iterative Learning Control: Analysis, Design, and Experiments*. PhD thesis, Linköping University, SE-581 83 Linköping, Sweden, 2000.

J. Öhr, S. Moberg, E. Wernholt, S. Hanssen, J. Pettersson, S. Persson, and S. Sander-Tavallaey. Identification of flexibility parameters of 6-axis industrial manipulator models. In *Proc. ISMA2006 International Conference on Noise and Vibration Engineering*, pages 3305–3314, Leuven, Belgium, September 2006.

H. Olsson. *Control Systems with Friction*. PhD thesis, Lund Institue of Technology, SE-221 00 Lund, Sweden, 1996.

A.V. Oppenheim and R.W. Schafer. *Digital Signal Processing*. Prentice Hall, Englewood Cliffs, New Jersey, USA, 1975.

M. Östring and S. Gunnarsson. LQG control of a flexible servo. In *Second conference on Computer Science and Systems Engineering in Linköping, CCSSE99*, pages 125–132, October 1999.

M. Östring, S. Gunnarsson, and M. Norrlöf. Closed-loop identification of an industrial robot containing flexibilities. *Control Engineering Practice*, 11:291–300, March 2003.

M. T. Pham, M. Gautier, and Ph. Poignet. Identification of joint stiffness with bandpass filtering. In *2001 IEEE International Conference on Robotics and Automation*, pages 2867–72, Seoul, Korea, May 2001.

M. T. Pham, M. Gautier, and Ph. Poignet. Accelerometer based identification of mechanical systems. In *2002 IEEE International Conference on Robotics and Automation*, pages 4293–4298, Washington, DC, May 2002.

R. Pintelon and J. Schoukens. *System identification: a frequency domain approach*. IEEE Press, New York, 2001.

D.B. Ridgely and M.B. McFarland. Tailoring theory to practise in tactical missile control. *IEEE Control Systems Magazine*, 19(6):49–55, 1999.

A. Robertsson. *On Observer-Based Control of Nonlinear Systems*. PhD thesis, Dept. of Automatic Control, Lund Institue of Technology, SE-221 00 Lund, Sweden, 1999.

P. Rouchon, M. Fliess, J. Lévine, and P. Martin. Flatness, motion planning and trailer systems. In *Proceedings of the 32nd Conference on Decision and Control*, pages 2700–2705, San Antonio, Texas, December 1993.

M. Ruderman, F. Hoffmann, and T. Bertram. Modeling and identification of elastic robot joints with hysteresis and backlash. *IEEE Transactions on Industrial Electronics*, 56(10):3840–3847, 2009.

W.J. Rugh. *Linear System Theory*. Prentice Hall, Upper Saddle River, New Jersey, USA, 1996.

H.G. Sage, M.F. De Mathelin, and E. Ostertag. Robust control of robot manipulators: A survey. *International Journal of Control*, 72(16):1498–1522, 1999.

V. Santibanez and R. Kelly. PD control with feedforward compensation for robot manipulators: analysis and experimentation. *Robotica*, 19:11–19, 2001.

L. Sciavicco and B. Siciliano. *Modeling and Control of Robotic Manipulators*. Springer, London, Great Britain, 2000.

A.A. Shabana. *Dynamics of Multibody Systems*. Cambridge University Press, Cambridge, United Kingdom, 1998.

Z. Shiller. Time-energy optimal control of articulated systems with geometric path constraints. In *Proc. 1994 IEEE International Conference on Robotics and Automation*, pages 2680–2685, San Diego, CA, 1994.

B. Siciliano and O. Khatib, editors. *Springer Handbook of Robotics*. Springer-Verlag, Berlin Heidelberg, 2008.

B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo. *Robotics - Modelling, Planning and Control*. Springer, London, Great Britain, 2010.

S. Skogestad and I. Postlethwaite. *Multivariable Feedback Control*. Wiley, New York, 1996.

J.-J. Slotine and W. Li. *Applied Nonlinear Control*. Prentice Hall, Englewood Cliffs, New Jersey, USA, 1991.

T. Söderström and P. Stoica. *System Identification*. Prentice-Hall Int., London, Great Britain, 1989.

M. W. Spong. Modeling and control of elastic joint robots. *Journal of Dynamic Systems, Measurement, and Control*, 109:310–319, December 1987.

M. W. Spong, S. Hutchinson, and M. Vidyasagar. *Robot Modeling and Control*. Wiley, 2006.

M.W. Spong. *Handbook of Control*. CRC Press, 1996. Chapter, Motion Control of Robot Manipulators, page 1339–1350.

J. Swevers, D. Torfs, M. Adams, J. De Schutter, and H. Van Brussel. Comparison of control algorithms for flexible joint robots implemented on a KUKA ir 161/60 industrial robot. In *91 ICAR. Fifth Conference on Advanced Robotics. Robots in Unstructured Environments.*, Pisa, Italy, 1991.

J. Swevers, W. Verdonck, and J. De Schutter. Dynamic model identification for industrial robots. *IEEE Control Systems Magazine*, pages 58–71, October 2007.

P. Tomei. A simple PD controller for robots with elastic joints. *IEEE Transactions on Automatic Control*, 36(10):1208–1213, 1991.

D. Torfs, J. Swevers, and J. De Schutter. Quasi-perfect tracking control of non-minimal phase systems. In *46th IEEE Conference on Decision and Control*, pages 241–244, Brighton, England, 1991.

J. Tsai, E. Wong, J. Tao, H. D. McGee, and H. Akeel. Secondary position feedback control of a robot. US Patent Application US2010191374, July 2010. URL `http://v3.espacenet.com/`.

E.D. Tung and M. Tomizuka. Feedforward tracking controller design based on the identification of low frequency dynamics. *Journal of Dynamic Systems Measurement and Control, Transactions of the ASME*, 115:348–356, 1993.

T.D. Tuttle and W.P. Seering. A nonlinear model of a harmonic drive gear transmission. *IEEE Transactions on Robotics and Automation*, 12(3):368–374, 1996.

J. Varso, K. Zenger, V. Hölttä, and H. Koivo. Advanced solution for a benchmark robot control problem. In *Proc. IEEE International Symposium on Computational Intelligence in Robotics and Automation*, pages 373–378, Helsinki, Finland, 2005.

D. Verscheure, B. Demeulenaere, J. Swevers, J. De Schutter, and M. Diehl. IEEE transactions on automatic control. *Time-Optimal Path Tracking for Robots: A Convex Optimization Approach*, 54(10):2318–2327, 2009.

Z. Wang, H. Zeng, D.W.C. Ho, and H. Unbehauen. Multiobjective control of a four-link flexible manipulator: A robust H-infinity approach. *IEEE Transactions on Control Systems Technology*, 10(6):866–875, 2002.

E. Wernholt. *Multivariable Frequency-Domain Identification of Industrial Robots*. PhD thesis, Linköping University, SE-581 83 Linköping, Sweden, 2007.

E. Wernholt and S. Gunnarsson. Nonlinear identification of a physically parameterized robot model. In *Proc. 14th IFAC Symposium on System Identification*, pages 143–148, Newcastle, Australia, March 2006.

E. Wernholt and S. Moberg. Frequency-domain gray-box identification of industrial robots. In *17th IFAC World Congress*, pages 15372–15380, Seoul, Korea, July 2008a.

E. Wernholt and S. Moberg. Experimental comparison of methods for multivariable frequency response function estimation. In *17th IFAC World Congress*, pages 15359–15366, Seoul, Korea, July 2008b.

E. Wernholt and S. Moberg. Nonlinear gray-box identification using local models applied to industrial robots. *Automatica*, 2010. Accepted for publication.

O. Yaniv and A. Pila. QFD of a multivariable nonlinear flexible manipulator: Benchmark problem. In *Proc. of the 6th IFAC Symposium on Robust Control Design*, Haifa, Israel, 2009.

W.-H. Zhu. Precision control of robots with harmonis drives. In *Proc. 2007 IEEE International Conference on Robotics and Automation*, pages 3831–3836, Roma, Italy, April 2007.

W.-H. Zhu. *Virtual Decomposition Control: Toward Hyper Degrees of Freedom Robots*. Springer-Verlag, Berlin Heidelberg, 2010.