Topics in Machining with Industrial Robot Manipulators and Optimal Motion Control

Olofsson, Björn

2015

Document Version:
Publisher's PDF, also known as Version of record

Topics in Machining with Industrial Robot Manipulators and Optimal Motion Control

Björn Olofsson

LUND
UNIVERSITY

Department of Automatic Control

PhD Thesis
ISRN LUTFD2/TFRT--1108--SE
ISBN 978-91-7623-434-1 (print)
ISBN 978-91-7623-435-8 (web)
ISSN 0280–5316

Department of Automatic Control
Lund University
Box 118
SE-221 00 LUND
Sweden

Printed in Sweden.
Lund 2015

Abstract

Two main topics are considered in this thesis: Machining with industrial robot manipulators and optimal motion control of robots and vehicles. The motivation for research on the first subject is the need for flexible and accurate production processes employing industrial robots as their main component. The challenge to overcome here is to achieve high-accuracy machining solutions, in spite of the strong process forces required for the task. Because of the process forces, the nonlinear dynamics of the manipulator, such as the joint compliance and backlash, may significantly degrade the achieved machining accuracy of the manufactured part. In this thesis, a macro/micro-manipulator configuration is considered to the purpose of increasing the milling accuracy. In particular, a model-based control architecture is developed for control of the macro/micro-manipulator setup. The considered approach is validated by experimental results from extensive milling experiments in aluminium and steel. Related to the problem of high-accuracy milling is the topic of robot modeling. To this purpose, two different approaches are considered; modeling of the quasi-static joint dynamics and dynamic compliance modeling. The first problem is approached by an identification method for determining the joint stiffness and backlash. The second problem is approached by using gray-box identification based on subspace-identification methods. Both identification algorithms are evaluated experimentally. Finally, online state estimation is considered as a means to determine the workspace position and orientation of the robot tool. Kalman Filters and Rao-Blackwellized Particle Filters are employed to the purpose of sensor fusion of internal robot measurements and measurements from an inertial measurement unit for estimation of the desired states. The approaches considered are fully implemented and evaluated on experimental data.

The second part of the thesis discusses optimal motion control applied to robot manipulators and road vehicles. A control architecture for online control of a robot manipulator in high-performance path tracking is developed, and the architecture is evaluated in extensive simulations. The main characteristic of the control strategy is that it combines coordinated feedback

control along both the tangential and transversal directions of the path; this separation is achieved in the framework of natural coordinates. One motivation for research on optimal control of road vehicles in time-critical maneuvers is the desire to develop improved vehicle-safety systems. In this thesis, a method for solving optimal maneuvering problems using nonlinear optimization is discussed. More specifically, vehicle and tire modeling and the optimization formulations required to get useful solutions to these problems are investigated. The considered method is evaluated on different combinations of chassis and tire models, in maneuvers under different road conditions, and for investigation of optimal maneuvers in systems for electronic stability control. The obtained optimization results in simulations are evaluated and compared.

Acknowledgments

I would like to acknowledge my supervisors, Prof. Anders Robertsson and Prof. Rolf Johansson, for continuous guidance during my PhD studies. Further, Dr. Klas Nilsson has provided valuable comments on the research and suggestions for research directions. My colleague within the COMET and SMErobotics projects, Olof Sörnmo, has been an inspiring research partner as well as traveling partner at project meetings and conferences around the world. Dr. Mathias Haage is acknowledged for good cooperation on the identification of joint-based robot models and simulation of joint dynamics.

In the ELLIIT consortium, Dr. Karl Berntorp, has been an excellent research partner and Prof. Bo Bernhardsson provided constructive comments on the research. Kristoffer Lundahl, Prof. Lars Nielsen, and Dr. Jan Åslund, Linköping University, are acknowledged for good cooperation on the research regarding optimal control of vehicles and their sharing of knowledge of vehicle and tire dynamics. Moreover, Prof. Lars Nielsen is also acknowledged for the joint research on path tracking for robot manipulators.

Jacob Antonsson is acknowledged for the cooperation on sensor fusion and state estimation for robots. Further, Mahdi Ghazaei is acknowledged for the joint research on MPC for real-time trajectory generation. Fredrik Magnusson has provided valuable assistance regarding the JModelica.org optimization platform.

I am also grateful to Ulrich Schneider, Dr. Arnold Puzik, and Manuel Drust at Fraunhofer IPA, Germany, for numerous fruitful workshops in the Robotics Laboratory in Stuttgart and discussions on the macro/micro-manipulator configuration for machining. Christian Lehmann and Marcel Halbauer, Brandenburg University of Technology, Germany, are acknowledged for the joint research on identification of robot-joint properties.

I also extend my acknowledgments to my colleagues at the Department of Automatic Control and in the Robotics Laboratory for interesting discussions on control theory and related applications, and to the technical and administrative staff at the department for assistance on various subjects related to my research.

Financial Support

Financial support is gratefully acknowledged from the European Commission's Seventh Framework Program under grant agreements COMET (ref. #258769) and SMErobotics (ref. #287787), as well as the Swedish Foundation for Strategic Research within the program ProViking, under grant ProFlexA PV08–0036. The author is a member of the LCCC Linnaeus Center, supported by the Swedish Research Council, and the ELLIIT Excellence Center, supported by the Swedish Government.

Contents

1

Introduction

This thesis considers two different topics: Machining with industrial robot manipulators and optimal motion control of robots and road vehicles. The common theme of these topics is the importance of accurate motion control in advanced applications. First, a robot manipulator in a machining scenario with significant process forces affecting the robot end-effector requires particular modeling, sensor equipment, and subsequent control design in order to achieve the desired workpiece accuracies. Second, accurate path tracking for robot manipulators is essential in many applications in production and manufacturing. In these scenarios, it is often desirable to perform the task time-optimally for production effectiveness, which implies maximum (or near maximum) utilization of the available actuation capabilities. Third, a road vehicle in a time-critical situation, where in particular the maximum of the tire forces that can be developed is utilized, it is imperative to be able to perform the optimal maneuver given actuator and internal state constraints.

1.1 Background and Motivation

Motion control of industrial serial-kinematic robots has been investigated in the scientific community for several decades. Hence, the positioning accuracy achieved with model-based feedforward control combined with joint-servo feedback controllers for a robot moving in free space is satisfactory for many applications, such as pick-and-place operations. In addition, certain methods and products exist for improving the absolute accuracy of the robot by means of kinematic calibration. For an introduction to robot modeling and control, see, *e.g.*, [Spong et al., 2006] and [Siciliano et al., 2009]. When considering machining applications traditionally employing dedicated machine tools—where the manipulator should form a kinematic chain with the tool and the workpiece—the effects of robot joint dynamics such as friction, backlash, and compliance influence the positioning accuracy significantly. This is a result of the process forces required for these kinds of tasks. The accu-

racy to be fulfilled in this kind of applications is often below 100 μm. These facts motivate the need for research on methods and strategies for increasing the accuracy of machining tasks performed with industrial robots. In this thesis, a macro/micro-manipulator configuration for milling scenarios is considered. The components comprising the experimental setup are modeled and a subsequent model-based control architecture is developed. The architecture is verified in extensive milling experiments. Another topic investigated is quasi-static joint models and subsequent parameter identification for position compensation in machining scenarios. Moreover, gray-box identification based on subspace methods for a class of flexible mechanical systems is considered in this thesis. This is an important problem in robotic machining and other tasks requiring contact during the operation, in order to enable model-based control design. The method is evaluated on the micro-manipulator employed in the robotic machining cell. Another important research problem in robotic machining and several other robot tasks requiring high position accuracy is sensor integration and sensor fusion. To this purpose, state estimation in robot workspace based on internal robot measurements and inertial measurement units (IMUs) is considered in this thesis as means to improve the accuracy of the estimated tool position and orientation in the robot workspace.

In the general perspective, optimal motion control is increasingly important for both robots and road vehicles. For industrial manipulators, a classical problem is time-optimal path tracking, whereby constraints on the joint input torques are considered [Bobrow et al., 1985; Shin and McKay, 1985]. Since the inputs are of bang-bang character in time-optimal control, such a control strategy is sensitive to both modeling uncertainties and online disturbances. It is thus clear that online feedback during runtime is essential. In this thesis, a robust control architecture to this purpose is discussed. One motivator for optimal control of road vehicles is to find improved safety systems and driver-assistance technologies, others are fuel optimization and lap-time minimization. Recent surveys on applications of optimal control for vehicles are presented in, *e.g.*, [Sharp and Peng, 2011; Limebeer and Rao, 2015]. Moreover, in order to increase the understanding of vehicle dynamics, optimal vehicle control in time-critical situations is an appealing approach in order to trigger extreme modes in the vehicle dynamics and to find performance limits. The challenge here is to find vehicle models, which contain a sufficient degree of details in order for the results to be useful, not only in a simulation setting but also for vehicle control in general. The overall aim of these kinds of studies in this thesis is to find vehicle behavior and control strategies, which can lead to the development of improved vehicle-safety systems. More specifically, optimal control of road vehicles in certain critical situations, resulting in aggressive maneuvering behavior, is investigated in this thesis. In particular, the tire and chassis modeling required for

optimal control applications is in focus. The considered method is applied to the cases of uncertain road conditions and braking maneuvers in electronic stability control (ESC) systems.

1.2 Publications

In this section, the publications on which this thesis is based are presented. The publications are divided into two parts; machining with industrial robot manipulators and optimal control of robots and vehicles. Preliminary versions of parts of the research presented in this thesis have been published in the Licentiate Thesis by the author:

Olofsson, B. (2013). *Topics in Machining with Industrial Robots and Optimal Control of Vehicles*. Licentiate Thesis ISRN LUTFD2/TFRT--3259--SE. Department of Automatic Control, Lund University, Sweden.

Machining with Industrial Robot Manipulators

Olofsson, B., O. Sörnmo, U. Schneider, A. Robertsson, A. Puzik, and R. Johansson (2011). "Modeling and control of a piezo-actuated high-dynamic compensation mechanism for industrial robots". In: *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*. San Francisco, CA, pp. 4704–4709.

Schneider, U., B. Olofsson, O. Sörnmo, M. Drust, A. Robertsson, M. Hägele, and R. Johansson (2014). "Integrated approach to robotic machining with macro/micro-actuation". *Robotics and Computer-Integrated Manufacturing* **30**:6, pp. 636–647.

Sörnmo, O., B. Olofsson, U. Schneider, A. Robertsson, and R. Johansson (2012). "Increasing the milling accuracy for industrial robots using a piezo-actuated high-dynamic micro manipulator". In: *Proc. IEEE/ASME Int. Conf. Advanced Intelligent Mechatronics (AIM)*. Kaohsiung, Taiwan, pp. 104–110.

The first and third publication were mainly written by B. Olofsson and O. Sörnmo, and equal contributions are asserted. The research leading to the publications was performed jointly by B. Olofsson, O. Sörnmo, and U. Schneider and a majority of the experiments were made by these authors together. B. Olofsson and O. Sörnmo were main responsible for the modeling and control development, whereas U. Schneider provided ideas for the control architecture and performed parts of the implementation. A. Puzik developed the considered micro-manipulator for online compensation. The research presented in the second publication was performed as a cooperation between U. Schneider, B. Olofsson, and O. Sörnmo, and all three authors assert equal

contributions and joint first-authorship. B. Olofsson and O. Sörnmo identified the dynamic models and developed a major part of the control architecture, whereas U. Schneider developed the experimental setup, implemented and tuned the control architecture, and performed many of the milling experiments. M. Drust, A. Robertsson, M. Hägele, and R. Johansson assisted in structuring and improving the manuscripts.

Olofsson, B., O. Sörnmo, U. Schneider, M. Barho, A. Robertsson, and R. Johansson (2012). "Increasing the accuracy for a piezo-actuated micro manipulator for industrial robots using model-based nonlinear control". In: *Proc. 10th Int. IFAC Symp. Robot Control (SYROCO)*. Dubrovnik, Croatia, pp. 277–282.

In this publication, B. Olofsson developed the models from experimental data and implemented the controllers for execution on the experimental setup. O. Sörnmo discussed the method with B. Olofsson. U. Schneider and M. Barho did the experiments and performed an initial investigation of the nonlinear dynamics of the micro-manipulator and possible compensation strategies. A. Robertsson and R. Johansson provided constructive comments on the research and assisted in structuring the manuscript.

Lehmann, C., B. Olofsson, K. Nilsson, M. Halbauer, M. Haage, A. Robertsson, O. Sörnmo, and U. Berger (2013). "Robot joint modeling and parameter identification using the clamping method". In: *Proc. IFAC Conf. Manufacturing Modelling, Management, and Control (MIM)*. St. Petersburg, Russia, pp. 843–848.

B. Olofsson and O. Sörnmo developed the measurement procedure presented in this publication based on an idea of K. Nilsson. C. Lehmann and M. Halbauer performed the experiments presented in the paper and developed parts of the experimental procedure. The remaining authors took part in the discussions leading to the realization of the method, in the experimental implementation, and in the proof-reading of the manuscript.

Olofsson, B., O. Sörnmo, A. Robertsson, and R. Johansson (2014). "Continuous-time gray-box identification of mechanical systems using subspace-based identification methods". In: *Proc. IEEE/ASME Int. Conf. Advanced Intelligent Mechatronics (AIM)*. Besançon, France, pp. 327–333.

This publication was developed by B. Olofsson as the main contributor. B. Olofsson formulated the algorithm and performed the simulations and evaluation of the method on experimental data. O. Sörnmo contributed with ideas for the evaluation of the method. A. Robertsson and R. Johansson assisted in the structuring of the manuscript.

Olofsson, B., J. Antonsson, H. Kortier, B. Bernhardsson, A. Robertsson, and R. Johansson (2015). "Sensor fusion for robotic workspace state estimation". *IEEE/ASME Trans. Mechatronics. Submitted.*

This manuscript was developed as a cooperation between B. Olofsson and J. Antonsson, and equal contributions are asserted. B. Olofsson performed the experiments on the robot and J. Antonsson implemented the major part of the estimation algorithms. H. Kortier provided ideas for the sensor calibration and B. Olofsson adapted them to the robotics setting. H. Kortier also assisted in improving the manuscript. B. Bernhardsson gave constructive feedback during the course of the research leading to the results presented. A. Robertsson and R. Johansson provided comments for improving the manuscript.

Optimal Motion Control

Olofsson, B. and L. Nielsen (2015). "Path-tracking velocity control for robot manipulators with actuator constraints". *Mechatronics. Submitted.*

B. Olofsson developed the details of the algorithm presented in this manuscript based on an idea of L. Nielsen. B. Olofsson also performed a major part of the simulations presented. L. Nielsen did a subset of the simulations, wrote parts of the manuscript, and assisted in the structuring of the same.

Berntorp, K., B. Olofsson, K. Lundahl, B. Bernhardsson, and L. Nielsen (2013). "Models and methodology for optimal vehicle maneuvers applied to a hairpin turn". In: *Proc. American Control Conf. (ACC)*. Washington, D.C., pp. 2142–2149.

Berntorp, K., B. Olofsson, K. Lundahl, and L. Nielsen (2014). "Models and methodology for optimal trajectory generation in safety-critical road–vehicle manoeuvres". *Vehicle System Dynamics* **52**:10, pp. 1304–1332.

Lundahl, K., K. Berntorp, B. Olofsson, J. Åslund, and L. Nielsen (2013). "Studying the influence of roll and pitch dynamics in optimal road-vehicle maneuvers". In: *Int. Symp. Dynamics of Vehicles on Roads and Tracks (IAVSD)*. Qingdao, China.

Olofsson, B., K. Lundahl, K. Berntorp, and L. Nielsen (2013). "An investigation of optimal vehicle maneuvers for different road conditions". In: *Proc. 7th IFAC Symp. Advances in Automotive Control (AAC)*. Tokyo, Japan, pp. 66–71.

B. Olofsson, K. Berntorp, and K. Lundahl performed the optimizations, whose results are presented in the first, second, and fourth publication, together and equal contributions are asserted. B. Olofsson and K. Berntorp

were main responsible for the development of the optimization methodology employed in these publications. K. Lundahl developed the tire models utilized in the optimal control formulation and the driver model used for initialization. K. Lundahl also performed the optimizations and analyzed the results that are presented in the third publication. L. Nielsen formulated the problem and wrote parts of the manuscripts. B. Bernhardsson and J. Åslund provided constructive comments on the method and assisted in analyzing the results and structuring of the manuscripts.

Lundahl, K., B. Olofsson, K. Berntorp, J. Åslund, and L. Nielsen (2014). "Towards lane-keeping electronic stability control for road-vehicles". In: *Proc. 19th IFAC World Congress*. Cape Town, South Africa, pp. 6319–6325.

This publication was developed based on an idea of L. Nielsen. K. Lundahl, B. Olofsson, and K. Berntorp designed the simulation study together and K. Lundahl was main responsible for the optimization runs and the analysis of the achieved results. J. Åslund participated in the analysis of the results. L. Nielsen wrote parts of the manuscript and took part in the structuring of the same.

Other Related Publications

The following publications, where the author also has made contributions in related areas, were decided not to be part of the present thesis.

Berntorp, K., B. Olofsson, and A. Robertsson (2014). "Path tracking with obstacle avoidance for pseudo-omnidirectional mobile robots using convex optimization". In: *Proc. American Control Conf. (ACC)*. Portland, OR, pp. 517–524.

Cano Marchal, P., O. Sörnmo, B. Olofsson, A. Robertsson, J. Gómez Ortega, and R. Johansson (2014). "Iterative learning control for machining with industrial robots". In: *Proc. 19th IFAC World Congress*. Cape Town, South Africa, pp. 9327–9333.

Ghazaei Ardakani, M. M., B. Olofsson, A. Robertsson, and R. Johansson (2015). "Real-time trajectory generation using model predictive control". In: *Proc. IEEE Int. Conf. Automation Science and Engineering (CASE)*. Göteborg, Sweden. *Accepted for Publication*.

Magnusson, F., K. Berntorp, B. Olofsson, and J. Åkesson (2014). "Symbolic transformations of dynamic optimization problems". In: *10th Int. Modelica Conf.* Lund, Sweden.

Olofsson, B., H. Nilsson, A. Robertsson, and J. Åkesson (2011). "Optimal tracking and identification of paths for industrial robots". In: *Proc. 18th IFAC World Congress*. Milano, Italy, pp. 1126–1132.

Sörnmo, O., B. Olofsson, A. Robertsson, and R. Johansson (2012). "Increasing time-efficiency and accuracy of robotic machining processes using model-based adaptive force control". In: *Proc. 10th Int. IFAC Symp. Robot Control (SYROCO)*. Dubrovnik, Croatia, pp. 543–548.

Sörnmo, O., B. Olofsson, A. Robertsson, and R. Johansson (2013). "Adaptive internal model control for mid-ranging of closed-loop systems with internal saturation". In: *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*. Tokyo, Japan, pp. 4893–4899.

Sörnmo, O., B. Olofsson, A. Robertsson, and R. Johansson (2015). "Learning approach to cycle-time minimization of wood milling using adaptive force control". *ASME J. Manufacturing Science and Engineering*. DOI: `http://dx.doi.org/10.1115/1.4030751`. *Accepted for Publication*.

1.3 Contributions and Outline

The main contributions of this thesis are:

- Modeling and control of a micro-manipulator for online position compensation in high-accuracy machining for production scenarios using industrial robot manipulators;
- An experimental verification of the micro-manipulator control architecture in a milling scenario;
- An integrated approach to robotic machining using macro/micro-actuation with accompanying model-based control architecture;
- Joint-based robot modeling for machining scenarios and subsequent parameter identification;
- Gray-box identification of flexible mechanical systems using subspace-based identification algorithms in continuous time, applied to the micro-manipulator for machining;
- State-estimation algorithms for robot manipulators aimed at online tool position and orientation estimates in the workspace;
- A control architecture developed for optimal and robust path tracking for robot manipulators subject to actuator constraints;
- A method for computation of optimal road-vehicle maneuvers in critical situations using realistic vehicle models, optimal control, and state-of-the-art numerical optimization software;

- A study of the influence of vehicle chassis and tire modeling on the resulting optimal maneuvers using the considered optimization methodology;
- Applications of the method developed for vehicle trajectory generation in the case of uncertain road conditions and for computing optimal maneuvers in ESC systems.

Outline

This thesis comprises two parts. Part 1 presents results and algorithms related to machining with industrial robot manipulators. Chapter 2 gives a background to the subject, provides a motivating machining example, and discusses the challenges of machining with industrial robots. Modeling and subsequent model-based control design for a macro/micro-manipulator configuration for milling tasks using industrial robots are presented in Chapter 3. The considered method to robotic machining and the control architecture are evaluated in milling experiments in aluminium and steel, whose results are presented and discussed in Chapters 4–5. Joint-based robot models and subsequent identification on an arbitrary robot manipulator are the topics of Chapter 6. In Chapter 7, gray-box identification of mechanical systems with inherent flexibilities, such as the micro-manipulator, is considered. Algorithms for workspace state estimation of the position and orientation of the robot tool are developed in Chapter 8. Finally, conclusions are drawn and different directions for future research are identified in Chapter 9.

The second part of the thesis, Part 2, discusses optimal motion control of robots and vehicles. An introduction and a background to the subject is given in Chapter 10. A control architecture for robust and optimal path tracking in the case of actuator limitations is presented in Chapter 11. In Chapter 12, a methodology for computing optimal trajectories for road vehicles in time-critical situations is developed. In addition, vehicle and tire modeling with focus on optimal control is considered. In Chapter 13, different applications of optimal trajectory generation for road vehicles are presented. In particular, optimization results are provided for certain combinations of vehicle chassis and tire models in aggressive maneuvers. In that chapter, also results for optimal maneuvering under different road conditions in a hairpin turn are presented. Moreover, the optimization methodology is applied for investigations of maneuvers to be performed by future control systems for vehicle stability. This part of the thesis is concluded in Chapter 14, where also aspects on future research are presented.

Part I

Machining with Industrial Robot Manipulators

2

Introduction to Robotic Machining

During the last decades, industrial robots have become an essential part of modern production and manufacturing. Traditionally, robots have been used for operations not requiring continuous contact with the environment, such as in material handling and painting. However, less than 1% of the industrial robots operational in the world are used for machining applications [IFR Statistical Department, 2012]. Such applications include milling and grinding, which require contact between the manipulator and the machining tool or workpiece. Milling is usually performed using dedicated machine tools, such as computer numerical control (CNC) machines, because of their high positioning accuracy and stiffness. As a result of the increased demands on efficiency and flexibility in industrial production over the past decades, the need for automated, flexible, and high-accuracy machining tasks has increased. In this context, usage of industrial robots is an appealing solution based on their flexibility in terms of reconfiguration possibilities, versatility, and the lower investment required, compared to the cost of a machine tool. However, because of the comparably limited stiffness and position-accuracy of industrial robots, and consequently their inability to execute tasks based on *Computer-Aided Design* (CAD) specifications with acceptable results, machining operations are not straightforward to perform, see, *e.g.*, [Zhang et al., 2005; Abele et al., 2007; Wang et al., 2009; Pan and Zhang, 2009]. Further, the nonlinear dynamics of the robot joints, such as backlash, friction, and nonlinear stiffness, are important for the Cartesian positioning accuracy, since the joint-servo feedback controllers are typically based on measurements on the motor side of the joint. Consequently, there is a possible difference between the arm-side position and the measured motor-side position for the joints. The dynamics of a robot joint is further discussed in Chapter 6, where also an identification procedure is developed. The Cartesian end-effector position and orientation are in many cases only estimated based on the forward kinematic

relations—*i.e.*, no explicit arm-side or workspace sensing is used. While certain methods and commercial products offered by robot manufacturers exist for obtaining very high position accuracy for movement of the robot end-effector in free space or under constant load, achieving this in the presence of dynamic process forces affecting the end-effector is much more challenging. Different approaches are developed in this thesis to increase the accuracy of robot tasks using external sensors and actuation in workspace. Insufficient position accuracy in machining with industrial robots is a well-known problem in manufacturing; an experimental investigation was presented in [Schneider et al., 2013b]. The accuracy tolerances in manufacturing processes are usually in the range of 100 μm or lower [COMET, 2015]. This can typically not be achieved using a conventional industrial robot in application scenarios where strong process forces are required to execute the desired task, mainly because of the compliance exhibited by the manipulator.

2.1 Motivating Machining Example

In order to illustrate some of the challenges in machining with industrial robots and motivate the research presented in this part of the thesis, a milling task executed in aluminium is considered in this section. The task is to remove material in a rectangular area, referred to as a pocket, see Figure 2.1.

The procedure for the milling task was as follows: First, the milling path was computed from CAD data of the robot cell and the workpiece using a computer-aided manufacturing (CAM) software and subsequently a robot program was generated. Second, the milling task was performed with an industrial robot of model IRB2400 from ABB [ABB Robotics, 2012], which was holding the workpiece. The machining spindle was attached to a base inside the robot workspace, see Figure 2.2. The spindle was running at a speed of 24 000 rpm and the milling tool had a diameter of 7 mm and had two teeth. For arm-side measurements of the position and orientation of the tool center point (TCP) coordinate system, defined at the robot end-effector, an optical tracking system from Nikon Metrology of model K600 [Nikon Metrology, 2010] was used. This tracking system provides data at a sample rate of 1 kHz during the complete milling process. In addition, a force/torque sensor was attached to the robot end-effector for measurements of the process forces. For accessing the sensor data in the robot controller, the ExtCtrl architecture [Blomdell et al., 2010; Blomdell et al., 2005; Nilsson and Johansson, 1999], a research interface to ABB S4CPlus and IRC5 controllers, was used.

The milling experiment was performed, whereby the results shown in Figure 2.3 were obtained. It is to be noted that the path planning is such that the pocket is manufactured in several milling runs, where the accumulated

Figure 2.1 Milling task where a rectangular area of material, referred to as a pocket, is to be removed.

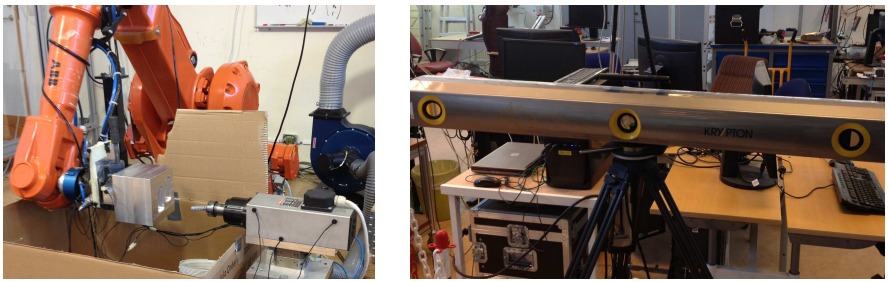

Figure 2.2 Experimental setup for milling experiments (left) at Lund University, including an optical tracking system (right) for high-accuracy measurements of robot position and orientation.

depth-of-cut is increased in every cycle. This strategy is common practice for machine tools and has for that reason also been adopted in the path-planning software utilized in this experiment. Further, the milling tool left the workpiece between each cycle, resulting in the characteristic force transients visible in the process-force data when re-entering the material prior to each cycle. Several conclusions can be drawn from the experimental data, exhibiting the challenges of machining with industrial robots:

- Uncertainty in the calibration of the robot—*i.e.*, in particular the kinematic parameters—and the aluminium block results in position and orientation errors of the workpiece which means that the milling tool is not completely orthogonal to the machined surface;

- The negative influence of the backlash in the robot joints, clearly visible in the machined surface in Figure 2.1 at the locations where the gears of the motors reverse motion direction and thus excite the backlash;

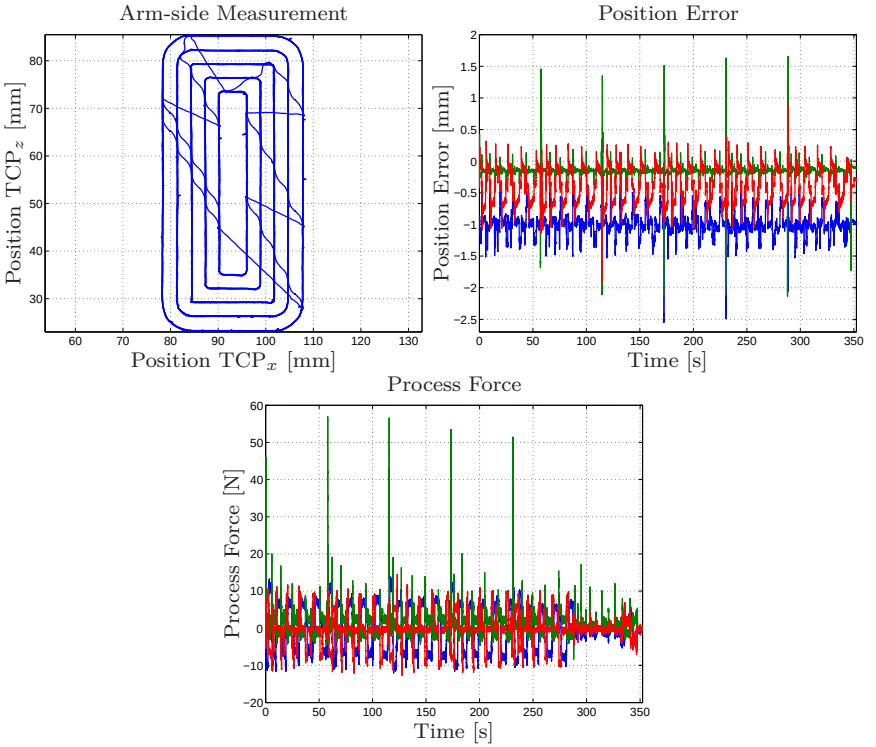

Figure 2.3 Measurements during a milling task with an industrial robot of model ABB IRB2400, see Figure 2.2. Subplots: Arm-side position measurements (upper left), position error (upper right), and process forces (lower center). Color scheme in upper right and lower plot: TCP$_x$–blue, TCP$_y$–green, and TCP$_z$–red.

- Position deviations occurring when the milling tool enters the material and consequently gives rise to significant force transients with frequency characteristics higher than the bandwidth of the position-control loop of the robot;
- The limited stiffness of the industrial robot—or more specifically the compliance of the robot joints—which results in position deviations when the process forces affect the robot end-effector.

Eigenfrequencies in Cartesian Space Considering the joint-based actuation for a serial-kinematic robot manipulator and that the high-frequency disturbances from the milling process enter the robot at the end-effector, not only the joint-actuation bandwidth of the robot is of importance but also the bandwidth for disturbance compensation at the TCP. The latter is signifi-

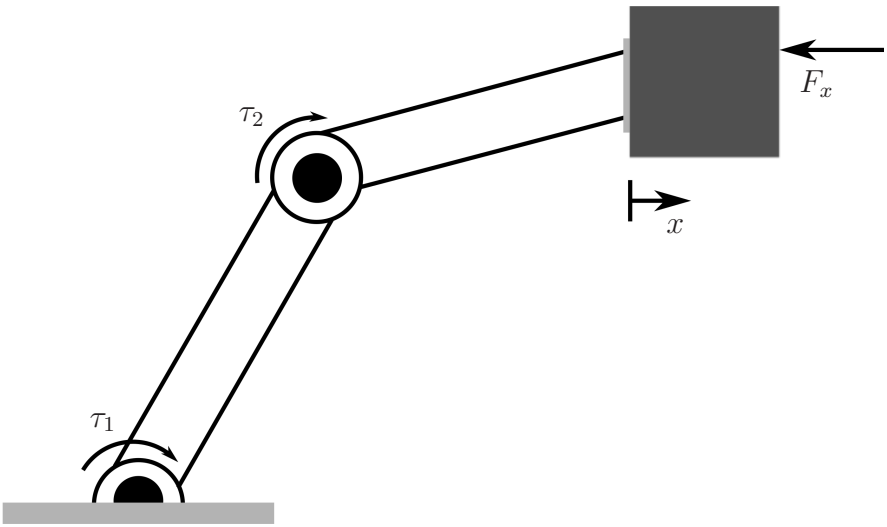

Figure 2.4 Schematic of a simplified robot manipulator with two degrees-of-freedom where joint actuation with torques τ_1 and τ_2 is performed. High-frequency force disturbances F_x from the machining process enter at the end-effector. The disturbances result in a negative position deflection x.

cantly different from the joint-actuation bandwidth because of the mechanical structure of the links. The non-colocation of actuation and disturbances (see, *e.g.*, [Sharon et al., 1993; Fasse and Hogan, 1995]) is schematically illustrated in one dimension for a simplified robot with two degrees-of-freedom in Figure 2.4. For illustration purposes, an impulse-response experiment was performed, where a force impulse was applied to the end-effector of the ABB IRB2400 robot and the resulting position deflection was measured simultaneously as the applied forces. The deflection of the robot was measured using the Nikon Metrology K600 optical tracking system. The result is shown in Figure 2.5. As observed in the plot, all Cartesian directions exhibit natural eigenfrequencies in the range of 10–25 Hz, with different damping along the Cartesian directions. This implies the limitations on the disturbance-rejection bandwidth in the position controller. The Cartesian eigenfrequencies of the robot also exhibit a dependency on the robot configuration.

2.2 The COMET project

Within the research project COMET[1] [COMET, 2015] started in 2010 and finished in 2013, the aim was to develop milling solutions for industrial robots

[1] Plug-and-produce COmponents and METhods for adaptive control of industrial robots enabling cost effective, high precision manufacturing in factories of the future.

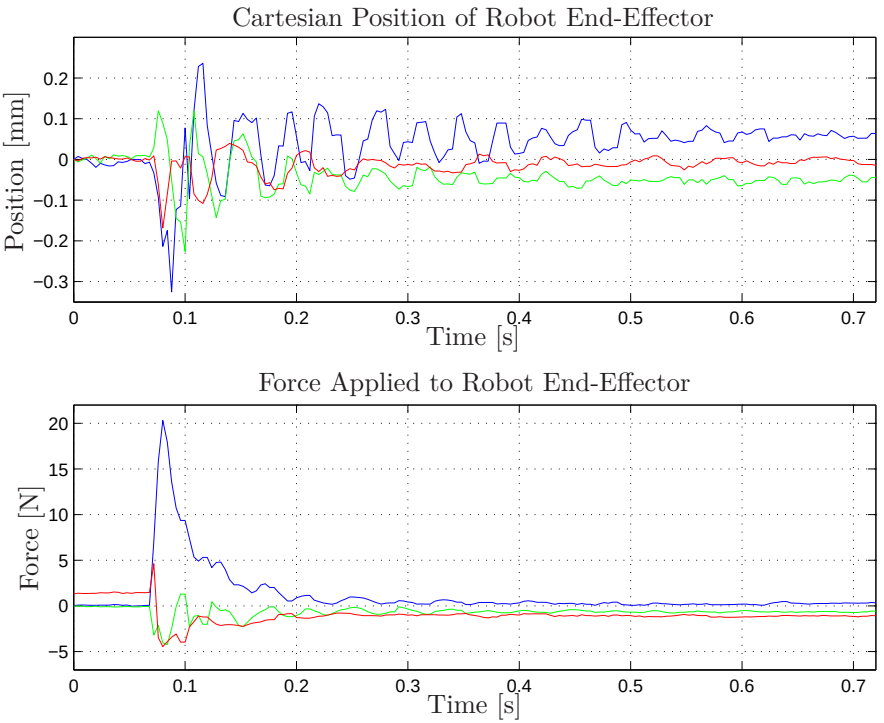

Figure 2.5 Impulse response obtained by applying a force on the robot end-effector and measuring the corresponding deflection. Color scheme: TCP_x–blue, TCP_y–green, and TCP_z–red.

achieving an accuracy better than 50 µm. This research was performed in a four-step procedure according to Figure 2.6. The four different parts of the project can be summarized as follows:

1. *Kinematic and Dynamic Models*
   To increase the machining accuracy, both kinematic and dynamic models of the robot were developed within the project. The kinematic models consist of the Denavit-Hartenberg parameters [Denavit and Hartenberg, 1955] as well as other geometric data of the robot. Further, the dynamic models were to describe the joint dynamics of the robot, significantly influencing the milling accuracy as observed in the milling experiment in the previous section. In particular, essential properties such as friction, backlash, and compliance were considered in the joint-based robot models. Further, methods for determining these properties of an industrial robot using automated procedures were investigated.

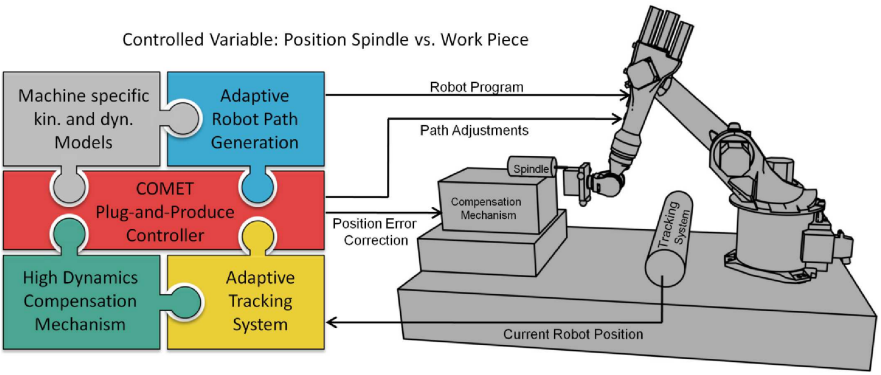

Figure 2.6 Graphical representation of the research aim of the COMET project, and its four corner stones [COMET, 2015].

2. *Model-Based Path Generation*
   Based on the kinematic and dynamic models, an improved strategy for offline path planning was developed. In the path-planning stage, the kinematic and dynamic models were utilized in order to plan a reference path utilizing CAM software which, given the identified models, results in the desired milling path when executed.

3. *Real-Time 6D Tracking System*
   As opposed to the two approaches described in the previous paragraphs based on offline compensation, also online strategies for improving the milling accuracy were investigated. To this purpose, online real-time 6D-compensation was developed, where the term 6D here refers to position and orientation (pose) of a rigid body. The poses of the robot workpiece were measured using optical tracking systems, whose measurements were fed to the robot controller and position corrections were performed in real-time.

4. *Micro-Manipulator for Milling*
   For milling tasks with industrial robots demanding very high position accuracy, a micro-manipulator where the machining spindle was attached was developed. The aim of the micro-manipulator was to compensate online for the 3D position deviations of the robot, as measured by the optical tracking system. By design, the micro-manipulator had a significantly higher bandwidth than the robot. Consequently, position deviations with higher frequency than the robot bandwidth were to be fed to the micro-manipulator—which has a geometrically limited compensation range—whereas low-frequency and static errors were handled by the robot itself.

Focus of This Thesis In this thesis, research results from the development of the control architecture for an integrated approach to robotic machining using macro/micro-manipulation are presented (see Chapter 3). Moreover, experimental results from application of the considered approach to milling are presented (see Chapters 4–5). In addition, methods aimed at robot-joint modeling and subsequent identification are developed and experimentally verified in Chapter 6.

2.3 The SMErobotics project

Within the research project SMErobotics[2] [SMErobotics, 2015], started in 2012, the aim is to develop robot systems with cognitive functionalities enabling robot-assisted manufacturing in small and medium-sized enterprises (SMEs). A key part of this research is the integration of sensor- and model-based feedback control in the robot task execution. Sensors are typically employed in modern robotic systems and the information from the respective sensor is combined using statistical sensor-fusion methods [Gustafsson, 2010] in order to estimate the states that are not explicitly measurable. In practice, sensors provide information that needs to be acted upon as part of the online task execution. A natural next step in a robotic system is learning—*i.e.*, inclusion of algorithms to the purpose of storing and subsequently utilize previously obtained information so as to improve the task execution sequentially. Here, also learning from human demonstration and interaction is essential, see, *e.g.*, [Calinon et al., 2007; Lee et al., 2014].

Focus of This Thesis In this thesis, research regarding gray-box identification for flexible mechanical systems (see Chapter 7), which is an essential component when designing controllers for robot tasks including contact operations. Another topic related to the SMErobotics project investigated in this thesis is sensor fusion for robotic state estimation (see Chapter 8); this is an example of how additional sensor data to the internal robot sensors can be used for extending the information available in the robot system during the online task execution.

2.4 Related Research

Previous approaches to increasing the position accuracy in robotic machining are for instance based on combinations of kinematic calibration [Roth et al., 1987; Schröer et al., 1997; Joubair et al., 2013] and stiffness modeling of the manipulator with subsequent offline or online compensation [Zhang et al., 2005; Abele et al., 2007; Wang et al., 2009; Pan and Zhang, 2009;

[2] The European Robotics Initiative for Strengthening the Competitiveness of SMEs in Manufacturing.

Reinl et al., 2011; Dumas et al., 2011; Tyapin et al., 2014]. In the kinematic calibration procedure, the kinematic parameters of the robot—*e.g.*, based on the Denavit-Hartenberg parametrization [Denavit and Hartenberg, 1955]—are typically determined with high accuracy using high-precision measurement systems without external load or with a constant load attached to the robot end-effector. In stiffness compensation methods, the manipulator stiffness matrix, in joint space or in workspace, is determined based on experimental data. Depending on the accuracy requirements, both joint and link flexibilities have been considered. Regarding the employment of the obtained stiffness models, different approaches have been suggested in the literature. If the models are employed offline, process forces during the task can be predicted and used for improving the path planning, thereby accounting for the *a priori* determined stiffness of the manipulator. These methods do, however, rely on accurate stiffness and process models. As an alternative, end-effector force-sensor data could be acquired online and used for real-time compensation and adaptation of the model in a designed feedback controller.

Other approaches to increasing the position accuracy in robotic processes are based on sensor data from high-precision 3D or 6D position-measurement sensors [Schneider et al., 2013c], such as a coordinate measuring machine (CMM) [Cuypers et al., 2009]. The sensor data can be used as a basis for end-effector position and orientation feedback for online corrections. The main limitation for application of such methods in machining and other contact operations is the sensor accuracy required, communication delays for sensor data in real-time applications, and noise in the measurements. Furthermore, the disturbance-rejection bandwidth at the end-effector of the robot manipulator is often limiting such approaches to increase the accuracy in machining. As a result of the cutting process in machining applications, high-frequency disturbances on the robot end-effector position are to be expected (compare with the machining example and subsequent discussion presented in Section 2.1), and therefore a high-bandwidth position control of the robot is essential in order to achieve sufficient accuracy of the machined parts.

Another set of methods proposed for increasing the accuracy of industrial manipulators is based on iterative learning control (ILC) [Arimoto et al., 1984; Norrlöf, 2000; Axehill et al., 2014]. Using arm-side position sensors or state estimates based on dynamic models, the accuracy of repetitive robot motions can be increased iteratively for many applications. It is to be noted that some of these methods were applied offline, and required an initial experiment to be performed, such that relevant sensor data could be collected for the subsequent compensations. However, also real-time approaches to ILC have been proposed, see, *e.g.*, [Xu et al., 2010]. Further, many previous applications of ILC in robotics described in literature primarily considered the case when the end-effector was moving in free space. This is not directly applicable in machining, where the required process forces are the dominant

source of position errors. ILC dedicated for machining with industrial manipulators was considered in [Cano Marchal et al., 2014].

Methods for contact-force control, learning, and adaptation in robotic machining, in particular milling in aluminium and wood, were presented in [Sörnmo et al., 2012b; Sörnmo et al., 2015] together with extensive experimental results. The results showed that a significant decrease of the required cycle time could be achieved with the proposed control design and online learning based on feedback from a force sensor attached to the robot wrist.

As an alternative to the approaches described in the previous paragraphs, the strategy to robotic machining considered in Chapters 3–5 in this thesis comprises workspace sensing using 6D position sensors, combined with macro/micro actuation. The concept of macro/micro-manipulator systems was suggested in [Sharon et al., 1993], and it was shown how the bandwidth of the end-effector position and force control can be increased beyond the natural eigenfrequencies of the macro-manipulator with appropriate control design. In [Sharon et al., 1993], the macro-manipulator was carrying the micro-manipulator in a serial configuration, whereas in this thesis the micro-manipulator is fixed in the workspace (*cf.* Figure 2.7 and Figure 2.9). The notions of macro- and micro-manipulators and actuation have been adopted in this thesis. The macro-manipulator has a large workspace, but has a limited position-control bandwidth. The bandwidth of the position control for an industrial manipulator is mainly limited by the eigenfrequencies of the manipulator structure, as experimentally indicated in Section 2.1. Typical values for the bandwidth of the end-effector position control for industrial manipulators are in the range of 10–30 Hz, depending on configuration [Schneider et al., 2013b]. In contrast, the micro-manipulator has significantly higher bandwidth, but a geometrically limited workspace. Hence, the micro-manipulator is to compensate for the high-frequency position deviations that occur during milling, which the macro-manipulator *per se* is unable to compensate for because of its limited disturbance-rejection bandwidth at the end-effector.

Piezo-actuated mechanisms based on flexure elements have been proposed for micro- and nano-manipulation earlier, see, *e.g.*, [Li and Xu, 2011; Liaw and Shirinzadeh, 2010]. Although the compensation mechanism considered in this thesis utilizes similar components in its mechanical design, there are certain differences. Previous designs were primarily developed for compensation in micro and nano manipulation, whereas the micro-manipulator considered in this thesis is designed for machining processes with industrial manipulators, where strong process forces are required to fulfill the specified task. Nevertheless, the control design in [Li and Xu, 2011] relates to the control algorithms for the micro-manipulator developed in this thesis. Further related research in this area is [Eielsen et al., 2015], where control design for high-accuracy tracking using repetitive control in the context of piezo-actuated, flexure-based mechanical systems was presented.

2.5 Macro/Micro-Manipulator Setup for Machining

In Chapters 3–5, the purpose is to develop an integrated control architecture aiming at high-accuracy robotic machining. This section provides a background and a description of the hardware components used, which are based on the publications [Sörnmo et al., 2012a; Schneider et al., 2014]. The goal in terms of machining accuracy with the robot cell is to reach a maximum error within ± 100 μm, which should be obtained also for stiff materials such as steel, where the required process forces are significantly stronger compared to, *e.g.*, aluminium. The long-term goal of the research presented in this thesis is to enable manufacturing with industrial robots, based on CAD specifications, achieving machine-tool accuracy of the produced parts.

For milling tasks where high-accuracy is required, a robot cell comprising a macro/micro-manipulator setup has been developed at the Fraunhofer Institute for Manufacturing and Engineering in Stuttgart, Germany. Two different cell layouts can be considered for robotic machining. First, the robot can carry the spindle and the workpiece is consequently attached to a fixed base in the workspace. In this configuration, the robot benefits from its large workspace and can machine large-scale workpieces. In the second configuration, the robot holds the workpiece, whereas the spindle is in a fixed position in the cell. This setup is suitable for smaller and lighter workpieces, since the workpiece geometry and mass are limited by the workspace and the payload of the robot. Further, the second configuration also allows the robot to perform handling operations, thus facilitating the integration in automated production lines. In the research presented in this thesis, the second configuration was chosen; thus, the spindle was attached to the micro-manipulator and the workpiece was held by the robot. This was motivated by the heavy weight of the micro-manipulator with the machining spindle attached. Two different robot-cell setups were considered for the machining experiments based on this configuration; one prototype version and a final setup. The final setup was developed based on experiences from a dynamic characterization and experiments on the prototype. The details of the respective setup are discussed next.

Prototype Robot Cell for Machining

The first developed robot cell for machining with macro/micro-actuation is shown in Figure 2.7. This cell was used as a prototype for evaluation of the machining accuracy with macro/micro-actuation. In this setup, the only compensation active was the motion performed with the micro-manipulator. As stated earlier, accurate positioning and high actuation bandwidth—high compared to the robot end-effector position-control bandwidth—are key features for the design of a micro-manipulator, see [Sharon et al., 1993] for a detailed discussion on the desired characteristics of macro- and micro-

manipulators. The objective in this thesis is to use the micro-manipulator in the machining setup for keeping the relative position between tool and workpiece according to the specified reference.

Micro-Manipulator A micro-manipulator prototype with three translational axes, with a serial actuation mechanism enabling position compensation in a Cartesian coordinate system, was previously proposed in [Puzik et al., 2009; Puzik et al., 2010; Puzik, 2011]. This version of the micro-manipulator was used as a prototype for the control design in this thesis. The micro-manipulator is depicted in Figure 2.7. The axes were driven by piezo-actuators and the design incorporated solid-state flexure elements, acting as a gear for increasing the compensation range, compared to the piezo-actuator extension. Piezo-actuators provide a high bandwidth transfer from applied voltage to extension, and can handle forces up to 30 kN. Hence, the micro-manipulator can withstand the process forces expected in a machining task. The process forces for the applications targeted with the considered robot cell are up to 1000 N. The process forces depend on machining type, workpiece material, depth-of-cut, and feed velocity. The maximum extensions of the piezo-actuators for the different axes were 100–180 µm and the gear ratio was between four and five, which resulted in a total compensation range of approximately 0.5 mm along each Cartesian actuation axis.

Robot-Cell Topology The experimental evaluation of the prototype robot cell was performed using an REIS industrial robot of model RV40 [Reis GmbH, 2011], with a maximum payload of 40 kg. The machining spindle was attached to the micro-manipulator and the robot held the workpiece, which in the experimental verification in Chapter 4 was a block of aluminium of type AlMg3,5. The setup was such that both face milling and peripheral milling, also referred to as radial milling, could be performed, see Figure 2.7. It is to be noted that these two strategies for milling are not equivalent, since the required cutting forces are different in magnitude and direction and hence affect the robot differently. Consequently, both face and peripheral milling experiments were performed in the subsequent evaluation in Chapter 4.

Interface and Sensors The micro-manipulator was equipped with strain gauges integrated into the piezo-actuators, measuring the corresponding extensions with temperature compensation and thus achieving a resolution of 0.7 µm. In addition, capacitive sensors which measure the Cartesian position of the end-effector with a resolution of 0.15 µm and a bandwidth of 2 kHz, were integrated into the micro-manipulator.

The micro-manipulator was interfaced with a dSPACE controller board of model DS1103 [dSPACE GmbH, 2007], where all sensor signals from the micro-manipulator were read and the signals to the actuators from the controller were sent. The developed controllers were executed in the controller

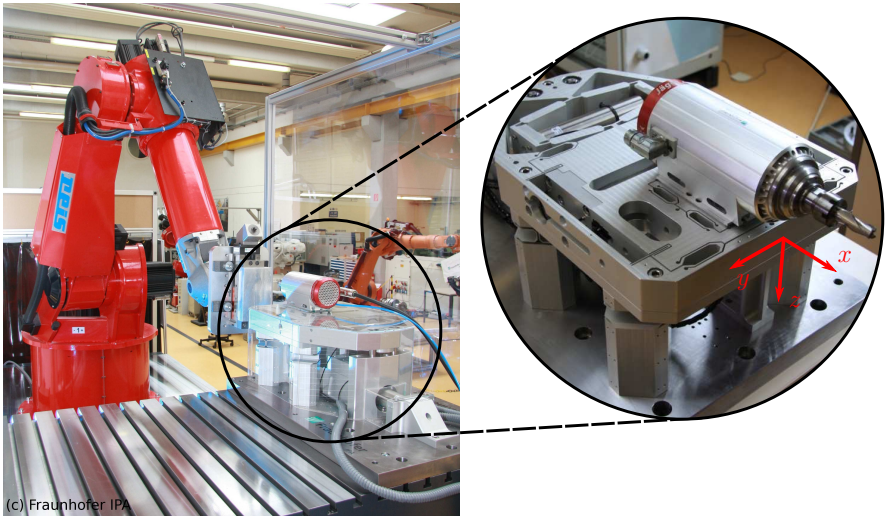

Figure 2.7 The prototype experimental setup at Fraunhofer IPA, Stuttgart, Germany, for online compensation of position errors during machining tasks. The workpiece is held by the robot and the milling spindle is attached to the micro-manipulator. A close-up of the micro-manipulator, as seen from the opposite side, is displayed to the right in the figure. Figure originally published in [Olofsson et al., 2011a].

board at a maximum sampling frequency of 10 kHz. The controllers were implemented in MATLAB Simulink and C-code was automatically generated by the Real-Time Workshop toolbox [MathWorks Inc. 2010]. The compiled C-code was then executed in the dSPACE system.

To the purpose of measuring the deflections of the robot in the milling direction—*i.e.*, the deflections which were to be compensated for by the micro-manipulator—a Keyence laser sensor of model LK-G87 [Keyence Corp. 2006], with a resolution of 0.2 μm and a sampling frequency of 10 kHz was used as a prototype tracking system. This sensor is based on the interferometer principle for measuring the distance to the aluminium workpiece attached to the robot.

Compensated and Uncompensated Milling In order to illustrate the benefit of the micro-manipulator, the milling experiments presented in Chapter 4 were performed both in a setting where compensation with the micro-manipulator was utilized and in a setting with the spindle rigidly attached to a fixed base—*i.e.*, in the latter setup no compensation was performed. The two experimental settings are illustrated in Figure 2.8.

In the experiments without compensation, the robot configuration was

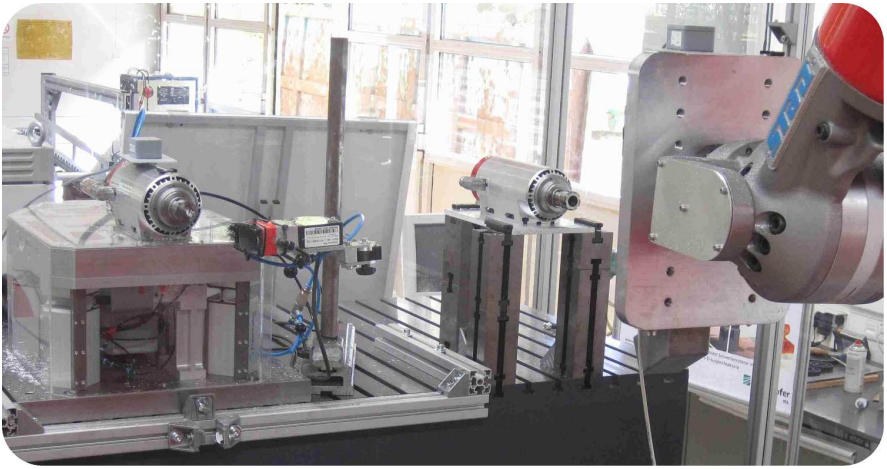

Figure 2.8 Prototype experimental setup at Fraunhofer IPA, Stuttgart, Germany, for evaluation of the performance of the considered micro-manipulator and accompanied position-control architecture. The micro-manipulator is seen to the left. The machining spindle to the right is rigidly attached to the base. The latter setup is utilized for milling experiments without compensation. Figure originally published in [Sörnmo et al., 2012a].

mirrored, with respect to the center plane of the robot, compared to the configuration chosen in the experiments with compensation. Consequently, the compliance properties of the robot in the two configurations are equivalent. Mirroring is important in order to make the compensated and uncompensated milling results comparable, since the Cartesian compliance exhibits a configuration dependency.

Complete Robot Cell for Machining

The final version of the robot cell at Fraunhofer IPA in Stuttgart, Germany, is depicted in Figure 2.9. This cell integrated online compensation with both the macro- and the micro-manipulator. The experimental setup contained an industrial robot manipulator (macro-manipulator), the micro-manipulator, an optical 6D tracking system, a machining spindle, and a CNC controller. Here, the different hardware components of the machining cell and their characteristics are briefly reviewed.

Robot and CNC Controller The foundation of the robot cell was a KUKA industrial robot of model KR125 [KUKA Roboter GmbH, 2013]. The joint positions of the robot were controlled using a Beckhoff TwinCAT CNC controller [Beckhoff Automation GmbH, 2013]. Based on the kinematics of the robot, Cartesian end-effector control could be achieved. Compared to a

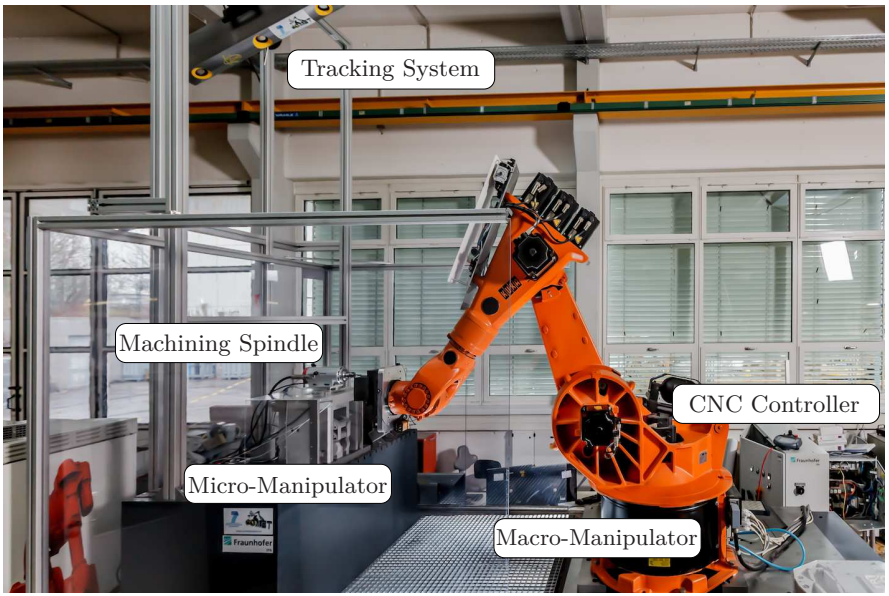

Figure 2.9 The final version of the developed robot cell for high-accuracy robotic machining at Fraunhofer IPA, Stuttgart, Germany. The robot cell includes a micro-manipulator and an optical tracking system. Figure originally published in [Schneider et al., 2014].

conventional robot controller, the CNC controller offers several advantages. First, path planning algorithms, which are optimized for machining tasks, are available. Second, from an implementation point-of-view, the most important feature is the open high-speed interface, which enables online position and orientation corrections based on external sensor data. The implemented robot controller was executed on a programmable logic controller (PLC). The integrated Codesys PLC programming environment provided possibilities for the implementation of complex controller structures, such as the architecture developed in Chapter 3.

Micro-Manipulator A new design of the micro-manipulator, which was revised with respect to the version shown in Figure 2.7 and comprising a parallel actuation principle, was shown to improve the dynamic properties of the mechanism significantly, see [Schneider et al., 2013a]. In particular, as a result of the reduction of the end-effector mass and a modified geometry of the solid-state flexure elements, a higher bandwidth was achieved. A photo of the mechanical design of the mechanism and the actuation axes is displayed in Figure 2.10. This version of the micro-manipulator was used in the final robot-cell setup, shown in Figure 2.9.

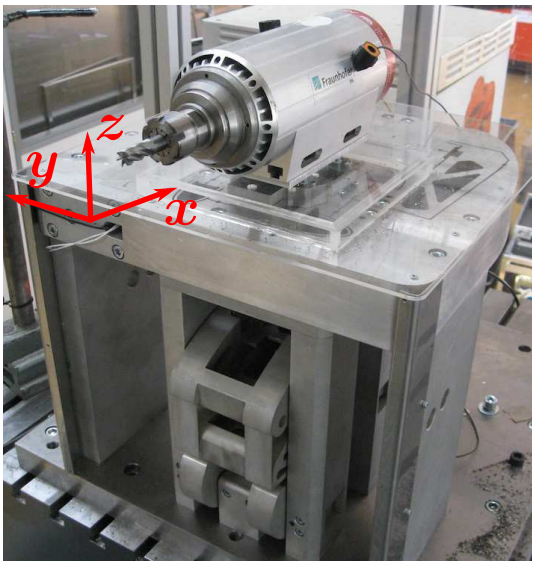

Figure 2.10 Revised design of the micro-manipulator with parallel actuation, with the Cartesian actuation coordinate system indicated in red. A machining spindle, with the tool and attached LED units for optical tracking, is mounted on the micro-manipulator end-effector. Figure originally published in [Schneider et al., 2014].

Similarly to the prototype setup, the revised micro-manipulator was equipped with two sets of sensors for control purposes. Integrated strain-gauge sensors in the piezo-actuators measured the extension of the respective actuator and capacitive sensors measured the distances along the Cartesian axes to a solid cube attached to the end-effector where the spindle was mounted.

Optical Tracking System In order to acquire workspace pose measurements, a Nikon Metrology K600 optical tracking system [Nikon Metrology, 2010] was employed because of its high-speed data streaming, large measurement volume, and the possibility to track both positions and orientations of several Cartesian coordinate systems simultaneously. As the optical measurements were subject to disturbances from the milling process, such as emitted metal chips, dust, and smoke, the coordinate systems were tracked using a redundant number of light-emitting diodes (LEDs) for increased robustness. The LEDs were attached to the rigid bodies to be tracked. For the machining application considered in this research, one coordinate system on the spindle and one coordinate system on the macro-manipulator end-effector were measured. The measurement frequency of the tracking system with this

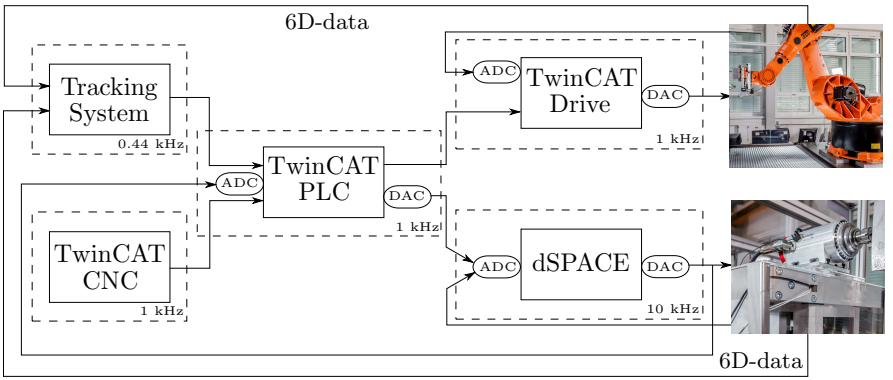

Figure 2.11 Hardware control units in the considered robot cell for machining, and their execution frequencies. The required analog to digital and digital to analog conversions are indicated as well. The 6D-data from the micro- and macro-manipulators were acquired using an optical tracking system measuring LED units attached to the respective manipulator. Figure originally published in [Schneider et al., 2014].

configuration of LEDs was 440 Hz and the specified accuracy (in terms of absolute measurements) for each LED unit measured was 100 µm. The sensor data from the measurement system were retrieved using a TCP data stream, connected to the CNC controller.

Machining Spindle The milling tool was attached to a Jäger Chopper 3300 high-frequency machining spindle [Alfred Jäger GmbH, 2013], which had a documented concentricity of less than 2 µm. The spindle speed was set to 28 000 rpm, which was justified by the fact that high rotation-velocities reduce process forces and the probability of exciting eigenmodes of the robot during the machining task.

Communication and Controller Execution Communication and timing are critical for the considered approach to machining with a macro-manipulator and additional external sensors and actuators, since delays between control units reduce the system performance significantly and might cause instability if too excessive. Figure 2.11 shows the relevant control units with digital-to-analog converters (DAC) and analog-to-digital converters (ADC). For each control unit, the execution frequency is specified. As in the prototype setup, the micro-manipulator controller was executed on the dSPACE signal processing system of model DS1103 [dSPACE GmbH, 2007]. The closed-loop control of the robot requires a high-speed interface for specifying joint-position references in real-time, which the used CNC controller provided, see Sections 3.5–3.6 in Chapter 3 for further details.

3

Modeling and Control of the Macro/Micro Setup

3.1 Introduction

In this chapter, modeling of the micro-manipulator and position-control design for the macro/micro-actuator setup for machining tasks with industrial manipulators are presented. Dynamic models of the micro-manipulator and a subsequent model-based control architecture are developed. This chapter is derived, in parts, from the publications [Olofsson et al., 2011a] and [Schneider et al., 2014].

The structure of this chapter is as follows: First, the mechanical design of the micro-manipulator is reviewed in Section 3.2. Subsequently, a dynamic characterization of the micro-manipulator is presented in Section 3.3, which provides the foundation for the modeling discussed in Section 3.4. Section 3.5 presents a model-based control architecture for position control of the micro-manipulator. The complete control architecture for the integrated macro/micro-actuator system is discussed in Section 3.6. The chapter is concluded with a summary in Section 3.7.

3.2 Micro-Manipulator Design

In order to provide a basis for the development of position-control algorithms for the micro-manipulator, its mechanical design is reviewed in this section. The focus is on the final version of the micro-manipulator, comprising a parallel actuation mechanism. The design is such that translational motion of the machining spindle is possible along all three Cartesian directions. However, there are no rotational degrees of freedom. Referring to Figure 2.10 in Chapter 2, the actuation axes are hereafter called x, y, and z, respectively. The movements of the piezo-actuators driving the mechanism are transferred to a corresponding translational movement of the spindle *via* a flexure mech-

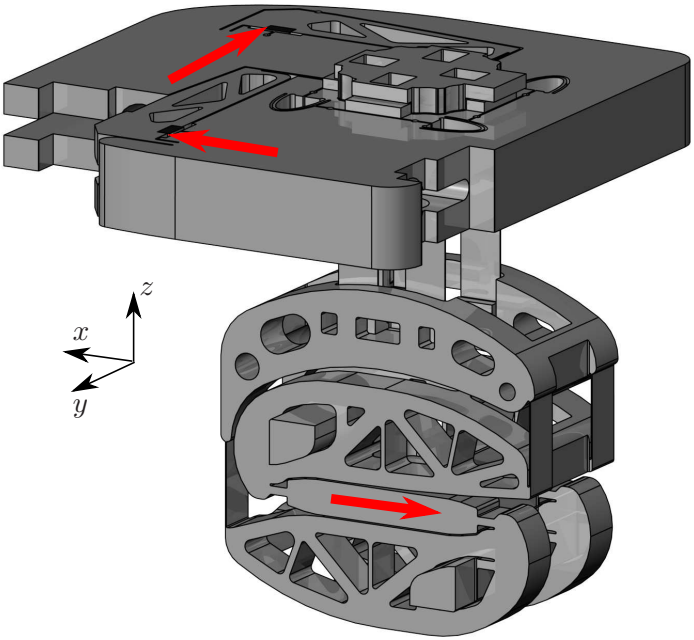

Figure 3.1 Actuation principle for the axes of the final version of the micro-manipulator (integrated in the robot cell shown in Figure 2.9 in Chapter 2). The piezo-actuators are indicated by red arrows in the drawing [Schneider et al., 2013a]. Figure courtesy of Fraunhofer IPA, Stuttgart, Germany.

anism. A mechanical drawing of the final version of the micro-manipulator is shown in Figure 3.1. The flexure elements are constructed using solid-state joints—*i.e.*, the joints are cut directly in the solid aluminium base plate. The flexure elements result in a gear ratio between actuator side and end-effector side along each of the motion axes. For further details regarding the mechanical design, see [Puzik, 2011; Schneider et al., 2013a].

The results presented for the modeling of the micro-manipulator in this chapter, were obtained from the final version of the micro-manipulator and the control architecture was aimed at the final robot-cell setup in Figure 2.9 in Chapter 2. For details regarding the modeling of the prototype version, see [Olofsson et al., 2011a].

3.3 Dynamic Characterization of the Micro-Manipulator

Because of the inherent resonant character of the mechanical design—which is a result of the solid-state flexure elements utilized—and the nonlinear dy-

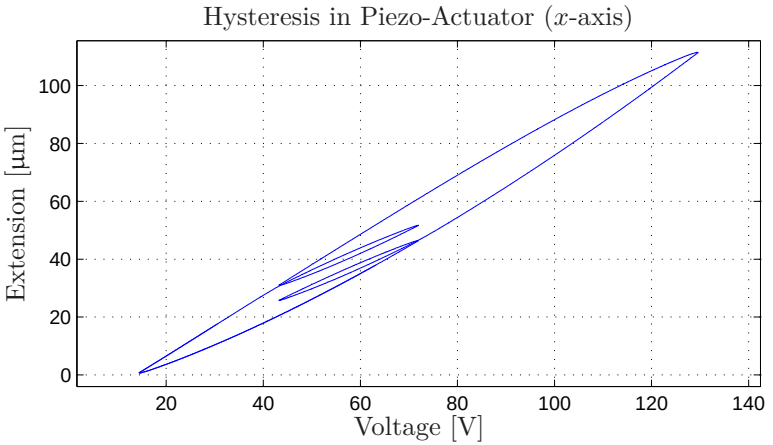

Figure 3.2 Characterization of hysteresis in the piezo-actuator dynamics along the x-axis of the micro-manipulator in the case of a linear, alternatingly increasing and decreasing, input with changing amplitude. Note the complex behavior of the hysteresis, which exhibits both rate and amplitude dependency.

namics which is a characteristic of piezo-actuators, accurate position control of the micro-manipulator without oscillations under milling-process disturbances is a challenging control problem.

Nonlinear Dynamics in the Piezo-Actuators

It is well-known that piezo-actuators exhibit a nonlinear relationship between the applied input voltage and the corresponding position extension, see, *e.g.*, [Al Janaideh et al., 2009; Sun and Yang, 2009]. Experiments were performed on the micro-manipulator in order to quantify the effect on the position accuracy of the nonlinear dynamics in the piezo-actuators. The experiments indicated that the main nonlinearities that needed to be handled were hysteresis and the creep phenomenon, where the latter means that the extension of the piezo-actuator is increasing over time for constant input voltage. The creep effect was quantified to an approximate rate of 0.02 μm/s for the respective piezo-actuator. Results from experiments where the input voltages to the piezo-actuators were alternatingly linearly increasing and decreasing are shown in Figure 3.2. It is obvious that the hysteresis needs to be handled actively for accurate positioning. Further, since the creep phenomenon in the actuators is a much slower process it is thus easier to handle. Although different in nature, both of these nonlinear effects can be reduced by using high-gain feedback, combined with integral feedback.

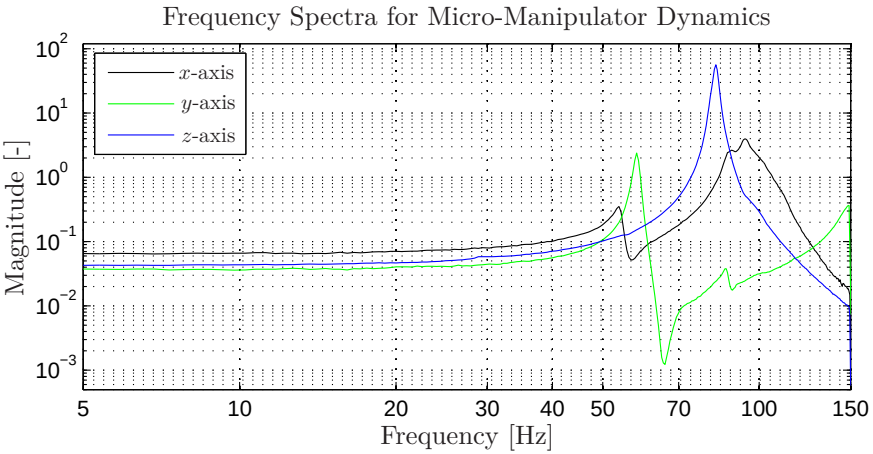

Figure 3.3 Estimated power spectral densities of the output signal in the Cartesian actuation directions of the micro-manipulator when exciting with a chirp signal (having an approximately flat power spectral density in the excitation frequency range).

Frequency Characterization of the Micro-Manipulator

In order to characterize the frequency properties of the mechanical design of the micro-manipulator, several frequency response experiments have been performed. The power spectral densities [Johansson, 1993] of the position outputs when exciting the inputs of the piezo-actuators with a chirp signal, are displayed in Figure 3.3 for the different actuation directions. The spectra were estimated using Welch's method in MATLAB. An important property of the system from a control point-of-view is the location of the first natural eigenfrequency. It is noted that the characteristics are quite different in the three Cartesian actuation directions. In particular, in the interesting frequency range of 0–150 Hz (determined based on the initial frequency sweeps), three major natural eigenfrequencies are visible for the dynamics along the x-axis, whereas two along the y-axis, and only one along the z-axis. The first eigenfrequency appears in the frequency range 50–80 Hz for all of the three axes. The locations of the eigenfrequencies are important since they limit the achievable bandwidth in the closed-loop control system. Increasing the bandwidth beyond poorly damped eigenfrequencies requires high control actuation and the sensitivity to model errors is increased significantly.

3.4 Modeling of the Micro-Manipulator

In order to design position-control algorithms, it is advantageous to perform modeling of the micro-manipulator prior to the design. Different methods

for modeling can be chosen. As a first approach, modeling based on physical first principles can be established, where the construction-specific parameters are either analytically calculated or experimentally identified. However, an analytical modeling approach is intractable for the current setup, since there is no straightforward mathematical description of the dynamics of the solid-state flexure elements used in the micro-manipulator design.

A black-box modeling approach [Johansson, 1993] is investigated in this chapter for modeling of the linear dynamics of the micro-manipulator. Such an approach results in satisfactory control performance given that the model captures the essential input-output dynamics of the system. A further modeling approach is gray-box identification where the model structure is fixed but certain model parameters are unknown; this modeling strategy is further investigated in Chapter 7 for the micro-manipulator dynamics.

Identification Based on Black-Box Models

Using system-identification methods [Johansson, 1993], mathematical models describing the dynamics of the micro-manipulator were determined. The axes can, in the modeling phase, be assumed to be decoupled provided that the mechanical design is such that the motions of the different axes are sufficiently independent. During the dynamic characterization of the micro-manipulator, minor cross coupling between the x- and z-axes was observed, which is further investigated using multi-input multi-output (MIMO) identification later in this section. The cross coupling is also further evaluated in Chapter 7 using gray-box identification. However, since the influence of the coupling is limited, the assumption on decoupling was made in the subsequent model-based control design. Consequently, each axis is considered as a system with one input and one output. Discrete-time state-space models of the innovation form

$$\begin{aligned} x_{k+1} &= \Phi x_k + \Gamma u_k + K e_k, \\ y_k &= C x_k + e_k, \end{aligned} \tag{3.1}$$

where $u_k \in \mathbb{R}^m$ is the input, $x_k \in \mathbb{R}^n$ is the state vector, $y_k \in \mathbb{R}^p$ is the output, K is the Kalman filter gain matrix, and e_k is a white-noise sequence, were considered. The system model matrices $\mathcal{S} : \{\Phi, \Gamma, C\}$ in the state-space representation were identified using one of the available implementations of subspace-based identification methods. The computer tools used were the System Identification Toolbox [Ljung, 2010] in MATLAB and the State-Space Model Identification (SMI) Toolbox [Haverkamp and Verhaegen, 1997]. In particular, the N4SID algorithm [Van Overschee and De Moor, 1994] and the MOESP algorithm [Verhaegen and Dewilde, 1992] were utilized. During the identification of the models, the gain matrix K in a Kalman filter [Kalman, 1960] for a minimum-variance estimation of the states in the model is also determined based on the noise properties of the identification data.

The subspace-based identification methods were found to result in models with superior fit to experimental data. In particular, the natural eigenfrequencies of the micro-manipulator were identified with significantly higher accuracy with subspace methods compared to identification of time-series models of autoregressive moving-average with exogenous input (ARMAX) type. For further details on time-series modeling, see, *e.g.*, [Madsen, 2008; Johansson, 1993].

Collection of Input-Output Data

The collection of experimental input-output data was performed in such a way that the input u_k was considered to be a scaled version of the input voltage to the piezo-actuator, whereas the output y_k was defined to be the Cartesian position of the micro-manipulator end-effector, as measured by the capacitive sensor (see Section 2.5 in Chapter 2).

When performing system identification, an appropriate input signal has to be chosen, such that the system is excited properly. In this research, a chirp-signal was chosen—*i.e.*, a sinusoid with constant amplitude and linearly increasing frequency—as input, since this signal gives excitation in a well-defined frequency range. Consequently, the start and end frequencies of the chirp-signal have to be chosen based on the frequency range of interest. Considering the frequency spectra shown in Figure 3.3, a suitable range of excitation is 0–150 Hz, see Chapter 8 in [Johansson, 1993].

Data Preprocessing and Model-Order Selection

Prior to the identification, the input-output data were processed such that the mean and a possible linear trend were removed. In addition, the data (acquired at a sampling rate of 1 kHz) were low-pass filtered and subsequently decimated in order to avoid aliasing. The sampling frequency for the identification data were chosen based on the location of the eigenfrequencies of the dynamics along the different actuation axes, in order to avoid over-sampling of the inputs and outputs. The singular values related to the Hankel matrices of the experimental input and output data were computed during the identification procedure using the N4SID or MOESP algorithms, see, *e.g.*, [Van Overschee and De Moor, 1996]. By plotting these singular values in a diagram with a logarithmic scale, the gap between the essential dynamics of the system and the noise level was identified. Based on this information, a sufficient order of the model could be chosen for each of the data sets.

Identified Models of the Micro-Manipulator

Discrete-time state-space models of the form (3.1) for the linear dynamics along the x-, y-, and z-axis of the micro-manipulator system were estimated based on experimental data. First, models of the open-loop system were de-

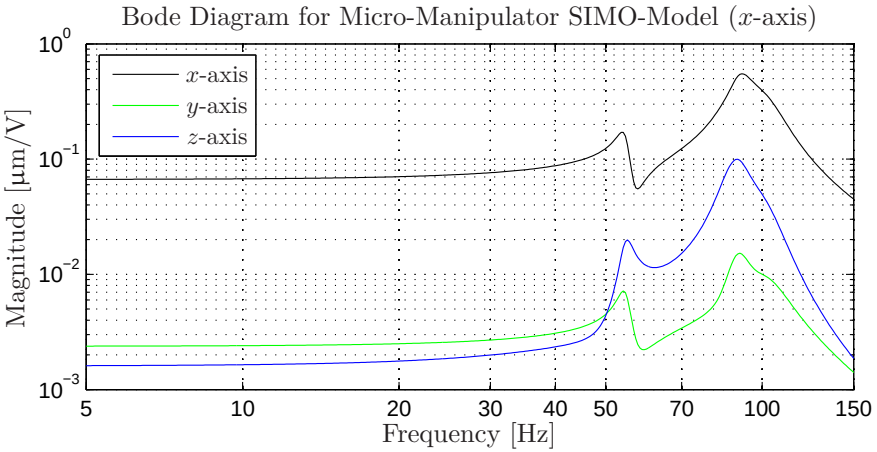

Figure 3.4 Magnitude plot for the Bode diagram of the discrete-time state-space SIMO-model of the micro-manipulator, with the scaled voltage to the x-axis piezo-actuator as input and the Cartesian end-effector positions as outputs.

termined. Second, models with the inner piezo-actuator control loops closed (see Section 3.5) were identified.

SIMO-Model Identification To the purpose of quantifying the cross couplings between the actuation axes of the micro-manipulator, identification of single-input multiple-output (SIMO)-models with the structure defined in (3.1) was performed. Experimental data were collected when exciting the piezo-actuators with chirp signals and measuring the corresponding Cartesian end-effector positions. The model input u_k was the input voltage to the actuator, scaled to a nominal interval, and the three outputs y_k were the end-effector positions. The magnitude plot for the Bode diagram of the identified model of order nine, obtained from the data when actuating along the x-axis, is provided in Figure 3.4. From the model, limited cross coupling between the x- and z-axes can be concluded, in particular when exciting close to the natural eigenfrequencies. This might lead to vibrations transversal to the actuation direction during machining. The corresponding magnitude plots for the Bode diagrams of the SIMO-models obtained with actuation along the y- and z-axes verify the coupling between the z- and x-axes, whereas they indicate that the y-axis is decoupled from both the x-axis and the z-axis.

Model Identification for Control Design During the dynamic characterization of the micro-manipulator and the SIMO-model identification in the previous paragraphs, minor cross couplings between the x- and z-axes were observed (see also Chapter 7). The assumption on decoupling was made in

the subsequent model-based control design. Instead, the cross coupling was handled as a disturbance in the controller. Consequently, for the control design, single-input single-output models were considered. The model orders vary for the different actuation directions, reflecting the number of natural eigenfrequencies, *cf.* the power spectral densities in Figure 3.3. The determined model orders for the micro-manipulator are 7, 6, and 2, for the x-, y-, and z-axis, respectively. The magnitude plots for the Bode diagrams of the identified models are shown in Figure 3.5. It is noted that there is good correspondence with the estimated power spectral densities of the outputs in Figure 3.3, when comparing the locations of the natural eigenfrequencies. The normalized root mean square errors (NRMSE), τ_{NRMSE}, is a measure of the fit of the models to the experimentally collected identification data, where 100% indicates complete model fit. Given N data points, this quantity is given by

$$\text{NRMSE} = 100 \times \left(1 - \frac{||\mathcal{W}_N - \widehat{\mathcal{W}}_N||_2}{||\mathcal{W}_N - \overline{\mathcal{W}}_N||_2} \right) \%, \tag{3.2}$$

where $\mathcal{W}_N$ is the validation output data, $\widehat{\mathcal{W}}_N$ is the output from the estimated model and $\overline{\mathcal{W}}_N$ is the mean of the validation output data. The NRMSE values for the identified models for the actuation along the x-, y-, and z-directions are 92.8%, 88.7%, and 96.0%, respectively. These values indicate that the identified models capture the essential dynamics of the micro-manipulator. Further, dynamic models of the format (3.1) were identified with the inner piezo-actuator control loops closed (see Section 3.5). The same procedure, algorithms, and evaluation strategy as for the open-loop models presented in this section were used. However, the models obtained from the identification using data from experiments with the inner loop closed are not presented in detail here.

3.5 Position Control of the Micro-Manipulator

The position-control problem of the micro-manipulator can be divided into two parts. First, the effects of the nonlinear dynamics in the piezo-actuators need to be reduced. Second, the oscillatory mechanical structure needs to be accurately position controlled. The specific control structure developed in this thesis to handle these properties is described next.

Inner Piezo-Actuator Control Loop

Three different strategies to control of the nonlinear dynamics of the piezo-actuators can be chosen: Model-based feedforward control, feedback control, or a combination of both. Several approaches to modeling of the hysteresis dynamics have been discussed in the literature. Three different categories

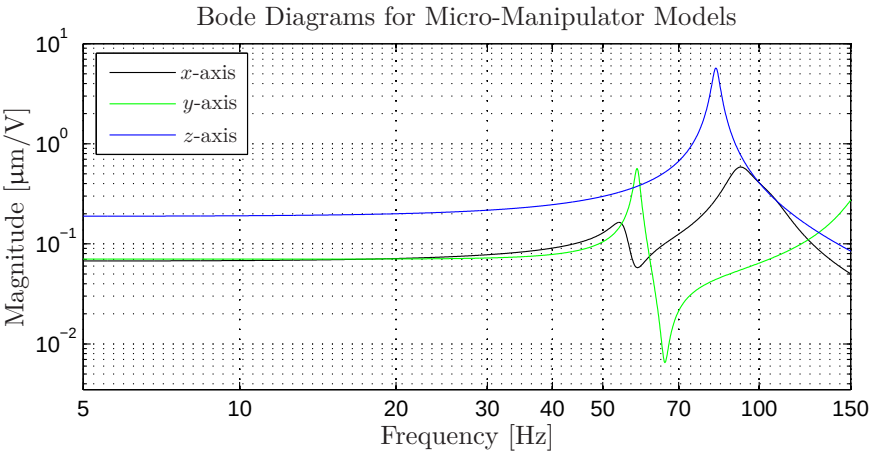

Figure 3.5 Magnitude plots for the Bode diagrams of the discrete-time state-space models identified using subspace methods, for the dynamics along the x-, y-, and z-axes of the micro manipulator. The locations of the natural eigenfrequencies can be compared to those in the power spectral densities in Figure 3.3. The differences in the low-frequency gain between the axes are explained by the different types of the respective piezo-actuator and the gear ratios.

of hysteresis models are common; approaches based on the Preisach model [Preisach, 1935; Ge and Jouaneh, 1996; Lei et al., 2011], the Prandtl-Ishlinskii model [Krasnosel'skii et al., 1989; Al Janaideh et al., 2008; Al Janaideh et al., 2009; Shan and Leang, 2009; Krejci and Kuhnen, 2001], and neural networks [Hastie et al., 2008; Xu, 1993]. By inverting the established model, a feedforward control integrated in the feedback controller has the potential of reducing the position error. Such a strategy was applied and investigated for the micro-manipulator control design in [Olofsson et al., 2012]. However, as the extensions of the piezo-actuators in the micro-manipulator were available for measurement with the strain gauges, an inner feedback loop can be closed around the nonlinear actuator and thus achieve sufficient control accuracy. The utilized controller is a PID controller, whose continuous-time control law can be stated, in its most simple form, according to

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau)\mathrm{d}\tau + K_d \frac{\mathrm{d}}{\mathrm{d}t} e(t), \tag{3.3}$$

where $e(t)$ is the difference between the reference value and the measured extension, and K_p, K_i, and K_d are controller parameters to be determined as part of the design procedure. The derivative part in the controller is low-pass filtered, in order to reduce the amplification of high-frequency noise

contaminating the measured signal from the strain gauges. Considering the intended application scenario for the micro-manipulator—*i.e.*, milling and other machining tasks—disturbances from the cutting process are to be expected. The PID controller also has to be accompanied by an anti-windup scheme [Åström and Hägglund, 2005], to handle the case when the controller saturates the actuators. Discretization of the continuous-time controller (3.3) for subsequent implementation in a digital signal processor is straightforward [Åström and Wittenmark, 1997].

In order to reduce the effects of the nonlinear dynamics in the piezo-actuators, the proportional gain K_p and the integral gain K_i should be increased as much as possible, while not resulting in a too high sensitivity to disturbances occurring during the milling. Further, the derivative part is important since it contributes with lead compensation in the system containing hysteresis. It will be shown by experimental results in Chapter 4 that this approach results in satisfactory performance of the control of the piezo-actuators for the intended application. The performance is in that context measured by the position accuracy achieved in the milling task execution.

Model-Based Feedback Control of the Micro-Manipulator

By utilizing the identified state-space models of the linear dynamics, a state-feedback control loop is designed for each of the three Cartesian actuation directions of the micro-manipulator. However, as discussed in Section 3.4, new models need to be identified after closing the inner feedback control loop for the piezo-actuators, where the reference signal to the inner PID control loop is considered as the input signal instead. With this approach, the effect of the nonlinear dynamics of the system is reduced and the nonlinear components are thus not degrading the performance of the identification of the remaining, approximately linear dynamics.

State feedback is an appropriate control structure for this kind of systems, since damping of the resonant modes in the micro-manipulator can be introduced by suitable control design. The control law for state feedback control of the system (3.1) can be written as

$$u_k = L(x_k^r - x_k) + u_k^f, \tag{3.4}$$

where the feedback gain matrix $L \in \mathbb{R}^{m \times n}$ is to be chosen, $x_k^r \in \mathbb{R}^n$ is the vector with reference values for the states, and $u_k^f \in \mathbb{R}^m$ is the feedforward control signal. The design procedure is to determine the gain matrix L by linear-quadratic (LQ) optimal control [Åström and Wittenmark, 1997], which provides a suitable parametrization of the trade-off between attenuation of the resonances in the system and the control signal utilization.

Since all states in the state-space models of the micro-manipulator are not available for direct measurement, a Kalman filter [Kalman, 1960] is intro-

duced for estimation of the states, based on the measured position signal, the system input, and the identified model. Consequently, the proposed control structure for the outer control loop is a linear-quadratic Gaussian (LQG) controller. The Kalman filter is organized as

$$\begin{aligned} \hat{x}_{k+1} &= \Phi\hat{x}_k + \Gamma u_k + K(y_k - C\hat{x}_k), \\ \hat{y}_k &= C\hat{x}_k, \end{aligned} \tag{3.5}$$

where the estimated states $\hat{x}_k$ and the estimated output $\hat{y}_k$ have been introduced, see [Åström and Wittenmark, 1997] for details. As previously mentioned, the Kalman filter gain matrix K for minimum-variance estimation of the states in the model is obtained from the identification procedure, since the noise model component is also included in the innovation model structure (3.1).

In order to eliminate stationary errors in the position-control loop, integral action is also introduced in the state feedback controller. Integral action can be introduced in different ways; either using a disturbance observer or using an additional integral state [Åkesson and Hagander, 2003]. Since it was assumed in the modeling phase that the systems were of SISO format, the integral action is thus achieved by extending the state vector with the integral state x_k^i according to

$$x_{k+1}^i = x_k^i + h(r_k - y_k), \quad x_0^i = 0, \tag{3.6}$$

where the sample period h and the position-reference signal r_k have been introduced. With this extra state, it is required that the state-feedback gain vector L is augmented with one element and subsequently redesigned to accommodate the introduced integral state. Different approaches can be chosen to handle the feedforward control signal. In the control architecture presented in this thesis, the feedforward control u_k^f is chosen as a direct term from the reference signal, *i.e.*, $u_k^f = l_r r_k$. The parameter l_r influences the response of the controller to changes in the reference signal.

To summarize the discussion on the position controller for the micromanipulator, the developed structure is displayed in Figure 3.6.

3.6 Control Architecture

To the purpose of establishing a macro/micro-manipulator configuration for milling with industrial robot manipulators, a controller for the micromanipulator tool position was developed in the previous section. In this section, the robot controller is described, and both of the manipulator controllers are integrated using a mid-ranging control strategy (see, *e.g.*, [Allison and Isaksson, 1998]) for achieving the desired frequency separation of the compensation.

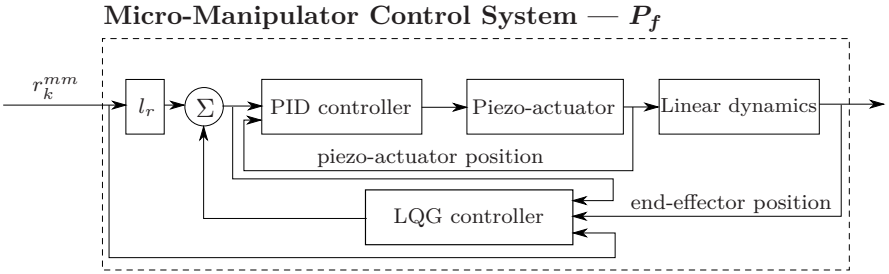

Figure 3.6 Control structure for the micro-manipulator. The figure illustrates the control strategy for one axis, but the control structures are identical (except parameters) for all three Cartesian actuation axes. The external input r_k^{mm} is the position reference for the micro-manipulator.

Robot Controller The geometric path to be tracked is planned offline using appropriate CAM software. The position and orientation deviations occurring during the milling are computed based on the measurements from the optical tracking system. The orientation deviations are compensated for by the macro-manipulator alone, whereas the position deviations were fed to a mid-ranging control structure, which separated the error in the frequency domain for subsequent compensation by the macro- and micro-manipulators jointly. As the position and orientation deviations during the milling were measured in Cartesian space, the macro-manipulator corrections computed by the mid-ranging control structure needed to be transformed to joint space. To this purpose, the Jacobian matrix, expressing the differential kinematics for the end-effector coordinate system of the robot, was used [Spong et al., 2006]. Assuming small position and orientation corrections, the Cartesian deviation $\delta x \in \mathbb{R}^6$ was transformed to joint space using the inverse of the Jacobian matrix $J(q) \in \mathbb{R}^{6\times 6}$, which is dependent on the joint-angle configuration $q \in \mathbb{R}^6$, according to

$$\delta q \approx J^{-1}(q)\delta x \tag{3.7}$$

where δq is the corresponding joint-angle correction. Given the expected range of the Cartesian position corrections, which are smaller than 1 mm, the approximation using the Jacobian matrix stated in (3.7) can be justified.

Micro-Manipulator Controller The micro-manipulator is controlled and actuated directly in Cartesian space with the strategy presented in Section 3.5. The complete model-based solution is illustrated for one axis in Figure 3.6, with the reference value r_k^{mm} as input and the corresponding end-effector position as output. The reference position r_k^{mm} for each actuation axis is determined by a mid-ranging control architecture, described in the next paragraph.

Mid-Ranging Control As mentioned earlier, the micro-manipulator has a limited total workspace of approximately 0.5 mm along each Cartesian axis, and thus, the manipulator may reach its actuation limits when performing advanced milling tasks, where strong process forces are required. Therefore, by employing the concepts of frequency separation for robots from [Sörnmo et al., 2013], a mid-ranging approach is considered in order to control the relative position between the macro- and micro-manipulators. In the implementation, each Cartesian axis is considered and controlled separately. The mid-ranging control strategy makes use of the higher bandwidth of the micro-manipulator, while striving to keep the position of the same close to the midpoint of its workspace. The purpose is that the actuation limits of the fast actuator, *i.e.*, the micro-manipulator in this case, should not be reached [Allison and Isaksson, 1998]. The control approach can be seen as a separation of the position error, computed from the tracking sensor data, in the frequency domain, that lets the fast micro-manipulator handle the high-frequency deviations, and the comparably slow macro-manipulator handle the low-frequency deviations and bring back the fast actuator to it. Several mid-ranging control architectures have been proposed previously, one of the most common is the valve position control (VPC) [Allison and Isaksson, 1998; Allison and Ogawa, 2003]. In this thesis, the VPC control structure is employed because of its widespread presence in applications in literature, and its appealing properties in terms of low complexity and robustness to model uncertainty.

The complete closed-loop control system for the integrated macro/micro manipulator control, using the VPC control architecture, is displayed in Figure 3.7, where C_f and C_s are controllers, here chosen as PI controllers. The systems P_f (displayed in Figure 3.6) and P_s represent the closed-loop Cartesian position-control systems of the micro- and macro-manipulators, respectively. The relative position between the manipulators along each Cartesian axis, which is measured in real-time during the task execution by the optical tracking system, is denoted p_k^r, the relative position reference is denoted r_k^r, and the desired setpoint for the mid-ranged input is denoted r_k^m. The latter is for the current setup chosen such that the micro-manipulator should work in the middle of its working range, for each actuation axis. The tuning guidelines that are provided in [Allison and Ogawa, 2003] are based on so called lambda-tuning for first-order models of P_f and P_s. For the considered setup, however, the models of the micro-manipulator dynamics in the three Cartesian directions range from order 2–7, see Section 3.4, and the open-loop dynamics contain poorly damped resonances. Even though a significant damping of the resonances in the micro-manipulator is achieved with the LQG controller, the closed-loop system naturally cannot be described using first-order systems. Hence, using the tuning guidelines proposed in previous references will not result in satisfactory performance. Therefore, the param-

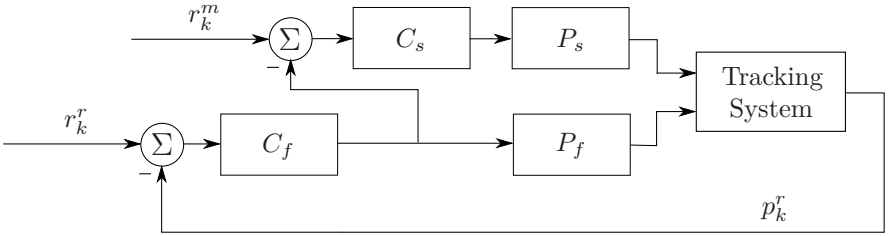

Figure 3.7 The complete control architecture, with integrated macro/micro feedback control for each Cartesian actuation axis, for the proposed milling cell, using the mid-ranging VPC control architecture as a basis. The superscripts m and r on the signals denote mid-range and relative, respectively.

eters obtained from the proposed design procedure in [Allison and Ogawa, 2003] were used as initial values, and were subsequently experimentally tuned in order to obtain the desired performance.

3.7 Conclusions

In this chapter, a model-based position-control architecture for a micro-manipulator for milling tasks with industrial robots was developed. Experimental data was collected, whereby models describing the linear dynamics of the system were determined. The control law was based on an LQG controller in order to damp the resonances of the micro-manipulator. Subsequently, a control architecture integrating the micro-manipulator controller with the robot controller and an optical tracking system for measuring the relative position and orientation between the macro- and micro-manipulators was presented.

4

Experimental Evaluation of the Micro-Manipulator

4.1 Introduction

To the purpose of experimental verification and evaluation of the proposed position controller for the micro-manipulator in Chapter 3, several milling tasks were executed with online position-error compensation. In this evaluation, the prototype experimental setup defined in Section 2.5 in Chapter 2 (see Figure 2.7) was employed, and thus only the micro-manipulator was used for compensation. To quantify the performance, the surface accuracies of the machined workpieces achieved with and without online compensation are compared and evaluated. This chapter of the thesis is derived, in parts, from the publications [Olofsson et al., 2011a] and [Sörnmo et al., 2012a].

The structure of this chapter is as follows: Section 4.2 presents the results from the performed milling tasks. The obtained experimental results are evaluated in Section 4.3. Further, based on the presented experiments, a discussion of the results and the proposed control architecture and setup for machining tasks is provided in Section 4.4. Finally, conclusions are drawn in Section 4.5.

4.2 Experimental Results

Initially, the micro-manipulator controller was applied in a milling experiment in aluminium in order to evaluate the performance of the controller. Subsequently, extensive milling experiments in aluminium were performed. The results of these experiments are presented next.

Experimental Verification of the Controller

In order to verify the operation of the control architecture, an initial milling experiment with online compensation along the x-axis of the micro-

manipulator was performed. The results of the experiment are displayed in Figure 4.1. It is clear that the controller is working satisfactory in this experiment, with a control error within ± 10 μm.

Experimental Validation in Milling Scenarios

With the prototype experimental setup described in Section 2.5 in Chapter 2, milling in a rectangular block of aluminium was performed. The industrial robot can be reconfigured such that milling is possible with compensation along all three directions of the micro-manipulator (see Figure 2.7 in Chapter 2). Results obtained during face milling in the x-actuation direction and peripheral milling in the y- and z-actuation directions of the micro-manipulator are presented in this thesis. The experiments were performed with a material feed-rate of 7.5 mm/s, a spindle speed of 28 000 rpm, and a depth-of-cut of 1 mm in the face-milling experiments and 1×10 mm in the peripheral-milling experiments.

Milling Experiments with Compensation

First, a set of milling experiments with online position compensation, utilizing the micro-manipulator, was performed. Each of the experiments in the different actuation axes of the micro-manipulator is described next.

***X*-direction** In the first setting, a face milling was performed, where the surface orthogonal to the x-axis of the micro-manipulator was to be machined. Consequently, the micro-manipulator is controlled in this direction. The result of the milling experiment is displayed in Figure 4.2. The control error is defined as the difference between the reference value to the micro-manipulator control system and the signal from the capacitive sensor measuring the position of the micro-manipulator end-effector along the x-direction.

***Y*-direction** The milling accuracy has further been evaluated in a peripheral-milling experiment, where the compensation was performed along the y-axis of the micro-manipulator. It should be noted that this milling task is different from the face milling presented in the previous paragraph, in the sense that the process forces affect the robot along different directions. Moreover, the experiment was designed such that the robot path is not completely perpendicular to the compensation direction. This situation can be considered as a result of a poorly calibrated workpiece or industrial robot. By utilizing the micro-manipulator, this effect can be compensated for online, since the motion of the robot is tracked in real-time using the laser sensor.

The result of the milling experiment is displayed in Figure 4.3. The control error displayed is defined analogously to the case with face milling along the x-direction of the micro-manipulator.

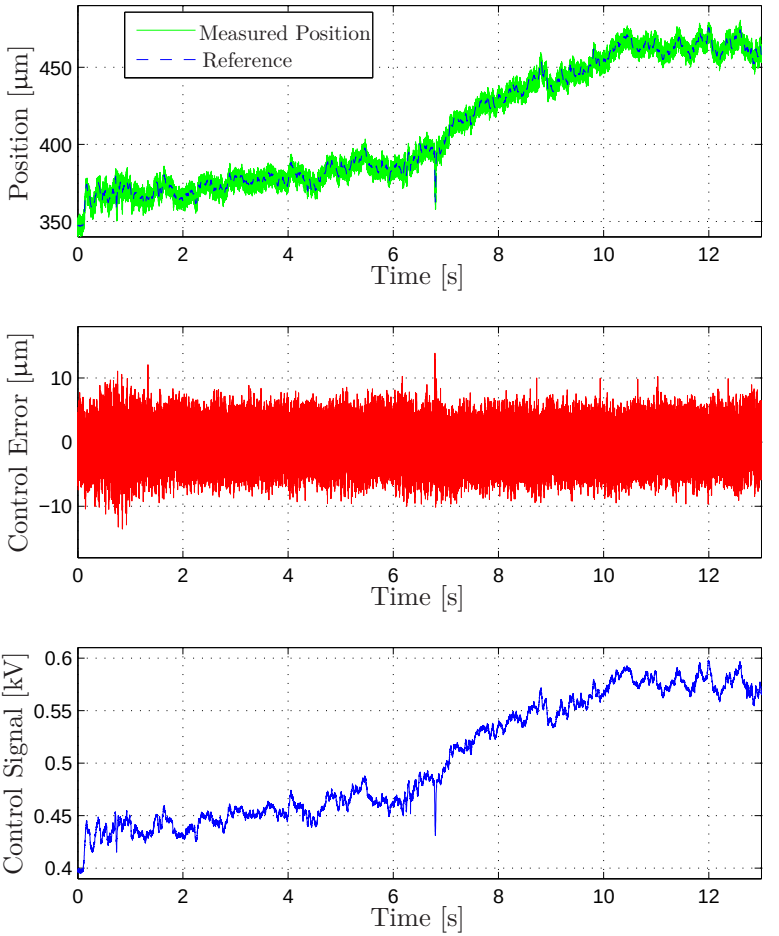

Figure 4.1 Experimental verification of the proposed control architecture in a milling scenario with compensation along the x-axis of the micro-manipulator. The control signal is the voltage applied to the piezo-actuator. The fast compensation achieved at approximately $t = 6.8$ s is noticeable, given that the error still is below ± 12 µm.

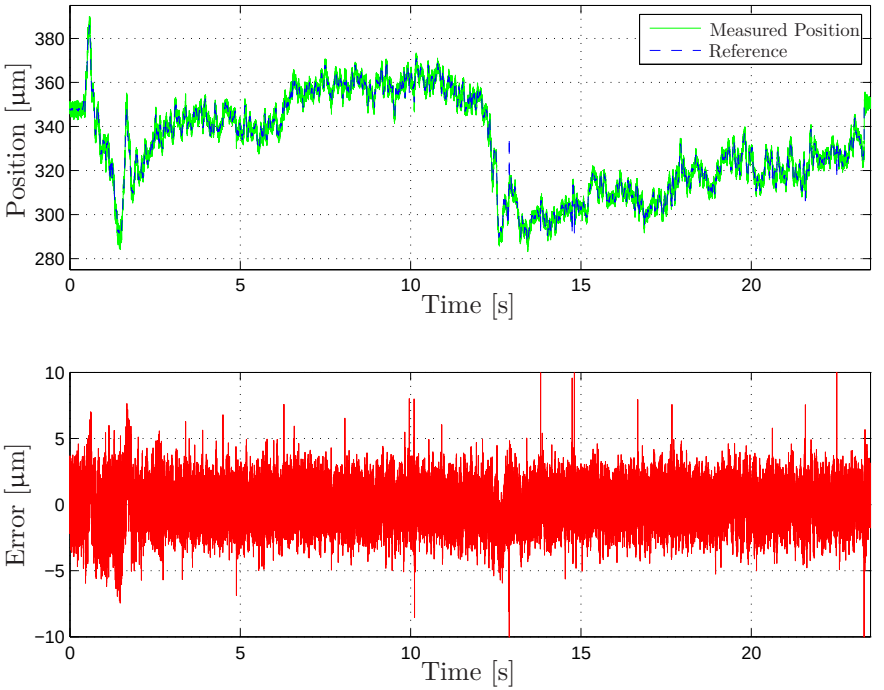

Figure 4.2 Reference value and measured position of the micro-manipulator during face-milling experiment with compensation along the x-direction (upper panel) and corresponding control error (lower panel).

***Z*-direction** The third experiment performed was a peripheral milling with compensation along the z-axis of the micro-manipulator. The experiment is similar to the one discussed in the previous paragraph; however, the compensation is performed along a different axis of the micro-manipulator and in addition the configuration of the robot is different. The control performance of the micro-manipulator in the milling experiment is displayed in Figure 4.4.

Milling Experiments Without Compensation

The same milling experiments described and presented in the previous subsection were repeated, but with the machining spindle rigidly attached to a base as shown in Figure 2.8 in Chapter 2—*i.e.*, no online position compensation was active. The results of the experiments are evaluated and contrasted to the results obtained with online position compensation in the next section.

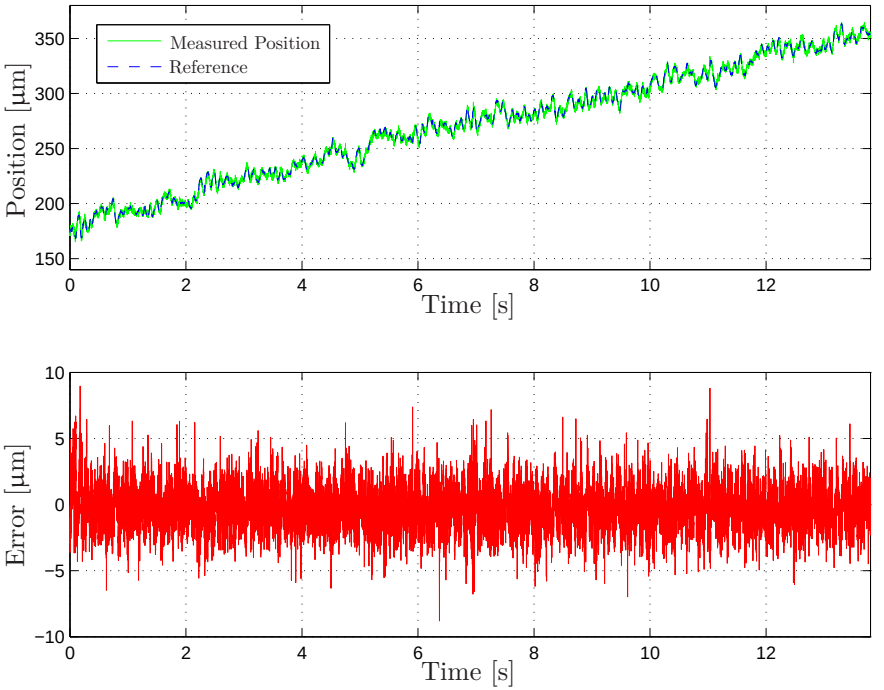

Figure 4.3 Reference value and measured position of the micro-manipulator during milling experiment with compensation along the y-direction (upper panel) and corresponding control error (lower panel).

4.3 Experimental Evaluation

In this section, the milling results obtained in experiments are evaluated using statistical methods. Further, the surface roughness of the machined aluminium workpieces are measured and compared, so as to quantify the milling performance.

Coherence Spectra

An important aspect to consider in the controller design is, if the nonlinear dynamics in the piezo-actuators influence the frequency characteristics of the position controlled micro-manipulator, or if the proposed control architecture is able to reduce its influence to a sufficient level. To that purpose, the quadratic coherence spectrum [Johansson, 1993], *i.e.*,

$$\gamma_{uy}(\omega) = \frac{|S_{uy}(i\omega)|^2}{S_{uu}(i\omega)S_{yy}(i\omega)}, \tag{4.1}$$

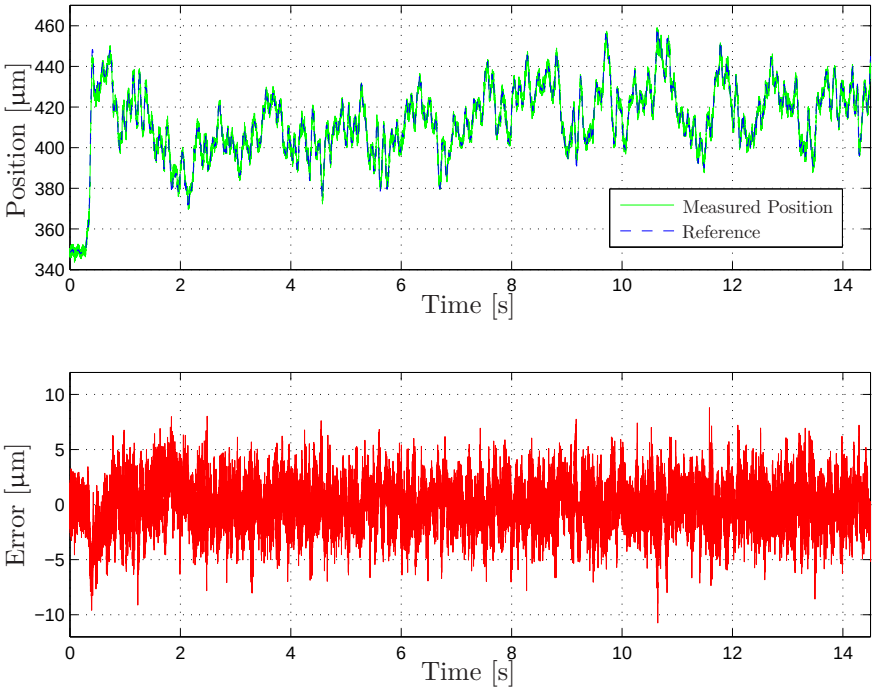

Figure 4.4 Reference value and measured position of the micro-manipulator during milling experiment with compensation along the z-direction (upper panel) and corresponding control error (lower panel).

where $S_{uy}(i\omega)$ is the power cross-spectrum between the input u and the output y, $S_{uu}(i\omega)$ and $S_{yy}(i\omega)$ are the autospectra for u and y, respectively, is investigated. The coherence spectra for the dynamics along the x-, y-, and z-axes of the micro-manipulator when exciting with a chirp signal are displayed in Figure 4.5. The diagrams have been obtained with the inner PID controller loops for the piezo-actuators active.

It is observed that the relation between the input and the output along the x-, y-, and z-directions appears to be linear in the major parts of the frequency range of interest. However, in the respective actuation direction, the effects of the natural eigenfrequencies of the structure, of which some exhibit low damping, are clearly visible. Hence, the parts of the frequency range where the coherence is well below one can be derived to these frequencies, *cf.* the frequency spectra for the prototype setup of the micro-manipulator in [Olofsson et al., 2011a]. A plausible interpretation of this observation is that when the micro-manipulator is in resonance, the nonlinear dynamics in the

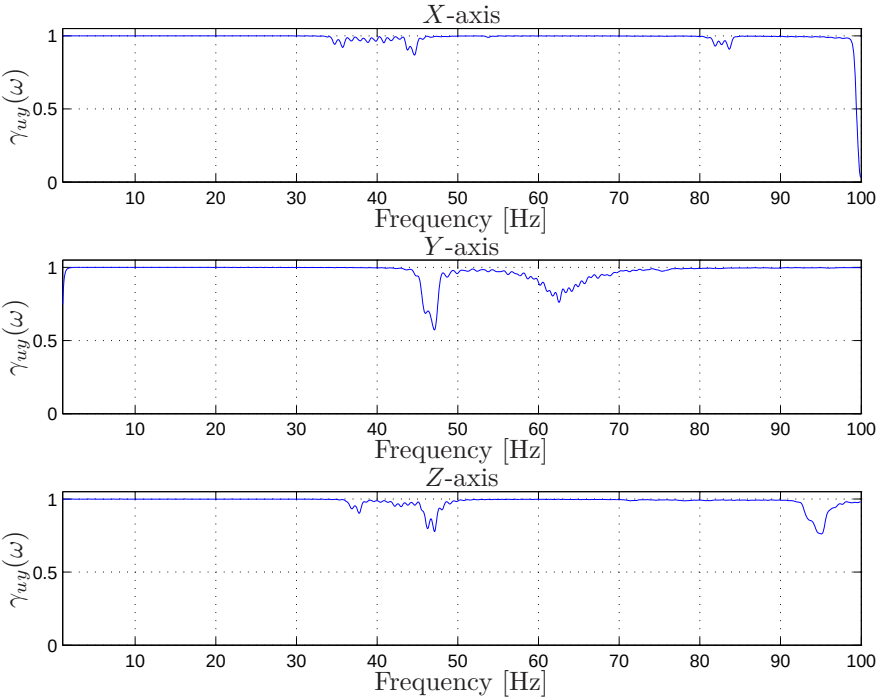

Figure 4.5 Estimated coherence spectra—obtained with the inner PID control loop for the piezo-actuators active—for the dynamics along the x, y, and z actuation directions of the micro-manipulator. Note that the data visualized are for the prototype version of the micro-manipulator, which is employed in this chapter.

structure is more prominent and the linear system description is insufficient. Nevertheless, since the major part of the frequency range exhibits a linear relation, it is an indication that the chosen modeling and control approach can be justified.

Frequency Analysis of the Control Error

From a control theory point of view, the results obtained from the milling experiments should be evaluated by examining if there is more information available in the control error—*i.e.*, separating the noise in the measurements from the possibly available information, which should be acted upon. To this purpose, auto-regressive moving average (ARMA) models as well as frequency spectra [Johansson, 1993] of the control errors presented in Figures 4.2–4.4 were estimated. The latter were estimated using Welch's method

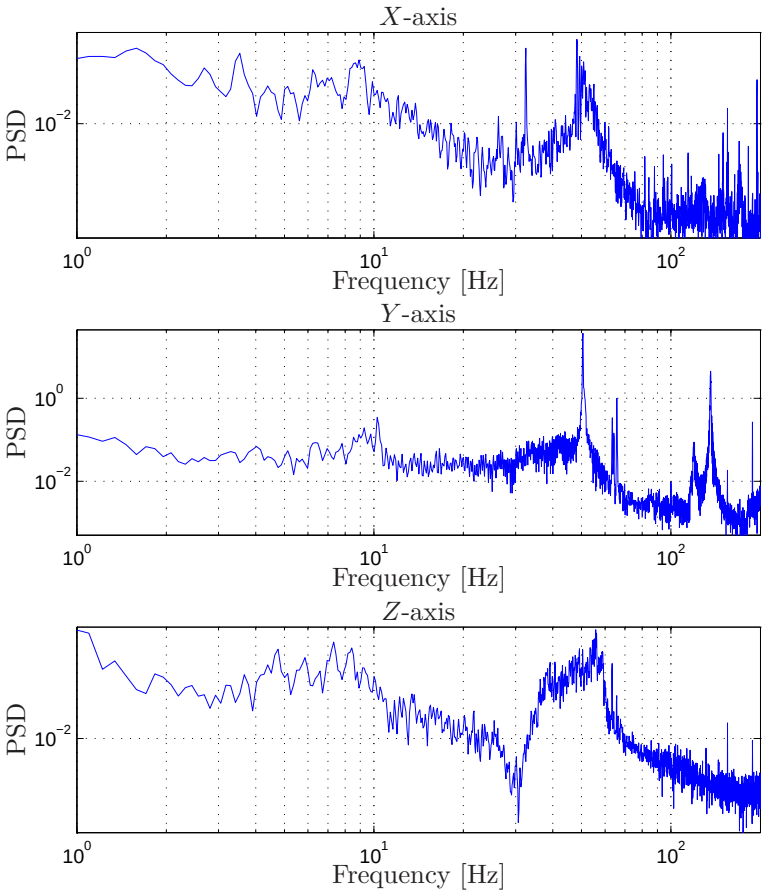

Figure 4.6 Estimated power spectral densities for the control error in Figures 4.2–4.4, measured during milling experiments with compensation along all actuation directions of the micro-manipulator.

in MATLAB. The estimated power spectral densities (PSD) for the control error in the performed milling experiments are displayed in Figure 4.6. The spectra are further discussed in Section 4.4.

Measurement of Milling Profiles

Since the main objective of the micro-manipulator is to achieve a high position accuracy of the machined surface of the workpiece, a Mahr surface measurement device of model M400 SD26 [Mahr GmbH, 2011] was utilized

to measure the surface roughness of the obtained profiles on the aluminium workpieces. The device is equipped with a sensitive probe which was sliding along the surface to be measured while the profile was recorded. Further, the measurement device was calibrated such that it had a measurement accuracy better than 1 μm.

Milling with Compensation The results of the surface-roughness measurements, for the three milling experiments in Figures 4.2–4.4 where online position compensation with the micro-manipulator was active, are displayed in Figure 4.7. The figures show the accuracy for randomly selected parts of the workpieces, each having a width of 25 mm. The measured profiles indicate that the milling accuracy in the x- and y-directions are within ± 7 μm and that the error of the measured milling profile is within approximately ± 12 μm in the z-direction of the micro-manipulator. Furthermore, it is noted that the measured profiles correspond well to the measurements from the capacitive sensors attached to the micro-manipulator, which are used for the position-control feedback. This correspondence provides experimental validation that the measured position of the micro-manipulator agrees with the actual position of the milling tool. Photos of the machined surfaces obtained in the experiments with compensation along the x-, y-, and z-directions are provided in Figures 4.9–4.11, respectively.

Milling Without Compensation The resulting surface roughness of the profiles from the uncompensated milling experiments is displayed in Figure 4.8. To evaluate the quality of the measured profiles from the experiments with online compensation, compared to the profiles obtained in milling without compensation, both the maximum peak-to-peak error e_m and the standard deviation σ_e of all profiles were computed. The standard deviation σ_e was computed as

$$\sigma_e = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(e_i - \bar{e})^2}, \tag{4.2}$$

where N is the number of data points and $\bar{e}$ is the mean of the error. Table 4.1 shows the maximum error of the profiles, computed as the minimum value subtracted from the maximum value, and the standard deviations from the nominal profile.

4.4 Discussion

Given the results in Table 4.1, it is evident that online compensation with the micro-manipulator has improved the milling accuracy significantly, compared to the uncompensated case. From the experimental evaluation presented in this chapter, it can be concluded that the control error in the

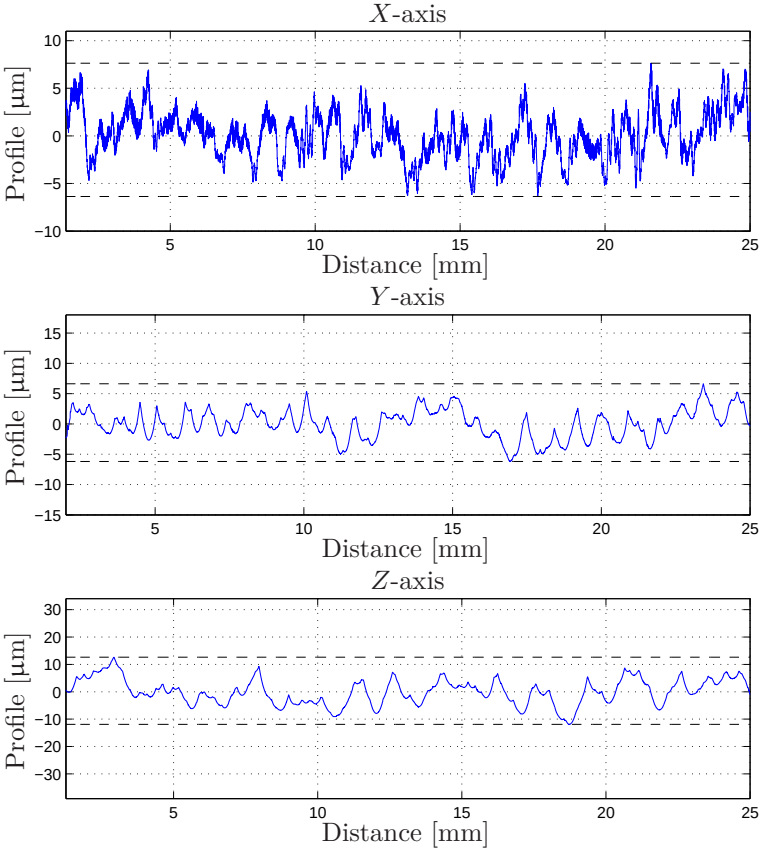

Figure 4.7 Position profiles of the machined surfaces after face milling along the x-direction and peripheral milling experiments with compensation along the y- and z-directions of the micro-manipulator. In all experiments, online position corrections with the micro-manipulator was performed along the corresponding actuation axis. The dashed lines indicate the maximum and minimum values.

micro-manipulator controller is below ± 12 µm for all axes, which is well below the desired accuracy of 100 µm. Moreover, the standard deviation of the milling profiles has been decreased by using the considered micro-manipulator and developed control structure. This means that the micro-manipulator not only increases the accuracy of the milling, but also that the stationary and low-frequency errors in the robot position can be handled. In the experiments, the required compensation range was lower than the total working range of the micro-manipulator because of the limited process forces when machining in aluminium. Thus, the micro-manipulator was sufficient in these

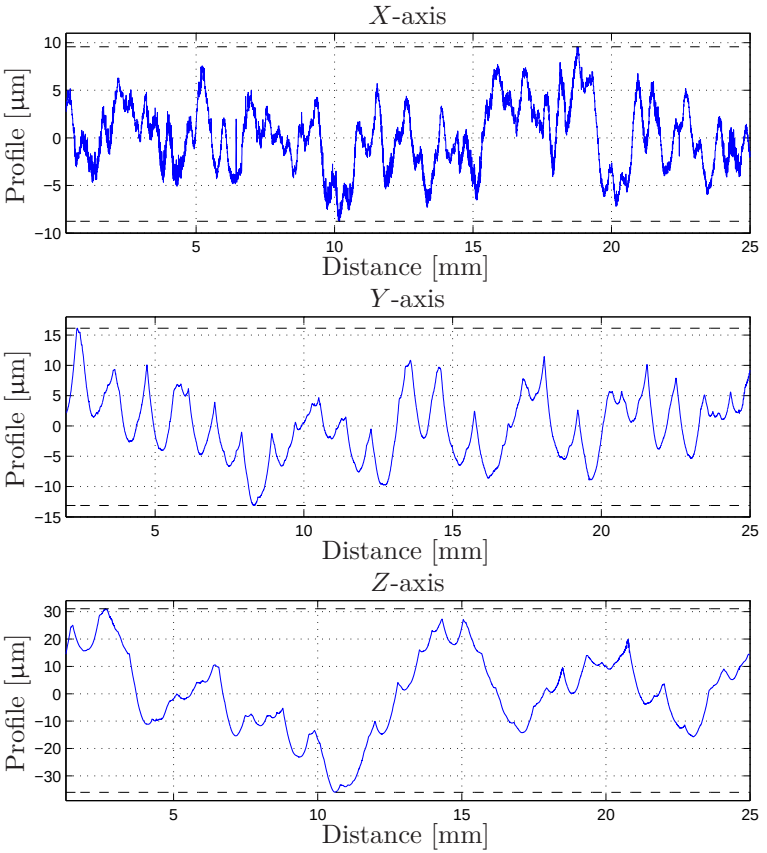

Figure 4.8 Position profiles of the machined surfaces after uncompensated milling experiments along the x-, y-, and z-directions of the micro-manipulator, respectively. The dashed lines indicate the maximum and minimum values. The profiles should be compared with the corresponding quantities in Figure 4.7, obtained with online compensation.

experiments. Nevertheless, to fully utilize the potential of the macro/micro control principle, the complete architecture with simultaneous compensation using the mid-ranging control architecture discussed in Section 3.6 in Chapter 3 needs to be used. Such experiments were performed with milling in steel using the final robot-cell setup and the results are presented in Chapter 5.

Several conclusions can be drawn from the frequency analysis of the control error. All frequency spectra of the control errors in Figure 4.6 exhibit peaks at approximately 10 Hz and at 50 Hz. The higher frequency is a disturbance from the power network system and hence inherent in an industrial environment with a multitude of potential disturbances. The lower frequency

Table 4.1 Maximum error e_m and standard deviation σ_e of milling profiles obtained with and without online position compensation utilizing the micro-manipulator.

Axis	e_m compensated (µm)	e_m uncompensated (µm)	Ratio
x	14.0	18.3	1.3
y	12.8	29.3	2.3
z	24.5	67.0	2.7
Axis	**σ_e compensated (µm)**	**σ_e uncompensated (µm)**	**Ratio**
x	2.8	7.6	2.7
y	2.5	5.6	2.2
z	4.7	14.9	3.2

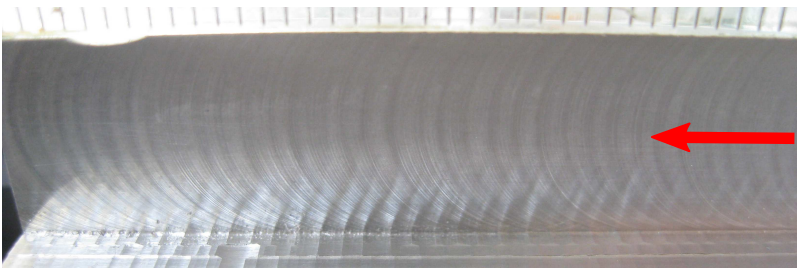

Figure 4.9 Workpiece after face milling on the surface indicated by the red arrow, with compensation along the x-direction of the micro-manipulator (*i.e.*, along the normal of the surface marked with the red arrow).

peak relates to the eigenfrequencies of the industrial robot in the corresponding Cartesian directions. This is experimentally confirmed by modal analysis of the REIS RV40 robot [Schneider, 2010]. However, while the peaks are visible in the frequency spectra, they are not prominent. This indicates that the micro-manipulator controller can attenuate one of the most important disturbances during the milling—*i.e.*, the natural eigenfrequencies of the robot. Further, the influence of the mechanical design of the micro-manipulator on the milling performance is visible in the frequency spectra presented in Figure 4.6. The zero in the transfer function for the dynamics along the z-direction of the micro-manipulator at 30 Hz is clearly visible in the corresponding frequency spectrum. Similarly, one of the natural eigenfrequencies of the micro-manipulator dynamics along the x-axis at 32 Hz is visible. The

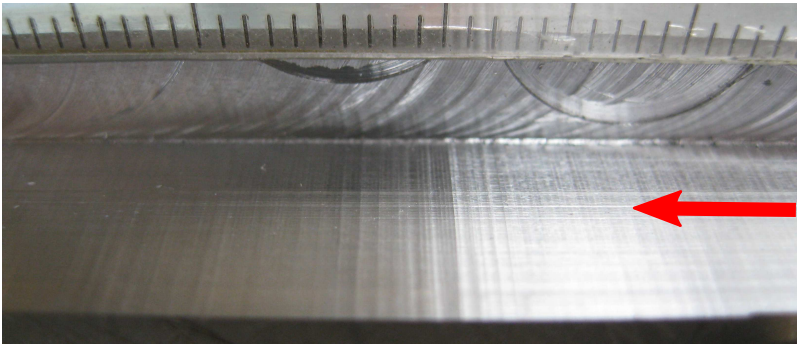

Figure 4.10 Workpiece after peripheral milling on the surface indicated by the red arrow, with compensation along the y-direction of the micro-manipulator (*i.e.*, along the normal of the surface marked with the red arrow).

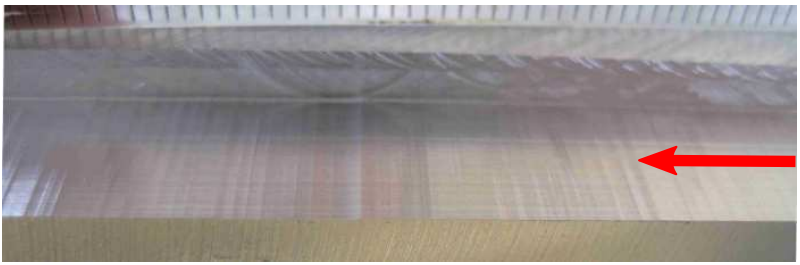

Figure 4.11 Workpiece after peripheral milling on the surface indicated by the red arrow, with compensation along the z-direction of the micro-manipulator (*i.e.*, along the normal of the surface marked with the red arrow).

bandwidth of the closed-loop position control for the micro-manipulator is consequently limited by the mechanical design. This limitation is relaxed using the improved version of the micro-manipulator in the final setup, employed in the machining experiments in Chapter 5.

4.5 Conclusions

This chapter has investigated how the milling accuracy of industrial robots could be improved by utilizing a high-bandwidth, piezo-actuated, micro-manipulator for demanding machining tasks. It was shown in an experimental validation procedure, comprising several different milling experiments with online position compensation with the micro-manipulator, that the developed

method offers significantly higher accuracy in terms of surface roughness, compared to the standard method for milling without online compensation.

5

Experimental Evaluation of the Macro/Micro Architecture

5.1 Introduction

In the previous chapter, the prototype version of the micro-manipulator was evaluated in milling experiments in aluminium, where only one-dimensional compensation with the micro-manipulator was employed. In this chapter, the complete control architecture with integrated macro/micro-actuation is evaluated in comprehensive milling experiments with full 6D compensation. The final version of the robot cell described in Section 2.5 in Chapter 2 (see Figure 2.9) was used for the experiments. The results are evaluated using statistical methods and the workpiece accuracy is evaluated using surface roughness and CMM measurements. This chapter is derived, in parts, from the experimental results presented in [Schneider et al., 2014].

This chapter of the thesis is organized as follows: The experimental protocol is defined in Section 5.2, and the obtained experimental results are presented in Section 5.3. Finally, the obtained results and the complete integrated approach to milling using macro/micro-actuation are discussed in Section 5.4, where conclusions also are drawn.

5.2 Experimental Protocol

Using the complete robot-cell setup described in Section 2.5 in Chapter 2 with the second generation of the micro-manipulator, a toroid with a rectangular cross section and a mid-diameter of 70 mm was machined with a depth of 0.5 mm in full width cut, see Figure 5.1. A Holex solid carbide milling tool with four teeth and a diameter of 8 mm was used. Further, a feedrate of

5 mm/s and a spindle speed of 28 000 rpm were used. The magnitude of the process forces required for this milling was approximately 100 N. Because of the optical tracking of the machining spindle and the robot end-effector, no lubrication or cooling, potentially obstructing the line-of-sight, were used. To the purpose of evaluation of the developed method to machining and comparison to previous approaches, three different configurations of the setup were considered and evaluated:

A. Position-controlled macro-manipulator with the CNC control system, and the spindle attached fixed to a rigid base.

B. Position-controlled macro-manipulator with the CNC control system, the spindle fixed to a rigid base, and online workspace position and orientation corrections of the macro-manipulator based on feedback from measurements obtained with the optical tracking system.

C. The complete setup defined in Section 2.5 in Chapter 2, including the macro- and micro-manipulators, the optical tracking system, and the mid-ranging control architecture presented in Section 3.6 in Chapter 3.

Two separate machining spindles were used during the course of the experimental validation; one was fixed to a rigid base and one was attached to the micro-manipulator (*cf.* Figure 2.8 in Chapter 2 illustrating the corresponding arrangement in the prototype setup). In Figure 2.9 in Chapter 2, the spindle attached to the micro-manipulator is shown. Similar to the milling experiments presented in Section 4.2 in Chapter 4, the spindle positions in the robot workspace were arranged such that the robot configuration was mirrored in the robot center plane in the two setups, in order to achieve comparable machining results.

5.3 Experimental Results

The toroid milling experiment was performed employing each of the Setups A–C. In the first part of this section, the results from Setup C are presented. The results from Setups A and B are further evaluated and compared in later subsections. The results obtained from the milling experiment with Setup C are displayed in Figures 5.2–5.3. Figure 5.2 shows the control performance of the micro-manipulator tracking during a representative part of ten seconds of the toroid milling, with compensation along each of the x-, y-, and z-axes. The definition of the axes is according to Figure 2.10 in Chapter 2, and the error is defined as the difference between the reference value r_k^{mm} sent to the micro-manipulator from the mid-ranging control architecture, see Figure 3.6 in Chapter 3, and the corresponding position of the micro-manipulator end-effector. Further, the relative Cartesian errors,

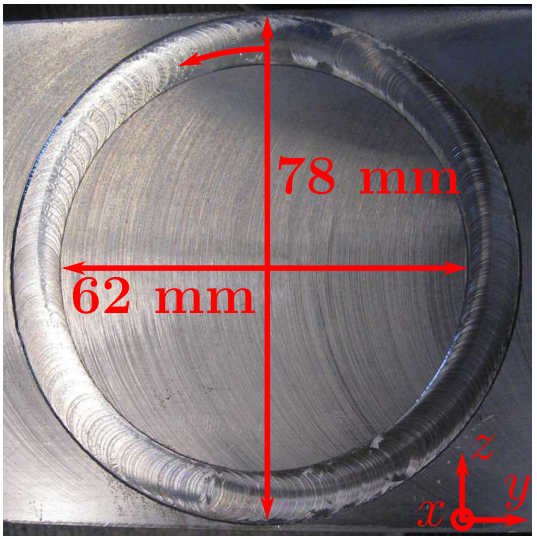

Figure 5.1 The milling geometry considered for evaluation of the proposed robotic machining cell. The milling path starts in the upper center point and continues in the counterclockwise direction.

defined as the difference between the origin of the tool coordinate system and the origin of the workpiece coordinate system and computed from the measurements from the optical tracking system, are displayed in Figure 5.3 for the first 20 seconds of the milling task.

As a first observation, the initial transient when the tool enters the workpiece can be seen in Figure 5.3 at approximately $t = 1.2$ s. Moreover, starting with the results for the x-axis, it should be noted that this direction corresponds to the face direction of the milling tool, see Figure 5.1, and is therefore orthogonal to the machined surface. Hence, the required compensation is smaller than in the other two Cartesian directions. Because of the alignment of the tool axis and this compensation axis, the x-axis is sensitive to milling-process disturbances and as a result, low-amplitude oscillations were present.

The major compensation with the micro-manipulator was performed along the y- and z-axes. The reference signal along the y-axis (see Figure 5.2) exhibits low-frequency oscillations of a comparably high amplitude, which were induced by the mechanics of the robot when moving linearly in the workspace. It is also to be noted that the effects of the robot mechanics exhibit configuration dependency. Nevertheless, the micro-manipulator was effective in compensating for the oscillation, which would be impossible to eliminate using the macro-manipulator solely. Considering the alignment of

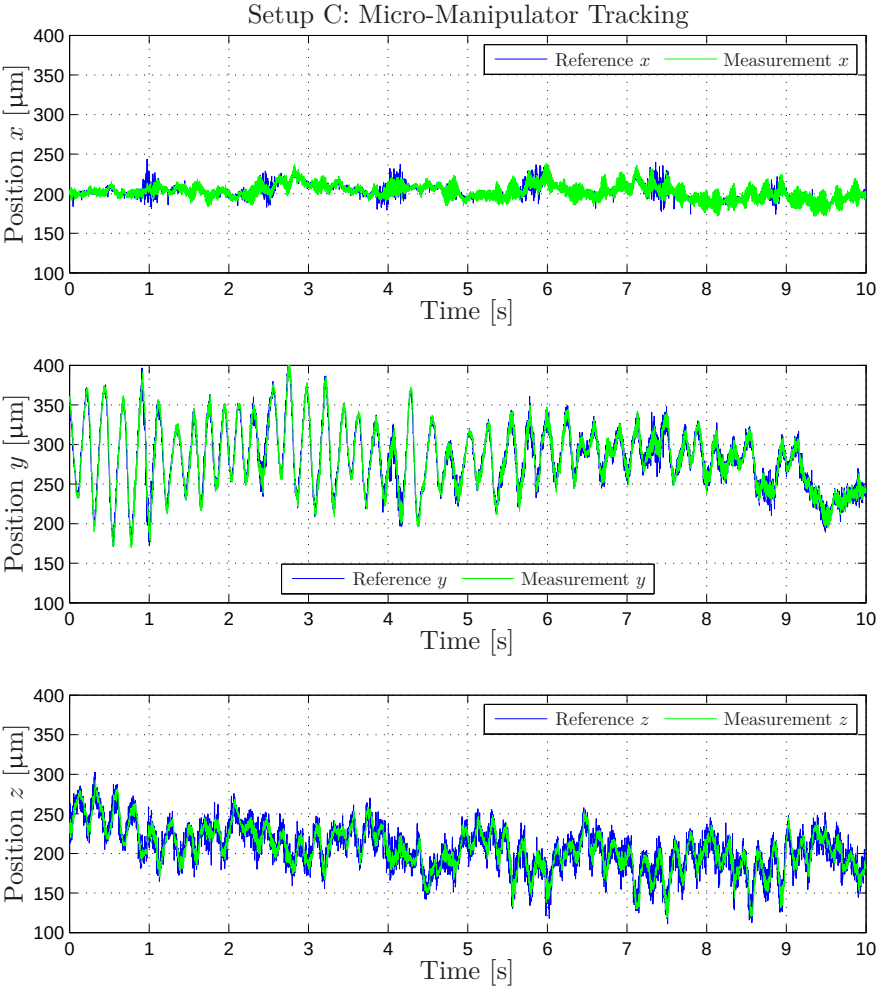

Figure 5.2 Micro-manipulator tracking performance during machining with compensation using Setup C, along each Cartesian axis x, y, and z.

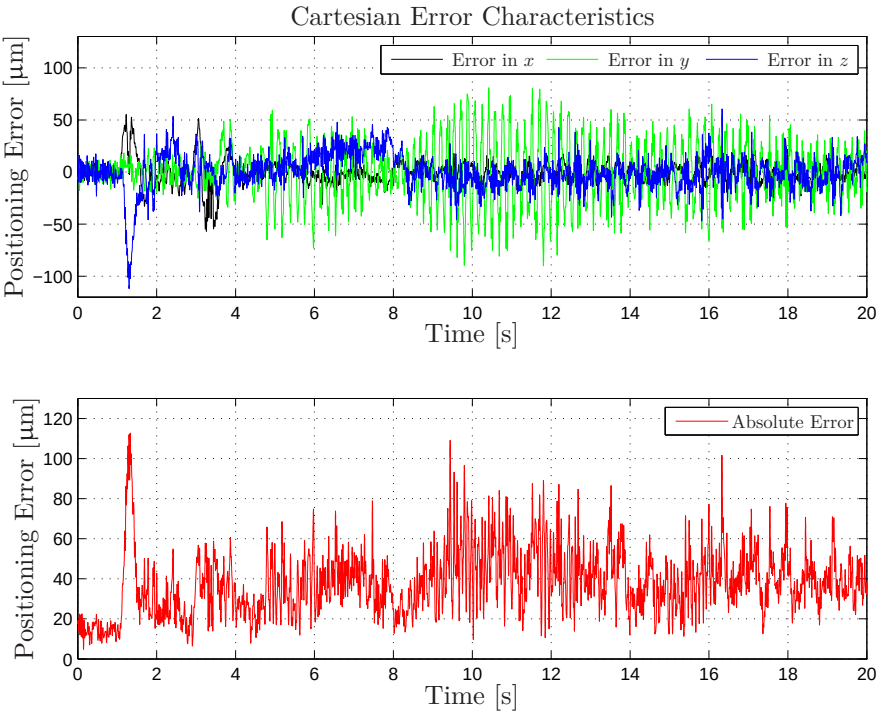

Figure 5.3 Relative position error between tool and workpiece during machining with macro- and micro-manipulator (Setup C), computed from the measurements by the optical tracking system. The lower plot shows the Euclidean norm of the Cartesian error. Note that the time frame of the milling experiment showed in this plot is not the same as the one in Figure 5.2.

the rotation axes of joint two, three, and five of the robot, see Figure 2.9 in Chapter 2 for the robot configuration in the milling process, the robot exhibits higher compliance along the z-axis compared to the x- and y-axes. However, the micro-manipulator shows good tracking behavior, with only minor longitudinal vibrations of the micro-manipulator.

Error Analysis

The experimental results obtained with Setup C were evaluated using statistical methods. The amplitudes of the error characteristics as well as the corresponding frequency spectra were investigated. First, the performance of the micro-manipulator tracking is considered. Referring to Figure 5.2, it can be observed that the micro-manipulator exhibits tracking behavior within

the desired range along all three Cartesian axes. Second, the power spectral densities of the Cartesian position errors shown in Figure 5.3 were investigated. The frequency spectra are displayed in Figure 5.4. Considering that the mechanical design of the micro-manipulator comprises solid state flexure elements, the damping of the oscillatory modes of the system is challenging. As a result, the micro-manipulator is sensitive to process disturbances with frequencies near the eigenfrequencies along the respective actuation axis. The first natural eigenfrequency of the micro manipulator, along each actuation direction, falls in the range from 50–100 Hz, see Section 3.4 in Chapter 3. It should be noted in the plots of the power spectral density in Figure 5.4, though, that the effects of these on the resulting performance is reduced. This implies that, as desired, a damping of the resonances in the construction was obtained with the designed model-based state-feedback controller. The major resonances of the micro-manipulator were clearly damped by the control design, even though a peak at approximately 95 Hz is visible in the error spectrum. Hence, it can be concluded that the sensitivity of the micro-manipulator around the eigenfrequencies resulted in low-amplitude oscillations of the machining tool with an amplitude of 10–30 μm, as can be observed in the data plot in Figure 5.2. In turn, these tool oscillations result in surface undulations on the machined surfaces. The control performance is here limited by the mechanical design of the micro-manipulator. Additionally, a periodicity with a frequency of approximately 30 Hz can be observed in the error spectrum. As this frequency is below 50 Hz and appears in all three Cartesian directions, it can be traced back to excitation of an eigenmode of the robot, which can be verified by modal analysis of the same. The observed peaks in the frequency spectra were found within 10–30 Hz, which is the range of typical eigenfrequencies of industrial robots.

Workpiece Geometry Measurements

In order to validate and quantify the milling results, the machined surfaces of the workpieces obtained using Setups A–C were measured. The 2D contours of the machined toroids were analyzed using a Werth CMM of model Videocheck HA400, which provides measurements with an accuracy lower than 0.5 μm [Werth GmbH, 2013]. By performing raster scanning, point clouds of the inner and the outer circular contours were captured and subsequently compared to the desired reference circles. A least-squares matching, assuming additive Gaussian noise on the measurements, was used to locate the measured CMM data, acquired in a local coordinate system, and the reference circles relative to each other in order to enable comparison.

The results of the surface measurements are found in the plots in Figure 5.5 for Setups A–C, respectively. The plots display the computed difference of the CMM-measured outer circle from the nominal reference circle.

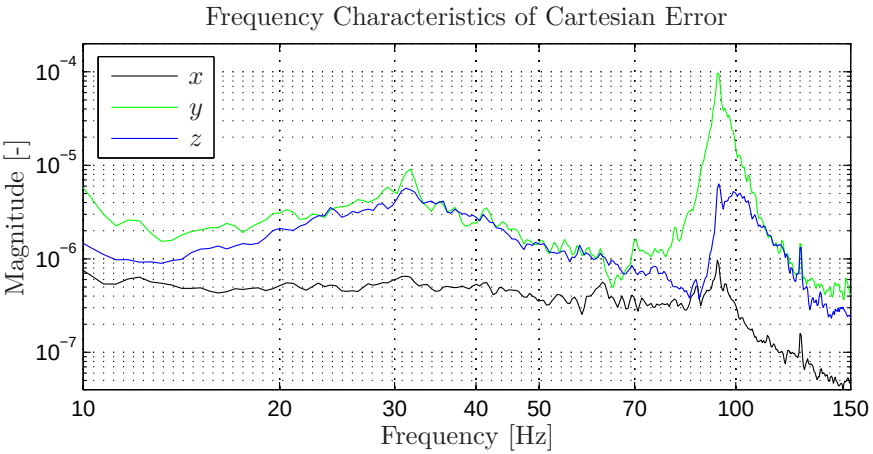

Figure 5.4 Power spectral densities of the Cartesian position errors—displayed in Figure 5.3—obtained during the machining of the toroid with Setup C.

The circumference measure starts at the location where the milling starts and traverses the toroid in the milling direction, see Figure 5.1. In addition, the measured inner and outer circular contours of the surfaces measured by the CMM are visualized and compared to the nominal circles in Figure 5.6. As a first observation, it can be noted in Figure 5.6 that the position deviations with Setup A are not constant during the machining of the circle. This is a result of the local stiffness properties of the robot combined with the changing direction of the process forces while traversing the toroid. Considering the performance increase using Setup C compared to the results from Setups A and B, see the two upper subplots in Figure 5.5, and the volumetric accuracy of 100 µm for the optical tracking system used, the machining accuracy of ±100 µm of the outer circle is a satisfying result. Further, when comparing the error of the toroid machined with Setup C to the corresponding results obtained using Setup A or Setup B, the significance of the different hardware and control components of the setups can be identified. A feedback loop, based on external sensor data from online position and orientation tracking of the workpiece, in the robot controller is sufficient for the compensation of static calibration errors and low-frequency errors, and thus reducing the error to below approximately ±200 µm (see Figure 5.5). A micro-manipulator, however, is essential for compensation of high-frequency process disturbances and reducing the error below an accuracy of ±100 µm.

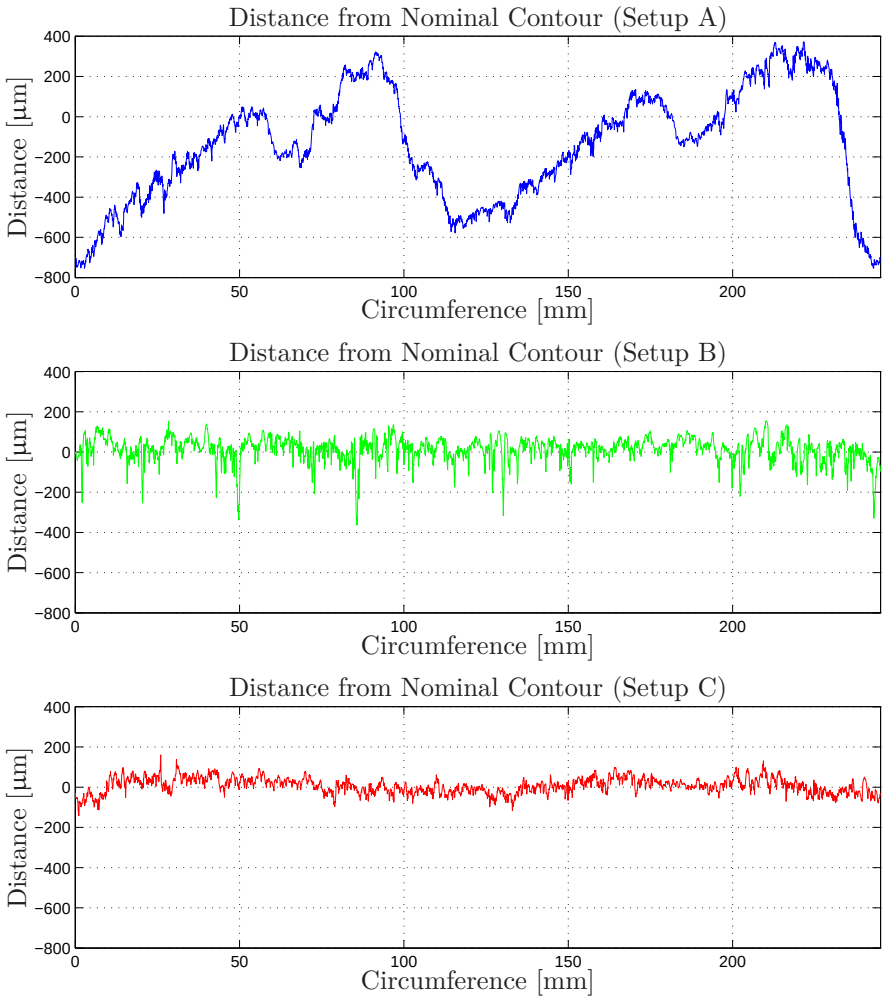

Figure 5.5 CMM measurements of the outer circular contour deviation, with the workpiece machined with Setups A–C. The circumference measure starts at the location where the milling starts, see Figure 5.1.

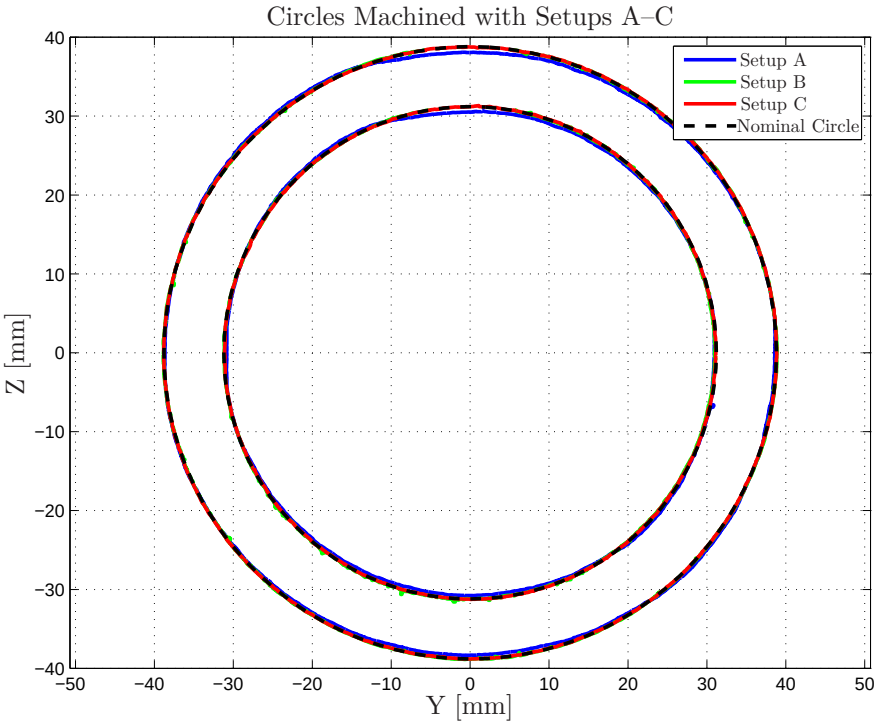

Figure 5.6 CMM measurements of the circular outer and inner contours, for the workpieces machined with the different Setups A–C.

Measurements of Surface Roughness

The CMM measurements of the circular contours on the machined workpiece can only evaluate the achieved performance along the y- and z-axes of the milling geometry. The performance along the x-direction, however, is also directly reflected on the machined surface. Therefore, the tactile measurement system, also employed in Section 4.3 in Chapter 4, of model M400/SD26 from Mahr [Mahr GmbH, 2011] was used in order the quantify the machining errors in the face direction of the tool. Figure 5.7 displays the surface roughness over a randomly chosen length of 25 mm for each workpiece. It is to be noted that the deviations on the surface were not as significant when compared to the circular contours visualized in Figure 5.6, which can be explained by the cutting-process dynamics in the machining. The major part of the material was removed in peripheral milling, and therefore radial excitement of the interaction between the tool and the workpiece was more prominent. This dynamic relation results in lower process forces, and hence reduced position deviations in the tool–axis direction. An interesting result is that the sur-

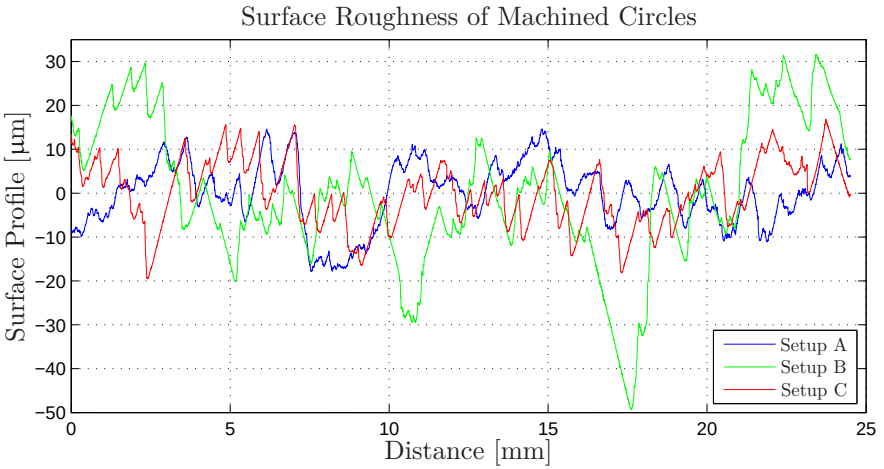

Figure 5.7 Surface-roughness measurements of the machined workpieces for the respective configuration, Setups A–C.

face obtained with Setup B is significantly worse than that of Setup A. It is plausible that this is related to the limited compensation bandwidth of the robot—*i.e.*, the high-frequency disturbances in position in the face direction of the tool are not possible to compensate for with this approach because of phase lag and, to some extent, communication delays. As a consequence, the resulting machined surface is worse than using a purely position-controlled robot without online compensation.

Summary of Experimental Results

In order to quantify and compare the performance of the setups considered, the mean absolute error (MAE) for the toroid machining was computed according to

$$\mathrm{MAE} = \sum_{k=1}^{N} \frac{||w_k^n - w_k^m||_1}{N}, \tag{5.1}$$

where k is the sample index, w_k^n is the nominal position, w_k^m is the measured position provided by the CMM, and N is the number of measurement points. The MAE provides an indication of the control-accuracy performance for the respective configuration. Further, the roundness of the outer circle was computed using the CMM measurements presented in Figures 5.5–5.6, according to the standard DIN ISO 1101 [ISO, 1985]. The surface roughness was quantified based on the data in Figure 5.7, using the standard deviation σ from the nominal surface. The resulting numbers obtained using the three Setups A–C are found in Table 5.1.

Table 5.1 Comparison of MAE and roundness of the machined toroids, and the standard deviations of the surface measurements. The numbers were computed from machining results obtained with the respective Setups A–C (numbers of MAE and roundness provided for the outer circle).

Setup	MAE (Circle) [μm]	Roundness (Circle) [μm]	σ (Surface) [μm]
A	255	1131	7.3
B	47	413	16.1
C	32	326	7.4

5.4 Discussion and Conclusions

As an alternative to joint-space position-controlled industrial robots for machining, workspace sensing combined with online position corrections with a macro/micro-actuation principle was considered in this thesis and evaluated experimentally in this chapter. The accuracy in terms of MAE was increased up to eight times, compared to the standard uncompensated case. High-accuracy sensing in workspace was realized using an optical tracking system. A prerequisite for this strategy to work well in practice is a high actuation bandwidth of the micro-manipulator and low latencies in the communication interfaces, in order to maintain stability during the machining process. Considering the open-loop bandwidth of the micro-manipulator, which is approximately 3–4 times the bandwidth of the closed-loop position control of the considered industrial robot at the end-effector, it is clear that the possibilities for compensation of high-frequency position errors during machining are significantly increased. This is not only the case for machining applications considered in this thesis, but also in other tasks that require contact between the robot and the tool or workpiece. In addition, as pointed out in [Sharon et al., 1993; Fasse and Hogan, 1995], workspace sensing as well as workspace actuation—*i.e.*, collocation of sensing and actuation—is beneficial in contact operations with mechanical manipulators in order to reach high-bandwidth control of the interaction between the tool and the workpiece. Another choice would be high-bandwidth inner-loop control with collocated sensing and actuation combined with a lower-bandwidth control with non-collocated sensing and actuation for the purpose of improved accuracy and closed-loop bandwidth, as proposed in [Vuong et al., 2009].

The approach to machining considered in Chapters 3–5 is hardware intensive, in particular compared to previous approaches to increasing the position-accuracy based on quasi-static and dynamic modeling of the robot stiffness. However, purely model-based approaches exhibits a dependency on

robot configuration, workpiece characteristics, and machining tool. Hence, the generality in application of the method considered in Chapters 3–5 should be considered as high, since no assumptions regarding these aspects have to be made. Instead, position deviations are measured online in the workspace during the machining process and are subsequently compensated for. In addition, the need for a calibration procedure for the force–deflection models, *i.e.*, stiffness models, is eliminated with the considered approach to machining.

Investigating the costs for the proposed robot cell for high-accuracy machining, it can be noted that the high-accuracy optical tracking system and the micro-manipulator are comparably expensive components. However, considering the rapid development and cost reduction for, in particular, optical tracking systems, the considered solution is still competitive as compared to the cost of a machine tool. The cost distribution for the micro-manipulator is between piezo-actuators (45%), manufacturing of the mechanical parts (35%), sensor equipment (15%), and other auxiliary equipment (5%).

Workspace control of industrial manipulators based on feedback from force data has previously been proposed for contact operations, see, *e.g.*, [Hogan, 1985; Vuong et al., 2009], in applications such as deburring [Jonsson et al., 2013] and drilling [Olsson et al., 2010]. Further, many robot manufacturers offer since some years such functionality for their industrial robots. The force-control performance is nevertheless limited by the mechanical bandwidth of the robot manipulator as well as the bandwidth of the internal joint-position controllers of the robot. Hence, the force control must be combined with appropriate mechanical actuation if high-frequency position deviations are to be compensated for. In the approach presented in this thesis, only position information was used for the workspace feedback. However, integration of a force sensor in the setup would be beneficial, in particular for monitoring the process forces during the machining process but also for incorporation in the feedback control architecture as a complement to the position information.

6

Robot Joint Modeling and Parameter Identification

6.1 Introduction

In this chapter, robot joint modeling with emphasis on machining scenarios is investigated. Further, a subsequent method for identification of joint backlash and stiffness is discussed. The identification method is based on rigidly clamping the robot end-effector to the environment and actuate the joint motors using a motion-control system. The approach for joint modeling is defined and a subsequent experimental evaluation is presented. This chapter of the thesis extends the preliminary method and results presented in the publication [Lehmann et al., 2013].

This chapter is outlined as follows: The problem formulation and previous research on robot joint modeling are discussed in Section 6.2. In Section 6.3, the method for robot joint parameter identification is outlined. A subsequent experimental verification providing results obtained with the method is presented in Section 6.4. Section 6.5 discusses the method and the obtained experimental results. The chapter is finally summarized in Section 6.6, where conclusions also are drawn.

6.2 Problem Formulation and Previous Research

The interest for robot joint modeling in the context of machining processes, is based on the possibility to predict or online estimate position deviations of the robot end-effector from the specified machining path because of process forces, see, *e.g.*, [Reinl et al., 2011; Abele et al., 2011]. For accurate prediction of the deviations, the robot has to be adequately modeled. Considering machining processes, the robot can be considered to be in a quasi-static state, and thus dynamic parameters such as link inertias and viscous friction in the joints are often of less interest. Rather, the robot joint dynamics including

static friction, backlash, and stiffness need to be explicitly considered since these account for a major part of the position errors in such applications. As a consequence, methods for determining the relevant model parameters on an arbitrary industrial manipulator are required. Achieving high position accuracy with robots manipulators in the presence of process forces is still an open research problem. This chapter considers a method for identification of relevant robot joint parameters for machining scenarios, which could be used as a basis for future position compensation in machining.

A previous method for kinematic calibration of robot manipulators that is based on clamping the robot end-effector to a stiff base and actuating the joints has been proposed in [Bennett et al., 1992]. In this chapter, that principle is utilized in order to measure the joint backlash and stiffness. It is also plausible that the extension of the previous method discussed in this chapter, will facilitate the practical use of the kinematic calibration procedure proposed in [Bennett et al., 1992].

Methods for measuring the static and dynamic joint friction in industrial manipulators have been investigated in the literature, see, *e.g.*, [Bittencourt et al., 2010]. Friction modeling in general is discussed extensively in [Olsson, 1996]. Further, methods and strategies for modeling and quantifying the backlash in robot joints have been discussed in [Hovland et al., 2002; Ruderman et al., 2009]. Stiffness modeling for robots is discussed in [Abele et al., 2007; Abele et al., 2008; Ruderman et al., 2009; Dumas et al., 2011]. One of the methods in [Abele et al., 2007] is using clamping of the joints preceding the joint to be measured in the kinematic chain to the purpose of decoupling when applying external load on the current joint. In the method in this chapter, however, the robot end-effector is clamped thus locking all degrees-of-freedom (DoF), including the joint to be measured, simultaneously and the joint motor itself is instead actuated using a control system. An extension of the methods in [Abele et al., 2007] considered in [Abele et al., 2008] concerns compliance orthogonal to the actuation direction of the joint. Position-error compensation for robots subject to external process forces is discussed in [Wang et al., 2009; Pan and Zhang, 2009]. A stiffness-modeling approach combining external load on the end-effector with local measurements on the links with an optical measurement system was proposed in [Tyapin et al., 2014].

Many of the methods for determining the joint stiffness in robots suggested in the references above, rely on applying a load on the robot TCP or load at certain links, while measuring the corresponding applied force using a force/torque sensor and the corresponding deflection of the robot using high-resolution measurement equipment, such as optical measurement systems. For measurement of the joint backlash, high-resolution encoders on the arm-side of the joints have been proposed. On the contrary to these methods, a method where the end-effector of the robot manipulator is rigidly clamped

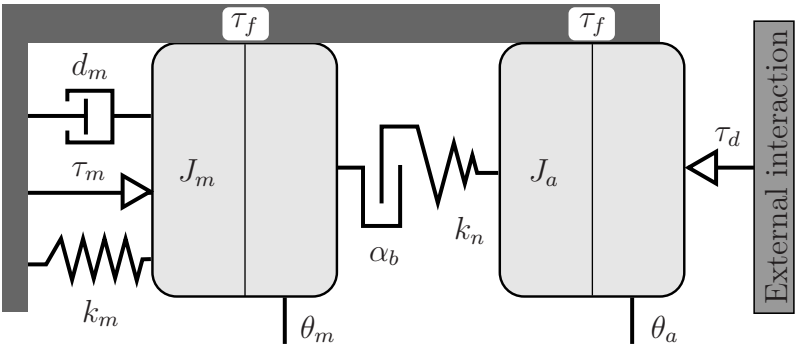

Figure 6.1 Robot joint model with nonlinear dynamics, depicted as a prismatic joint for simplicity. Also, the gear ratio is not visualized in the schematic figure.

to a stiff base, and each robot joint motor subsequently is actuated using the motion-control system, is considered in this thesis for identification of joint backlash and joint stiffness. With this method, only the motor position and the motor currents are required as measurements. Thus, this decreases the required costs for the experimental setup and the potential error sources can be reduced.

6.3 Method

The method to parameter identification for robot joints is based on simultaneously locking all DoF of the manipulator. In the presentation in this chapter, it is implicitly assumed that the manipulator has six DoF. The method is, however, not limited to this case, see Section 6.5. It is further assumed that the major compliances of the robot manipulator are the results from dynamics in the joints—*i.e.*, compliance orthogonal to the ordinary rotational direction of the joints [Abele et al., 2008], caused, *e.g.*, by the bearings and the links, is neglected in the adopted model here. Nevertheless, the effects of such dynamics on the measurement method are evaluated experimentally in Section 6.4, see also the following discussion in Section 6.5. The docking is achieved by attaching the robot end-effector to a base *via* a stiff connection. The considered joint dynamics is depicted graphically in Figure 6.1 and the notation employed in the figure and subsequently in this chapter is defined in Table 6.1. The joint parameters to be identified by the method in this chapter are the stiffness characteristics and the possible backlash angle. Joint friction is not treated in this thesis; for more details on friction modeling and identification for robots, see, *e.g.*, [Bittencourt et al., 2010].

Table 6.1 Variables and parameters in the nonlinear robot joint model visualized in Figure 6.1.

τ_m	—	Torque from controller to motor
τ_d	—	External disturbance torque
θ_m	—	Joint angle, motor side
θ_a	—	Joint angle, arm side
α_b	—	Backlash angle
k_n	—	Nonlinear spring function
τ_f	—	Friction torque
J_m	—	Actuator/motor inertia
J_a	—	Arm-side inertia
k_m	—	Stiffness accomplished by the controller
d_m	—	Damping accomplished by the controller

Clamping Procedure

The identification procedure is initiated by rigidly docking the robot end-effector. This can be performed using various strategies. In particular, contact-force feedback control is one approach. However, such an approach requires force sensing and is thus less attractive from an application point-of-view. Instead, partial detuning of the proportional gain and turning off the integral part in the joint controllers are employed as a method to make the robot slightly compliant in the critical final part of the docking phase. It is to be noted that the contact between the robot end-effector and the environment is stiff (since both the end-effector and the clamping device typically are made of metal), and thus pure position control of the robot with high-gain controllers is likely to result in high force transients. The partial detuning of the controllers is a commonly available feature of robot-control systems. An initial position of the robot is preprogrammed, where the approach phase to the target docking point can be started. Once the robot is docked, the three translational and the three rotational DoF are locked for motion.

Considering that the robot links are subject to gravitational forces during the experiment, a subset of the joint gear-boxes is loaded also when no external process forces affect the robot. Hence, a procedure is needed for achieving the state where each joint motor exerts zero torque on the joint. This state is reached by employing an iterative scheme, where each motor position is altered incrementally until zero torque is measured. Since the joint controllers have been detuned, such a strategy is possible to perform also with the robot end-effector clamped.

Measurement Sequence

With the robot end-effector rigidly clamped and thereby constrained, each joint motor is actuated separately in a sequence using the motion-control system of the robot. The actuation is performed such that the position reference sent to the joint controller is slowly varying around the zero-torque position of the actuated joint. Because of the fact that the joint controllers have been detuned, the effects of the quantization of the position steps in the joint controller are reduced. Assuming that the contact between the robot end-effector and the clamping device does not exhibit any compliance, the links of the robot are fixed during the measurement sequence. Thus, if the joint-motor positions and the corresponding joint motor currents (that are assumed to be proportional to the motor torque) are measured when actuating each joint, the torque/position characteristics can be determined. As a consequence, the identification method requires the joint-motor positions and currents to be available as logging signals from the robot controller. Note, however, that high-frequency access to these signals is not required considering the low-speed motions performed when moving the joint motors incrementally.

The measured joint deflection corresponds to the difference $\Delta\theta$ between the motor position θ_m and the corresponding arm-side position θ_a, since the latter is assumed to be fixed during the actuation. Thus, there is no need for external sensor equipment measuring the deflection of the links or robot end-effector as the internal robot sensors provide the necessary information for characterizing the joint properties. To summarize the method for determining the joint dynamics, the following steps define the procedure:

1. Move the robot close to the target docking position;

2. Activate detuning of the robot joint controllers and start the approach to the target position;

3. When the docking position has been reached, lock the connection between the robot and the environment;

4. Perform a search procedure such that all joint motors have zero torque applied;

5. Actuate each joint separately in a sequence by incrementally altering the corresponding position references to the motion controller, such that a hysteresis curve is traversed multiple times;

6. Record the joint motor positions and currents, corresponding to torque references, during the measurement sequence;

7. When the measurements are finished, the mechanical connection is opened and the robot is unclamped and can be moved away from the docking position.

Parameter Extraction

With the experimental data acquired during the measurement sequence defined in the previous subsection, a torque–position map can be constructed, from which the desired joint parameters can be extracted. Consider the schematic curve depicted in Figure 6.2. The main characteristics visible in the relationship between the position and the torque are the backlash when the gearbox torque changes sign and the compliance properties of the joint. In the figure, the stiffness is assumed linear (although possibly different when applying positive and negative torques, respectively). It is clear, though, that any nonlinear stiffness relationships can be captured with the outlined method to joint identification. A subsequent regression of the position and torque data can be used to determine a parametric relationship for the stiffness properties of the considered joint. Also, it is to be noted that the friction in the joint motors results in that the upper and lower part of the hysteresis curve are not coinciding.

Clamping to Non-Stiff Environments

In the ideal case, the robot end-effector is fixed during the complete measurement procedure. However, in practice, the environment might exhibit compliance and the result of such compliance is motion of the robot links during the measurement sequence. Since the method for parameter identification relies on fixed link positions during the measurements, this will result in an underestimate of the stiffness of the joint transmission. If the link motion can be measured using external equipment, this effect can obviously be compensated for. However, considering that a strategy not relying on external equipment is desirable, an approach based on an estimation of the stiffness of the environment—*i.e.*, the effective compliance for the mechanical fixture—is outlined here. Assuming the compliance matrix of the environment to be $K_e \in \mathbb{R}^{6\times 6}$, the deflection $\Delta x_e \in \mathbb{R}^6$ of the robot end-effector is

$$\Delta x_e = K_e f_e, \tag{6.1}$$

where $f_e \in \mathbb{R}^6$ is the force and torque applied on the end-effector. The end-effector force can be estimated from the joint torques τ_a (obtained from the motor-torque signals using the nominal gear ratios and with the contributions from friction subtracted) using the fundamental relation

$$f_e = J(\theta_a)^{-\mathrm{T}} \tau_a, \tag{6.2}$$

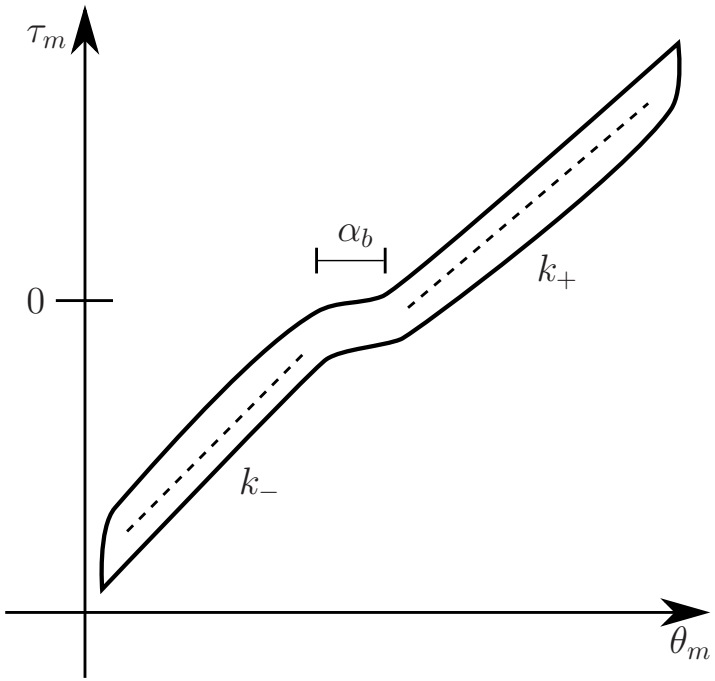

Figure 6.2 Schematic torque–position map for a robot joint with backlash and elasticity. The stiffness characteristics are visualized as linear springs with parameters k_+ and k_-, even though the method as such is not limited to linear compliance relationships.

where $J(\theta_a) \in \mathbb{R}^{6\times 6}$ is the Jacobian of the robot manipulator considered [Spong et al., 2006]. Note that this relation can be justified to be approximately true in the measurements, since quasi-static conditions hold during the experiments. Approximate values of the arm-side angles θ_a are obtained from the motor-side angles θ_m by using the nominal gear ratios. Of course, this only results in an approximation of the actual arm-side angle. In order to map the measurements of the deflections in Cartesian space to joint space, the relation

$$\Delta\theta_a = J(\theta_a)^{-1}\Delta x_e, \tag{6.3}$$

where $\Delta\theta_a \in \mathbb{R}^6$ is the deflection in joint space, is used. Hence, assuming that the orthogonal compliance of the robot joints is negligible and that the links are rigid, the link motion can approximately be quantified using this strategy during the measurements. Accordingly, the difference between motor angle and arm-side angle can be computed and used for characterization of the joint stiffness.

6.4 Experimental Results

The method for robot-joint identification in Section 6.3 was evaluated on an industrial manipulator from ABB of model IRB140 [ABB Robotics, 2014]. The experiments performed aimed at characterizing the backlash and stiffness properties of the first three main joints as well as the wrist joints of the robot manipulator. The two settings of the experimental setup are shown in Figures 6.3–6.4. The difference between the two were the surfaces used for the clamping in the experiments; the first comprises a table made of an aluminium frame containing a high density fiberboard (HDF) plate, see Figure 6.3, and the second a plate of solid steel where also the robot base was attached, see Figure 6.4. On the table, the point where the end-effector of the robot was attached to the environment is possible to move, because of the use of tool changers that are locked using compressed air. This enables verification of the consistency of the method by acquiring measurements in several configurations of the robot manipulator. In the case of using the steel plate, the robot end-effector plate is screwed directly on the same, thus eliminating the effects of possible compliance and backlash in the tool changer.

Experimental Protocol

The experiments were performed according to the procedure defined in Section 6.3, by actuating each joint separately in a sequence. The relevant data signals from the robot were recorded using the extended research interface ExtCtrl, with which the robot control system was equipped (see Section 2.1 in Chapter 2 for further details on this interface). The signals to be logged were the motor positions and the motor torques (obtained from the measured motor currents) of all six joints. The required robot motion and actuation were realized by a robot program implemented in the ABB RAPID robot-programming language. Note that logging of the data signals also can be performed on this interface level. Thus, the identification method does not rely on extended interfaces to the robot controller. Further, as previously indicated, the required sample rate when collecting the data is comparably low, since only low-speed motor motions are performed when in the clamped state.

For verification of the measurements, an optical tracking system from Nikon Metrology of model K600 [Nikon Metrology, 2010] for 6D measurements—*i.e.*, measurements of position and orientation of a rigid body—was used. Three LEDs were attached to the robot end-effector plate (defined to be the tool center point (TCP)) for measurement of the position and the orientation. In addition, LEDs were attached to the respective link connected to joints 1–6. This enables both verification of the method and evaluation of the compensation strategy accounting for clamping to compliant environments as discussed in Section 6.3. It is expected that the

Figure 6.3 Experimental setup for joint identification, comprising an ABB IRB140 6-DoF robot manipulator. The end-effector is rigidly attached to the table during the measurement procedure, thus locking all the DoF of the robot.

table used for attaching the clamping device and the robot-base attachment exhibit a certain degree of compliance in the first setup, see Figure 6.3. The steel plate, though, offers a significantly more stiff environment for clamping (see Figure 6.4).

Results

The experimental data collected from a set of experiments on the steel plate are visualized in Figure 6.5, where the motor torques are plotted as function of the motor positions. The motor positions have been detrended, such that they are centered around zero in order to enable comparison of different measurement configurations—*i.e.*, different positions of the clamping point in the robot workspace. In Figure 6.5, two curves are shown for each joint. The

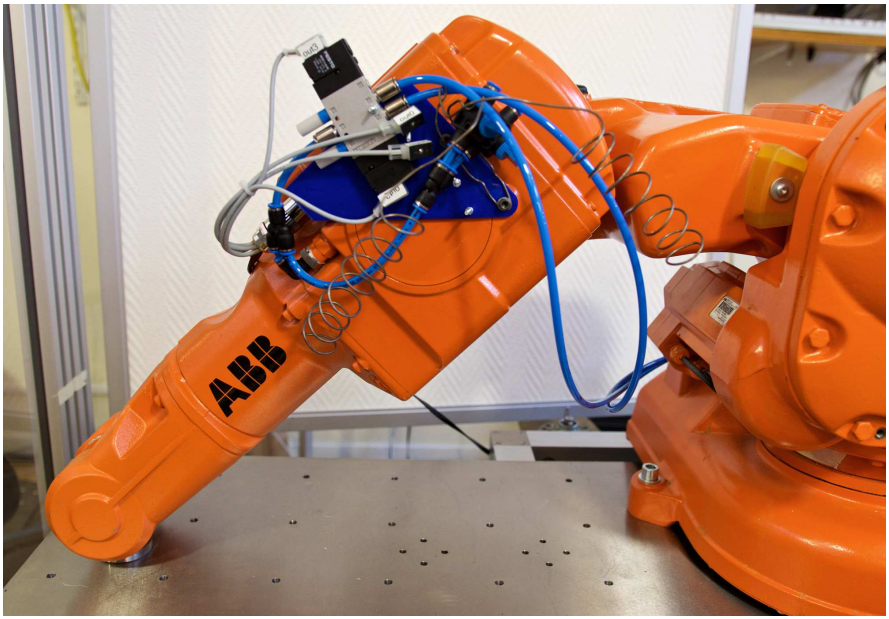

Figure 6.4 Modified version of the experimental setup in Figure 6.3, where the robot end-effector plate as well as the robot base were screwed directly on a plate made of solid steel. Compared to the first setup with the table in Figure 6.3, the steel plate exhibits significantly lower deflections when loaded during the experiments.

black curves correspond to the low-pass filtered measurement data, whereas the red curves exhibit the actual transmission characteristics. The latter were computed from the data acquired by the optical tracking system.

The 3D point measurements on the respective robot link from the optical tracking system were projected on the plane in which the joint actuates, and the actual link angle with respect to the preceding link was subsequently computed. The actuation plane in the coordinate system of the tracking system was computed *a priori* in a calibration step for each joint. In the sensor calibration, each joint was moved separately and the measurements from the tracking system were subsequently used in a singular-value decomposition. Using the eigenvectors corresponding to the two most significant singular values, the plane in which the joint actuates was determined. In addition, the radius from the measurement point to the rotation axis of the joint in the actuation plane was computed. This was needed for certain joints in order to transform Cartesian measurements from the tracking system to corresponding joint angles. For transforming measured angles on the arm-side to the motor-side of the joint, the nominal gear ratios were employed.

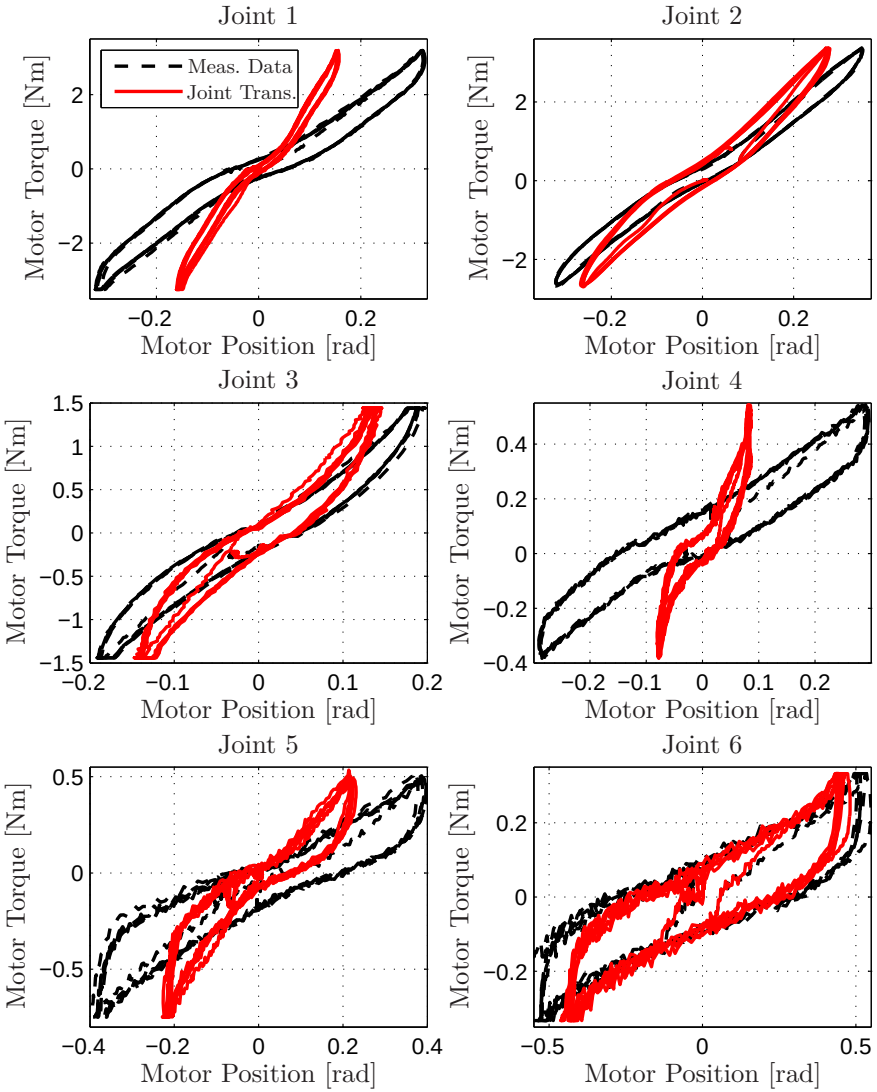

Figure 6.5 Measurement results for joint 1–6 of the ABB IRB140 robot when rigidly attached with the end-effector plate directly on a steel plate. The black curves are the measured data, whereas the red curves are the dynamics for the joint transmission obtained by compensating for possible link motion.

In the plots in Figure 6.5 visualizing the experimental data acquired, several observations can be made. Initially, it can be noted that the measurements are highly repetitive in the multiple cycles performed for each joint. This indicates that the method considered is consistent with respect to repeatability. Moreover, it can be seen that there are small discrepancies between the measured joint data and the resulting curves obtained using compensation for the link motion for certain of the joints. The main reasons were identified as a minor compliance of the environment, cross coupling between the wrist joints, and orthogonal joint compliances. The latter effect appears since a subset of the joint bearings were not stiff, and thus motion of the links occurred when certain joints were actuated. This was verified using the optical tracking system to be significant for the bearings of joint 3. That compliance manifests itself as a motion of the entire robot structure when actuating joint 1 when rigidly attached with the end-effector. The resulting effect is that the stiffness of the joint dynamics will be underestimated using only the measurements from the motor-side of the joint. However, this discrepancy could be resolved using the link measurements. Considering the first two wrist joints, *i.e.*, joints 4–5, slight discrepancies between the measured stiffness and actual joint dynamics are observed. These effects are explained by a combination of a mechanical cross coupling between the wrist joints and orthogonal joint compliance. It is clear that the measurements for joint 6 are close to the actual joint dynamics, which is expected since the robot end-effector is screwed directly on the steel plate.

In order to use the measurement data acquired for improving the accuracy of a machining task, the joint dynamics visualized in Figure 6.5 can be parametrized using three different segments; a linear approximation for the part with negative torque, one for the approximately horizontal backlash part, and a linear approximation for the part with positive torque (see Figure 6.2). It is to be noted that the backlash part is more or less pronounced for the different joints on the robot, resulting from the differences in the mechanical design of the same.

Non-Stiff Environments and Orthogonal Joint Compliances To further evaluate the influence of the stiffness of the environment and orthogonal joint compliance, extended measurements on the robot with the optical tracking system were performed. Clamping experiments on the table with the HDF table was thereby executed, thus resulting in more significant link deflections during the measurement sequence. In the presentation in this section, the focus is on joints 1–2, but the analysis for the remaining joints is analogous. Based on the measured data from the robot joint during the experiment, the 6D TCP deflection measurements were used for computing the corresponding link deflections according to the procedure outlined in Section 6.3 (*cf.* the relation (6.3)). Thus, the difference in angle between the

motor-side and the arm-side of the joint could be computed; the resulting stiffness relations are displayed in Figures 6.6–6.7 together with the robot measurement data and the actual joint dynamics computed from the link measurements. It is clear that compensation for the motion of the TCP only accounts for a part of the link motion. For joint 1, the compliance of the bearings of joint 3 had a significant influence. This was confirmed by explicit measurements of the relative motion between link 2 and 3 when exciting joint 1. Considering joint 2, the major link motion results from compliance in the attachment of the robot to the aluminium base frame. The conclusion to be drawn is that TCP measurements alone are not sufficient for quantifying the complete link motion in the case of both non-rigid environments and orthogonal joint compliance, even though a significant improvement can be achieved.

Consistency of Measurements In order to verify the invariance of the joint stiffness with respect to the configuration of the robot when performing the excitation, experiments with different clamping positions on the steel plate as well as on the table made of HDF were executed. This is of importance, since the joint-transmission dynamics is assumed to be an inherent property of the robot and the measurements should thus be independent of the configuration. Measurements were acquired at several different clamping configurations for all joints. A comparison of the measurements on joint 1 from four configurations is visualized in Figure 6.8. The observations that can be made are that the joint measurement curves (black and blue lines) exhibit a difference in properties. This is expected, since the influence of the non-stiff environment and the orthogonal joint compliance is dependent on the clamping configuration. However, more importantly is that the curves computed by compensating for the link motion virtually coincide, thus indicating the consistency of the measurements from the different robot configurations.

6.5 Discussion

The problem of identifying joint dynamics, in particular joint stiffness and backlash, for robot manipulators was considered in this chapter. In contrast to many previous methods for stiffness identification of robot manipulators, where external load are applied to the robot end-effector or links, the considered method relies on rigidly locking all DoF of the robot simultaneously by clamping the end-effector to a rigid fixture and actuate the joint motors by incrementally altering the position references to the motion controller. The advantage of such an approach is that the internal sensor data of the robot (motor position and current) can be used for characterizing the joint dynamics, thus eliminating the need for external high-accuracy measurement equipment. The method does not only provide information about the linear

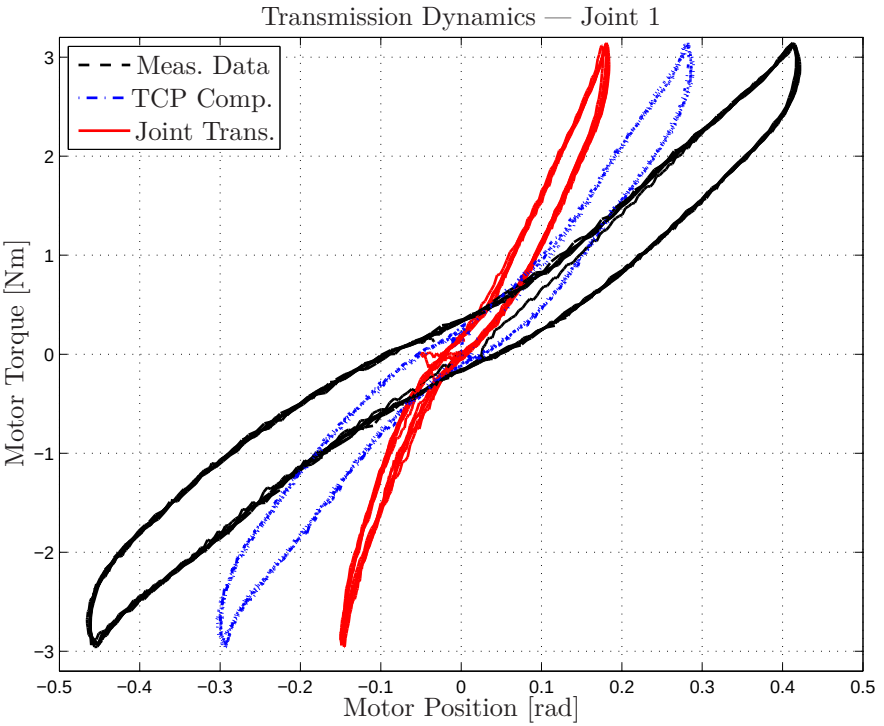

Figure 6.6 Measurement results for joint 1 of the ABB IRB140 robot with clamping on the HDF table, illustrating the effects of non-stiff environments and orthogonal joint compliance. The black curve is the measurement data, the blue curve is the corresponding data with compensation for motion of the TCP, whereas the red curve is the relationship for the actual joint-transmission dynamics.

joint stiffness, but also about possible backlash in the gearbox as well as nonlinear stiffness.

In the experimental verification, two different aspects potentially limiting the accuracy of the method were particularly investigated. First, the effects of non-stiff environments were experimentally evaluated using two different clamping setups and a strategy was outlined for compensating for deflections of the docking point. Clamping on the steel plate provided measurements close to the actual joint dynamics for several of the robot joints. Second, the effects of orthogonal joint compliance were identified. It was noticed that this has a significant effect when actuating certain joints in some robot configurations. With explicit link measurements these effects could be compensated for, but without such measurements the joint stiffness is underestimated.

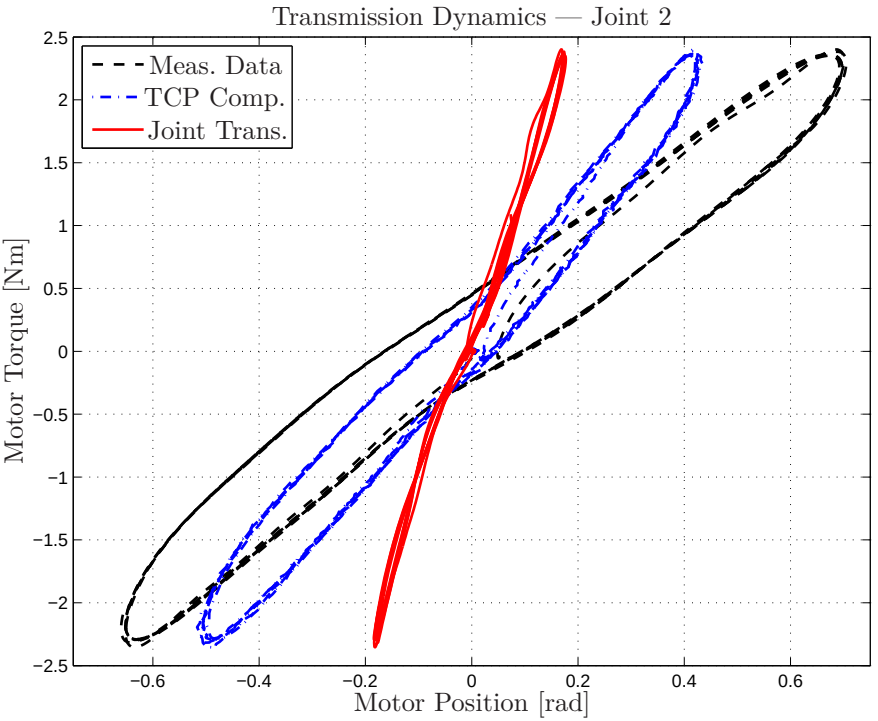

Figure 6.7 Measurement results for joint 2 of the ABB IRB140 robot with clamping on the HDF table, illustrating the effects of non-stiff environments. The black curve is the measurement data, the blue curve is the corresponding curve with compensation for motion of the TCP, whereas the red curve is the relationship for the actual joint-transmission dynamics.

Both during the clamping procedure itself and during the measurement sequence, the robot-joint position controllers were detuned. The reasons were twofold: First, it makes the robot slightly compliant in the initial docking phase and consequently avoids triggering of the supervision system for joint errors in the robot controller, thus enabling position control for the docking procedure. Second, in the measurement phase, the detuning decreases the effects of the quantization of the increments in the position references in the robot controller. Also, since the primary interest in this chapter is the inherent joint characteristics, the detuning decreases the influence of the joint-position controller gain on the experimental measurements of the torque–position relationship.

The method considered in this chapter relies on actuating each joint separately in a sequence using the same clamping position for the end-effector. An

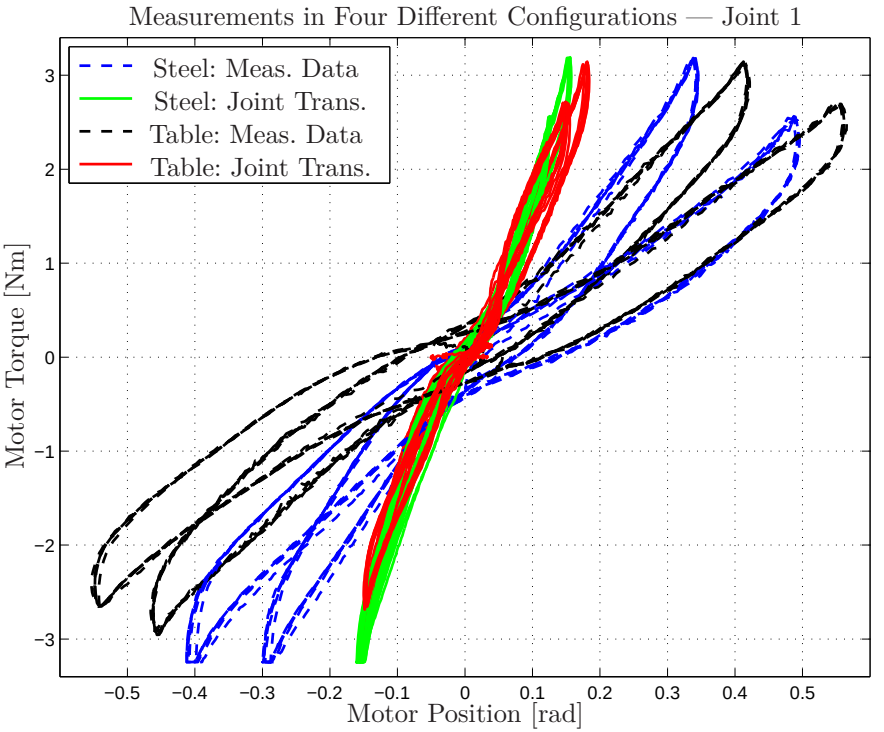

Figure 6.8 Measurement results for joint 1 on the ABB IRB140 robot, illustrating the invariance of the joint stiffness with respect to different measurement configurations. The black and blue curves are the measurement data from the joints, whereas the red and green curves are the relationships corresponding to the actual joint-transmission dynamics.

extension of the method would be to perform experiments in multiple configurations and thus acquire data in several clamping positions. This would enable the identification of the complete stiffness matrix for the robot manipulator (including the non-actuated DoF resulting from the compliance of the bearings and the links). Computing a least-squares solution using nonlinear optimization, where the stiffness parameters and the deflections of both actuated and non-actuated DoF are considered as optimization variables and the measured joint positions and torques are considered as inputs, enables identification of a linear approximation of the stiffness of the manipulator. This will then decrease the need for external measurement equipment for measuring the deflections resulting from orthogonal joint compliance.

6.6 Conclusions

This chapter considered robot-joint modeling, with focus on machining scenarios, and an approach to parameter identification was discussed and experimentally evaluated. The method was based on clamping the robot end-effector to a rigid base (and thus locking all DoF of the manipulator) and subsequently actuate each joint motor separately using the motion controller. The data acquired during the experiments were motor position and corresponding motor current. Experimental results were presented for a setup with an ABB IRB140 robot and different clamping environments. The results obtained make it plausible that the method is able to determine relevant joint data without non-standard and external measurement equipment.

7

Gray-Box Identification of Flexible Systems

7.1 Introduction

In this chapter, the problem of gray-box identification of dynamic models for flexible mechanical systems is considered. More specifically, the problem is approached by means of continuous-time system identification using subspace-based methods based on discrete-time input–output data. A method is developed, with the property that the structure of the model resulting from fundamental physical first principles is obtained and the parameter matrices have a clear physical interpretation. The method is used for investigating the cross coupling between the axes of the micro-manipulator discussed in Section 3.4 in Chapter 3, and to identify a multi-input multi-output model of the setup. This chapter of the thesis is derived, in parts, from the publication [Olofsson et al., 2014].

This chapter is outlined as follows: A background to the subject as well as an account of previous research are provided in Section 7.2, where also the problem formulation is defined. The considered method is discussed in Section 7.3. In Section 7.4, simulation results obtained with the method are presented. This is followed by experimental results in Section 7.5. A discussion of the obtained results and the method itself is provided in Section 7.6. Finally, conclusions are drawn in Section 7.7.

7.2 Background

System identification is a fundamental part of model-based control design as well as simulation and prediction of dynamic systems. The system identification problem is to define a model structure and subsequently determine the model from input–output data acquired from experiments on the process to be investigated [Johansson, 1993]. Typical factors to be considered

concern sufficiently exciting input signals (for a measure of this property, see the notion of persistency of excitation [Åström and Bohlin, 1966]), type of model, and model order. Traditionally, three different model structures have been considered for identification; time-series models [Madsen, 2008], transfer-function models, and state-space models. Methods for identification of transfer function and time-series models include, among others, the least-squares method and the maximum-likelihood method [Åström and Bohlin, 1966]. State-space models can be identified based on realization-based methods [Ho and Kalman, 1966; Juang and Pappa, 1985] and extensions of these called subspace-based identification methods [Verhaegen and Dewilde, 1992; Van Overschee and De Moor, 1994]. Also, Bayesian Monte Carlo methods have found applications in this area recently, see [Schön et al., 2011] for algorithms for identification of nonlinear state-space models. Given the discrete-time nature of input–output data acquisition, system identification is often approached by means of discrete-time methods. However, in certain situations, in particular when considering model structures resulting from physical first principles, the dynamic relations are more natural to describe in continuous time. Considering that the transformation between a discrete-time model and a continuous-time model is not a bijective mapping, identification of a discrete-time model and subsequent transformation is not straightforward. Hence, algorithms for identification of continuous-time state-space models based on discrete-time experimental input–output data have been proposed, see, *e.g.*, [Johansson et al., 1999; Haverkamp et al., 1997] for time-domain methods and [McKelvey et al., 1996] for frequency-domain methods.

In the modeling procedure, the models to be determined traditionally range from black-box models—*i.e.*, system models without predefined internal structure, where the input-output relation is the important property—to white-box models where the structure and parameters are completely determined based on first principles. Many of the proposed system identification algorithms consider estimation of black-box models from experimental data. An intermediate model category consists of gray-box models, see, *e.g.*, the case study in [Bohlin, 1994]. In such models, the structure is partially predefined, often as a result of the physical nature of the system to be modeled, but the parameters are unknown. In this chapter, identification of gray-box models for the compliance dynamics of mechanical systems—*i.e.*, the relation between the applied force and the corresponding deflection—is considered. The motivation for the interest in these kinds of models is mainly model-based control design—such as in impedance control in contact operations [Hogan, 1985; Olsson et al., 2004] and LQ/LQG optimal state-feedback control where a physical interpretation of the states in the model is essential—but also when investigating the structural properties of mechanical systems. More specifically, regarding methodology, a subspace-based identification algorithm for determining a continuous-time model from experimental data is developed

in this chapter. Previous research in this area includes [Cescon et al., 2009], where a black-box model approach in discrete time to the same problem was proposed. Gray-box identification for rotating systems and physical parameter estimation using optimization were investigated using frequency-based methods in [Tanaka et al., 2011; Tan et al., 2012], and identification of state-space models for compliance dynamics was treated in [Cavallo et al., 2007] using subspace methods in the frequency domain. Methods for estimation of gray-box models for industrial robots have been proposed in [Wernholt and Moberg, 2011; Moberg et al., 2014], and in [Gautier and Khalil, 1992; Gautier, 1997] with methods based on a least-squares approach. Identification of robot dynamics was also investigated in [Berglund and Hovland, 2000; Hovland et al., 2001] by application of the theory for inverse eigenvalues. Moreover, structural reformulations in subspace identification to the purpose of gray-box modeling, similar to the one used in this thesis, for general dynamic systems have been investigated in [Lyzell et al., 2009; Lyzell, 2012]. The main contribution of this chapter is the development and application of a time-domain subspace-based gray-box identification method for continuous-time models of flexible mechanical systems.

Problem Formulation

Introduce the notation $M \in \mathbb{R}^{n\times n}$, $D \in \mathbb{R}^{n\times n}$, and $K \in \mathbb{R}^{n\times n}$ for the mass, damping, and stiffness matrices, respectively, of the system to be modeled. From physical considerations, it is required that $M \succ 0$, $D \succeq 0$, and $K \succeq 0$, *i.e.*, the matrices are positive definite or semidefinite. The differential equations for the compliance dynamics, resulting from fundamental physical relationships, see [Egeland and Gravdahl, 2002], can be written as

$$M\ddot{y}(t) + D\dot{y}(t) + Ky(t) = u(t), \tag{7.1}$$

where $u(t) \in \mathbb{R}^n$ is the input and $y(t) \in \mathbb{R}^n$ is the output of the system, see Figure 7.1. Introducing the states

$$z(t) = \begin{pmatrix} y(t) \\ \dot{y}(t) \end{pmatrix} \in \mathbb{R}^{2n}, \tag{7.2}$$

the following state-space system can be established

$$\begin{aligned} \dot{z}(t) &= \begin{pmatrix} 0 & I \\ -M^{-1}K & -M^{-1}D \end{pmatrix} z(t) + \begin{pmatrix} 0 \\ M^{-1} \end{pmatrix} u(t), \\ y(t) &= \begin{pmatrix} I & 0 \end{pmatrix} z(t). \end{aligned} \tag{7.3}$$

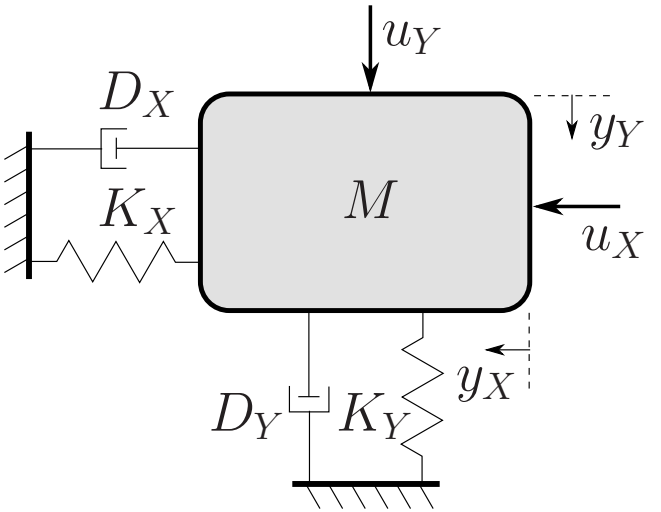

Figure 7.1 Schematic depiction of a spring-mass-damper system in two dimensions, with the applied forces u_X, u_Y as inputs and the corresponding deflections y_X, y_Y as outputs.

Consequently, the matrices in the state-space model should have the structure indicated in (7.3)—*i.e.*, the following format:

$$\begin{aligned} \dot{z}(t) &= \underbrace{\begin{pmatrix} 0 & I \\ A_{21} & A_{22} \end{pmatrix}}_{A} z(t) + \underbrace{\begin{pmatrix} 0 \\ B_2 \end{pmatrix}}_{B} u(t), \\ y(t) &= \underbrace{\begin{pmatrix} I & 0 \end{pmatrix}}_{C} z(t). \end{aligned} \tag{7.4}$$

The problem can then be stated as that of computing the matrices

$$\mathcal{S} : \{A \in \mathbb{R}^{2n \times 2n}, B \in \mathbb{R}^{2n \times n}, C \in \mathbb{R}^{n \times 2n}\} \tag{7.5}$$

in the model based on experimentally collected sampled input–output data $\{u_k\}_{k=1}^N$ and $\{y_k\}_{k=1}^N$, where k denotes the sampling instance and uniform sampling with period h is assumed. In addition, the physical parameter matrices M, D, and K should be estimated from the identification data.

7.3 Method

In this section, the method for identification of a gray-box compliance model is described and theoretically justified.

Model Transformation

The system (7.4) is rewritten[1] using a complex variable transformation *ad modum* [Johansson et al., 1999]. The motivation for the transformation is to avoid formulating the identification algorithm with the differential operator, which is known to be numerically challenging in the presence of noise. Applying the Laplace transform on (7.4) and assuming that the initial conditions are such that $z(0) = 0$, the transformed system can be written as

$$\begin{aligned} sZ(s) &= AZ(s) + BU(s), \\ Y(s) &= CZ(s). \end{aligned} \tag{7.6}$$

Introducing a variable transformation according to [Johansson et al., 1999] with the stable and causal relation

$$\lambda(s) = \frac{1}{s\tau + 1}, \quad \tau > 0, \tag{7.7}$$

where τ is the time constant, enables reformulation of the model to the following format

$$\begin{aligned} Z(s) &= (I + \tau A)\lambda(s)Z(s) + \tau B\lambda(s)U(s), \\ Y(s) &= CZ(s). \end{aligned} \tag{7.8}$$

Reformulation as an equation system in the time domain gives

$$\begin{aligned} z(t) &= A_\lambda[\lambda z](t) + B_\lambda[\lambda u](t), \\ y(t) &= Cz(t), \end{aligned} \tag{7.9}$$

where $[\lambda z](t)$ and $[\lambda u](t)$ denote the filtered signals $z(t)$ and $u(t)$, respectively, and

$$A_\lambda = I + \tau A, \quad B_\lambda = \tau B, \tag{7.10}$$

which for the current model (7.4) is

$$A_\lambda = \begin{pmatrix} I & \tau I \\ \tau A_{21} & I + \tau A_{22} \end{pmatrix}, \quad B_\lambda = \begin{pmatrix} 0 \\ \tau B_2 \end{pmatrix}. \tag{7.11}$$

It is straightforward to derive the following relations between the output, the transformed states, and the input using recursion

$$[\lambda^{i-1} y](t) = C[\lambda^{i-1} z](t), \tag{7.12}$$

$$\begin{aligned} [\lambda^k y](t) = {} & CA_\lambda^{i-1-k}[\lambda^{i-1} z](t) \\ & + \sum_{j=k+1}^{i-1} CA_\lambda^{j-k-1} B_\lambda [\lambda^j u](t), \quad 0 \le k \le i-2, \end{aligned} \tag{7.13}$$

[1] Even though the model (7.1) is linear in the parameters, the derivatives of the output, $\dot{y}(t)$ and $\ddot{y}(t)$, are typically not available for measurement and thus a least-squares solution is not directly applicable.

where $[\lambda^q y](t)$ means that $y(t)$ has been filtered with q serial-connected λ-filters and similarly for the input u and the state z. These relations enable the formulation of an extended linear model [Johansson et al., 1999] according to

$$\mathcal{Y} = \Gamma_z \mathcal{Z} + \Gamma_u \mathcal{U}, \quad \mathcal{Z} = \left([\lambda^{i-1} z](t)\right), \tag{7.14}$$

with

$$\mathcal{Y} = \begin{pmatrix} [\lambda^{i-1} y](t) \\ [\lambda^{i-2} y](t) \\ \vdots \\ y(t) \end{pmatrix}, \quad \mathcal{U} = \begin{pmatrix} [\lambda^{i-1} u](t) \\ [\lambda^{i-2} u](t) \\ \vdots \\ u(t) \end{pmatrix}. \tag{7.15}$$

Moreover, the extended observability matrix Γ_z is defined as

$$\Gamma_z = \begin{pmatrix} C \\ CA_\lambda \\ \vdots \\ CA_\lambda^{i-1} \end{pmatrix}, \tag{7.16}$$

and the matrix Γ_u is defined according to

$$\Gamma_u = \begin{pmatrix} 0 & 0 & \dots & 0 \\ CB_\lambda & 0 & \dots & \vdots \\ \vdots & \vdots & \ddots & 0 \\ CA_\lambda^{i-2} B_\lambda & CA_\lambda^{i-3} B_\lambda & \dots & 0 \end{pmatrix}. \tag{7.17}$$

Remark 1 The noise component of the dynamic model has been omitted in the transformation presented here; for details regarding the transformation of the noise component and representation of the model on innovations form, see [Johansson et al., 1999].

Subspace Identification

The identification of the matrices $\{A_\lambda, B_\lambda, C\}$ is based on the N4SID subspace algorithm [Van Overschee and De Moor, 1994; Van Overschee and De Moor, 1996]. However, to accommodate the predefined structure of the system to be identified, certain modifications are made. The strategy for obtaining the desired structure of the system matrices is related to the general approach for linear systems suggested in [Lyzell et al., 2009; Lyzell, 2012]. The filtered discrete-time input–output data are collected in the matrices

(similar to the Hankel matrices in the discrete-time case)

$$\mathcal{U}_N = \begin{pmatrix} [\lambda^{i-1}u]_1 & [\lambda^{i-1}u]_2 & \dots & [\lambda^{i-1}u]_N \\ [\lambda^{i-2}u]_1 & [\lambda^{i-2}u]_2 & \dots & [\lambda^{i-2}u]_N \\ \vdots & \vdots & \ddots & \vdots \\ [\lambda u]_1 & [\lambda u]_2 & \dots & [\lambda u]_N \\ u_1 & u_2 & \dots & u_N \end{pmatrix}, \tag{7.18}$$

where the sampled and filtered input at time t_k is denoted $[\lambda^q u]_k$ and an analogous construction $\mathcal{Y}_N$ is made for the filtered outputs $[\lambda^q y]_k$. Estimates of the system matrices, $\hat{A}_\lambda$ and $\hat{C}$, are then computed using Steps 1–5 of Algorithm 4.8 in [Van Overschee and De Moor, 1996], which provides the matrices up to a similarity transform. In order to fix the state space such that the desired form of the model in (7.4) is obtained, a state transformation $z \to Tz$, $T \in \mathbb{R}^{2n\times 2n}$, is made. Partition the state matrix estimates according to

$$\hat{A}_\lambda = \begin{pmatrix} \hat{A}_\lambda^{11} & \hat{A}_\lambda^{12} \\ \hat{A}_\lambda^{21} & \hat{A}_\lambda^{22} \end{pmatrix}, \quad \hat{C} = \begin{pmatrix} \hat{C}_1 & \hat{C}_2 \end{pmatrix}, \tag{7.19}$$

and then form the matrix T as follows

$$T = \begin{pmatrix} T_{11} & T_{12} \\ T_{21} & T_{22} \end{pmatrix}, \tag{7.20}$$

where

$$T_{11} = \hat{C}_1, \quad T_{12} = \hat{C}_2, \tag{7.21}$$

$$T_{21} = \frac{1}{\tau}(\hat{C}_1\hat{A}_\lambda^{11} + \hat{C}_2\hat{A}_\lambda^{21} - \hat{C}_1), \tag{7.22}$$

$$T_{22} = \frac{1}{\tau}(\hat{C}_1\hat{A}_\lambda^{12} + \hat{C}_2\hat{A}_\lambda^{22} - \hat{C}_2). \tag{7.23}$$

Based on the determined transformation matrix T, the estimated matrices $\hat{A}_\lambda$ and $\hat{C}$ are recomputed according to $\hat{A}_\lambda \to T\hat{A}_\lambda T^{-1}$ and $\hat{C} \to \hat{C}T^{-1}$. The transformed matrix $\hat{A}_\lambda$ then has the desired structure according to (7.11) and $\hat{C} = \begin{pmatrix} I & 0 \end{pmatrix}$. The extended observability matrix Γ_z and the estimates of the state sequence are subsequently recomputed based on the transformed estimates of the system matrices A_λ and C, as proposed in [Lyzell et al., 2009; Lyzell, 2012]. Using the recomputed matrix Γ_z and recomputed state estimates, the matrix B_λ is determined by solving the least-squares problem in Step 6 of Algorithm 4.8 in [Van Overschee and De Moor, 1996], with the additional constraint that the matrix should have the structure specified in (7.11). The upper block B_λ^1 of the matrix should be zero. Moreover, given that the matrix M is positive definite—and consequently the inverse M^{-1} as

well—it is clear that the lower sub-block B_λ^2 should be positive definite, *i.e.*, $B_\lambda^2 \succ 0$. These constraints are straightforward to enforce, since the determination of the estimate of the matrix B_λ is performed by solving a least-squares problem. The convexity is preserved when adding the linear constraint that the upper block of the matrix should be zero and the constraint that the lower block should be positive definite. As the final step, estimates of the original continuous-time system matrices $\hat{A}$ and $\hat{B}$ are determined from $\hat{A}_\lambda$ and $\hat{B}_\lambda$ using the bijective relations in (7.10). The final matrices then have the structure according to (7.4).

Remark 2 The considered method is not limited to the N4SID subspace algorithm. Other suggested algorithms, such as the MOESP algorithm [Verhaegen and Dewilde, 1992] or CVA algorithm [Larimore, 1990], can be modified for this gray-box identification purpose as well. As also discussed in [Lyzell et al., 2009; Lyzell, 2012], the fundamental property is that the algorithm is organized such that the system matrices are estimated in two main steps. In the first step, the matrices A_λ and C are computed to allow for fixing the state space on the desired form, and then the matrix B_λ is determined based on the recomputed extended observability matrix and estimated state sequence.

Remark 3 Analogously to the model transformation procedure presented earlier in this section, the noise model identification part has been omitted here. However, identification of this part is possible based on a description of the continuous-time model on innovations form, see [Johansson et al., 1999] for details.

Physical Parameter Estimation

In order to retrieve the physical parameter matrices, the mass matrix M is first estimated from the matrix B in the identified state-space model. It follows directly that the estimate is given by $\hat{M} = \hat{B}_2^{-1}$, which is positive definite because of the constraint enforced in the system identification procedure. With the mass matrix M estimated, the stiffness matrix K and damping matrix D are computed from the corresponding estimated blocks $\hat{A}_{21}$ and $\hat{A}_{22}$ in the system matrix according to

$$\begin{aligned} \underset{K}{\text{minimize}} \quad & ||K + \hat{M}\hat{A}_{21}||_F \\ \text{subject to} \quad & K \succeq 0 \end{aligned} \tag{7.24}$$

and

$$\begin{aligned} \underset{D}{\text{minimize}} \quad & ||D + \hat{M}\hat{A}_{22}||_F \\ \text{subject to} \quad & D \succeq 0 \end{aligned} \tag{7.25}$$

where $||\cdot||_F$ denotes the Frobenius norm. Requirements on positive semidefiniteness are imposed on K and D when solving the corresponding optimization problems. It is straightforward to verify that the problems (7.24)–(7.25) are convex [Boyd and Vandenberghe, 2004], and that a global minimum thus exists for each problem.

7.4 Simulation Results

In order to verify the considered approach to gray-box identification, simulated input–output data were determined from the multi-input multi-output (MIMO)-system (7.3) with $n = 2$ and the physical parameters chosen as follows

$$M = \begin{pmatrix} 5 & -1 \\ -1 & 1 \end{pmatrix}, \quad D = \begin{pmatrix} 5 & -0.01 \\ -0.01 & 1 \end{pmatrix}, \\ K = \begin{pmatrix} 100 & -5 \\ -5 & 100 \end{pmatrix}. \tag{7.26}$$

Two data sequences were simulated; one for identification and one for cross-validation. Each sequence contained a total of $N = 20000$ samples of input–output data with step inputs and a sampling period of $h = 0.005$ s, see Figure 7.2 for a visualization of the identification data. In addition, noise e_k was added to the measurements y from the distribution $e_k \in \mathcal{N}(0, \sigma^2 I)$ with $\sigma = 3 \cdot 10^{-4}$. The system can be considered as a two-dimensional spring-mass-damper system, with actuation along two directions X and Y, see Figure 7.1.

System Identification

The system matrices A, B, and C in a state-space model of order four were estimated based on the method outlined in Section 7.3. The time constant in the variable transformation (7.7) was chosen as $\tau = 0.15$, based on an iterative procedure. The convex optimization problems inherent in the identification procedure were solved using CVX [Grant and Boyd, 2008; CVX Research Inc. 2015] in MATLAB. Moreover, the maximum order of the filtering was selected as $i = 10$. The estimated system matrices are

$$\hat{A} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -24.4683 & -24.6485 & -1.3939 & -0.3884 \\ -19.1801 & -128.1179 & -1.3851 & -1.9230 \end{pmatrix}, \\ \hat{B} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0.2582 & 0.2590 \\ 0.2590 & 1.2926 \end{pmatrix}, \quad \hat{C} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}.$$

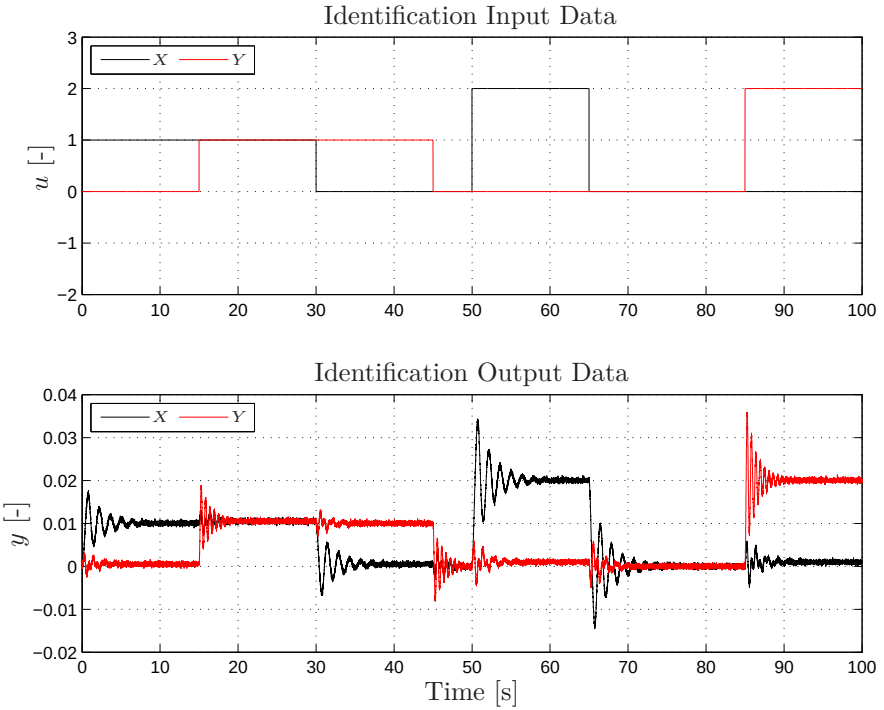

Figure 7.2 Simulated input–output identification data for the system with parameters in (7.26).

The validation data inputs were used as input to the estimated system with matrices $\hat{A}$, $\hat{B}$, and $\hat{C}$. A comparison of the validation data from the real system and the corresponding output $\hat{y}$ from the estimated system is shown in Figure 7.3. As a measure of the fit of the estimated model to the validation data, the normalized root-mean square error (NRMSE) values were used, see (3.2) in Chapter 3. This quantity was computed for the validation data for the respective axis X and Y. The values were 97.6% and 96.9%, respectively, which is considered as very good fit of the model to the validation data. To further compare the frequency characteristics of the actual and the estimated models, the magnitude plots of the Bode diagram are shown in Figure 7.4. Also here it is clear that the estimated system captures the essential dynamics of the original system, where in particular the natural eigenfrequencies and the transmission zeros of the system are identified with high accuracy.

Physical Parameters

To the purpose of estimating the physical parameters of the system, the mass matrix M was determined using the procedure defined in Section 7.3. Based

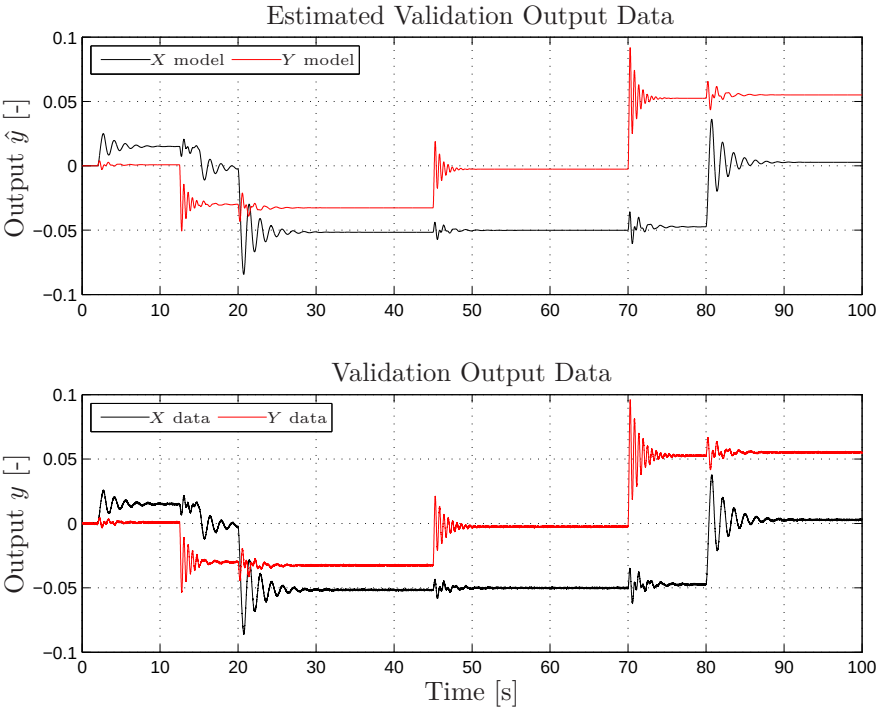

Figure 7.3 Comparison of validation output data for the system with parameters in (7.26) and the output from the estimated system model.

on $\hat{M}$, the estimates $\hat{K}$ and $\hat{D}$ of the stiffness matrix and the damping matrix, respectively, were computed by solving the optimization problems (7.24) and (7.25). For evaluation of the estimation accuracy, the obtained matrices were compared to the corresponding parameters for the real system in (7.26). The relative differences, measured with the Frobenius norm, are given by

$$\frac{||M-\hat{M}||_F}{||M||_F} = 0.0303, \quad \frac{||D-\hat{D}||_F}{||D||_F} = 0.125, \tag{7.27}$$

$$\frac{||K-\hat{K}||_F}{||K||_F} = 0.00108. \tag{7.28}$$

This indicates that also the mass matrix M and stiffness matrix K have been estimated with high accuracy and the damping matrix D with good accuracy.

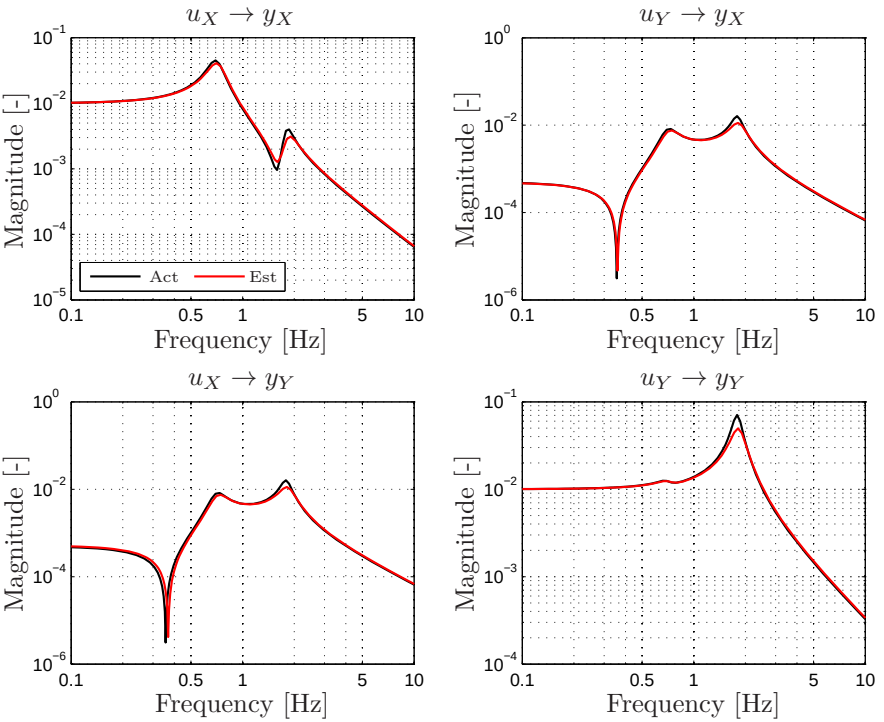

Figure 7.4 Comparison of the magnitude plots for the Bode diagrams of the actual system with parameters in (7.26) and the estimated MIMO-model.

7.5 Experimental Results

The considered identification method was further evaluated on data from an experimental setup. The system is the piezo-actuated micro-manipulator for robotic machining scenarios depicted in Figure 2.10 in Chapter 2, with actuation along the axes of a Cartesian coordinate system. As noted in the dynamic characterization in Section 3.3 in Chapter 3, the system exhibits significant resonances at particular eigenfrequencies and two of the axes, X and Z exhibit a noticeable coupling because of the mechanical design of the mechanism. This coupling was investigated using the method in Section 7.3. Consequently, a MIMO-system with the actuation forces u as inputs and the corresponding displacements y_X and y_Z along the X and Z axes, respectively, was identified. Since the actuating forces were not available for explicit measurement, it was assumed that the extensions of the piezo-actuators are proportional to the forces. Two sequences of experimental data were col-

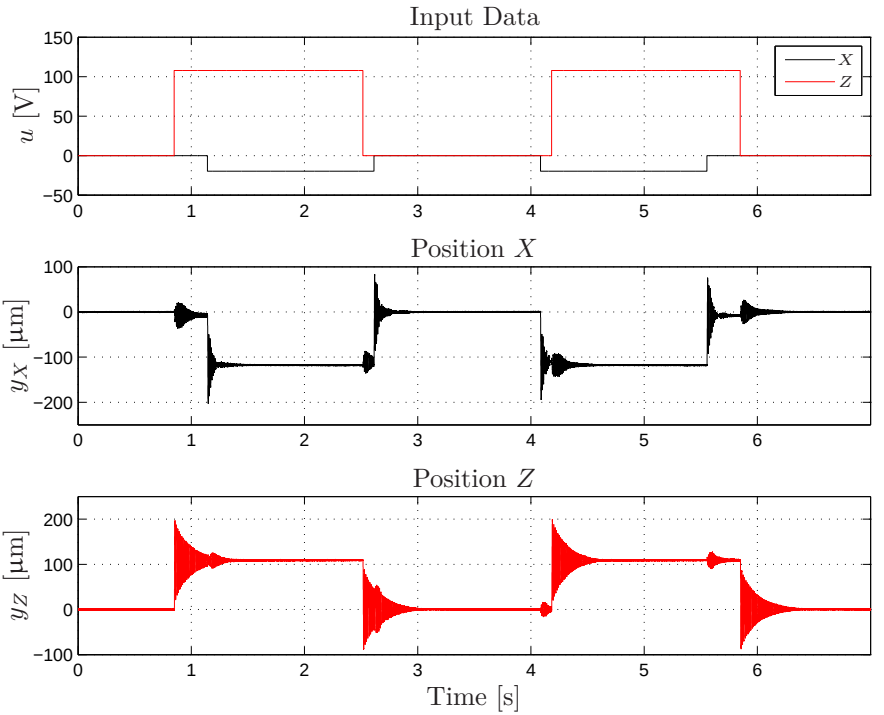

Figure 7.5 Experimental input–output data for the micro-manipulator. The dynamic coupling between the X- and Z-axes is clearly visible.

lected with a sampling period of $h = 0.0001$ s; one of the sets was used as identification data and the other set was used for cross-validation purposes. The sample rate was motivated by the observed natural eigenfrequencies, obtained by spectral analysis (see Section 3.3 in Chapter 3), in the order of 100 Hz and the fact that the data were to be processed with low-pass filters in the identification procedure. The inputs to the mechanical system were a sequence of steps with randomly chosen distances in time. The Cartesian extension of the end-effector was measured using the capacitive sensors and the collected experimental input–output data were detrended prior to system identification. The resulting identification data are displayed in Figure 7.5.

System Identification

A MIMO-model of the micro-manipulator of fourth order was estimated with the method in Section 7.3, using the identification data shown in Figure 7.5. The time constant $\tau = 0.00314$ was used in the operator transformation (7.7) for accommodating the frequency properties of the system at hand. The time

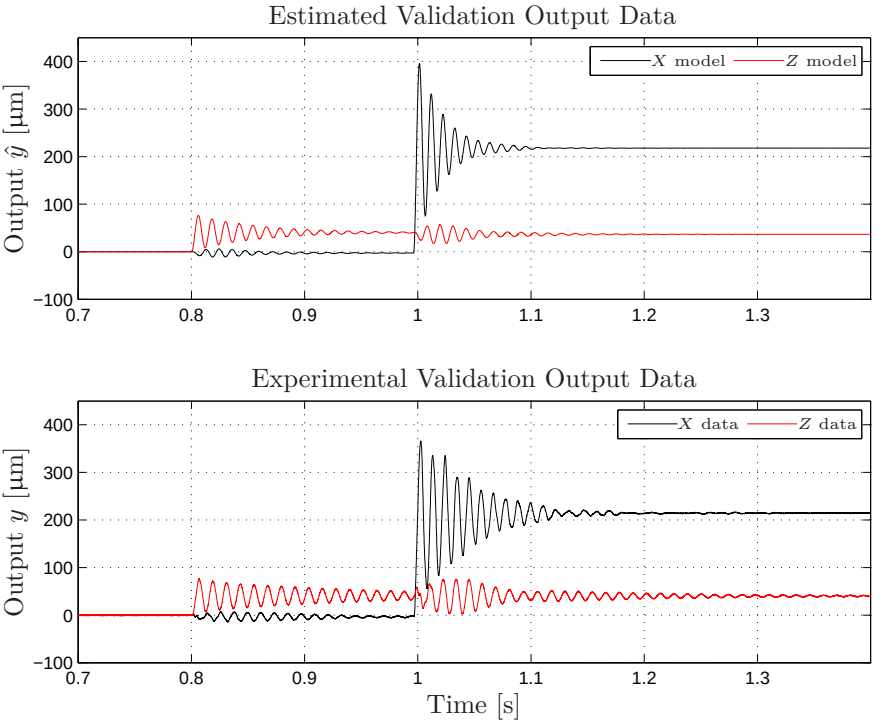

Figure 7.6 Comparison of experimental validation output data for the micro-manipulator and the output from the estimated system model.

constant was found by employing a linear search over a predefined interval and evaluating the corresponding model fit for the identification data series for each of the filter time constants. Moreover, the maximum filter order was selected as $i = 8$. For cross-validation of the estimated micro-manipulator model, the validation data series (containing a sequence of step inputs different from the identification data series) was used. The experimental validation output data and the corresponding output $\hat{y}$ from the estimated model for the same input signals are displayed in Figure 7.6 for a representative segment of the data series used. The NRMSE values for the computed model were 88.8% and 73.6% for the X and Z-axis, respectively, which indicates that the essential dynamics of the system is captured. The dynamics not captured by the model can be explained by higher-order harmonics, arising because of the nonlinear dynamics of the piezo-actuators used in the micro-manipulator design, and linear dynamics of higher order than the model assumes.

The frequency characteristics of the estimated model are represented by the magnitude plots for the Bode diagram in Figure 7.7. The natural eigenfre-

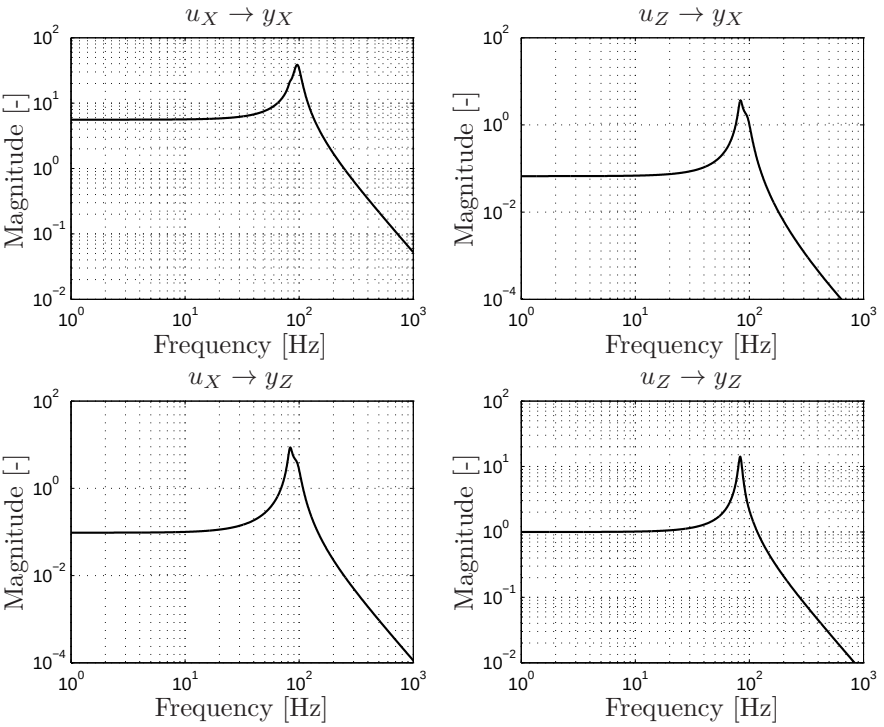

Figure 7.7 Magnitude plots for the Bode diagram of the estimated MIMO-model for the micro-manipulator.

quencies observed at 96.6 Hz and 83.0 Hz for the X- and Z-axis, respectively, correspond well to the eigenfrequencies extracted using spectral analysis of the experimental data. It is interesting to observe that it is the resonance along the Z-axis that results in the cross coupling between the axes—*i.e.*, actuation along the Z-axis results in oscillations along the X-axis as well with the eigenfrequency of the former.

Physical Parameters

The mass matrix M was estimated from the matrix $\hat{B}$ in the state-space model for the micro-manipulator. Moreover, using the estimated system matrix $\hat{A}$ in the model and the mass-matrix estimate $\hat{M}$, the stiffness matrix K and the damping matrix D were estimated as described in Section 7.3. The

estimates of the physical parameters are given by

$$\hat{M} = 1.0 \cdot 10^{-5} \begin{pmatrix} 0.0487 & 0.0001 \\ 0.0001 & 0.3669 \end{pmatrix}, \tag{7.29}$$

$$\hat{D} = 1.0 \cdot 10^{-3} \begin{pmatrix} 0.0407 & 0.0302 \\ 0.0302 & 0.1484 \end{pmatrix}, \tag{7.30}$$

$$\hat{K} = \begin{pmatrix} 0.1793 & 0.0147 \\ 0.0147 & 1.0049 \end{pmatrix}. \tag{7.31}$$

When analyzing the estimates of the physical parameters, the generalized eigenvalues s resulting in non-trivial solutions v (*i.e.*, generalized eigenvectors) to the quadratic eigenvalue equation $(Ms^2 + Ds + K)v = 0$ are of interest since they correspond to the eigenmodes of the system [Tisseur and Meerbergen, 2001]. In particular, the damped natural eigenfrequencies are given by the imaginary parts of the eigenvalues.

Undamped System First, the undamped system, *i.e.*, with $D = 0$, is considered, leading to the matrix equation $(Ms^2 + K)v = 0$. Solution of this equation provided the eigenvalues $s_1 = \pm 607.8i$ and $s_2 = \pm 522.2i$, corresponding to the undamped eigenfrequencies $\omega_1 = 96.7$ Hz and $\omega_2 = 83.1$ Hz. The eigenvectors v obtained are given by

$$v_1 = \begin{pmatrix} 0.999 \\ 0.0409 \end{pmatrix}, \qquad v_2 = \begin{pmatrix} -0.296 \\ 0.955 \end{pmatrix}. \tag{7.32}$$

Damped System Second, the complete damped system was considered. Solution of the generalized eigenvalue problem for the estimated mass, damping, and stiffness matrices gave the eigenvalues $s_1 = -44.2 \pm 605.2i$ and $s_2 = -17.8 \pm 522.6i$. These eigenvalues correspond to the damped natural eigenfrequencies $\omega_1 = 96.3$ Hz and $\omega_2 = 83.2$ Hz, which is well in agreement with the eigenfrequencies observed in the magnitude plots for the Bode diagram in Figure 7.7. Further, the eigenvectors v are given by

$$v_1 = \begin{pmatrix} -0.0745 \pm 0.995i \\ 0.0341 \pm 0.0529i \end{pmatrix}, \qquad v_2 = \begin{pmatrix} 0.204 \mp 0.351i \\ 0.0546 \pm 0.912i \end{pmatrix}. \tag{7.33}$$

Comparing the obtained eigenvectors for the undamped and the damped case, respectively, it is clear that information about the eigenmodes of the micro-manipulator is obtained in both settings. Note that, as expected, the eigenvectors for the damped case are complex in contrast to the undamped case. This is a result of that the damping leads to more complicated dynamics for the oscillations relating to the eigenmodes, compared to the undamped case where the eigenvectors are real.

7.6 Discussion

Gray-box identification of linear models for mechanical systems with flexible modes has been considered. The characterization of the dynamic force–deflection relationships does not only provide information about the mechanical system as such, but is also essential for model-based control design. Important examples here are contact operations for robot manipulators in manufacturing scenarios such as machining.

Using subspace-based methods for identification in this context is natural, since the systems to be modeled are of MIMO character and subspace methods have been found to be advantageous when modeling systems with closely spaced natural eigenfrequencies (and thus resulting resonances) [Johansson et al., 2000]. Since this is often the case with flexible systems—*cf.* the micro-manipulator system investigated in Section 7.5—this is an essential property of the method.

In order to allow for continuous-time identification in the time-domain, a variable transformation was made in (7.7). This transformation includes the choice of the filter time constant τ. The choice of this constant has been found essential in order to obtain models with satisfactory fit to the data and even stability of the algorithm. Intuitively, it is natural that the frequency content of the experimental identification data and the sampling rate of the same have implications on the choice. In the experimental results presented in Section 7.5, a linear search was used for finding the time constant maximizing the fit to the identification data in a predefined interval. Another option is to use multiple time constants in a logarithmically spaced interval for initial examination of the identification data. A drawback with this option is that the computational complexity increases.

Another aspect is if it is sufficient to use linear models, since it is known that, *e.g.*, robot manipulators in some cases exhibit nonlinear stiffness when applying strong external process forces at the end-effector (see Chapter 6). In addition, configuration-dependent compliance properties are expected. First, it is plausible that linear models are appropriate as initial approximations when designing position controllers in many applications. Second, within a certain limited range of input signals in a limited Cartesian workspace, the linear approximation is indeed valid. For modeling of configuration-dependent robot compliance dynamics, linear parameter-varying models can also be considered. Moreover, the method for identification of the nonlinear quasi-static stiffness-relationship discussed in Chapter 6 can be combined with the dynamic identification considered in this chapter.

Regarding the experimental results obtained by application of the method on the micro-manipulator, these can be used for further investigations of the mechanical design. For example, the computed eigenvalues and eigenvectors provided information about the eigenmodes of the coupled dynamics between

the X- and Z-axes. In particular, further analysis of the eigenvectors can be used as a basis for future mechanical design of micro-manipulators for robot machining.

In the problem formulation, it was assumed that the linear dynamics is of second order (*cf.* the relation (7.1)) and that all position states were available for measurement. This is a limitation when considering identification of systems with higher-order dynamics and non-actuated modes whose corresponding outputs are not easily measurable. It is an interesting aspect of future research to investigate how the method can be adopted to these extensions when determining the dynamic model from the input–output data.

7.7 Conclusions

This chapter has considered a method for identification of continuous-time gray-box models in the time-domain by using experimental input–output data. The method relies on subspace-based identification for computing the matrices of a state-space model and estimating the physical parameters. Moreover, the method was successfully evaluated both in simulation and in experiments, where the obtained models exhibited good fit to the data and the model-parameter matrices were feasible from a physical point-of-view.

8

Sensor Fusion for Robotic Workspace State Estimation

8.1 Introduction

As already discussed in Chapter 2 and Chapter 6, joint dynamics influence the achievable workspace position accuracy when performing tasks with robot manipulators. Such dynamic effects lead to differences between the motor-side angle of the joint and the corresponding arm-side angle. Therefore, the problem of robot pose state estimation in workspace by sensor fusion of the internal robot-joint measurements with inertial measurement unit (IMU) data is considered in this chapter. A prerequisite for this to be successful is accurate calibration of the sensors used. Therefore, a method for calibration of the IMU with respect to the robot end-effector is established, which is straightforward to apply on an arbitrary industrial robot manipulator. Two different workspace state-estimation algorithms are considered and evaluated experimentally. This chapter of the thesis is derived, in part, from the manuscript [Olofsson et al., 2015].

The structure of this chapter is as follows: Section 8.2 provides a background to the subject and discusses related previous research. In Section 8.3, the calibration procedure, the modeling, and the state-estimation algorithms are presented. The setup used for evaluation of the methods as well as the main experimental validation and results are presented in Section 8.4. In addition, an evaluation of the robustness to the choices of the filter parameters and the noise modeling is presented in this section. A discussion of the achieved results and the methods for sensor calibration and state estimation is provided in Section 8.5, followed by a summary and conclusions from the results in Section 8.6.

8.2 Background

A prerequisite for modern robot-manipulator control is external sensors providing information about both internal robot states and workspace states. The obtained state information is subsequently used in the control of the robot manipulator in advanced applications. In light of the discussion of nonlinear joint dynamics in Chapter 6 and eigenmodes in the links constituting the robot manipulator, workspace or arm-side sensors are important tools for accurate control of the robot tool pose.

Force/torque sensors have been the typical choice for acquiring data from the robot workspace during several decades [Siciliano and Villani, 1999] when considering machining applications [Sörnmo et al., 2012b; Sörnmo et al., 2015] and assembly tasks [Stolt et al., 2011], where contact between the manipulator and the workpiece is required. One option for acquiring workspace pose estimates is to use high-accuracy optical tracking systems for 3D or 6D measurements such as in Chapters 3–5. However, such measurement systems are expensive. Instead, statistical fusion [Gustafsson, 2010] of multiple external as well as robot-internal sensors (capable of delivering online measurements with the required sample rate) has been proposed as an alternative for estimating the workspace pose of the robot tool, see, *e.g.*, [Axelsson et al., 2012; Henriksson et al., 2009; Chen and Tomizuka, 2014; Norén, 2014]. More specifically, accelerometers and IMUs have been suggested in several publications, see references later in this section, to be used for estimation of the workspace position and in some cases orientation of the tool. This also requires modeling of the robot motion. Typically, a detailed dynamic model of the nonlinear dynamics of the robot is not accessible, or only known by the robot manufacturer. Robot modeling and subsequent identification for industrial robots are extensively discussed in [Moberg et al., 2014].

The focus of this chapter is twofold. First, pose calibration of IMUs attached to the tool of a robot manipulator with respect to the robot end-effector plate is discussed. Second, tool pose state estimation for serial-kinematic robot manipulators using sensor fusion of IMU data and pose estimates obtained from the forward kinematics of the robot is investigated. The state estimation is approached by means of two different algorithms; the first is based on the Extended Kalman Filter (EKF) [Särkkä, 2013; Johansson, 1993] and the second is based on the Rao-Blackwellized Particle Filter (RBPF) [Schön et al., 2005].

Previous Research and Contributions

Several approaches to statistical sensor fusion for state estimation of the pose of a rigid body have been proposed in the literature and thus the pose-estimation problem has been studied in great detail. One example using an array of accelerometers was presented in [Parsa et al., 2004]. In [Hol, 2011],

numerous examples of pose estimation using the EKF can be found. Further, [Li et al., 2013] and [Vernaza and Lee, 2006] utilize an RBPF for pose estimation. The problem of estimating only orientation has also been extensively studied, for applications in spacecraft tracking see, *e.g.*, the review in [Crassidis et al., 2007]. In this chapter, the focus will be on applications for robot manipulators. The differences between the methods proposed in this area are mainly in the sensors employed, in the complexity of the employed dynamic models, and in the state-estimation algorithms used. The Kalman Filter (KF) [Kalman, 1960], and nonlinear extensions of it such as the EKF, have been the traditional choice, even though the Particle Filter (PF) [Gordon et al., 1993] has been applied recently in this area, see [Axelsson et al., 2012; Rigatos, 2009]. The models employed are either based on kinematic relations in Cartesian space [Norén, 2014; Jeon et al., 2009] or include more complex dynamic modeling in joint space [Axelsson et al., 2012; Chen and Tomizuka, 2014; De Luca et al., 2007; Karlsson and Norrlöf, 2004; Lertpiriyasuwat et al., 2000]. However, even with the more complex models including elasticity, non-differentiable dynamics present in robot joints, such as backlash and friction, are difficult to account for, which makes also such models approximations of the actual system dynamics. Moreover, in the literature also the capabilities of direct measurement of the position in workspace, acceleration with an accelerometer, and angular rate-of-change with a gyroscope differ for the proposed methods. Another fundamental difference is the computational time for the suggested algorithms; if the state estimates are to be used for online control, real-time requirements must be put on the execution of the state-estimation algorithms. Some of the proposed sensor-fusion algorithms for robot state estimation, in particular those based on particle filters, do not permit real-time executions at sampling frequencies used in modern robot controllers. An alternative application area for these methods is ILC in an offline scenario [Arimoto et al., 1984; Axehill et al., 2014; Axelsson et al., 2013], where online measurements are not required given the batch-oriented nature of the algorithm.

An inherent problem in sensor fusion for robots using IMUs is calibration. More specifically, the translation and orientation of the IMU coordinate system must be determined with respect to a known coordinate system fixed on the robot. This problem was considered for accelerometers in the context of serial-kinematic robot manipulators in [Axelsson and Norrlöf, 2012]. Translation and orientation estimation as well as calibration of the internal IMU parameters were considered in [Magnussen et al., 2015] using a Stewart platform. Pose calibration of a pair of IMUs attached to an imperfect kinematic chain using nonlinear optimization was considered in [Birbach and Bauml, 2014], where modeling similar to the one in this chapter was employed. In [Hol, 2011], a general method for relative pose calibration of several IMUs was presented.

The contributions of this chapter are twofold: First, a procedure for pose calibration of one or several IMUs attached to an end-effector of a robot manipulator is developed, which relates to the general method for IMU pose calibration proposed in [Hol, 2011]. In addition, an experimental verification of the method is performed. The method is straightforward to apply on an arbitrary configuration of IMU sensors on the robot, and computes the parameters by first solving *Procrustes problem* [Golub and Van Loan, 1996] and then a linear least-squares problem. Second, two different methods for estimation of the workspace states of a robot manipulator are considered, requiring a minimum of *a priori* information about the robot dynamics. One of the algorithms is based on an EKF, while the other is a variant of an RBPF, where the noise of the IMU measurements are modeled using the multivariate Student's t-distribution instead of a Gaussian distribution. The full three-dimensional measurements of the angular rate-of-change from the gyroscope and the three-dimensional linear acceleration from the accelerometer are included, as well as all information from the forward kinematics of the robot. This enables workspace state estimation for an industrial manipulator with six degrees-of-freedom (position and orientation). Robot pose estimation in workspace based on kinematic models has been considered in [Jeon et al., 2009; Norén, 2014], but only using the EKF algorithm. The EKF algorithm is nevertheless included and used in this chapter as a reference method in the experimental evaluation. Even though RBPF-based pose estimation has been considered previously using IMU, GPS, and radio sensors, see, *e.g.*, [Li et al., 2013; Vernaza and Lee, 2006], the RBPF algorithm is here applied to the case of robot manipulators. Both the calibration procedure and the state-estimation algorithms are evaluated in extensive experiments on an industrial manipulator. The two state-estimation algorithms discussed are shown to be possible to execute on a standard PC as an external computational service within the real-time constraints of a robot controller running at a sampling period of 4 ms, thus permitting online access of the estimates in the robot controller.

8.3 Method

This section defines the IMU pose calibration procedure, presents the system modeling for state estimation and defines the filtering algorithms. Without loss of generality, three coordinate systems are defined on the robot; the world coordinate system Ψ_W at the robot base, the flange coordinate system Ψ_F at the robot flange (end-effector plate), and the body coordinate system Ψ_B on the IMU, which is assumed to coincide with the robot-tool coordinate system. The coordinate systems are depicted in Figure 8.1. The sensor signals assumed available for the calibration procedure and the state estimation

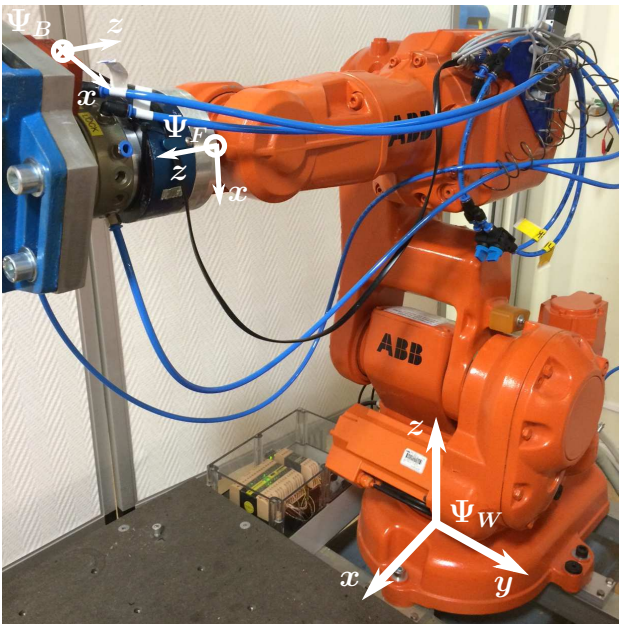

Figure 8.1 Figure defining the coordinate systems used in the calibration procedure and the state estimation. The robot visualized is the one used for the experimental evaluations. The world coordinate system Ψ_W is fixed in space, whereas the flange Ψ_F and body Ψ_B coordinate systems are moving with the robot flange and the IMU, respectively.

are joint motor-position measurements $\theta \in \mathbb{R}^6$ (obtained from the motor resolvers in the joints and using nominal values for the gear ratios of the joint transmissions), 3D accelerometer measurements $a_B^B \in \mathbb{R}^3$ on the robot tool, and 3D gyroscope measurements $\omega_B^B \in \mathbb{R}^3$ on the robot tool[1]. In addition, the forward kinematic relations $\xi(\theta)$ of the robot are assumed available, such that an estimate of the flange position $r_{WF}^W \in \mathbb{R}^3$ and orientation $q^{WF} \in \mathbb{R}^4$ (represented using a unit quaternion) can be computed based on the joint-motor positions θ.[2]

[1] Throughout this chapter, $(\cdot)_A^B$ refers to that a measurement is acquired at the origin of frame Ψ_A and that it is expressed in the coordinate system Ψ_B.

[2] For position vectors r_{AB}^C in this chapter, the superscript refers to the coordinate system Ψ_C and the subscript refers to that the vector is from the origin of frame Ψ_A to the origin of frame Ψ_B. For quaternions q^{AB}, and later for corresponding rotation matrices, the superscript refers to a rotation from frame Ψ_B to frame Ψ_A.

Calibration Procedure

The calibration procedure consists of determining the translation vector $r_{FB}^F \in \mathbb{R}^3$ from the origin of Ψ_F to the origin of Ψ_B and the rotation matrix $R^{BF} \in \mathbb{R}^{3\times3}$ from the frame Ψ_F to the frame Ψ_B. Throughout this chapter, it is assumed that the geometry of the robot is known and that all joint angles on the motor side are measured. The unknown calibration parameters of the sensor are inferred from experimental data. The calibration procedure is similar in modeling and structure to the two-step procedure proposed in [Hol, 2011], but is here derived in the setting of robot manipulators and with a different parametrization of the orientation and a total least-squares solution for the translation-vector estimation. The flange linear velocity $v_F^F \in \mathbb{R}^3$ and rotational velocity $\omega_F^F \in \mathbb{R}^3$ (*i.e.*, the twist) are determined based on the differential kinematics of the robot

$$\begin{pmatrix} v_F^F \\ \omega_F^F \end{pmatrix} = J(\theta)\dot{\theta}, \tag{8.1}$$

where $J(\theta) \in \mathbb{R}^{6\times6}$ is the Jacobian of the manipulator and $\dot{\theta}$ are the joint velocities (obtained by numerical differentiation and subsequent filtering of the joint-position measurements). Define the following functions of the calibration parameters

$$\begin{aligned} \mathcal{J}_B &= a_B^B + R^{BF} g^F \\ &\quad - \left(R^{BF} a_F^F + \omega_B^B \times (\omega_B^B \times R^{BF} r_{FB}^F) + \alpha_B^B \times R^{BF} r_{FB}^F\right), \qquad (8.2) \\ \mathcal{J}_F &= a_F^F - g^F - \left((R^{BF})^{\mathrm{T}} a_B^B - \omega_F^F \times (\omega_F^F \times r_{FB}^F) - \alpha_F^F \times r_{FB}^F\right), \qquad (8.3) \end{aligned}$$

where $\times$ denotes the vector cross product, $\alpha_F^F \in \mathbb{R}^3$ is the angular acceleration of the robot flange, $\alpha_B^B \in \mathbb{R}^3$ is the angular acceleration of the IMU (both quantities obtained by numerical differentiation of the corresponding velocities ω_F^F and ω_B^B and subsequent zero-phase non-causal[3] low-pass filtering with a Butterworth filter), and $g^F \in \mathbb{R}^3$ is the gravitational acceleration in the frame Ψ_F. Employing basic mechanics, it can be verified that $\mathcal{J}_B = \mathcal{J}_F = 0$ in the case of ideal measurements and calibration[4]. The estimates of the rotation matrix and the translation vector can thus be computed by minimizing the squares of the functions $\mathcal{J}_B$ and $\mathcal{J}_F$ in (8.2)–(8.3). Since R^{BF} is a rotation matrix, it must further satisfy the relation $(R^{BF})^{\mathrm{T}} R^{BF} = 1$.

[3] Note that the non-causality of the filter is in general not a restriction, since the sensor calibration is typically performed offline.

[4] Here, it is assumed that there are no biases on the accelerometer and gyroscope measurements. It is possible, though, to include a constant bias component for each axis in the calibration procedure. The time duration of the experiment for the calibration is assumed short, such that effects of time-varying biases can be neglected. See [Birbach and Bauml, 2014] for the case of time-varying biases.

To ensure this and to reduce the dimension of the parameter space, R^{BF} can be parametrized in the corresponding Euler angles[5]. Minimization of the squared functions in (8.2)–(8.3) results in a nonlinear optimization problem, which can be solved using, *e.g.*, the Levenberg-Marquardt algorithm [Levenberg, 1944; Marquardt, 1963]. The advantage of this method is that both the translation and the orientation are computed simultaneously. Nevertheless, the obvious limitation is that the numerical optimization might converge to local minima and thus provide a suboptimal solution. Therefore, another method is considered, where the parameters are computed in two steps. With the latter approach, the parameters of the rotation matrix R^{BF} are determined in an initial step.

Computation of the Rotation Matrix Considering the fact that the rotational velocity of a rigid body is the same at all points, the following relation holds

$$\omega_B^B = R^{BF}\omega_F^F. \tag{8.4}$$

Solving for R^{BF} in this equation while retaining the orthogonality of the matrix is the *Procrustes problem* [Golub and Van Loan, 1996]. Assuming that N samples of rotational velocity data have been collected in the matrix $\Gamma_B^B \in \mathbb{R}^{N\times 3}$ for the IMU data and correspondingly in the matrix $\Gamma_F^F \in \mathbb{R}^{N\times 3}$ for the robot data, the solution is given by, [Golub and Van Loan, 1996],

$$\hat{R}^{BF} = UV^{\mathrm{T}}, \tag{8.5}$$

where the matrices U and V are the result of a singular-value decomposition of the matrix $A = (\Gamma_B^B)^{\mathrm{T}}\Gamma_F^F$, *i.e.*, $A = U\Sigma V^{\mathrm{T}}$. An alternative approach to estimating the relative orientation between Ψ_F and Ψ_B is to use a quaternion parametrization instead of the rotation matrix, as proposed in [Hol, 2011].

Computation of the Translation Vector To compute the translation vector r_{FB}^F, it is noted that with the rotation matrix known, the relations (8.2)–(8.3) are linear in the desired parameters [Hol, 2011]. Hence, given N samples of the sensor and robot data, it is straightforward to formulate a linear model in the sought calibration parameter r_{FB}^F. Let Λ and λ be the matrices containing the measurement data, which are constructed from the relations $\mathcal{J}_B = 0$ and $\mathcal{J}_F = 0$ such that

$$\Lambda r_{FB}^F = \lambda. \tag{8.6}$$

When collecting the data for the identification, it is necessary to excite the robot along all the translational and rotational dimensions in order for the estimates of the rotation matrix and the translation vector to be consistent.

[5] Throughout this chapter, the XYZ convention is used for the Euler rotation angles.

Moreover, it is important not to excite the eigenmodes of the robot by using excessively high velocities, such that the relations (8.2)–(8.3) describing rigid-body motion hold. The effects on the calibration accuracy from friction in the joints are suppressed by the joint-position feedback controllers, and the influence of possible backlash in the gear-box for each joint is assumed negligible compared to the uncertainties in the IMU measurements.

It is straightforward to solve (8.6)—*i.e.*, computing an estimate of the translation vector r^F_{FB}—by employing the ordinary least-squares algorithm [Johansson, 1993]. However, considering that both the matrix $\Lambda \in \mathbb{R}^{6N\times 3}$ and the vector $\lambda \in \mathbb{R}^{6N}$ contain measurement data from the IMU and the robot forward kinematics with inherent uncertainty, a total least-squares solution [Golub and Van Loan, 1996] may be preferred compared to an ordinary least-squares solution. Forming the matrix $B = \begin{pmatrix}\Lambda & \lambda\end{pmatrix}$ and computing the singular-value decomposition $B = U\Sigma V^{\mathrm{T}}$, the estimate of the translation vector is given by, [Golub and Van Loan, 1996],

$$\hat{r}^F_{FB} = -V_{12}V_{22}^{-1}, \quad \text{where } V = \begin{pmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{pmatrix}. \tag{8.7}$$

Modeling for State Estimation

Two different models of the dynamics for the motion of the robot end-effector are established. The dynamics of the translational states and the measurements provided by the forward kinematics of the robot are modeled as being linear and are treated the same way for the two estimation algorithms considered. The measurement relation for the IMU and the dynamics of the orientation are inherently nonlinear, however, and the modeling of these therefore differs because of the nature of the two state-estimation algorithms.

Nonlinear models must be linearized in the EKF, and the EKF algorithm only allows Gaussian measurement noise [Kailath et al., 2000]. On the other hand, the RBPF does not require a linearized model and offers flexibility in the choice of noise modeling for the nonlinear states [Schön et al., 2005]. These properties of the RBPF allow a model of the orientation dynamics with fewer simplifications and assumptions than for the EKF, as well as the modeling of a subset of the noise processes as multivariate Student's t-distributions. This distribution is known to better accommodate for outliers in the data than the Gaussian distribution [Bishop, 2006].

First, the dynamic models for the RBPF and the EKF are defined. Then, the measurement models for the IMU and the forward kinematics are established for the two state-estimation algorithms. Finally, the complete models are listed for convenient reference.

Dynamic Model Introduce the states

$$x^l_k = \begin{pmatrix} p_k & v_k & a_k & b^a_k \end{pmatrix}, \tag{8.8}$$

where $p_k = (p_B^W)_k \in \mathbb{R}^3$ is the position, $v_k = (v_B^W)_k \in \mathbb{R}^3$ the velocity, and $a_k = (a_B^W)_k \in \mathbb{R}^3$ the acceleration of the tool, all given in the world frame Ψ_W, and $b_k^a = (b_B^{a,B})_k \in \mathbb{R}^3$ is the bias of the accelerometer given in the body frame Ψ_B. The translational states are modeled using a constant acceleration model, and the accelerometer bias is modeled using a random walk, see, *e.g.*, [Gustafsson, 2010]. Consequently, define the matrices

$$F^l = \begin{pmatrix} I_3 & T_s I_3 & \frac{T_s^2}{2} I_3 & 0 \\ 0 & I_3 & T_s I_3 & 0 \\ 0 & 0 & I_3 & 0 \\ 0 & 0 & 0 & I_3 \end{pmatrix}, \tag{8.9}$$

and

$$B^l = \begin{pmatrix} \frac{T_s^3}{6} I_3 & 0 \\ \frac{T_s^2}{2} I_3 & 0 \\ T_s I_3 & 0 \\ 0 & T_s I_3 \end{pmatrix}, \tag{8.10}$$

where T_s is the sampling period. The discretized dynamic model of the linear translational states and the accelerometer bias is then given by

$$x_{k+1}^l = F^l x_k^l + B^l n_k^l, \tag{8.11}$$

where $n_k^l \in \mathcal{N}(0, Q^l)$ is white Gaussian noise.

A unit quaternion parametrizing the relative orientation between Ψ_W and Ψ_B is introduced as

$$q^{BW}(t) = \begin{pmatrix} q^1(t) & q^2(t) & q^3(t) & q^4(t) \end{pmatrix}^{\mathrm{T}} \in SO(3). \tag{8.12}$$

To avoid additional states in the model, the angular velocity $\omega(t) = (\omega_B^B)(t)$ of the tool is treated as an input. Thus, let $\omega(t) = \begin{pmatrix} \omega^1(t) & \omega^2(t) & \omega^3(t) \end{pmatrix}^{\mathrm{T}} \in \mathbb{R}^3$ and introduce the matrix

$$\Omega(\omega) = \begin{pmatrix} 0 & \omega^3 & -\omega^2 & \omega^1 \\ -\omega^3 & 0 & \omega^1 & \omega^2 \\ \omega^2 & -\omega^1 & 0 & \omega^3 \\ -\omega^1 & -\omega^2 & -\omega^3 & 0 \end{pmatrix}, \tag{8.13}$$

where the dependence on time is implicit for notational convenience. The continuous-time dynamic model for the quaternion states is then given by, [Shuster, 1993],

$$\dot{q}^{BW}(t) = \frac{1}{2} \Omega(\omega(t)) q^{BW}(t). \tag{8.14}$$

Moreover, define the discrete-time function

$$f(\omega_k) = \left(\cos\left(\|\omega_k T_s\|/2\right) I_4 + \frac{\sin\left(\|\omega_k T_s\|/2\right)}{\|\omega_k\|} \Omega(\omega_k) \right). \tag{8.15}$$

Approximate zero-order hold (ZOH) sampling [Åström and Wittenmark, 1997] gives the discretized model as

$$q_{k+1}^{BW} = f(\omega_k) q_k^{BW}, \tag{8.16}$$

see, *e.g.*, [Törnqvist, 2008] for a derivation. Let $\mathrm{St}(\mu, \Sigma, \nu)$ denote the location-scale multivariate Student's t-distribution with mean μ, variance Σ, and ν degrees of freedom [Bishop, 2006]. Assuming Student's t-distributed noise on the measured input from the gyroscope, $\tilde{\omega}_k$, leads to the model

$$\omega_k = \tilde{\omega}_k - n_k^{\omega}, \tag{8.17}$$

where $n_k^{\omega} \in \mathrm{St}(0, Q^{\omega}, \nu^{\omega})$ are independent for all k and also with n_k^l for all k. Using this relation in (8.16) gives the dynamic model used for the quaternion in the RBPF. The IMU used in the experimental evaluation, see Section 8.4, had internal bias correction for the gyroscope and the measurements did not show any significant bias for the considered experiments, suggesting that removing the bias is a valid approximation. Thus, to simplify the algorithm, a bias term is not included for the gyroscope measurements for the RBPF.

For the EKF state-estimation algorithm, all noise processes must be Gaussian and the noise is assumed white. Further, the dynamic model and the measurement model must be linearized. In contrast to the RBPF, it is comparably straightforward to incorporate a bias term for the gyroscope measurement in the EKF. Therefore, by introducing such a bias term $b_k^{\omega} = (b_B^{\omega,B})_k$ (given in the body frame), the angular velocity in the EKF is modeled as

$$\omega_k = \tilde{\omega}_k - b_k^{\omega} - n_k^{\omega}, \tag{8.18}$$

where $n_k^{\omega} \in \mathcal{N}(0, Q^{\omega})$ are independent for all k and also with n_k^l for all k. Introduce the matrix

$$\Xi(q^{BW}) = \begin{pmatrix} q^4 & -q^3 & q^2 \\ q^3 & q^4 & -q^1 \\ -q^2 & q^1 & q^4 \\ -q^1 & -q^2 & -q^3 \end{pmatrix}, \tag{8.19}$$

where again the time-dependence is implicit. Linearizing (8.16) around the current state estimate and assuming that $\|\omega_k T_s\|/2$ is small, the small-angle approximation gives the relation, [Törnqvist, 2008],

$$q_{k+1}^{BW} = \left(I_4 + \frac{T_s}{2} \Omega(\omega_k) \right) q_k^{BW}, \tag{8.20}$$

which, using (8.18) and the fact that $\Omega(\omega_k)q_k^{BW} = \Xi(q_k^{BW})\omega_k$ according to [Shuster, 1993], is equivalent to

$$q_{k+1}^{BW} = \left(I_4 + \frac{T_s}{2}\Omega(\tilde{\omega}_k)\right) q_k^{BW} - \frac{T_s}{2}\Xi(q_k^{BW})\left(b_k^{\omega} + n_k^{\omega}\right). \tag{8.21}$$

A random walk is assumed for the gyroscope-bias dynamics,

$$b_{k+1}^{\omega} = b_k^{\omega} + T_s n_k^{b^{\omega}}, \tag{8.22}$$

where $n_k^{b^{\omega}} \in \mathcal{N}(0, Q^{b^{\omega}})$ is white Gaussian noise, independent of the other noise processes in the model for all k.

Measurement Model Let $R(q_k^{BW})$ be the rotation matrix formed by the quaternion q_k^{BW}, transforming a vector from the frame Ψ_W to the frame Ψ_B. For details on orientation representations, the reader is referred to [Shuster, 1993]. The measurement model for the accelerometer, in the frame Ψ_B, is given by

$$y_k^a = R(q_k^{BW})\left(a_k + g^W\right) + b_k^a + e_k^a, \tag{8.23}$$

where $e_k^a \in \mathcal{N}(0, R^a)$ is white noise independent of all other noise processes in the model, and g^W is the constant gravitational acceleration in the world frame Ψ_W.

The most natural way to model the measurements provided by the forward kinematics is to assume that the joint-space measurements are independent and identically distributed (iid). However, in that case the propagation of white noise through the nonlinear forward kinematics would have to be modeled, which is nontrivial and would make the algorithms unnecessarily complicated. Since the signal-to-noise ratios of these measurements are high, the correct noise modeling is not imperative for the performance of the model. The forward-kinematics estimates are thus approximated as being iid measurements of position, velocity, and orientation in the workspace directly. Even though it is known that the robot joints and links exhibit complex dynamics (see Chapter 6), another simplifying assumption introduced here is that the difference between the actual arm-side pose and the corresponding pose estimate computed from the motor-side measurements θ can be modeled as white noise. This simplification is justified in this chapter by the desire to be able to perform state estimation online. Using the previously stated assumptions, the relations for the forward-kinematics measurements (transformed from the flange frame Ψ_F to the body frame Ψ_B using the IMU calibration parameters $\hat{R}^{BF}$ and $\hat{r}_{FB}^F$) are given in the world frame Ψ_W by

$$y_k^p = p_k + e_k^p, \quad y_k^v = v_k + e_k^v, \quad y_k^q = q_k + e_k^q, \tag{8.24}$$

where $e_k^p \in \mathcal{N}(0, R^p)$, $e_k^v \in \mathcal{N}(0, R^v)$, and e_k^q are uncorrelated white noises independent of the noise processes in the dynamic models and the measurement

model of the accelerometer. For the EKF, $e_k^q \in \mathcal{N}(0, R^q)$, while for the RBPF the more robust Student's t-distribution is used instead as $e_k^q \in \mathrm{St}(0, R^q, \nu^q)$.

Summary of the Models The state-space models for the EKF and the RBPF are summarized in this section for completeness. For details on the noise distributions, the reader is referred to the previous subsection. To simplify the notation, let $q_k = q_k^{BW}$. Also, recall (8.8) that defines the vector x_k^l of the linear states.

EKF The EKF estimates the states in the vector

$$\begin{pmatrix} p_k & v_k & a_k & b_k^a & q_k & b_k^\omega \end{pmatrix}. \tag{8.25}$$

Define the EKF system matrix

$$F_k = \begin{pmatrix} F^l & 0 & 0 \\ 0 & \left(I_4 + \frac{T_s}{2}\Omega(\tilde{\omega}_k)\right) & -\frac{T_s}{2}\Xi(q_k) \\ 0 & 0 & I_3 \end{pmatrix}. \tag{8.26}$$

Using the translational dynamics (8.11) and the quaternion dynamics (8.20) together with the measurement equations (8.23) and (8.24) gives the complete state-space model for the EKF as follows:

$$\begin{pmatrix} x_{k+1}^l \\ q_{k+1} \\ b_{k+1}^\omega \end{pmatrix} = F_k \begin{pmatrix} x_k^l \\ q_k \\ b_k^\omega \end{pmatrix} + \begin{pmatrix} B^l & 0 & 0 \\ 0 & -\frac{T_s}{2}\Xi(q_k) & 0 \\ 0 & 0 & I_3 \end{pmatrix} \begin{pmatrix} n_k^l \\ n_k^\omega \\ n_k^{b^\omega} \end{pmatrix}, \tag{8.27}$$

$$\begin{aligned} \begin{pmatrix} y_k^p \\ y_k^v \\ y_k^a \\ y_k^q \end{pmatrix} &= \begin{pmatrix} 0 \\ 0 \\ R(q_k)\left(a_k + g^W\right) \\ 0 \end{pmatrix} \\ &+ \begin{pmatrix} I_3 & 0 & 0 & 0 \\ 0 & I_3 & 0 & 0 \\ 0 & 0 & I_3 & 0 \\ 0 & 0 & 0 & I_4 \end{pmatrix} \begin{pmatrix} p_k \\ v_k \\ b_k^a \\ q_k \end{pmatrix} + \begin{pmatrix} e_k^p \\ e_k^v \\ e_k^a \\ e_k^q \end{pmatrix}. \end{aligned} \tag{8.28}$$

RBPF The RBPF algorithm estimates the states in the vector

$$\begin{pmatrix} p_k & v_k & a_k & b_k^a & q_k \end{pmatrix}. \tag{8.29}$$

Introduce the matrix

$$C(q_k) = \begin{pmatrix} I_3 & 0 & 0 & 0 \\ 0 & I_3 & 0 & 0 \\ 0 & 0 & R(q_k) & I_3 \end{pmatrix}, \tag{8.30}$$

and the function

$$h(q_k) = \begin{pmatrix} 0 \\ 0 \\ R(q_k) g^W \end{pmatrix}. \tag{8.31}$$

The relations (8.11), (8.16), and (8.17) define the dynamic model, and the relations (8.23) and (8.24) describe the measurement equations for the RBPF. This gives the state-space model for the RBPF algorithm as follows:

$$x_{k+1}^l = F^l x_k^l + B^l n_k^l, \tag{8.32}$$

$$q_{k+1} = f(\tilde{\omega}_k - n_k^\omega) q_k, \tag{8.33}$$

$$y_k^q = q_k + e_k^q, \tag{8.34}$$

$$y_k = \begin{pmatrix} y_k^p \\ y_k^v \\ y_k^a \end{pmatrix} = h(q_k) + C(q_k) x_k^l + \begin{pmatrix} e_k^p \\ e_k^v \\ e_k^a \end{pmatrix}, \tag{8.35}$$

where

$$\begin{pmatrix} e_k^p \\ e_k^v \\ e_k^a \end{pmatrix} \in \mathcal{N}(0, R^l), \qquad R^l = \begin{pmatrix} R^p & 0 & 0 \\ 0 & R^v & 0 \\ 0 & 0 & R^a \end{pmatrix}. \tag{8.36}$$

State-Estimation Algorithms

The state-space model (8.27)–(8.28) is used directly in a standard EKF algorithm. The recursive relations defining the time- and measurement updates in the EKF algorithm are detailed in, *e.g.*, [Gustafsson, 2010; Särkkä, 2013].

The RBPF for (8.32)–(8.35) is derived from the general marginalized particle filter algorithm for mixed linear/nonlinear systems proposed in [Schön et al., 2005]. The state vector (8.29) is split into a linear part x_k^l, and a nonlinear part q_k. The model of the quaternion dynamics is independent of the linear states. The RBPF employs this property to estimate the posterior densities of the quaternion $p(q_k | \{y_m, y_m^q\}_{m=1}^k)$, $k \in \{1, \ldots, T\}$, using a particle filter. The densities are approximated by the empirical density of a set of N weighted particles, $\{w_k^i, q_k^i\}_{i=1}^N, k \in \{1, \ldots, T\}$, where w^i are the particle weights. The quaternion particles are propagated in time using the transition density $p(q_{k+1}|q_k)$, which is implicitly defined by (8.33). A point estimate of the quaternion at time k is computed as the principal eigenvector

of the matrix

$$\sum_{i=1}^{N} w_k^i \left(4q_k^i \left(q_k^i\right)^{\mathrm{T}} - I_4\right), \; k = 1, \ldots, T,$$

as suggested in [Morawiec, 1998] and employed in [Vernaza and Lee, 2006]. The linear states are estimated conditional on the quaternion particles using a conventional Kalman filter [Schön et al., 2005]. This means that all particles have their own set of estimates for the linear states. Introduce the notation $\mathrm{St}(x; \mu, \Sigma, \nu)$ for the density of a location-scale multivariate Student's t-distributed variable x with mean μ, covariance Σ, and ν degrees of freedom, and the notation $\mathcal{N}(x; \mu, \Sigma)$ for the density of a variable x with a Gaussian distribution having mean μ and covariance Σ. The full RBPF algorithm for (8.32)–(8.35), derived from [Schön et al., 2005], is listed in Algorithm 8.1.

8.4 Experimental Results

In this section, experimental results are provided both for the calibration algorithm and the state estimation using sensor fusion based on the EKF and the RBPF algorithms outlined in Section 8.3.

Algorithm 8.1 Rao-Blackwellized Particle Filter for (8.32)–(8.35).

1: **Input**: Measurement data $\{y_k, y_k^q\}_{k=1}^T$ and input data $\{\tilde{\omega}_k\}_{k=1}^T$.
2: **Initialize**: Set $\{w_0^i\}_{i=1}^N = 1/N$, sample $\{q_1^i\}_{i=1}^N \sim p(q_1^i)$ and set $\{\hat{x}_{1|0}^{i,l}, P_{1|0}^{i,l}\}_{i=1}^N = \{\bar{x}_1^l, P_1^l\}$.
3: **for** $k = 1, \ldots, T$ **do**
4: **for** $i = 1, \ldots, N$ **do**
5: Evaluate weights $\left(h_k^i = h(q_k^i),\, C_k^i = C(q_k^i)\right)$:

$$\begin{aligned} w_k^i = w_{k-1}^i &\mathrm{St}(y_k^q; q_k^i, R^q, \nu^q) \\ &\cdot \mathcal{N}(y_k; h_k^i + C_k^i \hat{x}_{k|k-1}^{i,l}, C_k^i P_{k|k-1}^{i,l} (C_k^i)^{\mathrm{T}} + R^l). \end{aligned} \tag{8.37}$$

6: **end for**
7: Normalize the weights to sum to one.
8: If needed, resample with replacement:
9: **for** $i = 1, \ldots, N$ **do**
10: Choose

$$q_k^i : \Pr\left(q_k^i = q_k^j\right) = w_k^j,\; j \in \{1, \ldots, N\}.$$

11: **end for**

12: Kalman Filter measurement update:
13: **for** $i = 1, \ldots, N$ **do**

$$S_k^{i,l} = C(q_k^i) P_{k|k-1}^{i,l} C(q_k^i)^{\mathrm{T}} + R^l, \tag{8.38}$$

$$K_k^{i,l} = P_{k|k-1}^{i,l} C(q_k^i)^{\mathrm{T}} (S_k^{i,l})^{-1}, \tag{8.39}$$

$$\varepsilon_k = y_k - h(q_k^i) - C(q_k^i) \hat{x}_{k|k-1}^{i,l}, \tag{8.40}$$

$$\hat{x}_k^{i,l} = \hat{x}_{k|k-1}^{i,l} + K_k^{i,l} \varepsilon_k, \tag{8.41}$$

$$P_k^{i,l} = \left(I - K_k^{i,l} C(q_k^i) \right) P_{k|k-1}^{i,l}. \tag{8.42}$$

14: **end for**
15: Particle Filter prediction:
16: **for** $i = 1, \ldots, N$ **do**

$$q_{k+1}^i \sim p(q_{k+1} | q_k^i, \tilde{\omega}_k). \tag{8.43}$$

17: **end for**
18: Kalman Filter time update:
19: **for** $i = 1, \ldots, N$ **do**

$$\hat{x}_{k+1|k}^{i,l} = F^l \hat{x}_k^{i,l}, \tag{8.44}$$

$$P_{k+1|k}^{i,l} = F^l P_k^{i,l} (F^l)^{\mathrm{T}} + G^l Q^l (G^l)^{\mathrm{T}}. \tag{8.45}$$

20: **end for**
21: **end for**

Experimental Setup and Implementation

The sensor calibration and state estimation were experimentally evaluated on an industrial robot manipulator, which is depicted in Figure 8.1. The robot is an ABB IRB140 [ABB Robotics, 2014], which was controlled by an IRC5 control system. The robot was equipped with the research interface ExtCtrl discussed in Chapter 2, which permits external access to the robot controller at a sample period of 4 ms. More specifically, the research interface enables low-level access to the joint position and velocity controllers at this sample period, while the measured joint positions and velocities and the corresponding reference joint torques were sent back to the external controller from the main control cabinet. It is to be noted, though, that with this interface enabled, the feedforward of the standard ABB motion control is not active. Thus, the robot motion is slightly less accurate in the experiments presented here than what would be the case with the complete motion control.

For access of external sensor data in the extended controller, the com-

munication was handled using the LabComm protocol [Dept. of Computer Science, Lund University, 2015]. The external control architecture also incorporates a server, to which clients sending sensor data using the LabComm protocol could establish a network connection. No buffering of sensor data in the communication was made. Instead, motivated by a control and sensor-fusion perspective, only the latest written data point was available in the robot controller at each sample.

Two external sensors were employed during the experiments; a tri-axial Xsens MTi-100 IMU [Xsens Inc. 2014] providing data at a maximum sample rate of 2 kHz, and a high-accuracy optical tracking system from Nikon Metrology of model K600 [Nikon Metrology, 2010] measuring the robot position with an absolute accuracy below 100 μm at a sample rate of 500 Hz with the current measurement configuration. The latter was only used for evaluation of the accuracy of the workspace position estimates. Considering that all external sensors sent data at a higher rate than the sampling frequency of the robot controller, synchronization of the sensor data and the internal robot controller data within 4 ms was achieved using the external research interface. This interface also permitted logging of all data signals (including sensor signals) with the sampling period of the robot controller.

The state-estimation algorithms were implemented in MATLAB for offline evaluation purposes. The computation times vary with the parameter choices. For the data and parameter sets considered in this research, each sample update in the EKF takes on average approximately 1 ms in MATLAB with an implementation where no measures to code optimization have been taken. A corresponding C-code implementation would thus clearly satisfy real-time constraints for online processing. The computations required for the RBPF were more extensive than those for the EKF. Therefore, the MATLAB Coder toolbox was used for transforming the initial implementation to C-code. The execution time for each sample update obviously depends on the number of particles in the RBPF. For the experiments performed, 50–300 particles were sufficient, resulting in average computation times of 0.7–3.9 ms for each sample update on a standard PC with an Intel i7 quad-core processor. Thus, executions of the algorithms as an external computational service to the robot controller clearly satisfy the real-time constraints for online application of RBPF-based state estimation at the sample rates required for modern robot controllers. Still, execution of the algorithms within the robot controller itself is challenging because of the limited available processor power and memory.

Calibration Results

To evaluate and quantify the performance of the calibration method defined in Section 8.3, data from several experiments were acquired on the robot setup.

Experimental Protocol For evaluation purposes, two different paths for the robot end-effector to be moved along were chosen for excitation. They are denoted as Path 1 and 2. These paths were located in different parts of the robot workspace in order to verify the consistency of the calibration method. The trajectory along each path was planned using three different linear and rotational velocities (denoted Velocity 1–3, with increasing velocity). The linear velocities of the robot end-effector were chosen equally spaced in the interval [0.2, 0.6] m/s. Thus, six different excitation trajectories were evaluated in total. The motion paths were chosen such that excitation (both linear and rotational) along all Cartesian directions of the robot was achieved. The data-acquisition experiments were executed repeatedly. Given the repeatability of the robot motion, each trial provided calibration results very close to each other and thus one data set for each combination of method and trajectory was used. The four different methods discussed in Section 8.3 for computing the estimates of the desired calibration parameters were applied and the results are subsequently compared. These are summarized as:

A. Levenberg-Marquardt algorithm for computation of both R^{BF} and r^F_{FB} simultaneously from (8.2)–(8.3), and no bias terms (Algorithm A);

B. Two-step procedure with an ordinary linear least-squares solution for r^F_{FB} and estimation of a constant bias-term for each of the accelerometer axes (Algorithm B);

C. Two-step procedure with an ordinary linear least-squares solution for r^F_{FB} and no bias terms (Algorithm C);

D. Two-step procedure with a total least-squares solution for r^F_{FB} and no bias terms (Algorithm D).

Results The resulting estimates of the rotation matrix R^{BF} (presented as corresponding Euler angles) and the translation vector r^F_{FB} for the different excitation trajectories and algorithms are provided in Table 8.1. Because of the similarity of the rotation-matrix estimates for different excitation velocities, the estimates of R^{BF} are only provided for Velocity 2. Comparing the parameter estimates obtained for the different algorithms and excitation trajectories, it can be observed that the estimates $\hat{R}^{BF}$ of the rotation matrix are highly consistent, whereas minor variations are visible for $\hat{r}^F_{FB}$ (in particular with regard to the velocity of the segments of the excitation trajectory). The mean of the estimates $\hat{r}^F_{FB}$ is

$$\begin{pmatrix} -0.0544 & 0.0295 & 0.125 \end{pmatrix}^{\mathrm{T}} \text{ m},$$

and the standard deviation is

$$\begin{pmatrix} 2.2 & 0.79 & 2.5 \end{pmatrix}^{\mathrm{T}} \text{ mm}.$$

Thus, the results are clearly consistent, and the accuracy is acceptable given the uncertainty of the acquired sensor data. It should also be noted that for this particular IMU, the bias estimation performed with Algorithm B does not influence the resulting estimation accuracy of the calibration parameters to any large extent.

In order to verify the estimated rotation matrix $\hat{R}^{BF}$, it is first noted in the experimental setup in Figure 8.1 that the Z-axes of Ψ_B and Ψ_F are approximately parallel but pointing in opposite directions. Thus, it is expected that the first two Euler angles are close to $-\pi$ and 0, respectively. This can be verified to be the case in Table 8.1. The third Euler angle is evaluated by rotating the IMU with the robot according to the estimated angle, such that the gravity vector is aligned with the X- and Y-axis of Ψ_B, respectively. The absolute values of the measured accelerations along the other axes were less than 0.054 m/s^2 in both cases, indicating that the orientation calibration is accurate. To the purpose of establishing a reference value of the translation vector r^F_{FB}, the corresponding distances were determined manually by measuring from the robot-flange origin to the IMU origin using a Vernier caliper. For practical measurement reasons, only the distance along the plane defined by the X- and Y-axes and the distance along the Z-axis of Ψ_F could be measured. The result was that the third component of $\hat{r}^F_{FB}$ was 0.124 m, whereas the Euclidean norm of the first and second components was 0.066 m.

Sensor-Fusion Results

The EKF and RBPF state-estimation algorithms defined in Section 8.3 were experimentally evaluated using four different data sets, obtained from experiments enumerated as 1–4. The algorithms were evaluated with respect to accuracy in the state estimation (using the measurements from the optical tracking system as reference) as well as in a sensitivity analysis using the acquired data sets. Motivated by that the workspace position is the primary interest in general, the results and evaluations in this subsection focus on this part.

Experimental Protocol For each experiment, the robot end-effector was programmed to move along a path in Cartesian space (measured in the frame Ψ_W in Figure 8.1) with specified orientation at certain waypoints along the path. Different paths were used for Experiments 1–4, and for each experiment the path was traversed three times in total, with the velocity multiplied by two in each repetition starting from 0.25 m/s for the linear velocity. During Experiments 1–2, the specified end-effector orientation was constant along the path. In contrast, during Experiments 3-4 the specified orientation varied along the path. The same comment regarding the repeatability of the robot motion as for the calibration applies to these experiments.

Table 8.1 Experimental results from the calibration procedure using different algorithm variants and excitation trajectories, as defined in the text. The configuration (Conf.) of excitation trajectory and method is defined as (Path No. & Algorithm). The rotation matrix is presented as the corresponding Euler angles (XYZ convention).

Conf.	Euler ang. $\hat{R}^{BF}$ [rad], Vel. 2	$\hat{r}^F_{FB}$, Vel. 1 [m]
1 & A	$\begin{pmatrix}-3.13 & -0.0009 & -1.21\end{pmatrix}$	$\begin{pmatrix}-0.0499 & 0.0297 & 0.123\end{pmatrix}^{\mathrm{T}}$
1 & B	$\begin{pmatrix}-3.13 & -0.0002 & -1.21\end{pmatrix}$	$\begin{pmatrix}-0.0502 & 0.0298 & 0.123\end{pmatrix}^{\mathrm{T}}$
1 & C	Same as 1 & B	$\begin{pmatrix}-0.0500 & 0.0298 & 0.123\end{pmatrix}^{\mathrm{T}}$
1 & D	Same as 1 & B	$\begin{pmatrix}-0.0503 & 0.0300 & 0.124\end{pmatrix}^{\mathrm{T}}$
2 & A	$\begin{pmatrix}-3.13 & 0.002 & -1.20\end{pmatrix}$	$\begin{pmatrix}-0.0552 & 0.0281 & 0.122\end{pmatrix}^{\mathrm{T}}$
2 & B	$\begin{pmatrix}-3.13 & 0.002 & -1.21\end{pmatrix}$	$\begin{pmatrix}-0.0555 & 0.0282 & 0.122\end{pmatrix}^{\mathrm{T}}$
2 & C	Same as 2 & B	$\begin{pmatrix}-0.0553 & 0.0285 & 0.122\end{pmatrix}^{\mathrm{T}}$
2 & D	Same as 2 & B	$\begin{pmatrix}-0.0555 & 0.0287 & 0.123\end{pmatrix}^{\mathrm{T}}$

Conf.	$\hat{r}^F_{FB}$, Vel. 2 [m]	$\hat{r}^F_{FB}$, Vel. 3 [m]
1 & A	$\begin{pmatrix}-0.0535 & 0.0294 & 0.127\end{pmatrix}^{\mathrm{T}}$	$\begin{pmatrix}-0.0570 & 0.0288 & 0.128\end{pmatrix}^{\mathrm{T}}$
1 & B	$\begin{pmatrix}-0.0536 & 0.0300 & 0.127\end{pmatrix}^{\mathrm{T}}$	$\begin{pmatrix}-0.0571 & 0.0295 & 0.128\end{pmatrix}^{\mathrm{T}}$
1 & C	$\begin{pmatrix}-0.0535 & 0.0300 & 0.127\end{pmatrix}^{\mathrm{T}}$	$\begin{pmatrix}-0.0571 & 0.0295 & 0.128\end{pmatrix}^{\mathrm{T}}$
1 & D	$\begin{pmatrix}-0.0537 & 0.0300 & 0.127\end{pmatrix}^{\mathrm{T}}$	$\begin{pmatrix}-0.0573 & 0.0296 & 0.129\end{pmatrix}^{\mathrm{T}}$
2 & A	$\begin{pmatrix}-0.0554 & 0.0306 & 0.127\end{pmatrix}^{\mathrm{T}}$	$\begin{pmatrix}-0.0548 & 0.0288 & 0.123\end{pmatrix}^{\mathrm{T}}$
2 & B	$\begin{pmatrix}-0.0554 & 0.0306 & 0.126\end{pmatrix}^{\mathrm{T}}$	$\begin{pmatrix}-0.0550 & 0.0289 & 0.122\end{pmatrix}^{\mathrm{T}}$
2 & C	$\begin{pmatrix}-0.0553 & 0.0308 & 0.127\end{pmatrix}^{\mathrm{T}}$	$\begin{pmatrix}-0.0549 & 0.0291 & 0.123\end{pmatrix}^{\mathrm{T}}$
2 & D	$\begin{pmatrix}-0.0555 & 0.0309 & 0.127\end{pmatrix}^{\mathrm{T}}$	$\begin{pmatrix}-0.0551 & 0.0292 & 0.123\end{pmatrix}^{\mathrm{T}}$

State-Estimation Results Here, the results from the data for Experiment 1 are presented in detail. The results for the other data sets were similar in terms of estimation performance, and are therefore not presented in detail. However, these results will be revisited when discussing the robustness of the considered algorithms. The robot path for Experiment 1 is presented in Figure 8.2, together with the actual tool position measured by the optical tracking system. The tool was in all experiments defined to be coinciding with the IMU origin, and all position and orientation estimates are provided with respect to the frame Ψ_W in Figure 8.1. Moreover, in all experiments a total number of 300 particles was used in the RBPF algorithm and the number of degrees-of-freedom in the multivariate Student's t-distribution was $\nu^\omega = \nu^q = 50$. The same parameters were chosen for the common parts of

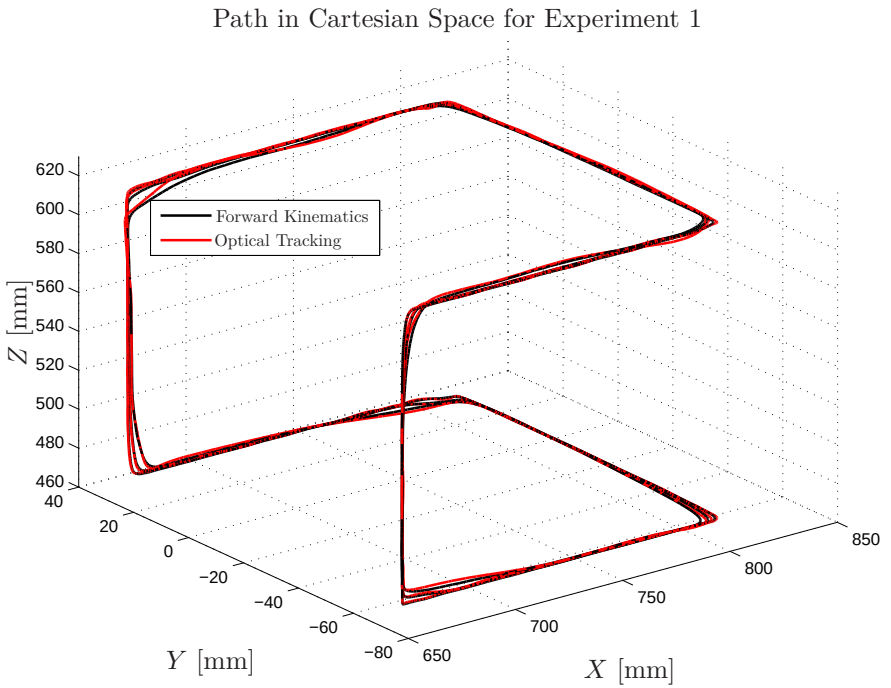

Figure 8.2 Position estimates from the forward kinematics and the actual traversed path in Experiment 1. The latter was measured by the optical tracking system.

the noise modeling in the EKF and the RBPF algorithms. Also, employing more particles in the RBPF was evaluated in simulation, but did not significantly improve the estimation, while increasing the average computation time. Given the stochastic nature of the RBPF, 100 Monte Carlo runs were performed for each data set in order to verify the consistency of the RBPF algorithm. The results are reported in this section as the average over these simulations.

In order to highlight the performance of the algorithms, the obtained results were analyzed both qualitatively and quantitatively. Three representative segments of the complete path and the corresponding estimates along the X-, Y-, and Z-axes are provided in Figure 8.3 for the RBPF. The results obtained with the EKF were similar from a qualitative perspective, and are therefore not presented in detail but instead further analyzed later in this section. The standard deviation for the position estimates over the Monte Carlo runs for the RBPF is below 70 µm for all Cartesian axes. Observing the results in Figure 8.3 closer it is seen that for the current experiment, the

Y- and Z-axes exhibit the most notable deviations of the forward-kinematics estimates from the reference measurements, and thus it is here that the IMU can contribute with the most additional relevant information to the forward kinematics $\xi(\theta)$. The position estimates obtained using the forward kinematics of the robot are also visualized in the figure for comparison. The estimates obtained with the sensor-fusion algorithm exhibit lower absolute error than the forward-kinematics estimates, in particular when considering the oscillations induced in the robot when moving at high velocities.

A comparison of the orientation estimates provided by the two algorithms for a segment of the data set acquired during Experiment 1, where the velocity of the robot is significant, is shown in Figure 8.4. The orientation is presented as the Euler angles corresponding to q^{BW}. The standard deviation for the quaternion estimates over the 100 Monte Carlo runs for the RBPF is below $4 \cdot 10^{-4}$ for all states. Thus, the standard deviations for both the position and the orientation indicate consistent state-estimation with the RBPF.

To quantify the performance of the algorithms, the estimation errors were computed for the estimated trajectories and compared to the error of the forward kinematics. The measurements from the optical tracking system were used as ground truth; thus the error is defined as the absolute values of the difference between the ground truth and the estimates. For illustration purposes, Figure 8.5 shows the absolute estimation error for the RBPF algorithm corresponding to the trajectories visualized in Figure 8.3. To quantify the performance of the estimation for the complete trajectory, *i.e.*, a movement with a duration of approximately 10 s, the root-mean square error (RMSE) along each Cartesian axis was computed. The results for the EKF and the RBPF, measured as the ratio between the RMSE for the state-estimation algorithm and the RMSE for the forward kinematics, are provided in Table 8.2. The performance increase, though, is significantly higher than the RMSE at the locations where the deviations are the largest and thus information from the IMU can contribute. As a result of the configuration of the robot in the experiment, it follows that the joint- and link flexibilities interact such that the largest deviations were to be expected along the Y- and Z-axes of the frame Ψ_W. Hence, the major improvements in the position estimation were observed along these axes. On the contrary, much less extra information to the forward-kinematics estimates can be extracted from the IMU along the X-axis.

Robustness Evaluation An important aspect when performing state estimation online is the sensitivity to the noise modeling and parameter choices in the algorithms. These decisions include choice of number of particles in the RBPF as well as the assumed noise covariance matrices for the process noise and the measurement noise. The latter can to some extent be extracted

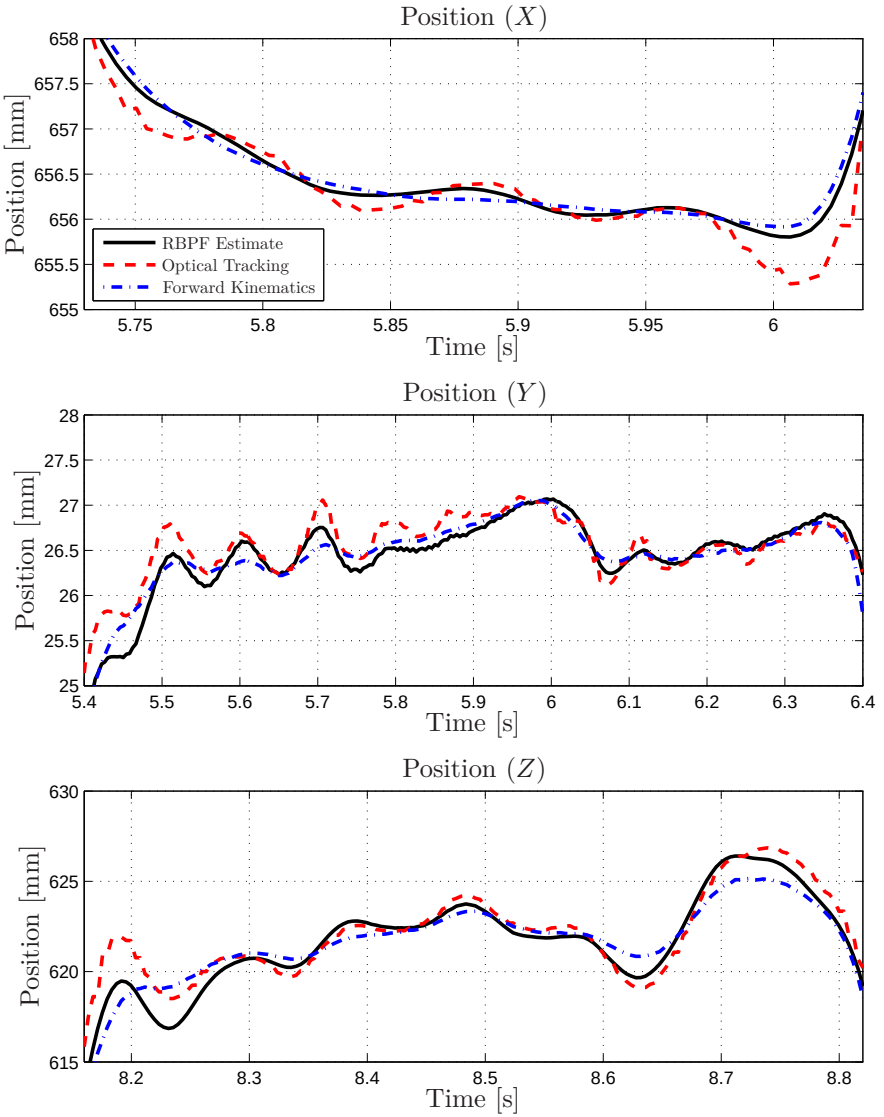

Figure 8.3 Comparison of the RBPF position estimates and the forward kinematics estimates of the robot for different representative segments of the path for Experiment 1. The actual position was measured with an optical tracking system. The RBPF algorithm gives improved performance compared to the forward kinematics. Note the different scales on the axes.

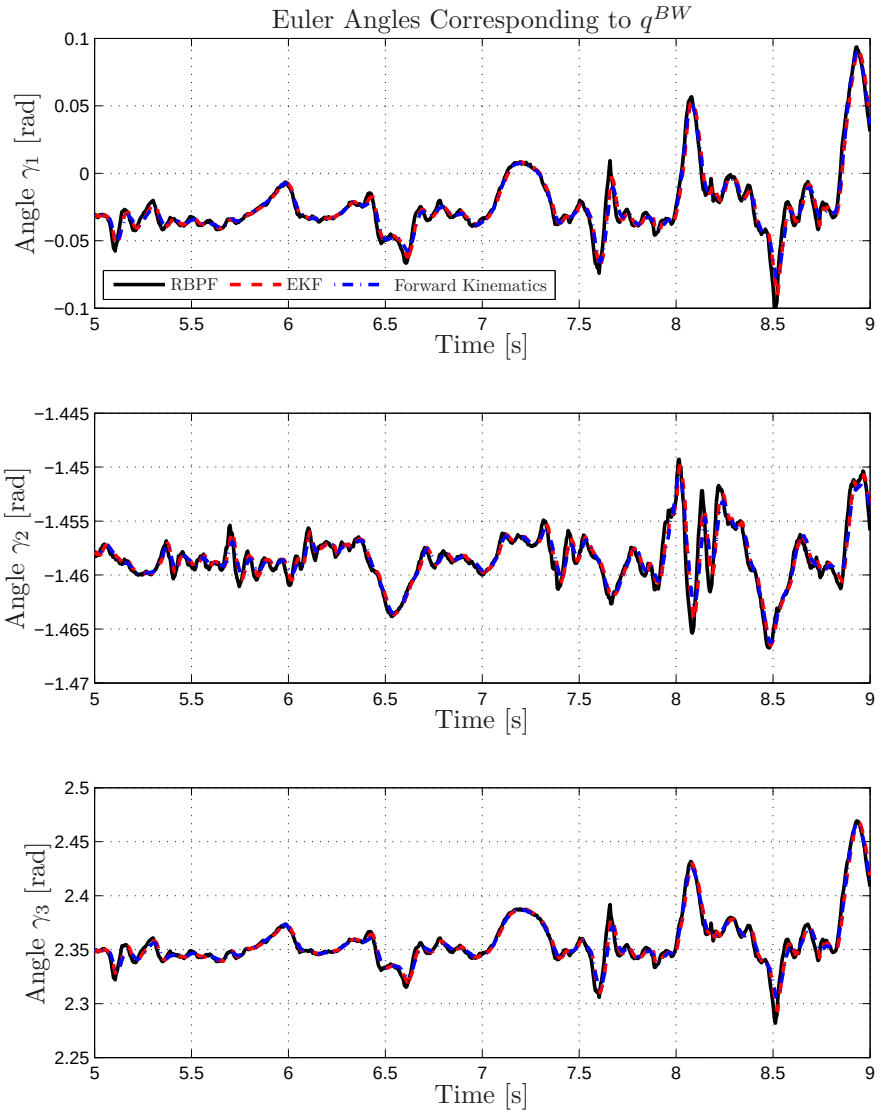

Figure 8.4 Estimated orientation of the frame Ψ_W with respect to the frame Ψ_B, parametrized in the Euler angles γ_1, γ_2, and γ_3, for a representative segment of the motion during Experiment 1.

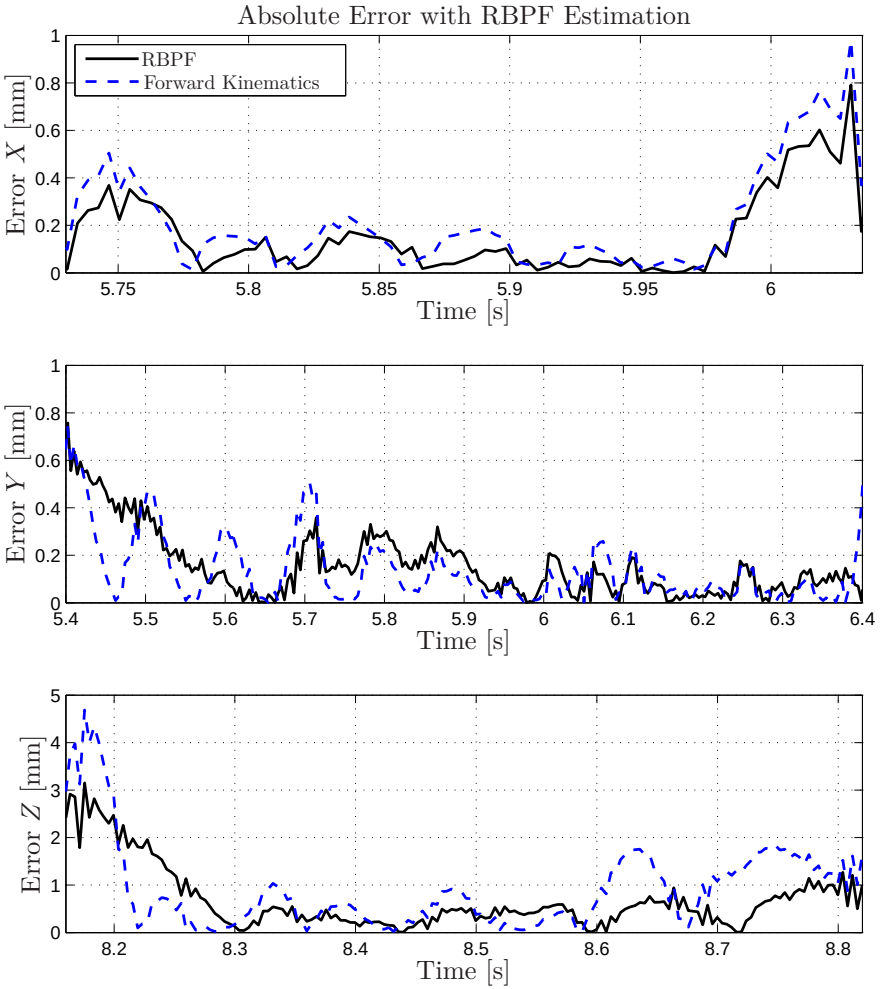

Figure 8.5 Absolute errors for the RBPF estimate and the forward kinematics for different representative segments of the data obtained in Experiment 1 (*cf.* the estimation results in Figure 8.3).

Table 8.2 Results for Experiment 1 using EKF and RBPF, measured as the ratio between the RMSE for the state-estimation algorithm and the RMSE for the forward-kinematics estimates.

Algorithm	X [%]	Y [%]	Z [%]
EKF	94.0	83.9	76.7
RBPF	94.1	83.7	77.0

Table 8.3 Robustness evaluation of the EKF and the RBPF algorithms, using the same assumptions for the noise parameters as in the evaluation of Experiment 1 in Table 8.2. The numbers are the ratio between the RMSE for the state-estimation algorithm and the RMSE for the forward-kinematics estimates.

Algorithm	Experiment	X [%]	Y [%]	Z [%]
EKF	Exp. 2	93.4	74.6	98.8
RBPF	Exp. 2	93.4	74.6	99.1
EKF	Exp. 3	98.4	91.7	81.7
RBPF	Exp. 3	98.4	91.6	81.8
EKF	Exp. 4	88.9	86.9	82.9
RBPF	Exp. 4	88.9	87.3	83.3

from measurements collected *a priori* during a test execution. In order to test the robustness of the state-estimation algorithms, the data sets from Experiments 2–4 were used with the same noise modeling and filter parameters as for Experiment 1. The results are collected in Table 8.3.

8.5 Discussion

This chapter of the thesis considered the problem of workspace state estimation for robot manipulators by sensor fusion of IMU data and the internal robot joint-motor position measurements. An inherent problem in such methods is the pose calibration of the sensors involved. A method for calibration of an IMU attached to the tool of a robot manipulator was considered, which was evaluated experimentally. The formulation of the calibration problem enabled the estimates to be computed without using nonlinear optimization (in contrast to, *e.g.*, [Birbach and Bauml, 2014]). Another advantage compared to, *e.g.*, the accelerometer calibration method in [Axelsson and Norrlöf, 2012], is that the method discussed does not require special motion patterns (only translational and orientational excitation along all Cartesian axes) and takes advantage of both accelerometer and gyroscope measurements for the calibration. Comparing the obtained calibration values from

the manual measurements of r_{FB}^{F} with the results obtained with the calibration algorithm in Table 8.1, it is clear that there is good correspondence within a few millimeters, which is in the range of the uncertainty involved in the manual measurements. The estimates of the rotation matrix exhibited very low variance between the different excitation trajectories, and the corresponding estimated Euler angles were in agreement with the physical setup and an experimental verification.

Relying on the calibrated IMU, two different state-estimation algorithms based on the EKF and the RBPF, respectively, were considered. The motion of the robot tool was modeled using a kinematic approach, thus minimizing the modeling effort required for application of the methods. In general, the forward and differential kinematic model information is available and can be employed. In contrast, identification of dynamic robot models requires more extensive experiments and subsequent processing. Kinematic models, similar to those in this chapter, have been considered for state estimation for robot manipulators previously in, *e.g.*, [Jeon et al., 2009; Norén, 2014], but only for state estimation with the EKF algorithm. Thus, a novelty in this chapter is the application of the RBPF to the case of pose estimation in workspace with kinematic models for robot manipulators. Kinematic models can serve as an evaluation of the performance that can be achieved with both the EKF and the PF when compared to approaches with more elaborate linear or nonlinear dynamic models as employed in, *e.g.*, [Axelsson et al., 2012; Henriksson et al., 2009; Chen and Tomizuka, 2014; De Luca et al., 2007]. The state-estimation accuracy and robustness are, of course, a trade-off between modeling complexity and applicability. Many of the previously suggested sensor-fusion algorithms in this area employ joint-space models, which increase the dimension of the state space. Since state estimation in workspace directly is considered in this chapter, the state-space dimension can be reduced if explicit joint-position estimates are not required. This enables estimation of the states for all six DoF of the robot tool in workspace.

Both algorithms for workspace state estimation in this chapter were shown to be possible to run in real-time on a standard PC. It was shown that the solution times were below 4 ms for both the EKF and the RBPF, while still providing higher estimation accuracy than the forward kinematics for all axes (see Tables 8.2–8.3). This is not surprising for the EKF, but the computation times reported previously for PF-based estimation in the robot-manipulator context (see, *e.g.*, [Axelsson et al., 2012; Rigatos, 2009]) are longer than the expected sampling period of a robot controller, thus not permitting online processing. The tools used for achieving real-time state estimation with the Particle Filter were twofold; first kinematic modeling was employed and second the Rao-Blackwellized PF was utilized. Using the RBPF can also be shown to result in lower variance of the estimate than the conventional Particle Filter, see [Lindsten et al., 2011]. Regarding the

estimation of the orientation, see Figure 8.4, it is clear that both algorithms performed similarly; this is expected because of the small orientation changes occurring during the motion in that experiment.

The performance of the state estimation is also a trade-off between model complexity and computational times. Still, the results from extensive experiments on a robot system showed a decrease of the estimation error for the 3D position, compared to the estimates provided by the forward kinematics of the robot. This shows that also kinematic models can be applied in robotic sensor-fusion applications with convincing results. Since nonlinear dynamic robot models are not commonly available, it is believed that the approach considered in this chapter extends the potential application areas for robot state estimation.

Since the state estimation is based on kinematic models and the forward-kinematics estimates are assumed to represent the arm-side position with additive noise (see Section 8.3), the robustness of the algorithms is an important aspect. It is standard procedure in PF-based methods to choose the covariance matrices slightly larger than what is justified by the measurement noise, in order to avoid particle depletion and account for possible model approximations, see, *e.g.*, [Gustafsson, 2010]. This strategy was used when parametrizing the noise modeling in this chapter. The performances of the EKF-based algorithm and the RBPF-based algorithm were investigated, and it was found that both exhibited similar sensitivity with respect to the noise modeling and filter parameters for different trajectories, see Table 8.3. Moreover, they both remained stable for different excitation trajectories.

Unlike the EKF, the RBPF is a theoretically more valid approach since it is consistent and unbiased. The RBPF algorithm also offers a more general framework for state estimation, because the measurement noise does not have to be Gaussian. This property was used to model the measurement noise from the IMU as multivariate Student's t-distributed, a distribution which is less sensitive to outliers than the Gaussian distribution. Note that the Gaussian distribution can be recovered by letting the parameter $\nu \to \infty$. Furthermore, since the distribution of the orientation states would deviate more from a Gaussian distribution in the cases of high speed and rapidly changing jerk of the motion, the RBPF should perform better than the EKF for extreme robot motions.

8.6 Conclusions

The problems of pose calibration of IMUs and workspace pose state estimation for robot manipulators have been approached. An algorithm for calibration of IMUs with respect to the robot flange was discussed. In addition, two different state-estimation algorithms (based on the EKF and the RBPF) that

can be executed in real-time for subsequent use in a robot controller were considered and evaluated in experiments. The experimental results from the calibration showed that the position and orientation of the sensor could be established with high accuracy (within a few millimeters for the estimated position with respect to the robot flange). The sensor-fusion algorithms provided results from experiments on a robot, with all six DoF moving, that increased the accuracy up to 25% compared to the robot forward kinematics (see Table 8.2). Considering the significant improvements without the need of accurate dynamic modeling, the method also offers robustness to uncertain and varying load conditions, which could be very relevant in, for instance, industrial settings. To summarize, this chapter investigated algorithms and presented experimental results for IMU calibration and state-estimation of the robot pose that are believed to contribute to the future applicability of online sensor fusion for robot manipulators in a broad sense.

9

Conclusions and Future Research

In this part of the thesis, methods for increasing the position accuracy of operations and task executions performed with industrial robots were presented and discussed. Both offline and online approaches were developed and experimentally evaluated. A common theme for the methods is model-based estimation and control combined with sensing and actuation in the robot workspace. For online control and actuation, a conventional industrial robot manipulator was combined with an external piezo-actuated compensation mechanism. This experimental setup enabled correction of errors in the relative position between the tool and the workpiece, beyond the bandwidth of the robot manipulator. A macro/micro-actuator feedback control architecture was developed, where an optical measurement system for online tracking of the robot and micro-manipulator end-effectors was used. In a subsequent experimental validation comprising milling in aluminium as well as in steel, the performance of the proposed approach was contrasted to machining results obtained using state-of-the-art methods. Moreover, dynamic modeling is an essential part of controller design. To this purpose, subspace identification methods were applied for gray-box modeling of the compliance dynamics of flexible mechanical systems, such as the micro-manipulator. The developed method was applied to the micro-manipulator, and the results provided new insights regarding both the fundamental frequency characteristics and the cross couplings (resulting from the mechanical design) between the actuation axes. As a further means to improve the position accuracy in the robot workspace, pose state estimation was considered using online sensor-fusion algorithms with IMU sensors on the robot tool. This proved to be successful in that the estimates provided by the algorithms exhibited lower error than the estimates obtained using only the forward kinematics of the robot with nominal model parameters. Such state-estimation methods are not only of interest in machining scenarios, but also in more general applications where

the eigenmodes of the robot are excited. Other important aspects here are that the methods do not require extensive robot modeling and can be applied online and thus employed for real-time compensation.

For the future, the online compensation strategies should be combined with offline compensation approaches. As a foundation for such developments, robot joint modeling and subsequent parameter identification were discussed. Models describing the joint characteristics in quasi-static operations were developed to the purpose of simulation and prediction of the joint motion in machining scenarios. Implementation of the resulting joint-based robot models (including the accompanying controllers) can be performed, *e.g.*, in the modeling language Modelica [Modelica Association, 2015]. A method for parameter identification, which was based on rigidly attaching the end-effector of the robot to a fixed base and actuate each joint motor using the control system, was developed and subsequently experimentally evaluated. Using the joint models and the measured parameters, prediction of the position deviations to be expected in the task execution would be possible already in the offline path-planning phase.

Another aspect of online task execution that has not been treated in this thesis is learning. Thus, future research include integrating the developed compliance models in, *e.g.*, the ILC methods for machining discussed in [Cano Marchal et al., 2014]. Also, adaptive contact-force control combined with learning [Sörnmo et al., 2012b; Sörnmo et al., 2015] could be combined with the models and sensor-fusion algorithms presented in Chapters 6–8 in this thesis for high-accuracy and high-performance machining with robot manipulators. As a summary, the results and methods presented in this part of the thesis are considered as steps toward the goal of having robot cells that are capable of machining with machine-tool accuracy by combining offline and online compensation strategies.

Part II

Optimal Motion Control

10

Introduction to Optimal Motion Control

This chapter provides an introduction to and motivates the interest in optimal motion control for robots and vehicles. Optimal motion control for robots is investigated in the context of path tracking, whereas optimal control of road vehicles, *e.g.*, is of interest for finding new safety systems and testing the limits of state-of-the-art dynamic modeling.

10.1 Optimal Motion Control for Robotic Systems

An integral part of the programming and task execution of robotic systems is the path and trajectory generation. A common task is to move the robot from point A to point B, without constraints on the path between the start and end points except to avoid known obstacles [Van Loock et al., 2013; LaValle, 2006]. However, in certain applications the path between the points is of explicit interest, and thus reliable path tracking is desired. Another scenario is that a global path planner provides the path, and a subsequent trajectory generation is to be made [Verscheure et al., 2009c]. Therefore, in intelligent and/or autonomous systems it is customary with a decoupled approach, *i.e.*, to segment motion control in the levels of path planning and path tracking [Kant and Zucker, 1986; LaValle, 2006] in a hierarchical structure to reduce the complexity of the complete motion-planning problem. To utilize the full capacity of the robot in a path-tracking application in industry, where the actuators are the limiting factors, a robust, near time-optimal path tracking considering the constraints on the actuators is desirable. Note that time-optimal does not *per se* imply high velocities, only that the maximum capacity of the actuators is used.

The task considered in Chapter 11 is for a controlled mechanical system to follow a predefined geometric path. A path is a curve in space, whereas for a trajectory the curve is time-parametrized, or alternatively, a corresponding

velocity profile is given. The fundamental difference between path tracking and trajectory tracking is consequently that the velocity along the path can be modified in the case of path tracking. Path tracking, or equivalently path following, is a fundamental control problem with many applications, and it is well-known for robot manipulators in tasks such as machining, welding, gluing, and cutting. Path tracking is also a major component in new developments in intelligent robotics, unmanned aerial vehicles (UAVs), and autonomous vehicles, and the current interest in robust algorithms for path tracking is considerable, see [Conte et al., 2004; Hoffmann et al., 2008; Hehn et al., 2012].

Trajectory Tracking versus Path Tracking

One way of approaching path tracking is to consider it as the task of tracking a sequence of trajectory points for a vector $x(t)$ of position coordinates in space, given as function of time t. Thus, path tracking is achieved by trajectory tracking. Trajectory tracking in this context means that the time frame, including the time when reaching the final state, is fully specified, and a common method is model-based feedforward control combined with online feedback using predefined trajectories.

In contrast, there are many examples where path tracking is possible but trajectory tracking is not, and a typical situation is when an actuator reaches its saturation limit. In all practical systems, there are always limitations on the available control authority, and then the only possibility may be to adjust the speed along the path, or phrased equivalently, to scale the time frame available for completion of the task. As a concrete example, when driving a car and the road exhibits high slip, the only way to stay on the road—*i.e.*, on the desired predefined path—may be to adjust the speed. Further, it is natural, as in the car-driving example, to think in terms of position along the path instead of time: The driver turns at a bend (path tracking) not after a certain time (trajectory tracking). In practice this means that the time frame is released, or equivalently phrased, that the speed along the path is adjusted. Nevertheless, there must still be coordination between the degrees of freedom (DoF) of the system to follow the desired path, and early research in this spirit is by means of dynamic scaling of the trajectories [Hollerbach, 1984].

10.2 Optimal Motion Control for Road Vehicles

Development of mathematical models and model-based control strategies for optimal road-vehicle maneuvers in time-critical situations have emerged as powerful tools during the past decade, see, *e.g.*, the survey in [Sharp and Peng, 2011]. Another recent survey on vehicular optimal control is presented

in [Limebeer and Rao, 2015], where the focus is on time and fuel optimization. An additional motivation for the interest in application of optimal control to vehicles is the possibilities to devise improved future safety systems for road vehicles and driver-assistance technologies. Advanced vehicle safety systems of today, such as the ESC systems, Anti-Lock Braking systems (ABS), and Active Slip Regulation (ASR) systems, see, *e.g.*, [Isermann, 2006], [Liebemann et al., 2005], and [Bauer et al., 2000], are still lagging behind the maneuvering performance achievable by professional race car drivers in critical situations, but the vision for improvement is there [Funke et al., 2012]. In particular, [Funke et al., 2012] presents control design with this perspective for autonomous vehicles driving close to the physical limits. Even though the solution of an optimal control problem depends on the particular choice of model and cost function, the fundamental behavior and control strategies found in the optimization can be used as inspiration for, or even integrated into, future safety systems. Hence, the research aim of the results presented in this thesis is the following:

- Develop a methodology for computing optimal trajectories for road vehicles in time-critical situations;
- Evaluate the strategy on different vehicle-maneuvering situations, so as to increase the understanding of vehicle dynamics in extreme situations;
- Investigate the influence of uncertain road conditions on the optimal maneuver;
- Analyze the obtained results to the purpose of design of new driver-assistance technologies with improved performance, compared to state-of-the-art systems of today.

The interaction between the vehicle and the road in an aggressive vehicle maneuver is complex, and consequently careful vehicle and tire modeling is required in order for the optimal control formulation to provide significant results. These aspects are extensively discussed in [Sharp and Peng, 2011]. In particular, if optimal control with time-optimality as criterion is considered, the control inputs and state variables are often at their limits and the vehicle chassis and tire models must provide reliable outputs even outside their normal range of operation. Another challenge with tire modeling is the experimental measurements of model parameters. An experimental evaluation presented in [Carlson and Gerdes, 2005] exhibits large variability of the measurement of the longitudinal tire stiffness—*i.e.*, the slope of the longitudinal force-slip curve. Further, when considering combined longitudinal and lateral slip, the tire-force modeling is even more demanding, since the tire model needs to capture the combinations of longitudinal and lateral slip that result in maxima and minima in the resulting tire force. In addition, it is of

interest to be able to model the tire forces on different surfaces, such as dry and wet asphalt, snow, and ice. To that purpose, scaling of nominal tire-model parameters has been proposed and experimentally measured, see [Pacejka, 2006; Braghin et al., 2006]. This strategy is investigated for computation of optimal maneuvers on different road surfaces in Chapter 13.

11

Control Architecture for Path Tracking with Robots

11.1 Introduction

In this chapter, a control architecture for high-accuracy and high-performance path tracking for robot manipulators is developed. The control structure is called path-tracking velocity control (PTVC) and builds on previous path-velocity control (PVC) algorithms [Dahl, 1992]. To utilize the freedom of velocity control along the desired path, it is clear that there are two different problems to be solved. One task is to control the traversal along the tangential direction of the path, where the objective typically is some optimal performance, *e.g.*, minimum-time or minimum-energy. The other task is to follow the path, *i.e.*, to coordinate the different DoF of the system such that the desired path is tracked. It is therefore natural to introduce a separation of the control of tangential motion along the path and motion toward the path (along the normal or binormal directions). This chapter of the thesis is derived, in parts, from the manuscript [Olofsson and Nielsen, 2015].

This chapter is organized as follows. A motivating example explaining fundamental concepts of the algorithm is presented in Section 11.2. The natural coordinates of a curve [Thorpe, 1979; Meriam and Kraige, 2012] is an appropriate framework for defining the tangential direction and the orthogonal directions of the path. This is described in Section 11.3, where also the definition of radius of curvature that is used in the control law is provided. The mathematical models of the mechanical systems under consideration are defined in Section 11.4. In this section, trajectory generation for such systems is also discussed. Then, based on natural coordinates, the control architecture PTVC with separate terms for tangential and orthogonal control is presented in Section 11.5. Subsequently, an analysis of the convergence properties is presented in Section 11.6. The algorithm is evaluated in Section 11.7 with

extensive simulations on different manipulators. The results obtained, possible extensions, and generalizations of the control architecture are discussed in Section 11.8, and finally conclusions are drawn in Section 11.9.

Previous Research on Path Tracking

Initial research on time-optimal trajectory generation for path tracking with robot manipulators was presented in [Bobrow et al., 1983; Bobrow et al., 1985; Pfeiffer and Johanni, 1987; Shin and McKay, 1985; Shiller, 1994b], and extensions with respect to dynamic uncertainties and singular control were proposed in [Shin and McKay, 1987; Shiller, 1994a]. Recently, a convex reformulation of the trajectory-generation problem for time-optimal path-tracking was suggested in [Verscheure et al., 2008; Verscheure et al., 2009c], together with efficient algorithms for computation of the optimal trajectories. Extensions with respect to convex-concave constraints were presented in [Debrouwere et al., 2013]. Methods for online trajectory generation for time-optimal path tracking were considered in [Verscheure et al., 2009a; Verscheure et al., 2009b]. Further application areas for the methods in [Verscheure et al., 2009c] were investigated in [Lipp and Boyd, 2014].

As stated in the previous subsection, path tracking can be achieved by trajectory tracking, conditioned that sufficient control authority is available. Within this class of approaches with a given time horizon, there are established algorithms to adjust the control inputs to obtain path tracking in the cases that there is some degree of repetitiveness of the path. As previously discussed in the thesis, ILC is one such successful strategy [Arimoto et al., 1984; Miyazaki et al., 1986; Norrlöf, 2000; Norrlöf, 2002]. The typical application scenario of ILC is offline in a batch-oriented structure. Often, ILC methods are limited to tasks where the trajectory is of fixed length, thus not permitting time scaling. Methods for relaxing this requirement were investigated in [Xu, 1998; Li et al., 2014]. ILC for optimal path tracking was considered in [Janssens et al., 2013]. Path tracking for mobile platforms was investigated in [Sarkar et al., 1993; Egerstedt et al., 2001]. Feedback linearization for trajectory planning was proposed in [Hauser and Banaszuk, 1997] and trajectory optimization in constrained environments was considered in [Bayer and Hauser, 2012]. Control laws for path tracking with mobile robots resulting in exponential convergence were proposed in [Sørdalen and Canudas de Wit, 1993]. The major difference between manipulators considered in this chapter and mobile robots is the non-holonomic constraints that might be necessary to consider for the latter category of systems.

The present research is a new approach to dynamic scaling of trajectories [Hollerbach, 1984], where the formulation is most related to the previous research by Dahl and Nielsen [Dahl and Nielsen, 1990b]. The algorithm proposed in [Dahl and Nielsen, 1990b] was formulated as online time scaling

of precomputed time-optimal trajectories, but could equivalently be phrased as online velocity control along the path and is here denoted path-velocity control (PVC) [Dahl, 1992; Dahl, 1994]. Extensions of the PVC algorithm in [Dahl and Nielsen, 1990b] and alternative approaches to high-accuracy path tracking have later been proposed in, *e.g.*, [Kieffer et al., 1997; Antonelli et al., 2003; Gerelli and Guarino Lo Bianco, 2008; Guarino Lo Bianco and Wahl, 2011; Guarino Lo Bianco and Gerelli, 2011; Antonelli et al., 2011; Guarino Lo Bianco and Ghilardelli, 2014], where the extensions mainly are with respect to the constraints on the system that can be accounted for and regarding developments for practical implementations of the algorithms in industrial systems. In particular, important extensions with respect to constraints in the robot workspace were proposed. Dynamic scaling of trajectories for robots with elastic joints was considered in [De Luca and Farina, 2002], and dynamic time scaling for generating energy-optimal trajectories for robots was proposed in [Wigström et al., 2013]. Another approach to path tracking for position servos with actuator limitations was considered in [Niu and Tomizuka, 2001]. Initial research on an alternative formulation of the path-tracking control problem with actuator constraints was presented in [Arai et al., 1994], where control along the orthogonal directions of the path was introduced and given priority over the control along the tangent of the path, thus not focusing on the coordination of the DoF. Another approach to orbital stabilization for nonholonomic underactuated systems based on transverse linearization, related to the control architecture in this chapter, was suggested in [Freidovich and Shiriaev, 2012]. Moreover, a predictive path parametrization for online path tracking was considered in [Bemporad et al., 1999], with a similar purpose as the PVC algorithm presented in [Dahl and Nielsen, 1990b].

Contributions

The main contribution of this chapter is a control architecture for robust path tracking for robot manipulators with actuator constraints. The path-tracking problem is approached using a separation between tangential and orthogonal control along the path. In the setting of the tangential direction $x_\perp$ and the orthogonal directions $x_\parallel$, the control along $x_\parallel$ utilizes PVC and builds on previous algorithms for time scaling [Dahl and Nielsen, 1990b]. In addition, a control law for achieving fast convergence along the orthogonal directions of the path, $x_\perp$, is developed. Combining these two algorithms results in the control architecture named PTVC, that employs coordinated feedback control both along $x_\perp$ and $x_\parallel$. This strategy, PTVC, achieves robust path tracking for a wide class of mechanical systems.

11.2 A Conceptual Example

The following example introduces some of the main approaches in this chapter for obtaining convergence of the path tracking along the transversal directions of the path in the case of model uncertainty and disturbances.

Tracking of a Circular Path

Consider the path-tracking task where the path to follow is a circle with constant radius R in a two-dimensional space. Let x_1 and x_2 be the positions along the respective coordinate axis. Different descriptions of the path will be valuable in the conceptual treatment of the task. The circle can be described by an equation

$$g(x_1, x_2) = x_1^2 + x_2^2 - R^2 = 0, \tag{11.1}$$

and it can be described in parametrized form as

$$x_1 = R\cos(\theta), \tag{11.2}$$
$$x_2 = R\sin(\theta), \tag{11.3}$$

where θ is the angle with respect to the positive x_1-axis. The latter relation is put in vector form by defining $f = \begin{pmatrix} x_1 & x_2 \end{pmatrix}^{\mathrm{T}}$. Note that f here denotes the coordinates in Cartesian space, while in Section 11.4 it instead denotes the path for the generalized coordinates. The path parameter s is introduced as the length along the path. Then, $\theta = s/R$ and

$$f(s) = \begin{pmatrix} R\cos(s/R) \\ R\sin(s/R) \end{pmatrix}. \tag{11.4}$$

The motion along the circle can also be expressed as a state-space dynamic system according to

$$\dot{x}_1(t) = \omega x_2(t), \tag{11.5}$$
$$\dot{x}_2(t) = -\omega x_1(t), \tag{11.6}$$

where ω is the angular velocity, defined by the relations $\omega = -\dot{\theta} = -\dot{s}/R$. Further, the variable r is introduced as the distance from the origin according to $r^2 = x_1^2 + x_2^2$. The path-tracking task is to keep $r(t) = R$. Considering (11.5)–(11.6), any model error or disturbance leading to $r(t) \neq R$, will result in the system continuing to evolve with a new faulty radius different from R.

To achieve path stability—*i.e.*, the property that $r(t)$ will converge to R after a disturbance—it is natural to introduce corrections orthogonal to the path. Here, this means in the direction of the gradient to (11.1), which is denoted ∇g. Justified by the analysis following in Section 11.6, one choice is

to introduce a path-stability term according to

$$\frac{1}{2}(R^2 - r^2)\nabla g = (R^2 - r^2)\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} x_1(R^2 - r^2) \\ x_2(R^2 - r^2) \end{pmatrix}. \tag{11.7}$$

Introducing the compensation term (11.7) in (11.5)–(11.6) leads to the resulting dynamics given by

$$\dot{x}_1(t) = \omega x_2(t) + x_1(t)\left(R^2 - r(t)^2\right), \tag{11.8}$$

$$\dot{x}_2(t) = -\omega x_1(t) + x_2(t)\left(R^2 - r(t)^2\right). \tag{11.9}$$

Eliminating x_1 and x_2 from the relations (11.8)–(11.9) is performed by multiplying the first and second equation with x_1 and x_2, respectively, and then adding and subtracting equations pairwise. Using that $r\dot{r} = x_1\dot{x}_1 + x_2\dot{x}_2$ for the first case and substituting $x_1 = r\cos(\theta)$, $x_2 = r\sin(\theta)$ for the second case, this procedure leads to the following decoupled set of differential equations

$$\dot{r}(t) = r(t)(R^2 - r(t)^2), \tag{11.10}$$

$$\dot{\theta}(t) = -\omega. \tag{11.11}$$

The differential equations above are solved independently. Starting with (11.10), the principle of separation of variables is used to obtain the positive solution, which is given by

$$r(t) = R\sqrt{\frac{1}{1 + ke^{-2R^2t}}}, \tag{11.12}$$

where k is a constant determined by the initial value $r(0)$. Here, it holds that $R - r = O(e^{-2R^2t})$, so it is clear that $r(t)$ converges to R exponentially as $t \to \infty$. The second differential equation, (11.11), gives

$$\dot{\theta}(t) = \frac{\dot{s}(t)}{R}, \tag{11.13}$$

which is just a rephrasing of the definition. Consequently, (11.8)–(11.9) result in exponential convergence to the path, while behavior along the path is still free to be controlled by a separate control law. This tangential behavior is described by the path parameter s and it can thus be used for tangential path-velocity control.

Outline of Algorithm Concepts

The key step in this chapter is to use the separation principle, indicated in the previous example, between tangential control and exponential path convergence along the directions normal to the path. For a given path, there

needs to be a sequence of local coordinate systems, defining the local variables corresponding to x_1, x_2, and the radius R. A natural choice here is to use the well established mathematical concept of natural coordinates for a path [Thorpe, 1979; Meriam and Kraige, 2012], as will be described in Section 11.3. Conceptually, the center of curvature will serve as the origin for the variables x and the radius of curvature will serve as R. For a general path, a well established concept in optimal trajectory planning [Bobrow et al., 1983; Bobrow et al., 1985; Pfeiffer and Johanni, 1987; Shin and McKay, 1985], and also in feedback architectures such as [Dahl and Nielsen, 1990b], is to reduce the multi-dimensional control problem to a one-dimensional problem by using the correspondence to (11.4)—*i.e.*, to parametrize the path f in a scalar path coordinate s. Conceptually, this means that the path is a constraint in x, and that s and its time-derivatives are used to compute the optimal velocity profile along the path. This parametrization is presented in Section 11.4, and is used for tangential control, which is equivalent to control along $x_\parallel$. The control architecture in Section 11.5 integrates the control along $x_\parallel$ from [Dahl and Nielsen, 1990b] with control along $x_\perp$, similarly to (11.7) in the example described earlier in this section.

11.3 Natural Path Coordinates

When studying motion along a path—*i.e.*, a curve in space—the concept of natural coordinates is often used. This formulation can be found in standard textbooks both in mechanics [Meriam and Kraige, 2012] and in mathematics [Thorpe, 1979]. The main results used later in this chapter are briefly recapitulated in this section, but the proofs are to a large extent omitted and the reader is referred to the mentioned references for details. The important role of tangential motion and the role of curvature along the normal direction should be noted.

Definition of Coordinate System

The natural coordinate system is usually derived by means of a trajectory, which from the geometrical point of view is a parametrized curve in a three-dimensional space. Thus, let the trajectory be given by the vector $\rho(t)$ according to

$$\rho(t) : [t_0, t_1] \subset \mathbb{R} \quad \text{and} \quad \rho \in \mathcal{C}^2[t_0, t_1], \tag{11.14}$$

which means that the trajectory is a parametric curve that is two times continuously differentiable on the parameter interval $[t_0, t_1]$. Moreover, the unit vector π is the tangent to the trajectory at each point and is defined using the velocity vector $v(t) = \dot{\rho}(t)$. The formal definition is:

Definition 1—Tangent Vector
Given a trajectory $\rho(t)$, the unit tangent vector to the trajectory is defined as

$$\pi = \frac{v}{||v||_2}. \tag{11.15}$$

□

This vector π is the first basis vector of the natural coordinate system. Since v is a function of the parameter t, which describes the trajectory in the geometric space, π itself is a function of the same parameter, so in general π is not constant along the trajectory. Let s be the length of the curve traversed in the interval $[t_0, t]$, *i.e.*,

$$s(t) = \int_{t_0}^{t} \sqrt{v(\zeta)^{\mathrm{T}} v(\zeta)}\, \mathrm{d}\zeta,\ t \leq t_1. \tag{11.16}$$

If $v \geq 0$, with equality only in isolated time instants, observe that the length is a non-negative, monotonically increasing, and real function of the parameter t and that

$$\dot{s} = \frac{\mathrm{d}s}{\mathrm{d}t} = \sqrt{v^{\mathrm{T}} v} = ||v||_2. \tag{11.17}$$

From this relation it follows that

$$\frac{\mathrm{d}\rho}{\mathrm{d}s} = \frac{\mathrm{d}\rho}{\mathrm{d}t}\frac{\mathrm{d}t}{\mathrm{d}s} = \frac{v}{||v||_2} = \pi. \tag{11.18}$$

This means that if the trajectory is parametrized in its length, then the velocity vector is the unit vector π, which is the tangent to the curve. The length of the curve in space is usually referred to as the natural parameter [Meriam and Kraige, 2012].

The second unit vector of the natural coordinate system is a consequence of an important property of π, which is given by the following theorem.

Theorem 1—Orthogonality
For the tangent vector π and its derivative with respect to t, it holds that

$$\pi \perp \dot{\pi}. \tag{11.19}$$

□

For the proof of this relation, see [Meriam and Kraige, 2012; Thorpe, 1979]. From this theorem, the following definition is natural:

Definition 2—Normal Vector
Given a trajectory ρ, the unit vector normal to the trajectory is defined as

$$n = \frac{\dot{\pi}}{||\dot{\pi}||_2}. \tag{11.20}$$

□

The plane spanned by (π, n) is called the osculating plane [Meriam and Kraige, 2012], which will be used later when deriving the controller in Section 11.5. Introduce the curvature κ by $\kappa = ||\dot{\pi}||_2$. Then, it holds that $\kappa \geq 0$, and it follows from (11.20) that

$$\dot{\pi} = \kappa n. \tag{11.21}$$

When a trajectory ρ is given and the curve is parametrized in the length s, then it follows from (11.17) that $||v||_2 = 1$ and that $\dot{v} = \kappa n$, as a consequence of (11.21). The physical dimension of κ is the inverse of a length and it is the curvature of the trajectory along the curve. This leads to the following central definition:

Definition 3—Radius of Curvature
The radius of curvature R is defined as

$$R = \frac{1}{\kappa}, \quad \text{if} \quad \kappa > 0. \tag{11.22}$$

□

Since $\kappa \geq 0$, the radius of curvature is non-negative. To complete the natural coordinate system with its third basis vector, the following definition is adopted:

Definition 4—Binormal Vector
For a given trajectory ρ, where π and n have been defined, the binormal unit vector is defined as the vector product

$$b = \pi \times n. \tag{11.23}$$

□

The unitary of the vector b is a direct consequence of the fact that π and n are perpendicular unit vectors. The complete formulation of the natural coordinate system is depicted in Figure 11.1.

Definition of $x_{\parallel}$ and $x_{\perp}$

The framework of natural coordinates, with the moving coordinate system (π, n, b), is not dependent on the original parametrization. Consequently, the parametrization is an intrinsic property of a curve in space, or equivalently an intrinsic property of a path. Relating to Section 11.2, the direction $x_{\parallel}$ is along the basis vector π and the directions $x_{\perp}$ are the orthogonal subspace, here spanned by the normal and binormal basis vectors (n, b). In a typical path-tracking task for a mechanical system, the centripetal forces will be along the direction of the normal vector n. As a consequence, control along the normal direction n is in general more crucial than the corresponding control along the binormal direction b. The radius of curvature $R(s)$, obtained as part of

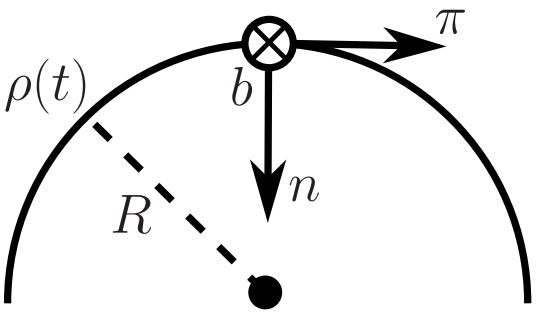

Figure 11.1 Axes of the natural coordinate system for an example curve ρ with constant radius R.

the procedure of describing the path in natural coordinates and in general varying along the path, is used as the local path descriptor as indicated in the example in Section 11.2.

11.4 Modeling Assumptions and Trajectory Optimization

This section defines the class of mechanical systems considered in this chapter. Moreover, the parametrization of the geometric path and nominal trajectory optimization, or more generally trajectory planning, are discussed.

Dynamics

The dynamics of the considered class of mechanical systems (having n_d DoF) is assumed to be possible to write on the following format

$$\tau = M(q)\ddot{q} + C(\dot{q}, q)\dot{q} + D(q)\dot{q} + G(q), \tag{11.24}$$

where $\tau \in \mathbb{R}^{n_d}$ are the input torques, $q \in \mathbb{R}^{n_d}$ are the generalized coordinates, $M(q)$ is the inertia matrix, $C(\dot{q}, q)$ is the Coriolis/Centripetal matrix, $D(q)$ is the viscous friction matrix, and $G(q)$ is the gravity vector [Siciliano et al., 2009; Spong et al., 2006]. The format of the dynamic equations in (11.24) is applicable for a large number of systems; important examples include rigid-body robot manipulators (with a fixed rigid base) and acceleration or torque-controlled position servos. It is assumed that the control objective is the position $x \in \mathbb{R}^3$ of the mechanical system in a Cartesian space.

Remark 1
The theory presented in this section can be extended to include flexible-joint models with nonactuated DoF, see [Dahl, 1992]. In practice, this means that the model will require an increased number of states, and that the system dynamics needs to be represented using time-derivatives of the path

parameter of higher order than two. The focus of the presentation in this chapter is on the rigid-body case. □

Path Parametrization

A path $f \in \mathbb{R}^{n_d}$ is assumed to be available, defining the desired motion of the n_d generalized coordinates q of the system. Note that this is the path for the generalized coordinates of the system, in contrast to the path in Cartesian space discussed in Section 11.2. As in the definition of a curve in Section 11.3, the path is assumed to be twice continuously differentiable. In general, the path is determined by a high-level path planner in Cartesian space and subsequently described in natural coordinates as outlined in Section 11.3. Hence, the inverse-kinematics relations are necessary for transformation to corresponding generalized coordinates. The end-time t_f of the time frame is in many cases unknown *a priori* and should thus be computed as part of the trajectory-planning procedure. In order to establish a time-invariant description of the path, the path coordinate $s \in [s_0, s_f]$ from Section 11.3 is adopted, see, *e.g.*, [Dahl, 1992]. This is formalized as

$$f(s), \;\; s \in [s_0, s_f]. \tag{11.25}$$

The interval for the path coordinate can be scaled to a nominal interval between zero and one, but the default choice here is the Euclidean length of the path. The corresponding time-derivatives of the path coordinate, $\dot{s}$ and $\ddot{s}$, are referred to as the path velocity and the path acceleration, respectively.[1]

Nominal Trajectory Optimization

In the control architecture developed in Section 11.5, it is assumed that a nominal trajectory is available. For task effectiveness, this nominal trajectory is natural to compute using a time-optimal criterion, with the maximum available inputs to the actuators as constraints. However, also other approaches to determine the nominal trajectory are possible to use together with the considered control architecture, see further discussion in Section 11.8.

It is well-known in the literature [Bobrow et al., 1983; Bobrow et al., 1985; Pfeiffer and Johanni, 1987; Shin and McKay, 1985] that the time-optimal path-tracking problem for systems of the format (11.24) can be efficiently solved by transforming the original problem in the time-domain with an open interval $[0, t_f]$ to a corresponding problem over the fixed horizon $[s_0, s_f]$ with a significantly reduced number of states. If the optimization criterion is chosen as the time for traversing the path, the cost function can be reformulated

[1] In the case of dynamic models with an increased number of states (such as the flexible-joint model discussed in Remark 1), also time-derivatives of higher order of the path coordinate are required.

as a function of the path velocity instead [Dahl, 1993] as follows

$$t_f = \int_0^{t_f} \mathrm{d}t = \int_{s_0}^{s_f} \frac{\mathrm{d}t}{\mathrm{d}s}\mathrm{d}s = \int_{s_0}^{s_f} \frac{1}{\dot{s}}\mathrm{d}s. \tag{11.26}$$

Employing the requirement that $q = f(s)$ along the desired path, it follows by using the chain rule that

$$\dot{q} = f'(s)\dot{s}, \tag{11.27}$$

$$\ddot{q} = f''(s)\dot{s}^2 + f'(s)\ddot{s}, \tag{11.28}$$

where $(\cdot)'$ is the derivative with respect to s. Further, it is straightforward to reformulate the system dynamics (11.24) as a function of the path coordinate and its time-derivatives using (11.27)–(11.28), see, *e.g.*, [Shin and McKay, 1985; Bobrow et al., 1985; Dahl, 1993], according to

$$\tau = \Gamma_1(s)\ddot{s} + \Gamma_2(\dot{s}, s), \tag{11.29}$$

where

$$\Gamma_1(s) = M(f(s))f'(s), \tag{11.30}$$

$$\begin{aligned}\Gamma_2(\dot{s}, s) = [M(f(s))f''(s) + C(f(s), f'(s))f'(s)]\dot{s}^2 + \\ D(f(s))f'(s)\dot{s} + G(f(s)).\end{aligned} \tag{11.31}$$

Introducing the state variable as $\beta(s) = \dot{s}^2$ and the input $\alpha(s) = \ddot{s}$, see [Dahl, 1992; Verscheure et al., 2009c], the optimization problem for time-optimal path-tracking is

$$\begin{aligned}\underset{\alpha(s),\beta(s)}{\text{minimize}} \quad & \int_{s_0}^{s_f} \frac{1}{\sqrt{\beta(s)}}\mathrm{d}s \\ \text{subject to} \quad & \tau(s) = \Gamma_1(s)\alpha(s) + \Gamma_2(\sqrt{\beta(s)}, s),\ \beta(s_0) = \beta(s_f) = 0, \\ & \beta'(s) = 2\alpha(s),\ \beta(s) \geq 0,\ \tau_{\min} \leq \tau(s) \leq \tau_{\max}.\end{aligned} \tag{11.32}$$

Finding a solution of the optimal control problem corresponding to (11.32) was originally performed by constructing the solution in the phase plane $(s, \dot{s})$ by utilizing the constraints on the path acceleration, see [Bobrow et al., 1985; Shin and McKay, 1985]. For numerical efficiency, the solution to (11.32) can directly be computed using parametric optimization methods such as collocation [Åkesson, 2007; Olofsson et al., 2011b]—*i.e.*, discretization of the complete optimization problem and subsequent solution using general-purpose solvers for large nonlinear programs (NLPs). Given certain constraints on the dynamic model (11.24), the problem is convex and can thus be solved efficiently with guaranteed convergence to a global minimum

[Boyd and Vandenberghe, 2004]. More specifically, it is shown in [Verscheure et al., 2008; Verscheure et al., 2009c; Debrouwere et al., 2013] that the optimization problem (11.32) is convex when $D(q) = 0$, because in this case the system dynamics (11.29) is affine in the state β and the input variable α, as can be seen from (11.30)–(11.31). Irrespective of the solution method, the result of the optimization procedure is a discretized time-optimal trajectory for tracking of the specified path f. The relation between the time t and the path coordinate s is given by

$$t(s) = \int_{s_0}^{s} \frac{1}{\sqrt{\beta(\zeta)}} \, \mathrm{d}\zeta, \;\; s_0 \leq s \leq s_f, \tag{11.33}$$

from which it is clear that there is a bijective relation between s and t.

Remark 2
The relation (11.33) together with (11.27)–(11.28) can be used to transform the resulting optimal joint trajectories from functions of the path coordinate to functions of time. □

11.5 Control Architecture

To execute a high-performance trajectory, such as the time-optimal resulting from the procedure outlined in the previous section, a direct application of the nominal trajectories as described in Remark 2 does not work satisfactory, as has been observed in [Dahl, 1992; Dahl and Nielsen, 1990b; Kieffer et al., 1997; Antonelli et al., 2011] and also pointed out in the discussion in Section 10.1 in Chapter 10. A major reason is that time-optimal path tracking implies that at least one of the control inputs is at its limit at each time point [Chen and Desrochers, 1989], which means that there is no control authority left to keep the system on the desired path in case of modeling uncertainties or disturbances. Thus, there is a need for a feedback mechanism able to accommodate for this by adapting the velocity while maintaining high-accuracy path tracking. The following sections describe the components of the PTVC algorithm.

Basic Control

Besides a given path f, it is assumed that a controller for set-point control and trajectory tracking is available. The controller acts on reference values q_r, $\dot{q}_r$, and $\ddot{q}_r$ for the generalized coordinates and their time-derivatives, and there are several possibilities for this basic controller. The particular choice is not critical for the presentation, but for illustration a common choice for mechanical systems, namely the computed-torque controller [Spong et al.,

2006; Siciliano et al., 2009], is employed. This control law is stated as

$$\tau = \hat{M}(q)\left(\ddot{q}_r + K_v(\dot{q}_r - \dot{q}) + K_p(q_r - q)\right) + \hat{C}(\dot{q}, q)\dot{q} + \hat{D}(q)\dot{q} + \hat{G}(q), \quad (11.34)$$

where K_v and K_p are controller parameters. Note that the control is based on estimates $\hat{M}$, $\hat{C}$, $\hat{D}$, and $\hat{G}$ of the system parameters.

Tangential Control

Introduce the *path parameter* σ as the actual length traveled in the online path traversal [Dahl and Nielsen, 1990b], *i.e.*, analogously to (11.16) and the path coordinate in the parametrization of the path (11.25). When the robot is on the path, the control is performed along the tangent, and thus the reference values for the controller are computed based on the geometric path f and the current value of the path parameter and its time-derivatives. Hence, the reference values for motion along the path are computed as follows

$$q_{r,\parallel}(\sigma(t)) = f(\sigma(t)), \quad (11.35)$$

$$\dot{q}_{r,\parallel}(\sigma(t)) = f'(\sigma(t))\dot{\sigma}(t), \quad (11.36)$$

$$\ddot{q}_{r,\parallel}(\sigma(t)) = f''(\sigma(t))\dot{\sigma}(t)^2 + f'(\sigma(t))\ddot{\sigma}(t), \quad (11.37)$$

where it is emphasized that the reference values correspond to motion along the tangent.

Transversal Control

When deviations from the nominal path occur during the motion, corrective actions transversal to the path are required. Since the controller is based on velocity control, the transversal control modifies the velocity references for the generalized coordinates. To the purpose of path stability, a correction similar to the control law (11.7) discussed in the example in Section 11.2 is used. However, also the cases of straight paths or paths with low curvature and motion in three dimensions need to be handled. This is treated in the following subsections.

Nominal Transversal Control The control law for the transversal control relies on the parametrization of the path in its natural coordinates, see Section 11.3. This means that the moving trihedral (π, n, b) has been defined along the path, and the basis vectors are consequently considered as functions of the path parameter in the controller. In addition to the Cartesian position x of the system, introduce the instantaneous center of curvature x_o in the Cartesian space. Let r be the Euclidean length of the projection of $x - x_o$ onto the subspace defined by π and n (*i.e.*, the osculating plane defined in Section 11.3), where x_o by definition is located in this plane. Thus, the following expression holds for r

$$r = ||x^{\pi,n} - x_o||_2 , \quad (11.38)$$

where $x^{\pi,n}$ is the projection of x on the osculating plane. Further, define the vector $n_{\text{op}} = x^{\pi,n} - x_o$ in the osculating plane (*cf.* the vector used in the stabilization (11.7) in the example in Section 11.2). Using these variables, the following term for the velocity reference along the orthogonal directions is introduced when deviating from the nominal path

$$\dot{q}_{r,\perp}(\sigma(t)) = K_\perp J(q)^{-1} \left((R^2 - r^2) n_{\text{op}} - (x - x_o)^{\mathrm{T}} b \; b \right), \tag{11.39}$$

where $K_\perp$ is a matrix with parameters to be chosen and $J(q)$ is the Jacobian of the considered manipulator—*i.e.*, the matrix describing the differential kinematics of the robot. The Jacobian is necessary for transforming the corrections required in Cartesian space to the corresponding joint-velocity reference values. The control along the binormal direction is computed as the projection of $x - x_o$ on the vector b and contributes to the velocity vector along the direction defined by the binormal. Note that R and the vector b in (11.39) depend on the current value of the path parameter σ and they are thus updated continuously along the path.

Path Segments with Low Curvature If the path is straight or the curvature is close to zero for certain parts of the path, the radius R is large or even infinite in the limit. Thus, the path correction in these cases needs to be treated differently than what is defined in (11.39). It is noted that the tangent π is a good local approximation of the path under these circumstances. Therefore, the following velocity reference is used

$$\dot{q}_{r,\perp}(\sigma(t)) = K_\perp J(q)^{-1} \left(-(x - x_r)^{\mathrm{T}} n \; n - (x - x_r)^{\mathrm{T}} b \; b \right), \tag{11.40}$$

in the case that the curvature $\kappa < \kappa_l$, where κ_l is a user-specified parameter for the threshold value. In (11.40), x_r is the current desired position along the path in Cartesian space (corresponding to $q_{r,\|}$). Consequently, in this case the control along the normal direction n is handled as the control along the binormal b.

Remark 3
The transversal control laws (11.39)–(11.40) assume, for notational convenience, a manipulator with three DoF and motion in 3D. The adaptation of the control laws to 2D motion is obvious, since the control is performed only in the osculating plane in that case. In the case of more DoF of the manipulator than three, the pseudo-inverse of appropriate parts of the Jacobian for the manipulator can be considered, see also Section 11.8. □

Reference Values and Their Interpretation

Combining the control along the tangential and the transversal directions leads to the following reference values for the generalized coordinates

$$q_r(\sigma(t)) = q_{r,||}(\sigma(t)), \tag{11.41}$$
$$\dot{q}_r(\sigma(t)) = \dot{q}_{r,||}(\sigma(t)) + \dot{q}_{r,\perp}(\sigma(t)), \tag{11.42}$$
$$\ddot{q}_r(\sigma(t)) = \ddot{q}_{r,||}(\sigma(t)). \tag{11.43}$$

The interpretation of the choice of reference values is as follows. For the generalized coordinates q, the reference values should always correspond to a point along the geometric path and consequently no modification is introduced in the case of path deviations. For the velocities $\dot{q}$, compensation is introduced according to (11.39)–(11.40) such that motion toward the desired path is obtained when deviations arise. Regarding the corresponding accelerations $\ddot{q}$, the reference values are chosen for motion along the nominal path. The reason for this being that the path-velocity control, see Section 11.5, limits the path acceleration and scales the nominal path velocity and acceleration trajectories based on the desired and available input torques. Thus, a fundamental mechanism of the algorithm is that the acceleration is adjusted automatically and therefore no modification of $\ddot{q}_r$ is required.

Parametrization of the Basic Controller

By using the expressions (11.41)–(11.43) for the reference values, all variables in the basic controller (11.34) can be parametrized in the path parameter σ and its time-derivatives. This leads to the expression

$$\tau = \beta_1(\sigma)\ddot{\sigma} + \beta_2(\dot{\sigma}, \sigma, \dot{q}, q), \tag{11.44}$$

where it is straightforward to verify that (11.34) can be written on the format (11.44) and to derive the corresponding expressions for β_1 and β_2. Also other control laws than computed torque have been parametrized according to this procedure as discussed in [Dahl, 1992], where examples include feedforward control combined with feedback using PD control and controllers with integral action.

Path-Tracking Velocity Control

With the complete control law parametrized in the path parameter σ, the final step is to determine a scheme for online modification of σ utilizing (11.44). The result is the algorithm called PTVC, and a graphical representation of the algorithm is shown in Figure 11.2. The lower blocks in the figure have already been defined, and the following sections describe the upper blocks for online control of σ, $\dot{\sigma}$, and $\ddot{\sigma}$. The trajectory scaling performed online using feedback from the robot is based on the PVC strategy proposed in

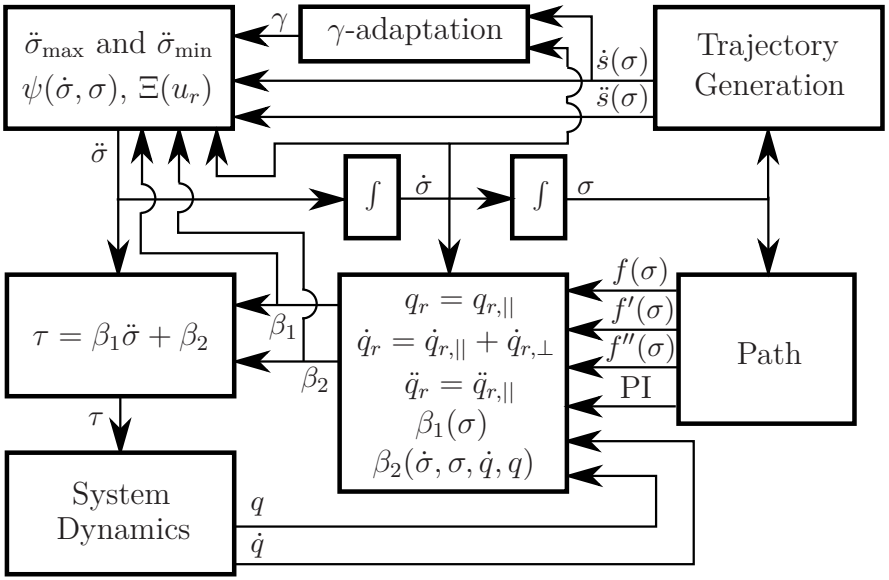

Figure 11.2 Structure of the PTVC algorithm. In addition to the information about the path f and its derivatives, the generation of the local coordinate systems and the time-varying radius R along the path is required (this is denoted path information (PI) in the block diagram).

[Dahl, 1992; Dahl and Nielsen, 1990a; Dahl and Nielsen, 1990b]. This algorithm determines the rate at which the path coordinate, and consequently the reference values q_r and their time-derivatives, is updated to avoid actuator saturation.

Bounds on Path Acceleration In order to ensure that each control input is in the available interval $[\tau_{\min}, \tau_{\max}]$, bounds on $\ddot{\sigma}$ can be computed according to [Dahl and Nielsen, 1990b]. Using the parametrization (11.44) of the control law for the basic controller, the bounds on $\ddot{\sigma}$ are for each input $i,\ i = 1, \ldots, n_d$, as follows

$$\ddot{\sigma}^i_{\max} = \begin{cases} \dfrac{\tau_{\min} - \beta_2^i}{\beta_1^i}, & \beta_1^i < 0 \\ \dfrac{\tau_{\max} - \beta_2^i}{\beta_1^i}, & \beta_1^i > 0 \\ \infty, & \beta_1^i = 0 \end{cases} \qquad \ddot{\sigma}^i_{\min} = \begin{cases} \dfrac{\tau_{\max} - \beta_2^i}{\beta_1^i}, & \beta_1^i < 0 \\ \dfrac{\tau_{\min} - \beta_2^i}{\beta_1^i}, & \beta_1^i > 0 \\ -\infty, & \beta_1^i = 0 \end{cases} \tag{11.45}$$

where, for notational convenience, it was assumed that all generalized coordinates have the same limitations on their corresponding control inputs. The computations can straightforwardly be modified to the case with dif-

ferent constraints on the input torques τ. To satisfy the constraints on all inputs, the bounds on the path acceleration are chosen as $\ddot{\sigma}_{\min} = \max_i \ddot{\sigma}^i_{\min}$ and $\ddot{\sigma}_{\max} = \min_i \ddot{\sigma}^i_{\max}$. Further, non-constant torque limits $\tau_{\max}$ and $\tau_{\min}$ in the algorithm can be used as they are. This is illustrated in a simulation example in Section 11.7 (see Figure 11.9), where state-dependent constraints are used to capture a decrease in maximum available torque because of a velocity-dependent electromotive force in the joint motor.

Trajectory Scaling and Feedback from Path Velocity Nominal values of the path velocity $\dot{s}$ and the path acceleration $\ddot{s}$, as functions of the path parameter, are obtained from the trajectory planning (see Section 11.4). In the case of model uncertainty, downscaling of the nominal optimal trajectories may be necessary. This is achieved by introducing a scaling parameter denoted γ, $0 < \gamma \leq 1$, depicted in the upper middle block in Figure 11.2. In practice, this means that the new nominal path velocity and path acceleration are $\gamma\dot{s}(\sigma)$ and $\gamma^2\ddot{s}(\sigma)$, respectively. This is equivalent to a scaling of the time frame, see [Dahl, 1992], thus implying that the time for the path traversal is increased when $\gamma < 1$. The scaling parameter is made adaptive by the update law proposed in [Dahl, 1992], which prescribes that

$$\dot{\gamma} = \begin{cases} \xi_{\text{sc}}\dot{\sigma}\left(\dfrac{\dot{\sigma}}{\dot{s}(\sigma)} - \gamma\right), & \gamma\dot{s}(\sigma) \geq \dot{\sigma},\ \ddot{\sigma} \text{ saturated} \\ 0, & \text{otherwise} \end{cases} \tag{11.46}$$

where ξ_{sc} is a positive constant and the initial value of the scaling parameter is $\gamma(0) = 1$.

In addition to the trajectory scaling, feedback from the desired path velocity is introduced according to [Dahl and Nielsen, 1990b; Dahl, 1992]. This feedback $\psi(\dot{\sigma}, \sigma)$ is given as function of the path velocity and the path parameter according to

$$\psi(\dot{\sigma}, \sigma) = \frac{\xi_{\text{fb}}}{2}\left(\gamma^2\dot{s}(\sigma)^2 - \dot{\sigma}^2\right), \tag{11.47}$$

where ξ_{fb} is the gain of the feedback and $\dot{s}(\sigma)$ is the precomputed nominal path velocity. When the path acceleration is not saturated, this feedback law results in asymptotic tracking of the desired path velocity, with the tracking time constant specified by ξ_{fb}, see [Dahl, 1992]. Thus, based on the new nominal acceleration $\gamma^2\ddot{s}(\sigma)$ and the feedback in (11.47), the desired path acceleration u_r is computed according to $u_r = \psi(\dot{\sigma}, \sigma) + \gamma^2\ddot{s}(\sigma)$. This desired path acceleration u_r is finally transformed to an achievable path acceleration $\ddot{\sigma}$ using

$$\ddot{\sigma} = \Xi(u_r, \ddot{\sigma}_{\min}, \ddot{\sigma}_{\max}), \tag{11.48}$$

where Ξ is the saturation function, with the second and third arguments as the lower and upper limit, respectively. These steps are represented by the upper left block in Figure 11.2.

Summary of the Control Architecture

To summarize the developed path-tracking control architecture (see Figure 11.2), the computational algorithm for the controller is as follows. Using the expressions (11.41)–(11.43) for the reference values combining tangential and orthogonal control, the quantities β_1 and β_2 in (11.44) are computed. Then, the desired path acceleration u_r is determined based on the (possibly scaled) nominal path acceleration and the path-velocity feedback in (11.47). The constrained path acceleration $\ddot{\sigma}(t)$ is finally obtained using (11.48), which by integration gives $\dot{\sigma}(t)$ and $\sigma(t)$. The scaling parameter γ is simultaneously updated according to (11.46).

11.6 Analysis of the Algorithm

The convergence properties of the control architecture are analyzed, both with respect to the trajectory control law and with respect to the path-tracking controller.

Stability in the Nominal Case

Assuming that trajectory tracking is realized with the computed-torque control law in (11.34) and a correct model of the system, the resulting dynamics for the tracking error $e = q_r - q$ is

$$\ddot{q}_{r,\|} - \ddot{q} + K_v(\dot{q}_{r,\|} - \dot{q}) + K_p(q_{r,\|} - q) = 0, \tag{11.49}$$

in the cases that $\dot{q}_{r,\perp} = 0$. This means that in the nominal case of no deviations from the path, (11.49) coincides with the standard differential equation for the error dynamics with computed-torque control [Spong et al., 2006]. Hence, it is clear that the dynamics of the closed-loop system (*i.e.*, the dynamics for the tracking error e) can be arbitrarily determined by choosing the parameters K_v and K_p. Hence, given appropriate choices of the gain matrices, this guarantees stability of the basic controller. The stability of the path-velocity control along the tangent of the path, PVC, was analyzed in [Dahl and Nielsen, 1990a], see also [Dahl, 1992], and the algorithm has later been extended and verified in subsequent publications (see references in Section 11.1). The convergence properties of the PTVC algorithm are the same as those for the PVC algorithm when on the desired path. However, also an investigation of the transversal behavior for the critical case when deviating from the path is needed, which is treated next.

Convergence Analysis of Path-Tracking Method

The transversal control aims to increase the path-tracking accuracy compared to the case with pure path-velocity control. In this section, the convergence

properties of the control along the normal vector n are analyzed. It is implicitly assumed that the parameters of the trajectory-tracking control law result in stable behavior (*cf.* the relation (11.49)), and that the inputs are not exceeding their saturation limits. As a prelude to the analysis, the example in Section 11.2 is revisited, where it was assumed that the radius R and the angular velocity ω were constant over time. It is observed that the transformation from (11.8)–(11.9) to (11.10)–(11.11) holds even if ω and R are time-varying. This can, for the normal direction n, be verified by differentiating r and using (11.8)–(11.9) in the obtained expression as follows:

$$\begin{aligned}\dot{r} &= \frac{\mathrm{d}}{\mathrm{d}t}\sqrt{x_1^2 + x_2^2} = \frac{x_1\dot{x}_1 + x_2\dot{x}_2}{\sqrt{x_1^2 + x_2^2}} \\ &= \frac{x_1\omega x_2 + x_1^2(R^2 - r^2) - x_2\omega x_1 + x_2^2(R^2 - r^2)}{r} \\ &= \frac{(x_1^2 + x_2^2)(R^2 - r^2)}{r} = r(R^2 - r^2),\end{aligned} \tag{11.50}$$

which is the same for constant and time-varying ω and R. The step in the example in Section 11.2 that does not hold in the general case is the solution of (11.10), which for a time-varying R has a different solution than (11.12). It is thus interesting to investigate the behavior of the fundamental path-convergence equation (11.50), which states that

$$\dot{r}(t) = r(t)\left(R(t)^2 - r(t)^2\right), \tag{11.51}$$

for reference functions having different frequency characteristics. The relation in (11.51) is a so called Bernoulli differential equation, which also can be solved analytically. To illustrate the behavior of this equation for certain choices of the reference trajectory $R(t)$, (11.51) is numerically solved for

$$R(t) = 2 + \sin(\omega t), \tag{11.52}$$

using $\omega = 0$, 0.5, 1, and 2 rad/s. The results of the simulations are shown in Figure 11.3. The observation to be made in the simulations is that if the path curvature $R(t)$ varies moderately (in the simulations interpreted as the choice of ω), then the tracking is very close. In the cases where the curvature varies quickly, path deviations naturally occur (or equivalently, $R(t)$ and $r(t)$ start to differ). The interesting question following this conclusion is whether this property can be quantified. Note that $R(s)$ (here considered as a function of the path coordinate) is given by the geometric properties of the specified path and is consequently in general precomputed. The path-velocity control of $\dot{s}$ determines how fast $R(s)$ is to be updated, so any tracking accuracy can be obtained by decreasing the speed of the system along the specified path. Consequently, the variable ω, or more generally the speed along the

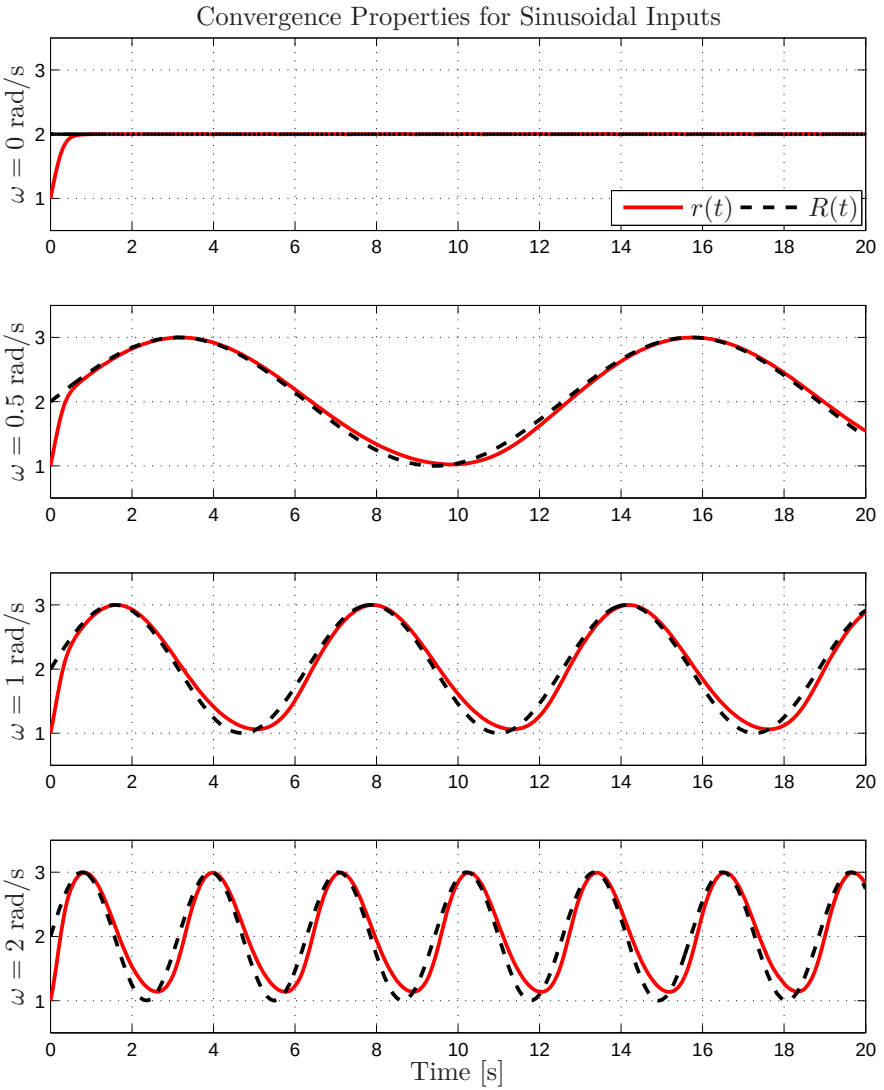

Figure 11.3 Convergence of $r(t)$ in the case of the reference trajectory $R(t) = 2 + \sin(\omega t)$, for different values of the angular velocity ω.

tangent of the path, can always be controlled such that r tracks R within any specified limit.

To further analyze the convergence properties of the transversal control, the path deviation δ is introduced as follows

$$r(t) = R(t) + \delta(t). \tag{11.53}$$

Differentiating (11.53) with respect to time, substituting (11.51) and (11.53) to eliminate r, and finally solving for δ gives the differential equation

$$\dot{\delta}(t) = -\dot{R}(t) - \delta(t)\left(2R(t)^2 + 3R(t)\delta(t) + \delta(t)^2\right). \tag{11.54}$$

This equation can be approximated for small values of δ according to

$$\dot{\delta}(t) = -\dot{R}(t) - 2R(t)^2\delta(t). \tag{11.55}$$

The dynamics of the path deviation $\delta(t)$ is thus driven by $\dot{R}(t)$ and $R(t)$. Hence, the path tracking intuitively depends on the amount of variation in $R(t)$. If $R(t)$ is piecewise constant, then the tracking has exponential convergence because $\dot{R}(t) = 0$, which for small δ gives the path-deviation dynamics $\dot{\delta}(t) = -2R(t)^2\delta(t)$. If R is time-varying, $r(t)$ will track $R(t)$ with a deviation given by (11.54). To illustrate this property in simulations, the reference trajectories were chosen as the functions $R(t) = 2 \pm e^{-t}$ and $R(t) = 2 \pm 1/\log(e+t)$, with the former exhibiting faster dynamics than the latter. Figure 11.4 shows the results from the numerical solution of (11.51) for the different choices of $R(t)$. In the simulations, $R(0)$ and $r(0)$ switch between 1 and 3 to illustrate convergence both from an initial negative error and an initial positive error. As expected from the approximate differential equation (11.55) for the dynamics of $\delta(t)$, a larger $R(t)$ should give faster convergence, and this is confirmed in the cases when $R(t)$ is approaching its final value from $R(0) = 3$, compared to starting in $R(0) = 1$.

11.7 Simulation Results

To evaluate and quantify the performance of the PTVC architecture for path tracking, extensive simulations were performed and three of these are illustrated in this section. The first two examples concern a manipulator with two DoF moving in a two-dimensional plane, illustrating different paths and actuator constraints. The third simulation considers an anthropomorphic manipulator [Siciliano et al., 2009] with three DoF, which enables arbitrary motions in a three-dimensional space for the robot end-effector. The computed-torque strategy (11.34) was used in all simulations as the basic controller, and the control law was parametrized in the natural frequency ω_0 such that

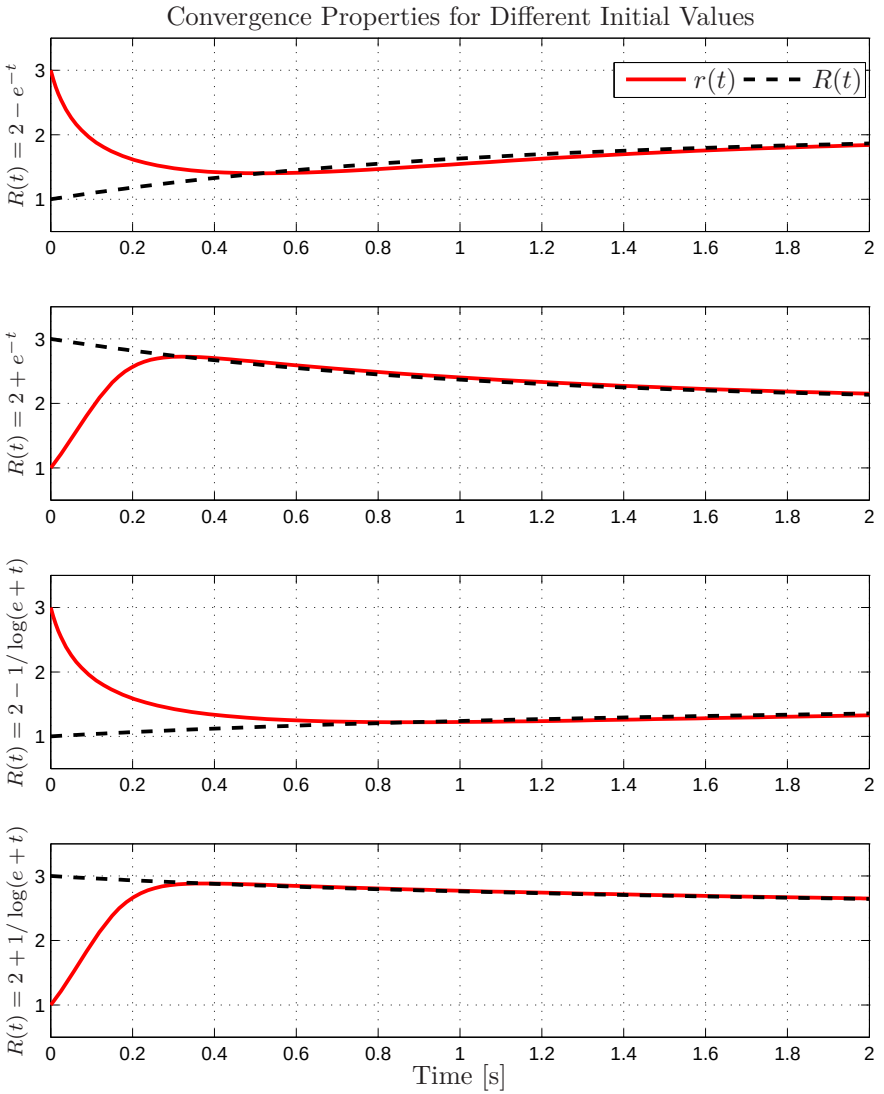

Figure 11.4 Convergence of $r(t)$ for different choices of the reference trajectory $R(t)$ and initial values $r(0)$.

$K_v = \text{diag}(2\omega_0, \ldots, 2\omega_0)$ and $K_p = \text{diag}(\omega_0^2, \ldots, \omega_0^2)$, where $\text{diag}(\cdot)$ is a diagonal matrix with the specified elements along the diagonal. This choice makes the convergence of the error (11.49) to exhibit critical damping [Spong et al., 2006]. In the simulations presented here, $\omega_0 = 8$ rad/s. In all simulation examples, the complete PTVC algorithm with both adaptive trajectory scaling (11.46) and path-velocity feedback (11.47) was used. The parameters in the algorithms were $\xi_{\text{sc}} = 1$ and $\xi_{\text{fb}} = 0.5$.

For comparison, the path-tracking accuracy is evaluated in all examples using both existing PVC—*i.e.*, based on the tangential control only—and using the PTVC architecture with combined tangential and explicit transversal control.

Tracking of a Circle with Planar Robot

The planar robot in the first example has equal mass of the links according to $m_1 = m_2 = 1$ kg and the length of each link is $l_1 = l_2 = l = 2$ m. Further, the center-of-mass is located at a distance of $l/2$ from the joint rotational center and each link has a moment of inertia about an axis through the center-of-mass according to $I = 1$ kgm^2. The constant of gravity is $g = 9.81$ m/s^2. The dynamics of a planar robot (neglecting friction in the joints) moving in a two-dimensional plane subject to gravity is given by

$$\tau = M(q)\ddot{q} + C(\dot{q}, q)\dot{q} + G(q), \tag{11.56}$$

where the elements of the matrices M and C and the vector G are as follows:

$$M_{11} = \frac{m_1 l^2}{4} + m_2 l^2 \left(\frac{5}{4} + \cos(q_2)\right) + 2I, \tag{11.57}$$

$$M_{12} = \frac{m_2 l^2}{2} \left(\frac{1}{2} + \cos(q_2)\right) + I, \tag{11.58}$$

$$M_{21} = M_{12}, \quad M_{22} = \frac{m_2 l^2}{4} + I, \tag{11.59}$$

and

$$c = -\frac{m_2 l^2}{2} \sin(q_2), \tag{11.60}$$

$$C_{11} = c\dot{q}_2, \quad C_{12} = c(\dot{q}_1 + \dot{q}_2), \tag{11.61}$$

$$C_{21} = -c\dot{q}_1, \quad C_{22} = 0, \tag{11.62}$$

$$G_1 = \left(\frac{m_1}{2} + m_2\right) gl \cos(q_1) + \frac{m_2 gl}{2} \cos(q_1 + q_2), \tag{11.63}$$

$$G_2 = \frac{m_2 gl}{2} \cos(q_1 + q_2). \tag{11.64}$$

The forward and inverse kinematics as well as the differential kinematics defining the Jacobian $J(q)$ of the manipulator are straightforward to derive, and are thus omitted in this presentation. The robot base is located at the origin, $(0, 0)$ m, and the desired path is to move along a circle centered in $(0.5, 1.5)$ m with a radius of 1 m. The starting point is $(0.5, 0.5)$ m and the motion is performed counter-clockwise. The inverse kinematics of the robot is used for transforming the desired path in Cartesian space to joint positions q defining the path f to be used in the trajectory generation. The trajectory generation was performed by solving the optimal control problem (11.32) using the convex optimization method presented in [Verscheure et al., 2009c], which is possible since joint friction is neglected in the model. The maximum and minimum available torques are $\tau_{\max} = 40$ Nm and $\tau_{\min} = -40$ Nm, respectively, for each robot joint. The numerical implementation was made in MATLAB and the convex optimization problem was solved using the CVX toolbox [Grant and Boyd, 2008; CVX Research Inc. 2015].

When applying the nominal optimal trajectory in the PVC and PTVC architectures, model errors were introduced in the simulations such that the actual link masses were $m_1 = 1.1$ kg and $m_2 = 1.2$ kg, *i.e.*, higher link masses than expected. The allowed torque utilization was set to ± 45 Nm, which is slightly higher than in the optimization of the trajectory. Moreover, the parameter $K_\perp = 10 I_{2\times 2}$ in the simulation.

As pointed out earlier in Remark 2 and the first paragraph of Section 11.5, a direct application of the optimal solution as reference values $q_r(t)$, $\dot{q}_r(t)$, and $\ddot{q}_r(t)$ to the basic controller (11.34) does not work satisfactory, but is nevertheless presented as a reference solution in this first example. The PVC and the PTVC algorithms and the robot motion were simulated using MATLAB Simulink.

The resulting path-tracking accuracy in Cartesian space is shown in Figure 11.5. Pure trajectory tracking leads, as expected, to large deviations from the desired path. The PVC, in contrast, manages to maintain control authority, and the computed-torque controller can thus to some extent reduce the path-tracking error even without explicit control along the orthogonal directions. However, it is clear that the PTVC with explicit path corrections is significantly more effective than the PVC in handling the path deviations occurring in the lower left corner of the plane as a result of the model uncertainty. The input torques τ are shown in Figure 11.6 and it can be seen that, at all times, one input is close to its limit demonstrating that the close tracking is not achieved by conservatively decreasing the velocity and traversing the path slower than necessary. This preservation of high performance is further visualized in Figure 11.7, showing the path velocity $\dot{\sigma}$ together with the nominal time-optimal solution. Observing the differences in the path velocity for the considered algorithms, it is obvious that both PVC and PTVC reduce the speed along the path in order to handle the model error. In this example,

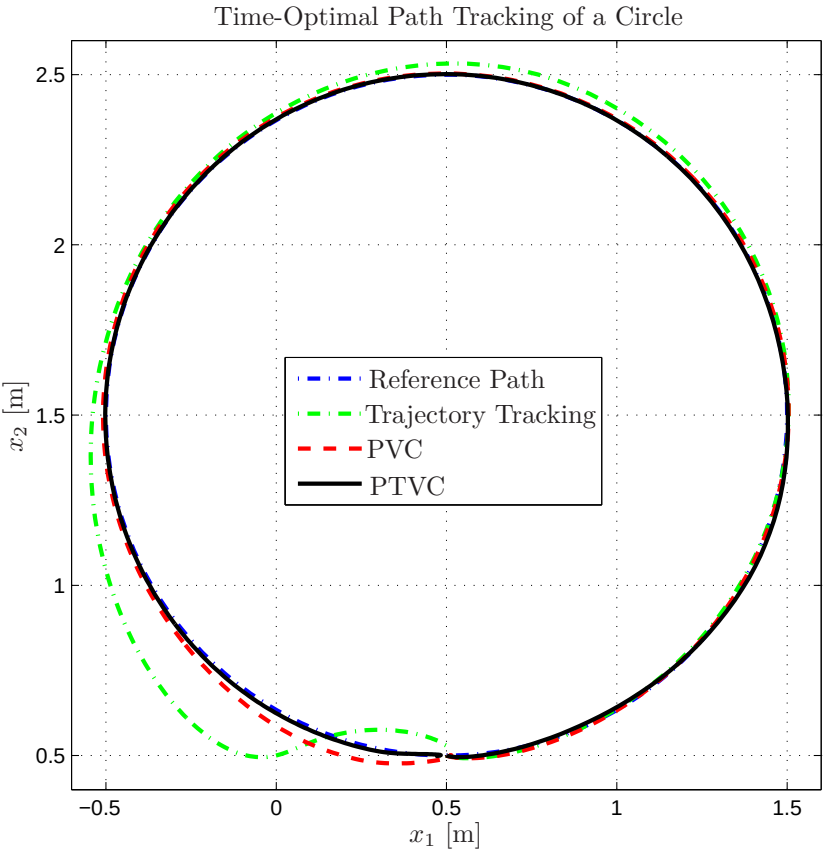

Figure 11.5 Robot end-effector positions x_1 and x_2 during time-optimal path tracking for a planar robot moving along a circle in a two-dimensional plane, see Section 11.7.

PTVC requires slightly longer time to finish the path, but the path-tracking accuracy is increased.

Tracking of Circle Segments and Straight Line with Planar Robot

The planar robot, considered in the previous simulation, is further used to demonstrate the behavior of the control architecture for another type of path. The path is evaluated in the case when the robot has velocity-dependent actuator limits, despite that the trajectory planning assumed constant limits. The tracking task is a path composed of two circular segments with radius 0.8 m, joined by a straight line (see Figure 11.8). Hence, the curvature is zero during parts of the path and the transversal control needs to be handled using

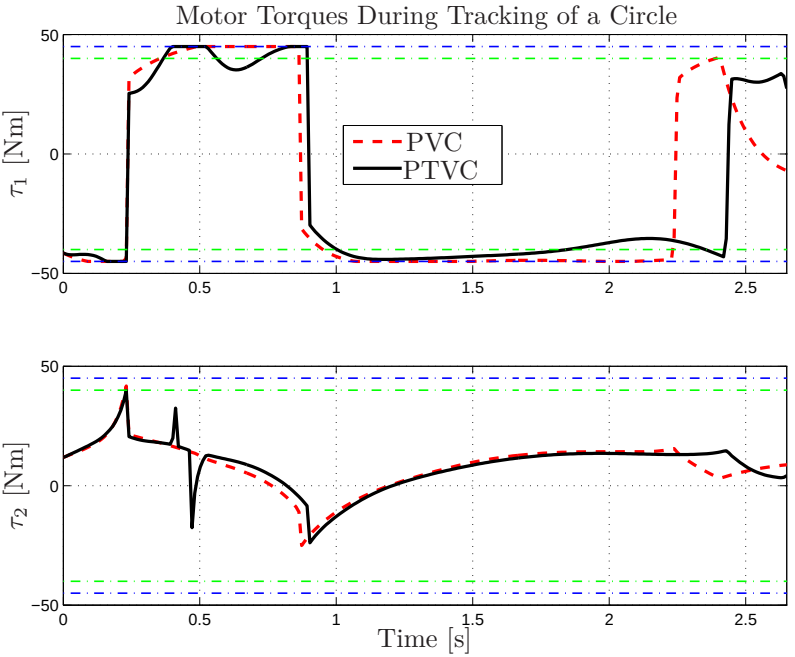

Figure 11.6 Input torques τ during time-optimal path tracking for a planar robot, corresponding to Figure 11.5. The torque limits used in the trajectory planning (green) and in the actual system (blue) are indicated by the horizontal lines.

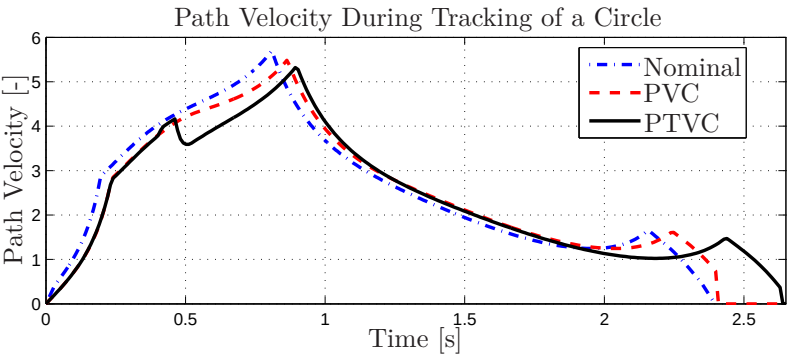

Figure 11.7 Path velocity $\dot{\sigma}$ during time-optimal path tracking for a planar robot, corresponding to Figure 11.5.

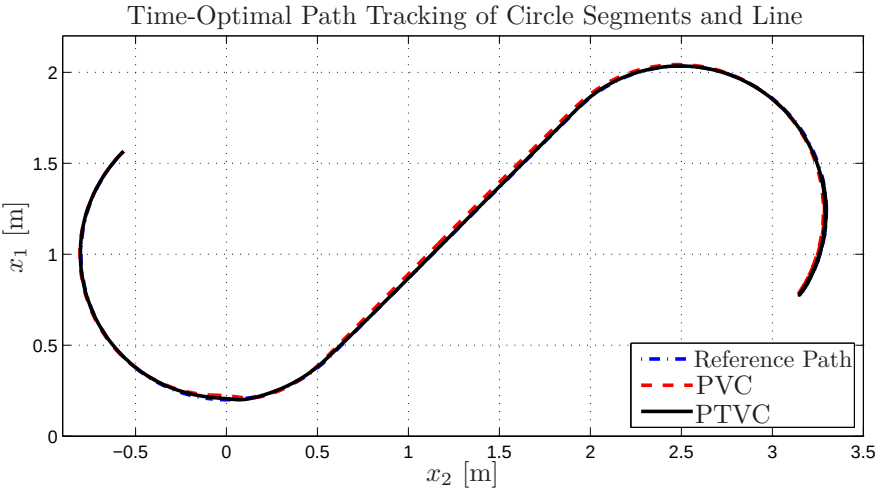

Figure 11.8 Robot end-effector positions x_1 and x_2 during time-optimal path tracking for a planar robot moving along a path composed of circular segments joined by a straight line, see Section 11.7.

the alternative control law defined in Section 11.5 for this path segment. The nominal model parameters and the maximum and minimum input torques are the same as in the previous simulation. The parameter for the feedback in the transversal control was $K_\perp = 10I_{2\times 2}$ in the simulations. The time-optimal trajectory planning was performed using the same procedure as for the previous example.

In the evaluation, model errors were introduced such that $m_1 = 0.8$ kg and $m_2 = 1.15$ kg, *i.e.*, one link was lighter and one link was heavier than what was assumed in the model. Further, velocity-dependent constraints were introduced to model the effect that the maximum torque decreases with increasing joint velocity because of the electromotive force. These constraints were given as follows

$$\tau_{i,\max} = 45 - 5|\dot{q}_i|, \ i = 1, 2,$$
$$\tau_{i,\min} = -45 + 5|\dot{q}_i|, \ i = 1, 2.$$

The resulting paths in Cartesian space from time-optimal path tracking are presented in Figure 11.8 for PVC and PTVC. In both cases, the path starts in the upper left corner. The results show that both PVC and PTVC achieve high accuracy, but that the transversal control in PTVC increases the accuracy of the path tracking both in the circular parts and the straight part of the path. The input torques τ are shown in Figure 11.9.

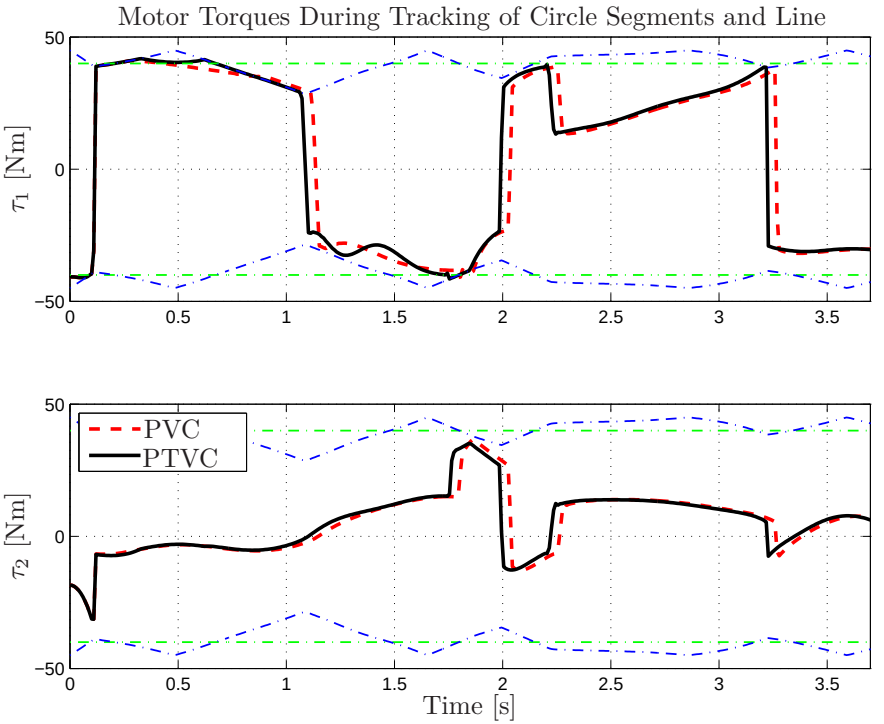

Figure 11.9 Input torques τ during time-optimal path tracking for a planar robot, corresponding to Figure 11.8. The torque limits used in the trajectory planning (green) and in the actual system (blue) are indicated by the horizontal lines. Note the velocity-dependent torque limits in the simulation, aiming at modeling the decrease of the maximum and minimum available torque for increasing velocity.

Note in particular the state-dependent constraints on the actuators. Considering that the trajectory generation is based on constant torque limits, downscaling of the time-optimal trajectories is necessary and therefore automatically performed by the control architecture to satisfy the constraints of the actual system. The path velocity in the respective simulation is visualized in Figure 11.10. Comparing the obtained path velocity with the nominal time-optimal, it is clear that both algorithms downscale the trajectories in order to handle the model uncertainty and the state-dependent constraints. It can be observed that the differences between PVC and PTVC are minor with respect to the path velocity throughout the maneuver, but in this case the PTVC exhibits slightly lower end-time than the PVC.

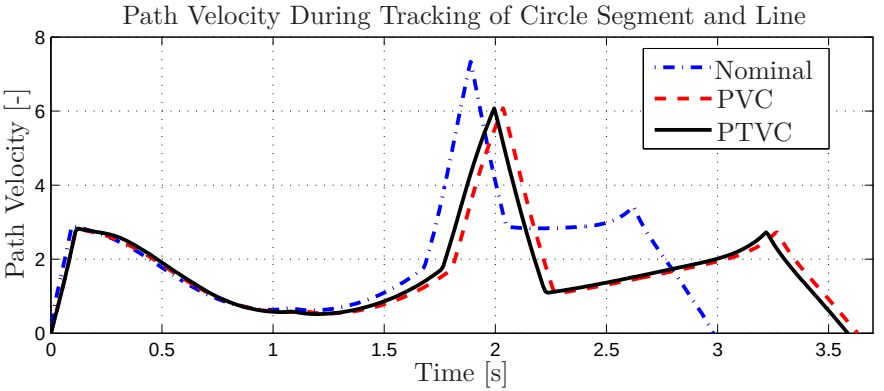

Figure 11.10 Path velocity $\dot{\sigma}$ during time-optimal path tracking for a planar robot, corresponding to Figure 11.8.

Tracking of Path in 3D with Anthropomorphic Robot

In order to investigate the performance on a system with more complex dynamics than the planar robot, the path-tracking algorithm is further evaluated on an anthropomorphic robot manipulator with three DoF [Siciliano et al., 2009]. The masses of link two and three were $m_2 = m_3 = 1$ kg and the lengths of link two and three were $l = 2$ m. The moment-of-inertia matrices for the links were assumed diagonal with elements $I = 1$ kgm^2. The dynamic equations for this robot are straightforward to derive using appropriate software for symbolic manipulations of algebraic expressions, according to the strategy defined in textbooks on robot dynamics, see [Spong et al., 2006; Siciliano et al., 2009]. The format is the same as for the planar robot, see (11.56), but the elements of the matrices are omitted here. Also in this simulation, no viscous friction in the joints was assumed. The robot base is located at the origin, $(0, 0, 0)$ m, and the desired path consists of segments of circles in three dimensions, starting in the point $(1, 0, 1.5)$ m. The time-optimal trajectories were generated using the same procedure as in the two previous simulation examples, with the same maximum and minimum torque limits $\tau_{\max}$ and $\tau_{\min}$. In the simulations, a modeling error was introduced such that $m_3 = 1.3$ kg, *i.e.*, the mass for the third link is 30% higher than what is assumed in the model. As in the previous simulations, the limits on the available torque are ± 45 Nm. The parameter for the feedback in the transversal control was $K_\perp = 50 I_{3\times 3}$ in the simulations.

The simulation results comparing the path-tracking accuracy of PVC and PTVC in Cartesian space are shown in Figure 11.11. It can be observed that PTVC exhibits good path-tracking performance for the considered path in three dimensions. The input torques τ during the motion are provided in

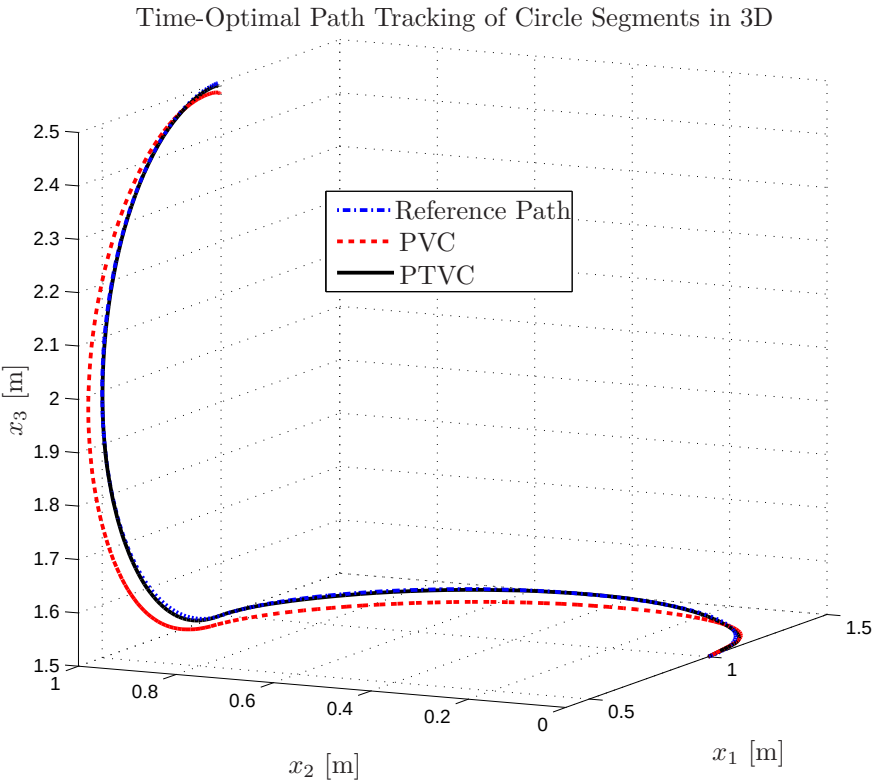

Figure 11.11 Robot end-effector positions x_1, x_2, and x_3 during time-optimal path tracking for an anthropomorphic robot moving along a path in a three-dimensional space, see Section 11.7.

Figure 11.12, together with the actuator limits. As previously discussed, a necessary condition for time-optimality of the path traversal is that at least one input torque is at its limit at each time point [Chen and Desrochers, 1989]. Observing the results in Figure 11.12, this property is indeed indicated. In addition, the path velocity for the PVC and the PTVC algorithms is visualized in Figure 11.13. It can here be observed that the PTVC algorithm both achieves higher path-tracking accuracy and requires slightly shorter time for finishing the path traversal.

11.8 Discussion

As observed in the simulation results presented in Section 11.7, the control architecture for combined path tracking and path-velocity control, PTVC,

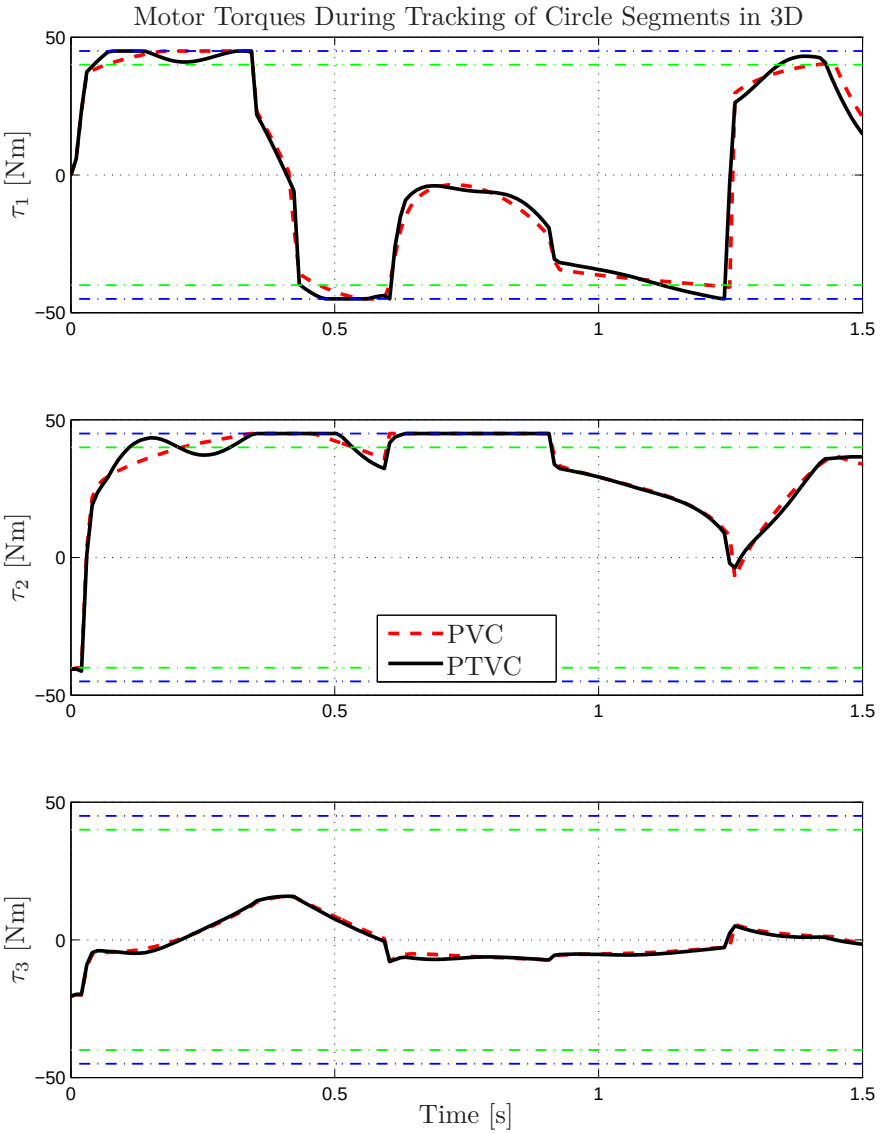

Figure 11.12 Input torques τ during time-optimal path tracking for an anthropomorphic robot, corresponding to Figure 11.11. The torque limits used in the trajectory planning (green) and in the actual system (blue) are indicated by the horizontal lines.

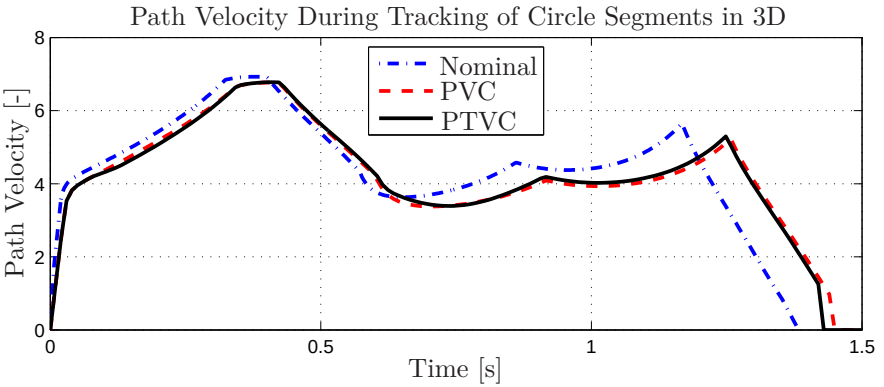

Figure 11.13 Path velocity $\dot{\sigma}$ during time-optimal path tracking for an anthropomorphic robot, corresponding to Figure 11.11.

exhibits convincing results in terms of tracking accuracy. This is not obtained by decreasing the velocity conservatively—instead high-performance character with good torque utilization is preserved. It is thus in place to give a number of remarks and observations regarding the generality of the method, as well as a discussion of limitations and possible extensions of the control architecture.

Definition of the Geometric Path

The only requirement on the geometric path to be tracked is the smoothness, as defined in Section 11.4. Should there be sharp corners, where the tangent to the path is not defined, there are two options. The trajectory planning may in this case be allowed to locally modify the path such that the corners are smoothed according to predefined specifications. An example here is a spline description with limits on the path deviation. The other possibility is to be strict on the path tracking and then the strategy is to move to the corner and stop, restart, and then continue. With these strategies, any realistic geometric path can be tracked.

Both paths in the second and third examples in Section 11.7 consist of different segments, as seen in Figures 11.8 and 11.11. In line with the presentation in Sections 11.2 and 11.3, each part of the path has its own local coordinate axes in the osculating plane corresponding to that segment. The pieces of the path are connected directly, without any smoothing, and thus it is noteworthy how well the tracking is performed for interconnected segments.

Generality Regarding the Trajectory Planning

In the simulation examples considered in Section 11.7, the trajectory planning was performed using a time-optimal criterion, see Section 11.4. It should be stressed that any trajectory-planning algorithm can be used with the PTVC. The only requirement is that the computed trajectory can be specified by the path velocity as a function of the path coordinate (*i.e.*, the distance along the path). This is in almost all cases possible by using the transformations (11.27)–(11.28), or if needed also time-derivatives of higher order. Naturally, time-optimal path tracking is of interest in many applications in order to maximize task effectiveness. However, due to potential wear of the actuators, modified criteria with combinations of time-optimal and energy-optimal can be used instead [Verscheure et al., 2008; Ghazaei Ardakani et al., 2015]. In [Grundelius, 2001], it is shown that solution of the minimum-energy optimal control problem over a fixed time-horizon, where the final time is chosen slightly longer than the corresponding time-optimal, is beneficial for robustness of the control and the wear of the system. PTVC has no limitations with regard to the optimization criterion, so any trajectory planning can be used provided the monotonicity/bijectivity between the path coordinate s and time t.

Controller for Trajectory Tracking

The computed-torque control law was used for tracking of the computed reference values for the generalized coordinates when simulating the experiments presented in Section 11.7. However, in principle any appropriate controller can be used. The only requirement is that the chain rule can be applied as in (11.35)–(11.37), such that the controller can be parametrized on the desired form (11.44). This is in general not a restrictive assumption, as already discussed in Section 11.5.

Insights have been gained in the tuning of the parameters in the trajectory-tracking controller, such as the PD-controller parameters K_v and K_p in the computed-torque control law (11.34). Of course, increasing the gain of the controller leads to faster convergence, but in time-optimal path tracking at least one actuator is close or at its saturation limit so an increased gain will in many cases lead to a decrease of the path velocity compared to the time-optimal (see Section 11.5). However, an advantage with the PTVC algorithm is that high-gain controllers are not necessary for path tracking since the corrections are introduced for explicit path convergence along the orthogonal directions of the path, by modifying the velocity references.

Additional Degrees of Freedom

For robot manipulators, it is common that not only the path of the tool is specified, but also its orientation. Even though the 3D path has been the pri-

mary objective of the control in this chapter, it is straightforward to extend the path-tracking algorithm to also achieve tracking of the desired orientation of a coordinate system associated with the system (important examples include the robot tool). Such an extension requires the consideration of the full 6D motion. For 3D motion, the path is a sequence of positions, whereas for 6D it is a sequence of positions and orientations. Using the inverse kinematics, the path $f(s)$ for the generalized coordinates (now 6D) is defined exactly as in (11.25). The parallel direction $x_{\parallel}$ along the tangent of the path defines a one-dimensional manifold where PVC is applied as before. For the orthogonal dimensions $x_{\perp}$, now typically 5D, the deviations from the desired path are handled similarly as defined in (11.39)–(11.40) in Section 11.5.

Additional Sensors in Workspace

Another interesting feature of PTVC is the ability to include workspace position measurements in the motion controller. As discussed in Chapter 2, the joint motor positions are typically the only sensors available in an industrial robot manipulator. Inclusion of explicit workspace measurements thus has high potential (see Chapter 8). This can easily be incorporated in the developed architecture, since the path-tracking corrections in (11.39)–(11.40) in Section 11.5 could be based on actual workspace measurements (*e.g.*, using the state-estimation developed in Chapter 8) instead of the estimates provided by the forward kinematics.

Extensions

An interesting extension of the PTVC algorithm is to consider other mechanical systems such as road vehicles and mobile robot platforms. Since the main requirements are the bijectivity between s and t for the planned trajectory, and that the controller can be parametrized using the chain rule, which is almost always the case, the basic formulation is straightforward. The main difficulties remaining are those related to non-holonomic constraints and combined longitudinal and lateral wheel friction, *i.e.*, the dynamics can not be written on the form (11.24). For certain non-holonomic systems, this is not a problem and this was remarked in Section 11.4 for the flexible-joint manipulator. In other cases, questions remain to be resolved, and more specifically, what needs to be done in an extension concerning mobile platforms and vehicles is to ensure that the path-velocity control avoids exceeding the maximum possible interaction forces between the wheels and the ground.

11.9 Conclusions

A control architecture, PTVC, for combined path tracking and path-velocity control has been developed. The algorithm does not conservatively decrease

the velocity to stay on the path. In contrast, it preserves the high performance in terms of velocity and execution time, while providing robustness to unmodeled dynamics and actuator limitations. The main property of the controller is a separation of the control along the tangential direction of the path (path-velocity control) and the control along the transversal directions of the path (orthogonal path-tracking control). Formally, the separation is achieved in the framework of natural coordinates of a path, by treating the tangential direction in the osculating plane differently than its orthogonal directions. Compared to trajectory tracking or existing PVC, the results obtained showed that the PTVC exhibits better path-tracking behavior. Still, it maintains high performance of the task execution with significant torque utilization, such that the traversal time is in the same order as for the PVC, or in some cases even better, which is an important observation.

12

Optimization Methodology for Road-Vehicle Maneuvers

12.1 Introduction

In this chapter, a method for solving optimal motion control problems for road vehicles in time- or safety-critical situations is developed. The challenge is to find the correct combinations of vehicle models and optimization formulations. To verify the method, different maneuvers are investigated. In all of these maneuvers, an optimal criterion is considered. This results in that the vehicle—and in particular the tires—perform at the limits of their capacity. Hence, the solutions obtained can give an indication of the maneuverability of the vehicle in the particular situation investigated. Further, different chassis- and tire-modeling principles are employed and compared to each other from an optimal control perspective. This chapter of the thesis is derived, in parts, from the publication [Berntorp et al., 2014b].

The structure of this chapter is as follows: The problem formulation and specific aim of the research presented are discussed in Section 12.1. Vehicle and tire modeling and the particular models investigated in this research are presented in Section 12.2, followed by the formulation and solution method for the studied optimal maneuvering problem in Section 12.3. Finally, conclusions are drawn in Section 12.4.

Objectives and Contributions

The objective of the research presented in this chapter is to utilize recent advances in optimization tools to develop a platform for study of optimal vehicle-maneuvering problems. The purpose is to demonstrate the usefulness of the platform, and to obtain insightful solutions where one specific interest is in future onboard systems for control and safety. Regarding methodology, this means that the control-oriented goal is to find a formulation that gives insight into improved safety systems that benefit from the recent develop-

ments in sensor and computing-power technology in vehicles—*e.g.*, future driver-assistance systems performing closer to what the most experienced drivers can do. To that end, the optimal maneuvers in different scenarios under different conditions are studied in Chapter 13.

As already noted, it is stated in [Sharp and Peng, 2011] that modeling is a crucial part of optimal control of vehicles. Different versions of the well known single-track (ST) and double-track (DT) chassis models have been used in several safety systems and optimal vehicle-maneuver studies, see [Chakraborty et al., 2013; Yi et al., 2012; Dingle and Guzzella, 2010; Velenis, 2011; Timings and Cole, 2013] for examples with ST and [Sundström et al., 2010; Andreasson, 2009] for examples with DT. An ST model sufficiently captures the planar dynamics and has the advantage of lower computational complexity because of its reduced number of states. On the other hand, a DT model incorporates the dynamics for motion in space and can, *e.g.*, account for lateral load transfer of the chassis between the inner and outer wheels in cornering. In general, however, such complex models require more computation time when solving the corresponding optimization problem. It is thus a natural objective to study a spectrum of chassis models and compare both the solutions and the computational complexity, instead of focusing on a particular vehicle model on a certain road surface. Finally, there is the aspect of modeling of the interaction between the tire and the road. Such models are reproductions of the situation under which they were measured or may be an average over different conditions, and they may exhibit significant differences. Here, it is an objective to investigate a spectrum of tire–road models from simple to more descriptive (but with increased computational complexity). To demonstrate the value of a platform for study of optimal vehicle-maneuvering problems, it must be verified that the platform provides sensible solutions for a spectrum of models with different characteristics. The platform has been used with several model combinations, and in this thesis six different combinations of chassis and tire models are presented, all of which are common in the literature. The obtained results are discussed and analyzed in detail in Chapter 13. The chosen model configurations are the ones that are considered most interesting for the analysis and understanding of the balance between accuracy and computational demand for future automotive safety systems.

Problem Formulation

The goal of the research presented in this chapter and the following is twofold. The first goal is control-oriented and consists of finding the time-optimal vehicle trajectory when maneuvering through a time-critical situation, with the vehicle being subject to various constraints, which are motivated by physical limitations of the driver, the vehicle, and the road geometry. The second goal

is model-oriented and aims at investigating whether different combinations of chassis and tire models yield fundamentally different solutions, not only in the cost function in the optimization but also in the internal vehicle behavior. Using optimal control for computing vehicle trajectories leads often to input signals that are at their limits and, *e.g.*, time-optimal control solutions tend to push the vehicle and tire models more to extreme performance levels than simulations typically do. Hence, it is plausible that conclusions about model behavior can be made from such a comparison. It is also of particular interest to analyze the results from a safety-system perspective—*i.e.*, what driving behavior and model characteristics can be extracted from the results. Differential-algebraic equation (DAE) systems of the format

$$\begin{aligned} \dot{x}(t) &= G(x(t), y(t), u(t)), \\ 0 &= h(x(t), y(t), u(t)), \end{aligned}$$

where $G(x(t), y(t), u(t))$ and $h(x(t), y(t), u(t))$ are twice continuously differentiable nonlinear functions of the vehicle differential variables x, algebraic variables y, and control inputs u are considered. The employed vehicle models differ with regard to both chassis and tire aspects.

Previous Research

Optimal control problems for vehicles in time-critical situations have been studied previously, see [Chakraborty et al., 2011; Velenis and Tsiotras, 2005; Chakraborty et al., 2013] for different examples concerning T-bone collisions and cornering. In [Velenis and Tsiotras, 2005], the influence of longitudinal load transfer on the optimal maneuver was also investigated. Aggressive vehicle maneuvers and related control design was investigated in [Velenis, 2011]. The influence of different road surfaces and wheel-actuation strategies on time-optimal maneuvers was investigated in [Tavernini et al., 2013] using a ST chassis model. Control laws for vehicle emergency-maneuvers were developed in [Dingle and Guzzella, 2010] based on an analytical optimal control approach, and the method was applied to different situations for investigating the benefit of independent steering and driving on the wheels. Certain assumptions on the vehicle dynamics were imposed, and roll and pitch dynamics for the chassis were neglected. Optimal lane-change maneuvers were theoretically investigated in [Shiller and Sundar, 1998], using both analytical and numerical methods. In particular, the minimum distance at which an approaching obstacle can be avoided was determined, given an initial velocity and the optimal feasible maneuver. Methods for computing the time-optimal race-car line were presented in [Kelly and Sharp, 2010; Casanova, 2000], which were based on numerical optimal control methods. In contrast to the classical nonlinear optimal control approach to vehicle-maneuver optimization, an approximate linearization approach leading to a sequence of convex

optimization problems (one problem for each time point in a discrete grid along the spatial path of the vehicle) was proposed in [Timings and Cole, 2013] and applied to a turn maneuver. Methods for constraint-based path and trajectory planning for optimal avoidance maneuvers were developed in [Anderson et al., 2010; Anderson et al., 2012]. In [Yi et al., 2012], aggressive pendulum-turn maneuvers, performed by professional race-car drivers, were investigated and related high-performance modeling was developed. Further, [Sundström et al., 2010; Andreasson, 2009] discussed optimal control of over-actuated vehicles, where similar optimization tools as those employed in this chapter were utilized. A method for optimal actuation allocation to the individual wheels in yaw control of road vehicles was proposed in [Tøndel and Johansen, 2005], and an expansion of the research comprising a two-level strategy for active steering and adaptive control allocation was presented in [Tjønnås and Johansen, 2010]. Further, an optimal yaw-control law for road vehicles was developed in [Esmailzadeh et al., 2003]. In [Lundahl et al., 2013] it was reported that simplified vehicle models, such as the ST model, identified from experimental data managed to replicate the behavior of real vehicles. However, this was based on less aggressive driving situations, and not using optimization as a criterion for determining the control inputs. The method considered in this chapter has also been applied for investigation of critical truck maneuvers in [Lundahl et al., 2015].

12.2 Modeling

The vehicle-dynamics modeling presented in this section incorporates the chassis motion modeling (having a varying number of DoF) and the tire-force modeling. Further, calibration of the tire models is discussed and a subsequent investigation of the qualitative behavior of the studied tire models is presented.

Chassis Models

Three different chassis models of varying complexities are used. The first, most complex, model is a DT model with roll (ϕ) and pitch (θ) dynamics and both longitudinal and lateral load transfer.[1] This chassis model is illustrated in Figure 12.1. The model has five degrees of freedom, namely two translational and three rotational. The chassis rotational motions in the roll, pitch, and yaw directions are characterized by the vehicle chassis inertias I_{xx}, I_{yy}, and I_{zz}, respectively.

[1] Motivated by a passenger-vehicle perspective in the analysis, aerodynamic modeling in the chassis dynamics is neglected. It can, however, easily be introduced in the modeling framework if rally or racing applications are to be investigated.

DT Model The suspension system is modeled as a rotational spring-damper system. Consequently, the moment τ_ϕ produced by the suspension system in the roll direction is given by

$$\tau_\phi = (K_{\phi,f} + K_{\phi,r})\phi + (D_{\phi,f} + D_{\phi,r})\dot{\phi}, \tag{12.1}$$

and correspondingly for the moment τ_θ in the pitch direction according to

$$\tau_\theta = K_\theta\theta + D_\theta\dot{\theta}, \tag{12.2}$$

where K and D are parameters. Throughout this chapter, the indices $\{f, r\}$ are used for denoting the front and the rear wheel pair, respectively, and $\{1, 2, 3, 4\}$ denote the individual wheels. The dynamic equations for the longitudinal load transfer are given by

$$(F_{z,1} + F_{z,2})l_f - (F_{z,3} + F_{z,4})l_r = K_\theta\theta + D_\theta\dot{\theta}, \quad \sum_{i=1}^{4} F_{z,i} = mg \tag{12.3}$$

where $F_{z,i},\ i \in \{1, 2, 3, 4\}$, denote the time-dependent normal forces, m is the vehicle mass, l_f, l_r are defined in Figure 12.1, and g is the constant of gravity. The lateral load transfer is determined by the relations

$$-w(F_{z,1} - F_{z,2}) = K_{\phi,f}\phi + D_{\phi,f}\dot{\phi}, \tag{12.4}$$

$$-w(F_{z,3} - F_{z,4}) = K_{\phi,r}\phi + D_{\phi,r}\dot{\phi}, \tag{12.5}$$

where w is defined in Figure 12.1.

The derivation and complete details of the translational and orientational dynamic equations for the double-track model are omitted here; for details the reader is referred to [Berntorp, 2013; Berntorp, 2014].

ST-Pitch Model The second model is a single-track model, where pitch dynamics has been added (ST-pitch). The dynamics incorporate the same modeling of the suspension system in the pitch direction as for the DT model. The dynamic equations for this model are conceptually found from DT by lumping the left and right wheel on each axle together and setting the roll angle to zero. This results in the following equations of motion along the translational directions

$$\dot{v}_x - v_y\dot{\psi} = h\left(\sin(\theta)\left(\dot{\psi}^2 + \dot{\theta}^2\right) - \cos(\theta)\ddot{\theta}\right) + \frac{F_X}{m},$$

$$\dot{v}_y + v_x\dot{\psi} = -h\left(\sin(\theta)\ddot{\psi} + 2\cos(\theta)\dot{\theta}\dot{\psi}\right) + \frac{F_Y}{m},$$

and the following along the rotational directions

$$(I_{zz} + I_{xx}\sin(\theta)^2)\ddot{\psi} = M_Z - h\sin(\theta)F_Y,$$

$$\ddot{\theta}I_{yy} + D_\theta\dot{\theta} + K_\theta\theta - mgh\sin(\theta) = -h\cos(\theta)F_X + \dot{\psi}^2\sin(\theta)\cos(\theta)\Delta I_{xz},$$

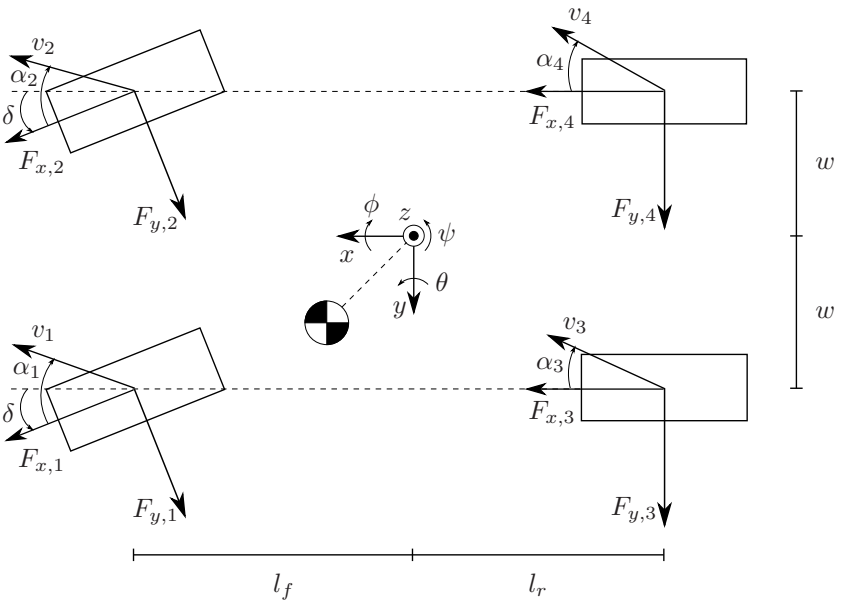

Figure 12.1 The double-track (DT) model with pitch and roll dynamics. Note that the geometric lateral slip angles α are shown in the figure.

where $\Delta I_{xz} = I_{xx} - I_{zz}$ and F_X, F_Y, and M_Z are the lumped total forces and moment, derived from the DT model detailed in [Berntorp, 2013; Berntorp, 2014]. The load-transfer equations in the pitch direction are given by (12.3), where the wheel forces on each axle are lumped together for the ST-pitch chassis model.

ST Model The third chassis model is a single-track model (ST), illustrated in Figure 12.2. This model lumps together the left and right wheel on each axle, and roll and pitch dynamics are neglected. Thus, the model has two translational and one rotational DoF. The model dynamics is straightforward to derive, see, *e.g.*, [Schindler, 2007], and are given by

$$\dot{v}_x - v_y\dot{\psi} = \frac{1}{m}(F_{x,f}\cos(\delta) + F_{x,r} - F_{y,f}\sin(\delta)) = \frac{F_X}{m}, \tag{12.6}$$

$$\dot{v}_y + v_x\dot{\psi} = \frac{1}{m}(F_{y,f}\cos(\delta) + F_{y,r} + F_{x,f}\sin(\delta)) = \frac{F_Y}{m}, \tag{12.7}$$

$$I_{zz}\ddot{\psi} = l_f F_{y,f}\cos(\delta) - l_r F_{y,r} + l_f F_{x,f}\sin(\delta) = M_Z, \tag{12.8}$$

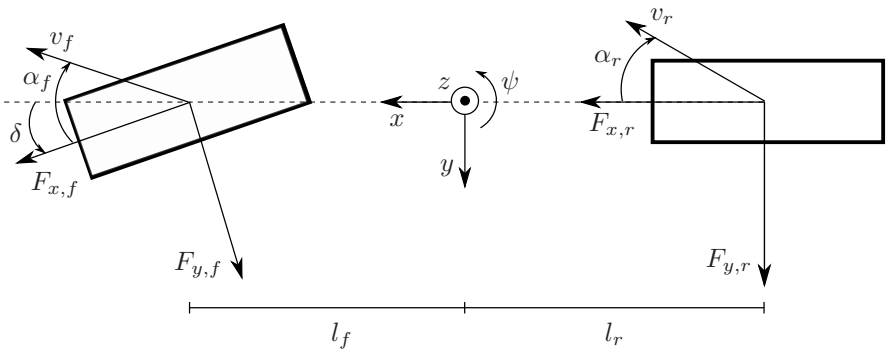

Figure 12.2 The single-track (ST) model utilized as a base model in the formulation of the optimal control problem in this chapter. Note that the geometric lateral slip angles α are shown in the figure.

where F_X, F_Y, and M_Z are the global forces and moment. The nominal normal force F_{z0} resting on the respective wheel in steady state is given by

$$F_{z0,f} = mg\frac{l_r}{l}, \quad F_{z0,r} = mg\frac{l_f}{l}, \tag{12.9}$$

where the length of the wheel base is defined as $l = l_f + l_r$.

Wheel and Tire Modeling

The slip angles α_i and slip ratios κ_i are defined as in [Pacejka, 2006]:

$$\dot{\alpha}_i \frac{\sigma}{v_{x,i}} + \alpha_i = -\arctan\left(\frac{v_{y,i}}{v_{x,i}}\right), \tag{12.10}$$

$$\kappa_i = \frac{R_w\omega_i - v_{x,i}}{v_{x,i}}, \ i \in \{f, r\} \text{ or } \{1, 2, 3, 4\}, \tag{12.11}$$

where σ is the relaxation length, R_w is the wheel radius, ω_i is the wheel angular velocity for wheel i, and $v_{x,i}$ and $v_{y,i}$ are the longitudinal and lateral wheel velocities for wheel i with respect to an inertial system, expressed in the coordinate system of the wheel. The wheel dynamics[2], necessary for slip-ratio computation, are given by

$$T_i - I_w\dot{\omega}_i - F_{x,i}R_w = 0, \quad i \in \{f, r\} \text{ or } \{1, 2, 3, 4\}. \tag{12.12}$$

Here, T_i is the driving/braking torque and I_w is the wheel inertia. When developing a platform for investigation of optimal maneuvers, it is of interest

[2] In the wheel-dynamics modeling, it is assumed that the vehicle has an open differential, motivated by a passenger-vehicle perspective in the study. However, it has been verified that the considered optimization framework can handle other differential settings, such as a locking differential on the rear axle and a limited-slip differential, as well.

to handle and compare different tire characteristics, and thus to cope with different tire models. The nominal tire forces F_{x0} and F_{y0}—*i.e.*, the forces under pure slip conditions—are computed with the Magic Formula model [Pacejka, 2006], given by

$$F_{x0,i} = \mu_{x,i} F_{z,i} \sin(C_{x,i} \arctan(B_{x,i}\kappa_i - E_{x,i}(B_{x,i}\kappa_i - \arctan B_{x,i}\kappa_i))), \tag{12.13}$$

$$F_{y0,i} = \mu_{y,i} F_{z,i} \sin(C_{y,i} \arctan(B_{y,i}\alpha_i - E_{y,i}(B_{y,i}\alpha_i - \arctan B_{y,i}\alpha_i))), \tag{12.14}$$

for each wheel $i \in \{f, r\}$ or $\{1, 2, 3, 4\}$. In (12.13)–(12.14), μ_x and μ_y are the friction coefficients and B, C, and E are model parameters.

Under combined slip conditions—*i.e.*, both κ and α are nonzero—the longitudinal and lateral tire forces F_x and F_y will depend on both slip quantities. How this coupling is described can have significant effect on the modeled vehicle dynamics. In an optimal maneuver, the computed control inputs will result in the best combination of longitudinal and lateral force, and these forces are coupled via the physics of the tire. Even though detailed experiments, like the ones in [Carlson and Gerdes, 2005] for longitudinal stiffness, are lacking for the complete longitudinal-lateral tire interaction, there are large differences in the characteristics, see [Isermann, 2006; Pacejka, 2006; Kiencke and Nielsen, 2005; Rajamani, 2006; Schofield, 2008]. As a consequence, it is of interest to investigate how the adopted tire model influences the resulting computed optimal maneuver. Also, different tire models may interact with certain chassis models such that the obtained results are not realistic. Two different tire-model categories are chosen for the study in this chapter and the following, which are described next.

Combined Forces based on the Friction Ellipse A straightforward model of combined forces is based on the friction ellipse, and is described by the elliptical constraint

$$F_{y,i} = F_{y0,i}\sqrt{1 - \left(\frac{F_{x0,i}}{\mu_{x,i} F_{z,i}}\right)^2}, \; i \in \{f, r\} \text{ or } \{1, 2, 3, 4\}, \tag{12.15}$$

where F_{x0} is used as an input variable, see for example [Wong, 2008]. However, in the setup considered in this chapter, it is desirable to use the driving and braking torques as input, see (12.12), since this is a quantity that can be controlled in a physical setup of a vehicle. The main limitation with the friction ellipse model is that the longitudinal force does not explicitly depend on the lateral slip α, which is not realistic. With lateral slip present (*i.e.*, α nonzero), it is possible to use a related, more involved model, which is also based on the friction ellipse [Kiencke and Nielsen, 2005]. However, (12.15) is used because it represents a straightforward combined-force model that is

used in the vehicle optimal control literature [Andreasson, 2009; Sundström et al., 2010]. This model is denoted the friction-ellipse (FE) model.

Representing Combined Slip with Weighting Functions Another, more comprehensive approach to tire modeling proposed in [Pacejka, 2006], which is inspired by the Magic Formula and explicitly accounting for the effect on the tire force by the longitudinal and lateral slip, is to scale the nominal forces (12.13)–(12.14) with weighting functions $G_{x\alpha,i}$ and $G_{y\kappa,i}$, which depend on α and κ. The relations along the longitudinal direction are, [Pacejka, 2006],

$$H_{x\alpha,i} = B_{x1,i}\cos(\arctan(B_{x2,i}\kappa_i)), \tag{12.16}$$
$$G_{x\alpha,i} = \cos(C_{x\alpha,i}\arctan(H_{x\alpha,i}\alpha_i)), \tag{12.17}$$
$$F_{x,i} = F_{x0,i}G_{x\alpha,i},\ i \in \{f,r\} \text{ or } \{1,2,3,4\}, \tag{12.18}$$

and the corresponding relations along the lateral direction are given by

$$H_{y\kappa,i} = B_{y1,i}\cos(\arctan(B_{y2,i}\alpha_i)), \tag{12.19}$$
$$G_{y\kappa,i} = \cos(C_{y\kappa,i}\arctan(H_{y\kappa,i}\kappa_i)), \tag{12.20}$$
$$F_{y,i} = F_{y0,i}G_{y\kappa,i},\ i \in \{f,r\} \text{ or } \{1,2,3,4\}, \tag{12.21}$$

where B and C are model parameters. Throughout this chapter, (12.16)–(12.21) are denoted the weighting-functions (WF) model.

Calibrating Tire Models for Comparison

When comparing an optimal maneuver based on two different tire models, it is not obvious how to calibrate the models with respect to the specific tire to get comparable solutions to the optimal control problem. As an example, Figure 12.3 shows the resulting tire forces for two tire models; the first is parametrized using FE and the second is parametrized using WF by employing the experimental data presented in [Pacejka, 2006]. To equalize these models in comparative studies, one way would be to have the same average resultant force, whereas another way would be to equalize the longitudinal stiffness. For the particular tire models considered in this chapter, the same parameters have been used in the relations for the nominal longitudinal and lateral forces in (12.13)–(12.14), *i.e.*, for pure slip conditions the two tire models agree. In the calibration procedure, the parts of the tire-model parameters depending on the time-varying normal force are neglected. Instead the parameters are determined from the normal forces present when the vehicle is in steady-state motion.

Qualitative Behavior of Tire Models The characteristic surfaces defining the relation between tire force and the slip quantities are used as a basis

for the analysis. These plots are referred to as Force-Slip (FS)-diagrams. This 3D surface is defined as the resulting force

$$F_{\text{res},i} = \frac{\sqrt{(F_{x,i})^2 + (F_{y,i})^2}}{F_{z,i}},\ i \in \{f, r\} \text{ or } \{1, 2, 3, 4\}, \tag{12.22}$$

as a function of the longitudinal slip κ and slip angle α. The resultant force is normalized with the normal force to enable comparison of models with and without dynamic load transfer. The model based on the weighting functions is parametrized according to the parameters found from experiments in [Pacejka, 2006], representing a tire behavior when driving on dry asphalt. The friction-ellipse model also uses the parameters in [Pacejka, 2006] for the nominal tire forces. Figure 12.3 shows how the resulting tire force for the front wheel varies over slip angle and slip ratio for the friction-ellipse and the weighting-functions models with the parameters presented in Table A.2 in Appendix A. Studying Figure 12.3 gives a basis for a discussion of the behavior of the tire models in an optimal maneuver—*e.g.*, the models give different force characteristics for combined slip, where the most prominent difference is that FE predicts a significantly larger force for combined slip of high values than what WF does. Further, the characteristic peaks in F_{res} obviously influence the behavior of the tire-force model significantly.

Another fundamental difference between the models is seen in Figure 12.4, where the lateral force is plotted against the longitudinal force for FE and WF. Differences between the two approaches are clear. In particular, the longitudinal force increases monotonically with decreasing lateral force for FE, which is not the case for WF. Typically, experimental results exhibit similar behavior as WF, even though the friction ellipse has been used successfully in the case of not too aggressive maneuvers.

12.3 Optimal Control Problem

Based on the chassis and tire dynamics described in Section 12.2, the optimal solutions for different maneuvering situations are to be determined. These optimal trajectories are computed as the solution to an optimal control problem and considering the physical setup of the problem, it is clear that a solution exists given that the initial velocity of the vehicle is chosen sufficiently low. The resulting optimal control problems are more challenging than they initially appear, since, *e.g.*, the time-optimality implies that the tire-friction models in certain cases may operate outside the area of which they have been validated. Also, solving dynamic optimization problems numerically where the time horizon is free is in general more demanding than solving a problem with fixed time horizon, since it adds additional degrees of freedom. To the purpose of increasing the robustness of the convergence,

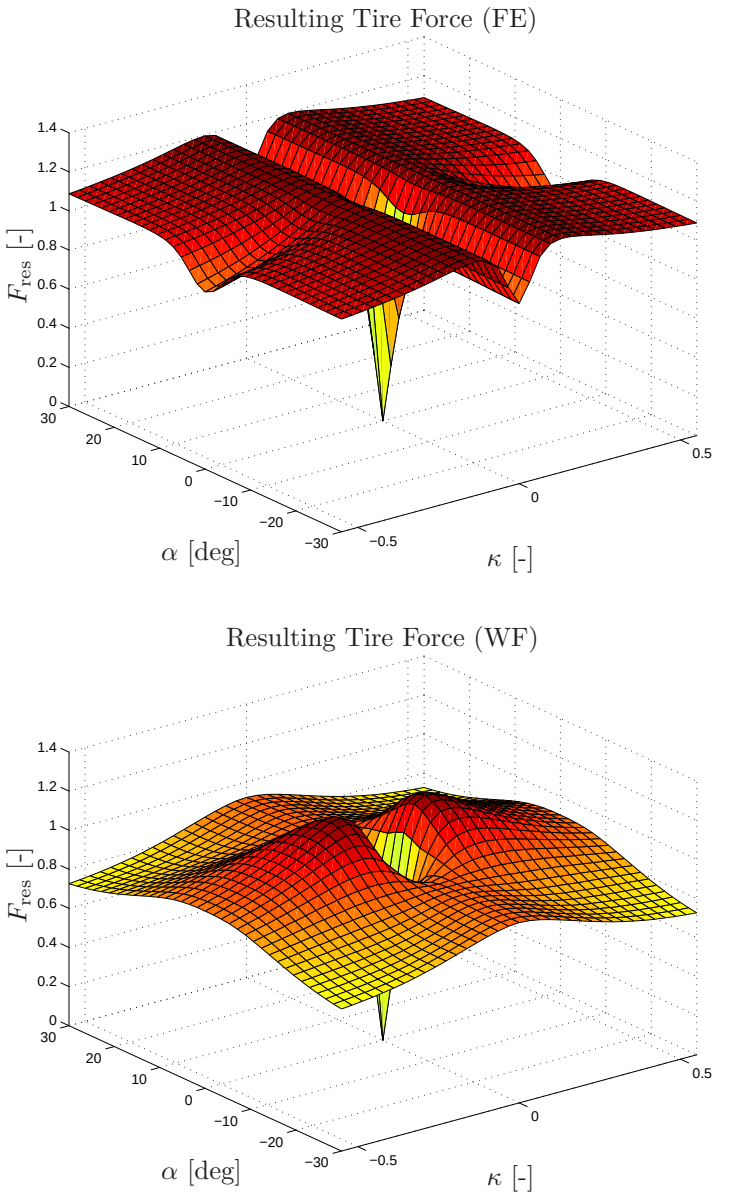

Figure 12.3 Resulting tire force F_{res} for the front wheel with a friction-ellipse model (upper) and a weighting-functions model (lower), with experimental parameters from [Pacejka, 2006] according to Table A.2 in Appendix A. The force is normalized with the steady-state normal force.

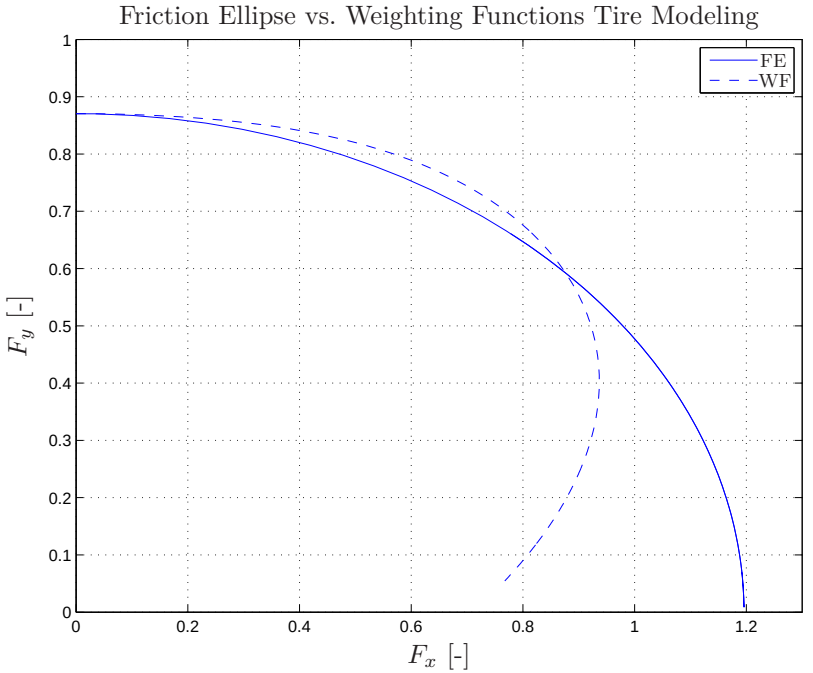

Figure 12.4 Lateral tire force F_y plotted as function of the longitudinal force F_x (normalized) for the friction ellipse (FE) and weighting functions (WF) models, respectively, with $\alpha = 7.6$ deg, $\kappa \in [0, 1]$. The models predict different behavior for large values of the longitudinal tire force; in particular, the longitudinal force increases monotonically with decreasing lateral force for FE.

an initialization procedure based on driver models has been developed as part of the optimization platform, which is described later in this section. Moreover, scaling of the optimization variables to the same nominal interval (based on *a priori* assumptions on their maximum values) is essential to avoid ill-conditioned matrices in the optimization procedure.

Formulation of the Optimal Control Problem

The chassis motion models are formulated as differential-algebraic equation systems according to $\dot{x} = G(x, y, u)$, as described in Section 12.1. The wheel driving and braking torques $T = \begin{pmatrix} T_f & T_r \end{pmatrix}$, as well as the steer angle δ of the front wheels are considered as inputs. For simplicity, it is assumed that the front wheels have the same steer angle in the double-track models. Moreover, considering that the analysis in this and the next chapter has a focus on optimal maneuvers for passenger vehicles in safety-critical situations, it is

justified to assume that the double-track models have one wheel-torque input for each axle. The inputs are equally distributed between the wheels at the respective axle, *i.e.*, $T_1 = T_2 = T_f/2$ and $T_3 = T_4 = T_r/2$. This is equivalent to separate front and rear braking systems and an open differential. Further, the tire-force model is written as the equation system $h(x, y, u) = 0$. The optimization problem is formulated over the time horizon $t \in [0, t_f]$ and the objective of the optimization is here assumed to be to minimize the final time t_f of the maneuver. Accordingly, the dynamic optimization problem to be solved is stated as follows:

$$
\begin{aligned}
\text{minimize} \quad & t_f \\
\text{subject to} \quad & T_{i,\min} \leq T_i \leq T_{i,\max},\ i \in \{f, r\} \text{ or } \{1, 2, 3, 4\} \\
& |\dot{T}_i| \leq \dot{T}_{i,\max},\ i \in \{f, r\} \\
& |\delta| \leq \delta_{\max}, \quad |\dot{\delta}| \leq \dot{\delta}_{\max} \\
& x(0) = x_0, \quad x(t_f) = x_{t_f}, \\
& f(X_p, Y_p) \leq 0 \\
& \dot{x} = G(x, y, u), \quad h(x, y, u) = 0
\end{aligned}
\tag{12.23}
$$

where x_0 are the initial conditions for the differential variables, x_{t_f} are the desired values for the differential variables at the final time $t = t_f$, and (X_p, Y_p) is the position of the center-of-mass of the vehicle. In practice, the conditions at $t = t_f$ are only applied to a subset of the model state variables. Further, $f(X_p, Y_p)$ is a mathematical description of the road constraint for the center-of-mass of the vehicle for the respective maneuver. These constraints in the geometric two-dimensional XY-plane can, *e.g.*, formulated as super-ellipses with different radii and degrees.

Implementation and Solution

The models of the chassis and tire dynamics were implemented using the modeling language Modelica [Modelica Association, 2015]. Utilizing Optimica [Åkesson, 2008], which is an extension of Modelica for high-level description of optimization problems based on Modelica models, the implementation of the vehicle and tire dynamics described in Section 12.2 and the optimal control problem (12.23) was straightforward.

Because of the complex nature of the nonlinear and nonconvex optimization problem (12.23), analytical solutions were intractable. Instead, numerical methods based on simultaneous collocation [Biegler et al., 2002] of the continuous-time problem (12.23) were utilized. The collocation procedure and solution of the optimization problem were performed using the open-source software platform JModelica.org [Åkesson et al., 2010; JModelica.org, 2015]. The user interaction with this platform is illustrated in Figure 12.5. In the particular scheme employed in the utilized optimization platform,

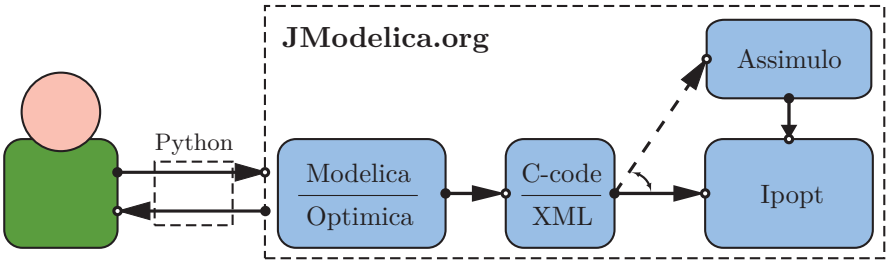

Figure 12.5 Illustration of the usage of the optimization software JModelica.org. The user utilizes the scripting language Python in order to communicate with JModelica.org. The Modelica and Optimica models are compiled to C-code and an XML description. The NLP is subsequently numerically solved using the solver Ipopt. Simulation of DAE systems for initialization and verification purposes is possible using the software package Assimulo [Assimulo, 2015], which is integrated in JModelica.org.

the trajectories of the optimization variables were described using Lagrange polynomials. The collocation procedure results in a discrete-time nonlinear program (NLP), where the interpolation coefficients of the polynomials are the optimization variables. For further details on the collocation procedure, see [Åkesson, 2007]. The resulting NLP is solved using Ipopt [Wächter and Biegler, 2006], which is a numerical solver based on a primal-dual interior-point method developed for large and sparse optimization problems. For the optimal control problems considered in Chapter 13, 150 discretization elements were used and each element contained three collocation points. The selection of discretization parameters was based on a strategy where the optimal control problem was solved iteratively with different parameters. The Jacobian and the Hessian related to the problem were required in the Newton iterations in the optimization procedure. Considering the complexity of the employed chassis and tire models, finite-difference approximations of these quantities in quasi-Newton methods such as the Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm [Dennis and Schnabel, 1983] are not numerically stable, especially not for the DT model. Instead, calculation of the required derivatives with numerical precision—*i.e.*, in the order of 10^{-16}—was performed using automatic differentiation [Griewank, 2000]; this procedure reduced convergence times several orders of magnitudes and increased numerical stability as compared with the approach using approximate derivatives. This observation is in agreement with the conclusions in the survey [Sharp and Peng, 2011].

Initialization Procedure

Robust convergence to a solution of the NLP in Ipopt relies on proper initialization. A simulation of a driver model is used in combination with the

Table 12.1 Solution times and number of iterations required for solving the time-optimal maneuver problem in a hairpin turn for the considered model configurations (see Chapter 13 for further details).

Model	Solution Time [s]	No. of Iterations
ST FE	8.0	111
ST WF	12.3	101
ST-pitch FE	16.9	110
ST-pitch WF	7.8	78
DT FE	137.8	340
DT WF	144.2	287

vehicle model in order to obtain initial trajectories for the model variables in the numerical optimization procedure. The driver model is designed such that the vehicle tracks the middle of the road while following a predefined velocity profile. The driver model, operating the steer angle δ and the rear wheel torque T_r, is based on the lane-keeping controller in [Rajamani, 2006]:

$$\delta = \delta_{ss} - k_1 e - k_2 \dot{e} - k_3 \xi - k_4 \dot{\xi}, \tag{12.24}$$

$$T_r = T_{r,ff} - k_5 (v - v_{\text{nom}}) \tag{12.25}$$

where δ_{ss} is the steady-state steer angle in the hairpin turn, e is the lateral deviation from the desired path, ξ is the angular deviation from the desired heading direction, $T_{r,ff}$ is the feedforward term for the rear torque input, v_{nom} is the desired velocity profile, and $\{k_i\}_{i=1}^{5}$ are parameters of the driver model. The controller parameters k_1–k_4 are chosen such that the eigenvalues of the closed-loop system are placed as suggested in [Rajamani, 2006]. The desired velocity v_{nom} is tracked by controlling the rear wheel torques with the feedforward part $T_{r,ff}$, computed from $\dot{v}_{\text{nom}}$, and a term proportional to the deviation from the velocity profile.

Using the developed initialization procedure and a standard PC with an Intel QuadCore i7 CPU having 16 GB of RAM in an implementation using only one of the cores, the solution times and number of iterations reported in Table 12.1 were obtained for a hairpin maneuver, which is the most challenging maneuver considered in Chapter 13. Obviously, the number of iterations and computation times are dependent on the complexity of the model configuration and the maneuver. Noticeable is that the DT chassis model requires approximately one order of magnitude longer solution time than the ST chassis model.

12.4 Conclusions

This chapter discussed a method for determining optimal vehicle maneuvers in time-critical situations using a state-of-the-art numerical optimization platform. Vehicle and tire models frequently encountered in the literature were analyzed from an optimization perspective and an optimal control problem was subsequently formulated. Tools for solving this type of optimal control problems were discussed. Based on the JModelica.org platform, solutions to the NLPs can be obtained using collocation for discretization of the continuous-time optimal control problems.

13

Applications of Method for Optimal Road-Vehicle Maneuvers

13.1 Introduction

In the previous chapter, a method for determining optimal road-vehicle maneuvers was discussed. In this chapter, three different applications of the strategy to determine the optimal maneuvers are investigated. The first is an evaluation and quantification of the implications of different choices of the chassis and tire models on the resulting optimal maneuvers. The second application is an investigation of the influence of the road condition on the time-optimal maneuver in a hairpin turn. The third application is a study on how different braking strategies in ESC systems relate to the vehicle performance in the time-critical situation when entering a curve with too high velocity. This chapter is derived, in parts, from the publications [Berntorp et al., 2014b; Olofsson et al., 2013; Lundahl et al., 2014].

The structure of this chapter is as follows: The impact of chassis and tire modeling on the computed optimal maneuvers is investigated in Section 13.2. In Section 13.3, the optimal maneuvers in a hairpin turn under uncertain road conditions are studied. Optimal maneuvers for ESC systems are investigated in Section 13.4. Finally, the chapter is summarized and conclusions are drawn in Section 13.5.

13.2 Investigation of the Impact of Chassis and Tire Modeling

In this section, the results achieved by solving the optimal control problem (12.23) in Chapter 12 are presented for different vehicle-model configurations

in a number of maneuvering situations. The aim is to quantify and evaluate the influence on the resulting optimal maneuvers in different driving situations. For each investigated maneuver, the following chassis- and tire-model configurations were evaluated:

- ST with FE for tire modeling—*i.e.*, the single-track model without pitch and roll dynamics, and the friction ellipse for tire modeling.
- ST with WF for tire modeling—*i.e.*, the single-track model without pitch and roll dynamics, and the weighting functions for tire modeling.
- ST-pitch with FE for tire modeling—*i.e.*, the single-track model with pitch dynamics, and the friction ellipse for tire modeling.
- ST-pitch with WF for tire modeling—*i.e.*, the single-track model with pitch dynamics, and the weighting functions for tire modeling.
- DT with FE for tire modeling—*i.e.*, the double-track model with pitch and roll dynamics, and the friction ellipse for tire modeling.
- DT with WF for tire modeling—*i.e.*, the double-track model with pitch and roll dynamics, and the weighting functions for tire modeling.

The numerical values for the vehicle-model parameters used in this study are provided in Table A.1 in Appendix A. The corresponding parameters for the tire-force models are found in Table A.2 in Appendix A. The tire-model parameters have been derived from the empirical parameters presented in [Pacejka, 2006].

Maneuvers

Three time-critical maneuvers were chosen for evaluation of the approach to trajectory generation. The motivation for choosing multiple maneuvers is firstly to verify that the developed platform can handle different situations and vehicle behavior, and secondly to enable comparison of the resulting solution for the considered model configurations. The first maneuver is a 90°-turn, which is important in, *e.g.*, evaluation of ESC systems in lane-keeping scenarios. The second maneuver is a hairpin turn, which is selected because it tests several aspects of the tire and chassis modeling. The third maneuver is a double lane-change maneuver, where the dimensions of the track was chosen congruent to the ISO standard 3888-2 [ISO 3888-2:2011, 2011]. This maneuver is, in particular, common for testing the possibilities for collision avoidance if obstacles are approaching the road, but is also important for evaluation of the roll stability of a vehicle.

Optimization Prerequisites

For the evaluations, the maximum allowed wheel angle $\delta_{\max}$ and wheel-angle change rate $\dot{\delta}_{\max}$ were set to 30 deg and 60 deg/s, respectively, which are reasonable parameters, both seen from physical and driver limitations. For all considered maneuvers, the start $(X_{p,0}, Y_{p,0})$ and final vehicle position (X_{p,t_f}, Y_{p,t_f}) were set to be in the middle of the road. The initial velocities were $v_0 = 70$ km/h in the 90°-turn maneuver, $v_0 = 25$ km/h in the hairpin-turn maneuver, and $v_0 = 80$ km/h in the double lane-change maneuver. Further, the lower and upper constraints on the torque inputs were chosen as

$$T_{i,\min} = -\mu_{x,i} R_w mg, \ i \in \{f, r\} \text{ or } \{1, 2, 3, 4\}, \tag{13.1}$$

$$T_{r,\max} = \mu_{x,r} R_w F_{z0,r}, \tag{13.2}$$

$$T_{f,\max} = 0, \tag{13.3}$$

which implies a rear-wheel driven vehicle. The constraints on the derivative of the torque inputs were chosen as $\dot{T}_{i,\max} = 2.5\mu_{x,i} R_w mg, \ i \in \{f, r\}$. Note that the friction coefficients and the other tire-model parameters on the left and right wheels on the respective axle are assumed to be equal in the DT models. The choice of torque limitations originates from that the maximum braking torques that can be applied on the wheels are significantly larger than the corresponding acceleration torques. Further, the driving-torque limit was set to prevent excessive wheel spin, corresponding to large slip ratios. This can be justified because the employed empirical tire models are based on tire-force measurements that for experimental reasons are only possible to obtain for a limited area in the plane spanned by the longitudinal and lateral slip. Moreover, the wheel velocities were limited to be nonnegative, since solutions with wheel backspin were not desired. To increase the probability of convergence of the solver to a solution, the longitudinal forces were constrained to $|F_x| \leq \mu_x F_z$ and correspondingly for the lateral forces. Note that both the force and the wheel-velocity constraints are mathematically redundant.

The analysis of the achieved results presented in this section is focused on the 90°-turn and the hairpin-turn maneuvers. The results from the double lane-change maneuver are commented on and compared to the results obtained for the two other maneuvers in the discussion later in this section.

Optimal Trajectories in the 90°-turn Maneuver

An analysis of the resulting time-optimal maneuvers for the 90°-turn is presented. First, the computed time-optimal trajectories are explained, followed by four paragraphs discussing different aspects of the maneuver; the geometric vehicle paths, the vehicle-model variables, the global tire forces, and how the available tire forces are used for the particular maneuvers.

The vehicle start position was set to $(X_{p,0}, Y_{p,0}) = (37.5, 0)$ m, which is in the lower right corner in Figure 13.1, and the vehicle was aligned with the road direction, $\psi_0 = \pi/2$. The target vehicle position was set to $(X_{p,t_f}, Y_{p,t_f}) = (0, 37.5)$ m, where the vehicle heading should be in the road direction, $\psi_{t_f} = \pi$. Figures 13.2–13.3 display the computed trajectories for the time-optimal maneuvers for the different chassis- and tire-model configurations in the 90°-turn. The variable v is the norm of the vehicle velocity vector, given by

$$v = \sqrt{v_x^2 + v_y^2},$$

and β is the body-slip angle defined as

$$\beta = \arctan\left(\frac{v_y}{v_x}\right).$$

Figure 13.4 shows the global forces F_X and F_Y. In addition, the yaw moment M_Z generated from the tire forces, *i.e.*, the moment about an axis orthogonal to the road, is visualized. These quantities are displayed as functions of the driven distance s to enable comparison of the results for the different model configurations. Figures 13.5 and 13.6 show the FS-diagrams for ST with WF and DT with FE, respectively. The combination of longitudinal and lateral slip in the time-optimal solution is plotted on the surface. Moreover, Table 13.1 provides the times t_f for completing the maneuver with the respective model configuration. The computed values of the objective value t_f vary approximately 5% at most, which occurs between the ST-pitch model with FE tire model and DT with WF tire model. It is noticeable that ST-pitch and DT exhibit larger discrepancies in the time for completing the maneuver for the respective tire model than ST. This is most certainly a result of the load transfer incorporated in the former models, which results in significant variations in the normal load on the wheels during the maneuver. No significant trend in the time for completing the maneuver is observed with regard to chassis model. However, the friction ellipse seems to result in shorter end times for ST-pitch and DT. This is because the resulting force for the friction ellipse is always larger than that for the weighting functions when combined slip is present, *cf.* Figure 12.3 in Chapter 12. In this maneuver, combined slip is developed; hence, the friction-ellipse modeling results in larger forces and thus increased acceleration and deceleration.

Path Trajectories The path trajectories shown in Figure 13.1 are similar from a qualitative perspective. The largest discrepancies of the path trajectories occur between ST-pitch with FE and ST-pitch with WF during the exit phase, and are approximately 15% of the road width. It is interesting to note that the two ST-pitch configurations result in different strategies when exiting the turn; ST-pitch with WF results in the most narrow curve-taking,

Table 13.1 Time for completing the maneuver for each model configuration in the 90°-turn.

Model	Maneuver Completion Time
ST FE	4.28 s
ST WF	4.28 s
ST-pitch FE	4.12 s
ST-pitch WF	4.21 s
DT FE	4.30 s
DT WF	4.35 s

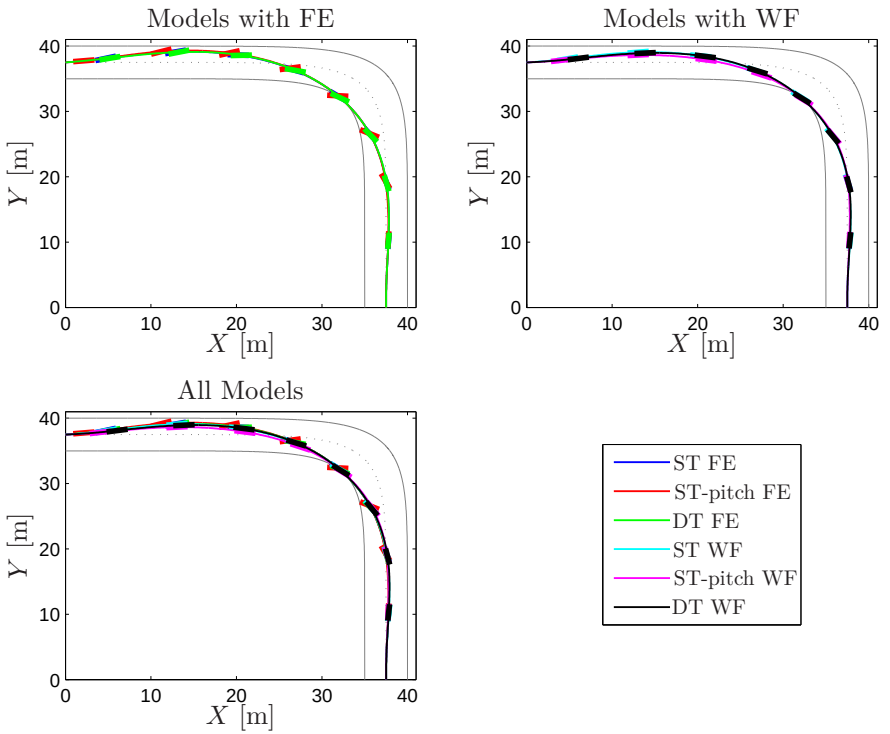

Figure 13.1 Time-optimal path trajectories in the 90°-turn obtained for the respective model configuration. The colored bars represent the vehicle heading every half second.

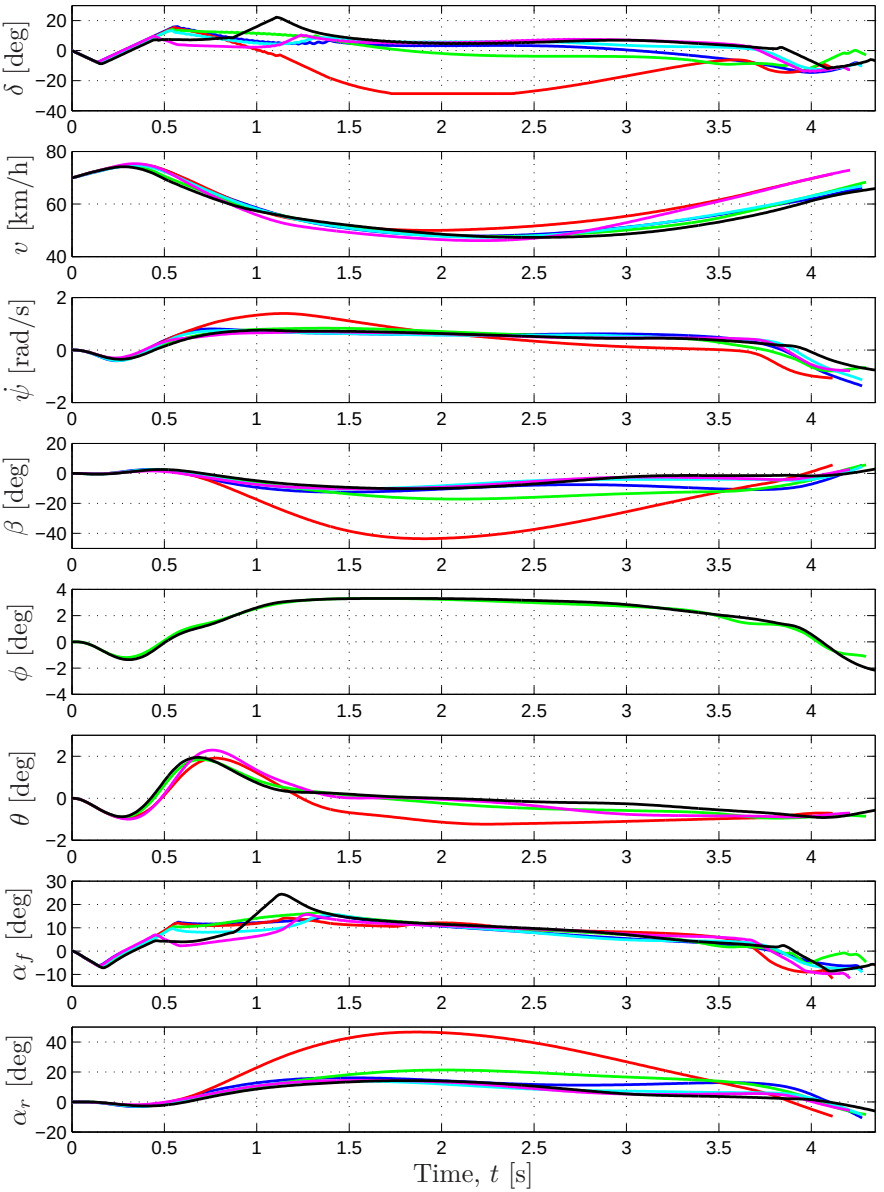

Figure 13.2 Time-optimal solutions obtained for the 90°-turn, with the respective model configuration. For the DT models, the average lateral slip angle for the right and left wheels is shown for each axle in order to enable comparison of the solutions. Same color scheme as in Figure 13.1.

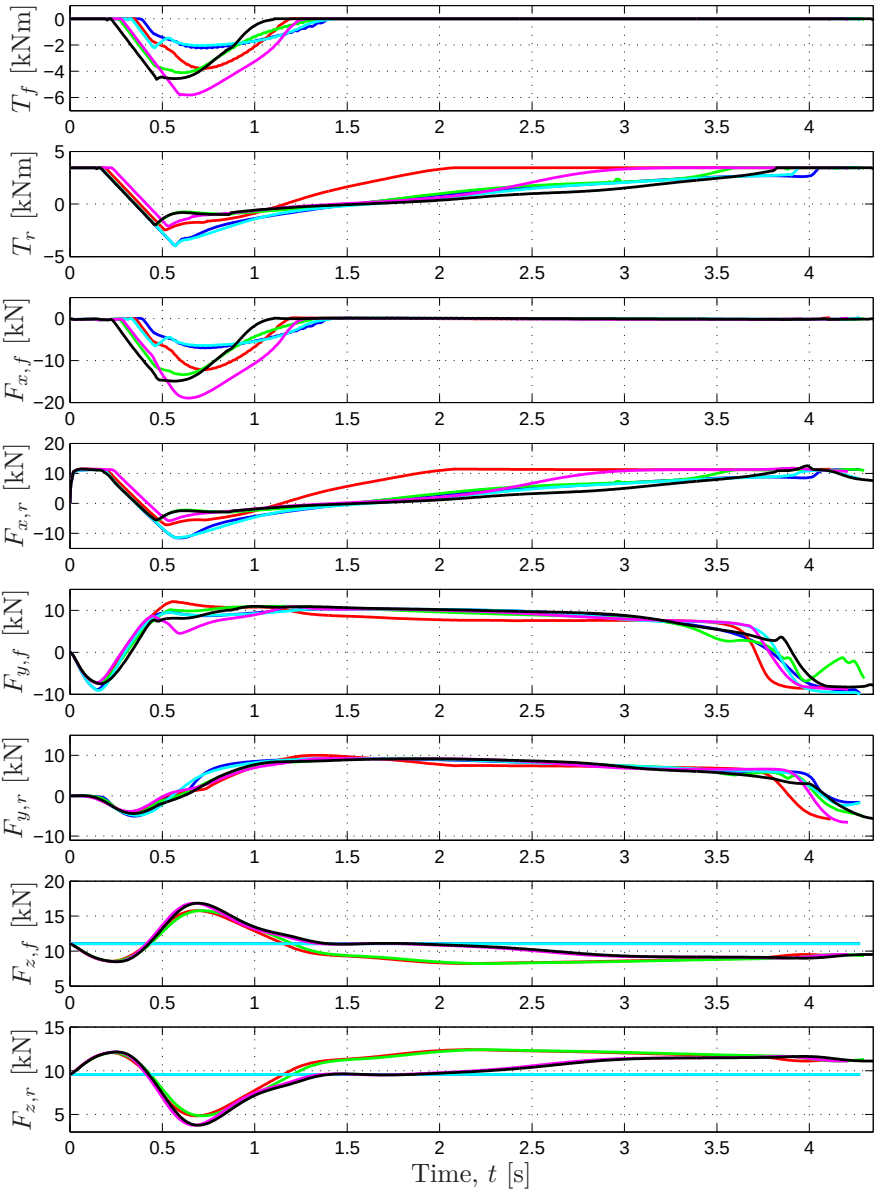

Figure 13.3 Time-optimal solutions obtained for the 90°-turn, with the respective model configuration. Same color scheme as in Figure 13.1. To allow comparison between the models, the forces and torques for the DT models have been lumped together, yielding one longitudinal force, one lateral force, and one torque per vehicle axle.

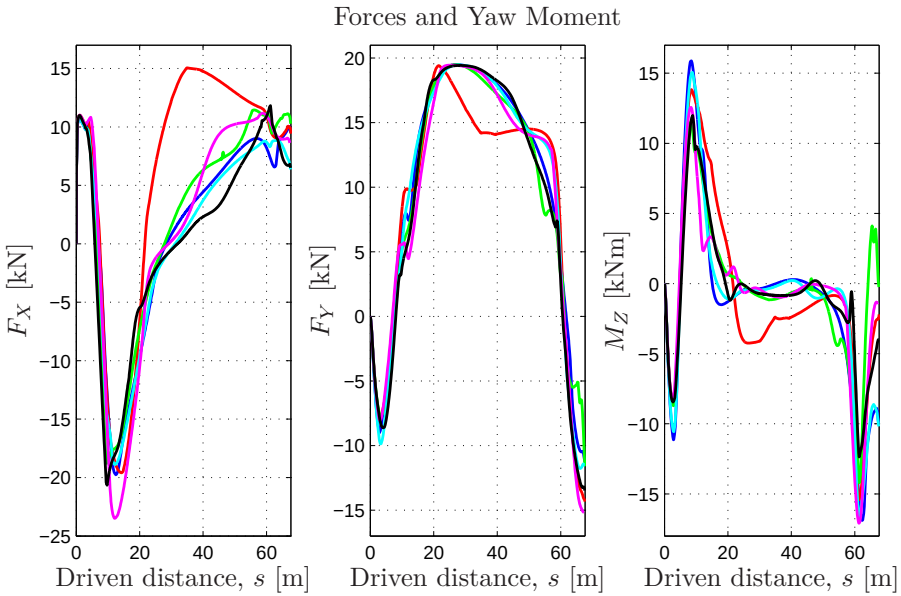

Figure 13.4 Longitudinal force F_X, lateral force F_Y, and yaw moment M_Z, developed by the tires in the 90°-turn, illustrated as functions of the driven distance s. Same color scheme as in Figure 13.1.

whereas ST-pitch with FE results in the most wide. However, irrespective of the tire model, the differences between the chassis models ST and DT are minor throughout the maneuver. Moreover, the geometric paths are virtually indistinguishable for the FE model configurations (even though the slip behavior is significantly different) but exhibit more variation for the WF tire models. It is plausible that this is a result of the difference in the tire force predicted by the respective model for combined slip; for the FE model, approximately the same tire force is obtained for different slip combinations. This is not the case for WF, where the tire force rapidly decreases for increasing slip quantities, and hence the load transfer has more significant effects on the optimal path in the curve taking.

Trajectories for the Model Variables The first observation when investigating the optimal trajectories for the model variables in Figures 13.2–13.3 closer is that the solution obtained for ST-pitch with FE has a completely different behavior than the solutions obtained for the other model configurations. In particular, the slip behavior is much more excessive with this model configuration, which can be observed in the plots for β (peaking at a significant angle of 44 deg) and α_r. The observed slip behavior is also consistent with the steering angle δ, where the constraints on the angle are active for

a short period around $t = 2$ s. A plausible explanation for this is that the longitudinal load transfer in combination with the characteristics of FE leads to that the largest forces are achieved when the vehicle both accelerates and slides. The tire forces for FE and WF are compared in Figure 12.3 in Chapter 12. It could also be argued that the longitudinal force is used in the curve for centripetal acceleration, since the friction coefficient in the longitudinal direction is larger than in the lateral direction. For DT, however, heavy centripetal acceleration implies lateral load transfer, resulting in less available tire force for the inner wheels. This, in turn, limits the driving or braking effort if wheel spin or even wheel lock should be avoided. Disregarding the solution obtained for ST-pitch with FE and considering all the remaining model configurations, the solutions are similar for several variables, such as ϕ, θ, $\dot{\psi}$, and β.

Investigating the optimal vehicle trajectories further from a qualitative perspective, Figures 13.2–13.3 show that the different models result in characteristics that are similar with regard to several aspects. Prior to turning toward the corner, all the resulting optimal solutions exhibit a slight rightward maneuvering while accelerating. This is followed by a braking phase, where both the front and the rear wheels are used. Initially in the braking phase, a significant braking torque is applied, which is gradually reducing as the vehicle approaches the turn, see T_f and T_r in Figure 13.3. Larger lateral forces are generated in the turn. Half-way through the turn, at $t \approx 2$ s, all solutions comprise an increasing driving torque, which accelerates the vehicle out of the turn. In the final stage when exiting the turn, all solutions utilize maximum driving torque as expected.

It can be observed that the steer angle varies considerably between the models. At approximately $t \approx 0.7$ s, a smaller δ is applied for the ST-pitch and DT WF, than for the remaining model configurations. For the ST-pitch and DT WF, the longitudinal load transfer is utilized to achieve a strategy with more emphasis on braking when entering the turn, with the lateral force being slightly smaller. Hence, a lower δ is consistent with these observations. Shortly after, approximately at $t = 0.8$–1.2 s, the steer angle increases sharply in the solutions for the ST-pitch and DT WF. The effect is clearly visible for DT in the front slip angle in Figure 13.2. This behavior is not observed for the corresponding models with FE for tire modeling. Given the resulting forces developed at the front wheels at this time, there exist two different strategies to achieve the observed behavior: Either by utilizing front-wheel braking together with a moderate steer angle, or by only applying a large steer angle and achieve the longitudinal contribution from $F_{y,f} \sin(\delta)$ alone. The latter seems to be what is utilized for ST-pitch and DT WF.

Tire Forces The total forces that are developed by the tires, displayed in Figure 13.4, mirror the behavior observed in the plots with the internal

model variables and path trajectories; the ST-pitch model with FE exhibits largest discrepancies. Considering the remaining configurations, F_Y and M_Z are similar from a qualitative perspective. The solutions exhibit quantitative differences in the magnitude of M_Z during the initial and exit phases of the maneuver. This observation is also in agreement with the tire-force plots in Figure 13.3. There are larger numerical discrepancies in F_X, in particular during shorter periods of the maneuver.

Summary of Differences The most significant differences between the resulting solutions appear, not unexpectedly, for the control inputs and variables closely coupled to the longitudinal dynamics of the vehicle, such as T_f, T_r, and v in Figures 13.2–13.3, and F_X in Figure 13.4. Comparing the ST models with the DT chassis models, the double-track models reduce front-wheel braking slightly earlier, see T_f in Figure 13.3. This is probably a consequence of T_f being equally distributed between the front wheels for the double-track models. Thus, when braking and cornering simultaneously, the inner wheels will have less load and thus risk to enter a state of wheel lock for large braking torques. Similarly, during the exit phase of the maneuver where lateral load-transfer still is present, a too large driving torque will result in significant spin on the inner rear wheel. Therefore, a smaller driving torque is applied in the solution obtained for DT, in particular compared with the optimal solutions obtained for the ST-pitch chassis models.

Force-Slip Diagrams The FS-diagrams displayed in Figures 13.5 and 13.6 for the ST WF and for the DT FE model configurations, respectively, provide further information on the maneuver execution. These plots display slightly different slip characteristics for the considered model configurations. For ST, the optimal solution comprises a control strategy such that pure slip is favored, especially for the front wheel. The DT model, having dynamically varying normal forces on the wheels, naturally exhibits different slip trajectories for the left and right wheels. From the FS-diagrams, it is also clear that the slip on the rear wheels is more pronounced as a result of the vehicle being rear-wheel driven.

Optimal Trajectories in the Hairpin Maneuver

In this section, the computed time-optimal maneuvers in the hairpin turn are presented. The analysis of the results for the hairpin maneuver is structured equivalently to the results for the 90°-turn. The start position in the hairpin turn was $(X_{p,0}, Y_{p,0}) = (-5, 0)$ m for all model configurations, and the initial and final heading angles were aligned with the road direction. Figures 13.7–13.9 show the vehicle path and the most relevant model variables for all six model configurations in the hairpin maneuver. The global forces and yaw moment are displayed in Figure 13.10, whereas the FS-diagrams for the DT chassis model with WF and FE tire models are presented in

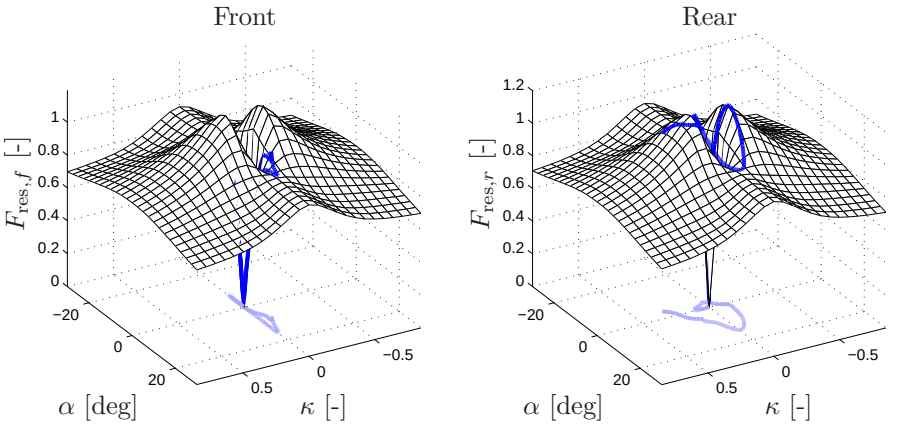

Figure 13.5 Resultant normalized tire forces for ST with WF tire modeling in the 90°-turn.

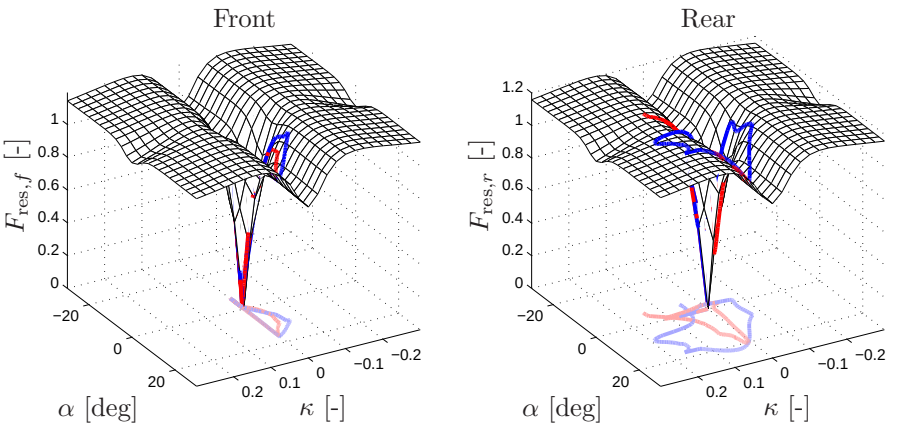

Figure 13.6 Resultant normalized tire forces for DT with FE tire modeling in the 90°-turn (blue – left wheel, red – right wheel).

Table 13.2 Time for completing the maneuver for each model configuration in the hairpin turn.

Model	Maneuver Completion Time
ST FE	8.47 s
ST WF	8.49 s
ST-pitch FE	8.19 s
ST-pitch WF	8.32 s
DT FE	8.48 s
DT WF	8.61 s

Figures 13.11–13.12, respectively. Table 13.2 displays the computed times required for completing the maneuvers for the different model configurations. The objective function t_f deviates approximately 0.4 s when comparing all six model configurations, corresponding to a maximum difference of 5%. The models with roll or pitch dynamics show the largest differences for the respective tire model. This is caused by that these additional dynamic effects, as expected, lead to large wheel-load variations during the maneuver. Similar to the results for the 90°-turn, the models employing FE for tire modeling exhibit shorter maneuver completion times. This is because FE leads to larger resulting forces than WF for combined slip, as seen in Figure 12.3 in Chapter 12. With load transfer, this difference is even more pronounced. The times for completing the maneuver computed with the ST-pitch models are shorter than for the other model configurations. The reason is that ST-pitch, as does DT, benefits from the longitudinal load transfer when accelerating. However, ST-pitch does not take roll dynamics into account, whereas a high cornering velocity will result in lateral load transfer caused by the roll dynamics in DT. This leads to reduced loads on the inner wheels for the double-track models, which is the same phenomenon as observed in the 90°-turn.

Path Trajectories From Figure 13.7, it can be observed that the geometric paths are pairwise equal from a qualitative perspective. For example, the paths for the ST models are comparably symmetric in shape with respect to the center line of the turn. The paths for the ST-pitch and DT models are clearly asymmetric, with the ST-pitch models resulting in a wide exit out of the turn, and opposite for the DT models. The results seem to indicate that the qualitative behavior is more dependent on the chassis model than the tire model. However, considering the similarity of the obtained paths for the FE tire model and the differences in the paths for the WF models (in direct analogy with the results obtained in the 90°-turn), it is clear that the modeling of the tire forces for large combined slip values are important for the optimal geometric path.

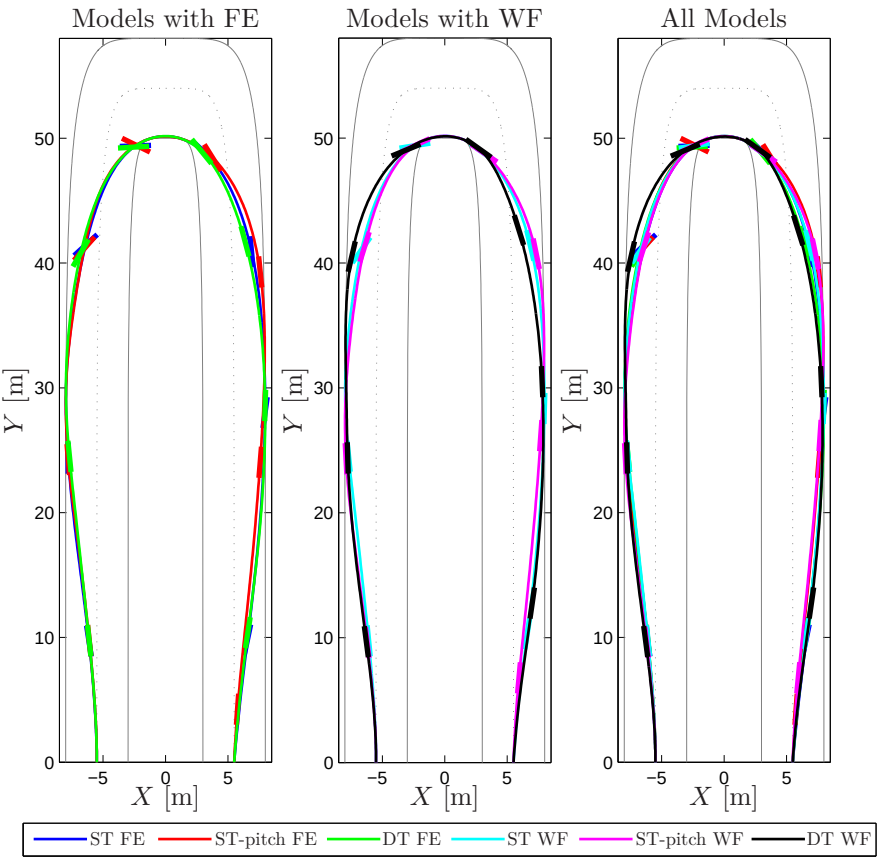

Figure 13.7 Time-optimal path trajectories obtained for the hairpin turn, with the respective model configuration. The colored bars represent the vehicle heading every second.

Trajectories of the Model Variables The trajectories of the model variables in Figures 13.8–13.9 show that all models have similarities in terms of internal variables: The vehicle starts by applying full driving torque simultaneously as it turns in order to allow a wider curve taking. When entering the curve, the vehicle starts to brake with all wheels. This braking is applied approximately until the vehicle reaches the half-way point. Furthermore, all models result in solutions with vehicle slip. The longitudinal forces, and thus also the velocities, are also similar in size and shape. However, there are also fundamental differences between the resulting solutions for the different models. Inspecting the body-slip angle β, it can be concluded that the models with FE have significantly larger slip, when comparing the same chassis

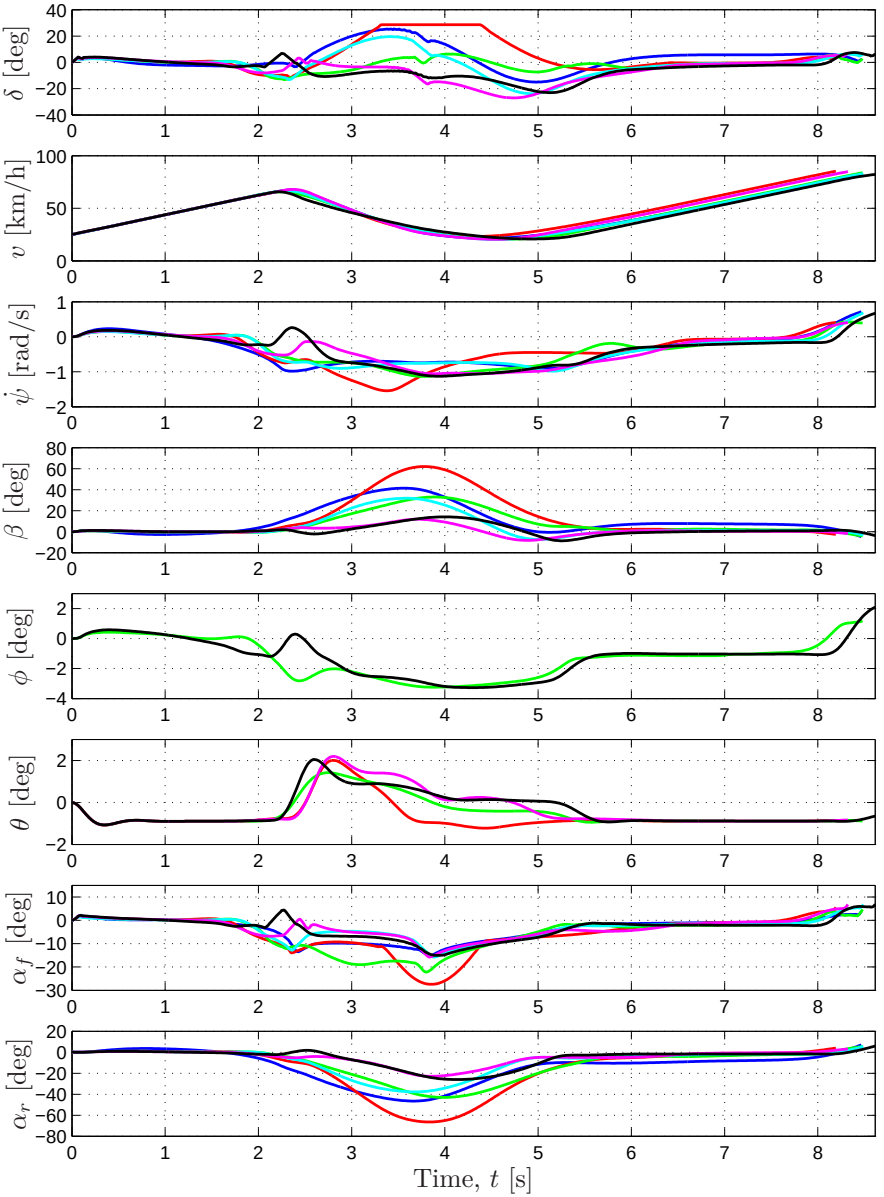

Figure 13.8 Time-optimal solutions obtained for the hairpin turn, with the respective model configuration. For the DT models, the average lateral slip angle for the right and left wheels is shown for each axle in order to enable comparison of the solutions. Same color scheme as in Figure 13.7.

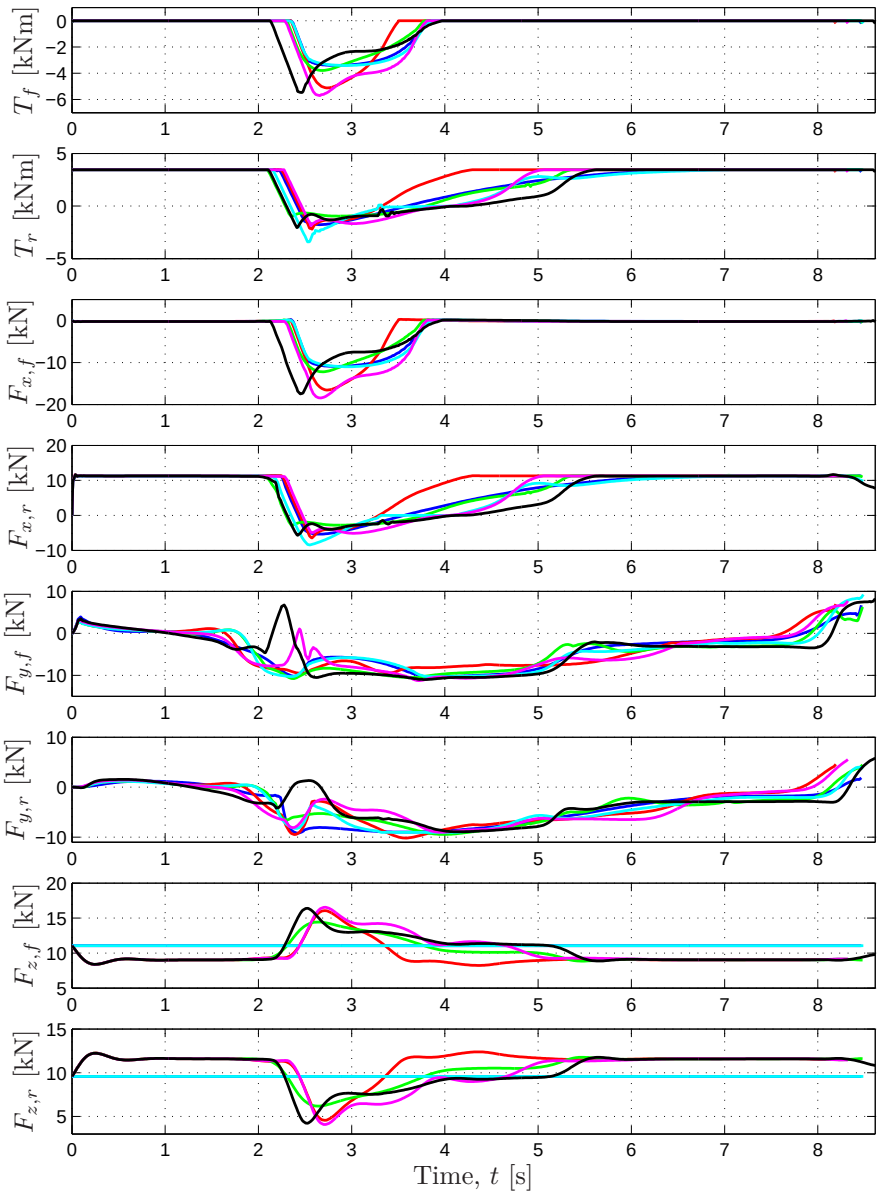

Figure 13.9 Time-optimal solutions obtained for the hairpin turn, with the respective model configuration. For DT, the total forces and torques on each axle are shown. Same color scheme as in Figure 13.7.

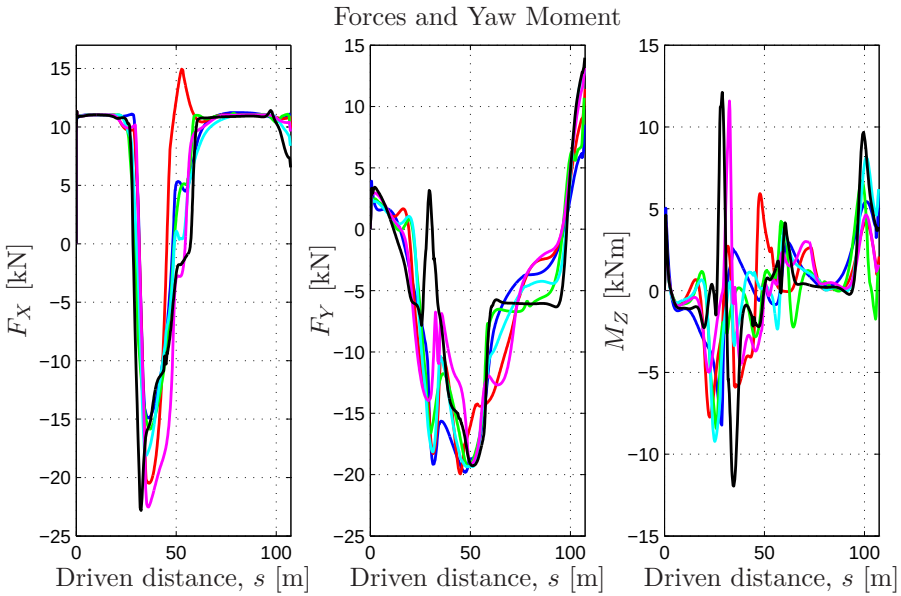

Figure 13.10 Longitudinal force F_X, lateral force F_Y, and yaw moment M_Z, developed by the tires in the hairpin turn, illustrated as functions of the driven distance s. Same color scheme as in Figure 13.7.

models. It can, *e.g.*, be noted that β for the ST-pitch FE model has its maximum value at approximately 60 deg. As mentioned previously, the reason for FE giving larger slip is that the resulting force for FE is higher than the corresponding force for WF when having combined slip. These differences are coupled to the respective steer angles; δ for ST-pitch with FE is at its upper boundary for approximately 1 s. For DT, and to some extent ST-pitch with the WF tire model, there is an abrupt change in δ when reaching the leftmost part of the maneuver, at around 2 s. The tire forces become even smaller for the wheels on the rear axle (especially the rear inner wheel with the DT chassis model) with longitudinal or lateral load transfer, since there will be variations in the normal load on the wheels when cornering or braking. Consequently, the total decelerating force will be smaller. One way to suppress this is to decrease α_f and α_r closer to zero, which can be achieved by the mentioned sudden change in δ. This behavior is not exhibited by the DT or the ST-pitch FE model configurations.

Tire Forces Figure 13.10 shows the total tire-forces F_X and F_Y, resolved in the road-surface plane, as functions of the driven distance. In addition, the yaw moment M_Z generated from the tire forces is visualized. The qualitative behavior of the longitudinal forces exhibits similarities for all models, except

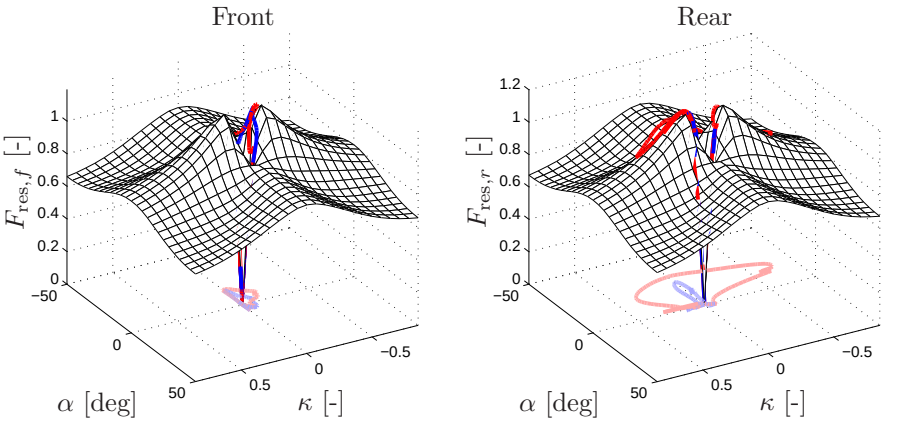

Figure 13.11 Resultant normalized tire forces for DT with WF tire modeling in the hairpin maneuver (blue – left wheel, red – right wheel).

for ST-pitch with FE. The behavior of the lateral forces also exhibits some similarities, even though numerical differences occur. A significant difference is the peak in the lateral force that is visible for DT with WF approximately between $25 < s < 35$ m. This discrepancy is a result of the change in δ as discussed earlier. There are also qualitative discrepancies between the models in M_Z, where ST-pitch WF, DT WF, and ST-pitch FE are significantly different in behavior during the turn part of the maneuver. Note that in this maneuver, the overall behavior for DT FE is similar to ST in all three plots, whereas DT with WF is fundamentally different in behavior during the critical part of the maneuver (*i.e.*, when in the turn).

Force-Slip Diagrams Figures 13.11 and 13.12 show the FS-diagrams for DT with WF and FE, respectively. It is clear that larger longitudinal slip values are obtained with WF, whereas higher lateral slip is achieved with the FE model. This observation is related to that large slip values are required for the WF model in order to obtain small lateral forces. However, for the FE model, longitudinal slip values of 0.1–0.2 are sufficient for this purpose. In addition, since the longitudinal force does not depend on the lateral slip in the adopted FE model, in contrast to the WF model, the observed behavior is expected. With FE, changes in the resulting force seem to have a tendency of moving in orthogonal directions; when κ changes values, the lateral slip α is approximately constant, and vice versa (see Figure 13.12). However, this is partly a result of that the wheel-torque constraints have been reached (especially for negative slip ratios), and is not only a result of the discussed properties of the FE tire model.

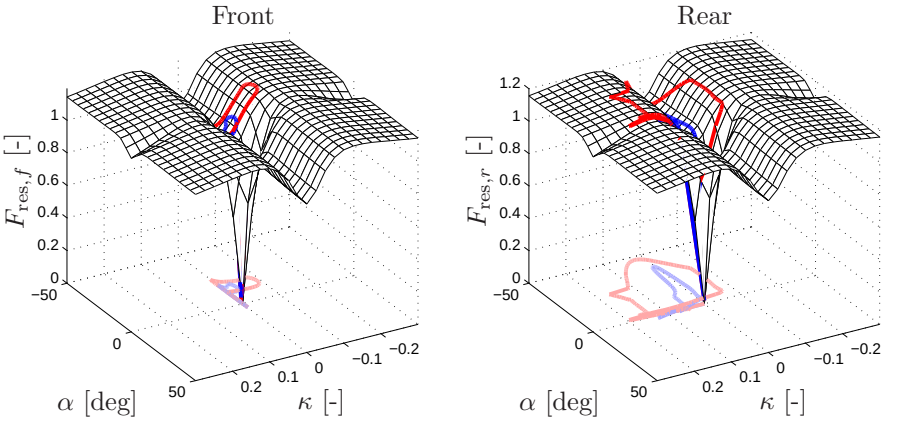

Figure 13.12 Resultant normalized tire forces for DT with FE tire modeling in the hairpin maneuver (blue – left wheel, red – right wheel). In contrast to Figure 13.11, the FE tire model results in smaller longitudinal slip but larger lateral slip. This implies that the desired forces are achieved by sliding rather than spinning or locking the wheels.

Discussion

The time-optimal trajectories in three different time-critical maneuvers were computed for several configurations of chassis and tire models. Two of them, the 90°-turn and the hairpin turn, were investigated in detail in this section. The double lane-change maneuver provided results that lead to similar conclusions as those for the two reported maneuvers, and this is an important verification that the method and numerical tools discussed in Chapter 12 can handle a variety of situations relevant for vehicle-safety systems. Moreover, as, *e.g.*, the results for the solution times in Table 12.1 in Chapter 12 show, the computed solutions give valuable insight into the balance between model detail and computational complexity, which is important for developments toward online solutions for optimal driver-assistance systems.

The obtained results provided a solid basis for discussion of the behavior of different vehicle models in time-critical situations. Several interesting properties were found: The differences in behavior for M_Z between the investigated models is interesting because this variable is often used as a high-level input in safety systems, such as in yaw-rate controllers and rollover-prevention systems [Tøndel and Johansen, 2005; Tjønnås and Johansen, 2010; Berntorp, 2008; Schofield, 2008]. Thus, it seems that the choice of models can potentially lead to fundamentally different control strategies, where, *e.g.*, whether to use WF instead of FE with the DT chassis model seems crucial. The characteristics of the two tire models are fundamentally different. The

WF model results in much smaller forces for combined slip compared to the case when only one of the slip quantities is nonzero. For the FE model, however, the largest forces are attained for combined slip. It is hard to conclude which of the models that are most suitable to use for trajectory generation in extreme maneuvering. The WF model has been experimentally verified for normal driving conditions on specific road surfaces. However, for large combined slip, it is difficult to explain why the resulting force should be significantly smaller than for pure longitudinal or lateral slip of similar magnitude. Similarly, it may not be reasonable that the largest forces should be achieved for combined slip, such as the FE model predicts. This is a result of that the longitudinal force is not affected by the lateral slip. Regarding the chassis models, the employment of ST or DT seems to have impact on the employed control strategy. The intuitively obvious choice would be to use DT, because it is a more advanced model. However, the optimal solutions that are obtained indicate that the models with load transfer are highly dependent on the choice of tire model, where the different characteristics for combined slip have large impact on the solutions. Thus, it is possible that certain combinations of chassis and tire models are inappropriate and lead to nonphysical behavior, as discussed in [Sharp and Peng, 2011] and stated in Section 12.1 in Chapter 12. Based on the optimal solutions computed for the ST-pitch chassis models, this model is not included in the following discussion about the implications in practice.

Considering that the computed time-optimal maneuvers result in at-the-limit behavior of the vehicle, it can be argued that the observations made for the results presented in this section have potentially important implications for future safety systems. The resulting behaviors in the optimal solutions are similar in several key aspects for both maneuvers, as observed in Figures 13.2–13.3 and Figures 13.8–13.9—*e.g.*, variables traditionally used for detecting loss of maneuvering stability, such as the yaw rate, the slip angle, and the roll angle, only show minor discrepancies. In contrast, the input torques differ significantly during parts of the maneuver. In the 90°-turn, however, the overall lateral forces and yaw moments generated by the tires (see Figure 13.4) for the considered models have similar characteristics from a qualitative perspective, but with numerical differences in between for certain model configurations. The discrepancies do not seem to have much impact on the other model variables, though. Moreover, model parameters such as the road-friction coefficients, vehicle mass, and tire parameters are uncertain. Hence, safety bounds on supervisory variables have to be set conservatively, for example resulting in early braking in order to for sure avoid impact, and the model deviations occurring for low-order models can thus be suppressed in an online implementation with feedback. The observations are important, because they imply that variables traditionally considered as high-level inputs in safety systems, such as M_Z, may be generated by optimization using

models of low complexity (such as the ST chassis model). These high-level inputs can then be utilized as inputs to a low-level optimizer that benefit more from complex road interaction models for distributing the desired torque to the respective wheel. This observation, together with the increased amount of sensor data and computational power available in modern road vehicles, opens up for the use of simplistic models when designing the future online optimization-based safety systems.

13.3 Optimal Maneuvers under Different Road Conditions

In this section, the method in Chapter 12 is applied to a case where the road condition is uncertain—*i.e.*, the road is potentially covered with snow or ice. The road surface has, naturally, significant influence on how large tire forces that can be realized, and hence on the optimal control strategy for the vehicle in a maneuver. Scaling of nominal tire-force models for describing the behavior on different surfaces is described in [Pacejka, 2006] and was experimentally measured and verified in [Braghin et al., 2006]. Comparisons of optimal maneuvers for different road conditions have been made, see, *e.g.*, [Chakraborty et al., 2011], where the friction coefficient in the tire-force model was varied and the resulting optimal solutions were computed. The more extensive results presented in this section are based on [Olofsson et al., 2013] and investigate the optimal vehicle maneuvers under different road conditions using a scaling of the complete tire model. A similar study with comprehensive tire modeling was independently presented in [Tavernini et al., 2013] for a different set of surfaces. In this section, a comparison between the optimal maneuvers obtained by varying the friction coefficient only and using the complete scaling proposed in [Pacejka, 2006] is also presented. Related results for a more extensive set of chassis models under different road conditions are also presented in the thesis [Berntorp, 2014].

One step further toward increasing the performance of vehicle-safety systems is to study the behavior of a vehicle in a time-critical maneuver under varying road-surface conditions, *e.g.*, dry asphalt, wet asphalt, ice, and snow. Therefore, an investigation of the time-optimal vehicle maneuver under different road conditions in the hairpin turn, also considered in the previous section, was performed. The time-optimal control problem defined in the previous chapter is solved and the results thereof are presented for the different investigated road conditions. With this study, it is plausible that the understanding of vehicle dynamics in at-the-limit maneuvering situations under road-surface uncertainties can be increased and that the results can be used as a basis for increasing the robustness of future vehicle-safety systems.

Modeling

For modeling of the vehicle chassis and tires, the principles developed in Chapter 12 are employed. Since the influence of the road condition is on the interaction between the tire and the road, the focus of the modeling is on the tire-force aspects. The vehicle chassis dynamics is modeled with a single-track model with roll dynamics, together with a wheel model and a tire model based on the Magic-Formula with weighting functions for describing combined longitudinal and lateral slip behavior [Pacejka, 2006].

Vehicle Modeling The ST chassis model is used when computing the time-optimal maneuvers, see Figure 12.2 in Chapter 12. A rotational DoF about the x-axis—*i.e.*, the roll dynamics—has been included in the model. The coordinate system is located in the ground plane at the xy-coordinates of the center-of-mass for zero roll angle. The roll dynamics has been included in the chassis model in order to verify that the vehicle is not overbalancing in the aggressive hairpin maneuver on high-friction surfaces. With the same notation as in Chapter 12, the dynamic equations for the chassis model are

$$m\dot{v}_x = F_X + mv_y\dot{\psi} - mh\sin(\phi)\ddot{\psi} - 2mh\cos(\phi)\dot{\phi}\dot{\psi}, \tag{13.4}$$

$$m\dot{v}_y = F_Y - mv_x\dot{\psi} - mh\sin(\phi)\dot{\psi}^2 + mh\ddot{\phi}\cos(\phi) - m\dot{\phi}^2 h\sin(\phi), \tag{13.5}$$

$$\ddot{\psi} = \frac{M_Z - F_X h\sin(\phi)}{I_{zz}\cos^2(\phi) + I_{yy}\sin^2(\phi)}, \tag{13.6}$$

$$\begin{aligned} I_{xx}\ddot{\phi} = {} & F_Y h\cos(\phi) + mgh\sin(\phi) + \dot{\psi}^2\Delta I_{yz}\sin(\phi)\cos(\phi) \\ & - K_\phi\phi - D_\phi\dot{\phi}, \end{aligned} \tag{13.7}$$

$$F_X = F_{x,f}\cos(\delta) + F_{x,r} - F_{y,f}\sin(\delta), \tag{13.8}$$

$$F_Y = F_{y,f}\cos(\delta) + F_{y,r} + F_{x,f}\sin(\delta), \tag{13.9}$$

$$M_Z = l_f F_{y,f}\cos(\delta) - l_r F_{y,r} + l_f F_{x,f}\sin(\delta), \tag{13.10}$$

where $\Delta I_{yz} = I_{yy} - I_{zz}$.

Wheel and Tire Modeling The wheel dynamics presented in Chapter 12 is adopted in this section for formulation of the optimal control problem. However, the relaxation length σ is neglected in the investigation presented in this section for simplicity. The vehicle and wheel parameters are given in Table A.1 in Appendix A. The tire forces are modeled using the WF model. Considering the fact that the ST model with lumped wheels are utilized here, the asymmetric parameters in the original complete tire model in [Pacejka, 2006] have been recomputed such that the resulting model in (12.13)–(12.14) and (12.16)–(12.21) in Chapter 12 is symmetric with respect to the slip angle α and slip ratio κ.

Tire-Force Model Calibration It is clear that the road condition has a fundamental impact on the tire forces that can be realized. Based on a set of

tire parameters for a nominal surface, [Pacejka, 2006] proposes to use scaling factors λ_i in (12.13)–(12.14) and (12.16)–(12.21) in Chapter 12 in order to describe different road conditions. This method was used in [Braghin et al., 2006], where the scaling factors representing surfaces corresponding to dry asphalt, wet asphalt, snow, and smooth ice were estimated based on experimental data. In the investigation performed in this section, the scaling factors experimentally determined in [Braghin et al., 2006] are adopted as a basis for calibrating tire models approximately corresponding to the tire forces on the different surfaces. The empirical tire-model parameters presented in [Pacejka, 2006] are used to represent dry asphalt. With the dry-asphalt model as base model, the scaling factors are introduced according to

$$\lambda_{\text{dry}} = 1, \quad \lambda_{\text{wet}} = \frac{\lambda^*_{\text{wet}}}{\lambda^*_{\text{dry}}}, \quad \lambda_{\text{snow}} = \frac{\lambda^*_{\text{snow}}}{\lambda^*_{\text{dry}}}, \quad \lambda_{\text{ice}} = \frac{\lambda^*_{\text{ice}}}{\lambda^*_{\text{dry}}}, \tag{13.11}$$

where λ is the scaling factor used in this study and λ^* is the scaling factor presented in [Braghin et al., 2006]. Considering that a different set of nominal parameters is used in the study in this section, and that there are uncertainties in the estimation of the original scaling factors based on experimental data in [Braghin et al., 2006]—especially for larger slip values, where the experimental conditions are challenging—some inconsistent characteristics appear for the snow and ice models. The original tire model on the snow surface predicts a longitudinal force F_x that changes sign for increasing slip ratios, which is avoided by slightly adjusting the scaling factor for C_x. For the model representing the ice surface, multiple sharp and narrow peaks in the resultant force occurred. This was adjusted by recomputing the scaling factor influencing the relation (12.19) in Chapter 12, as well as the parameters B_{x2} and B_{y2}. In addition, the lateral curvature factor E_y was adjusted to smoothen the sharp peak, originating from the relations (12.13)–(12.14) in Chapter 12. The complete set of tire-model parameters used to represent the different surfaces are provided in Table A.3 in Appendix A. A subset of these parameters has been calculated with expressions depending on the normal force F_z on the wheel. Consequently, the front and rear parameter values differ—*e.g.*, the friction coefficients $\mu_{x,f}$ and $\mu_{x,r}$—because of the asymmetric location of the center-of-mass of the vehicle chassis. The resulting tire forces for the respective road surface are visualized in Figures 13.16–13.19 later in this section.

Results

To the purpose of investigating the time-optimal maneuver in the hairpin turn for the different road conditions, the optimal control problem (12.23) defined in Chapter 12 was solved for each of the surface models discussed. The road in the current study is 5 m wide and the vehicle is rear-wheel driven.

The limitations on the driving and braking torques and the tire forces were chosen as follows:

$$T_{f,\max} = 0, \tag{13.12}$$
$$T_{f,\min} = -\mu_x F_{z,f} R_w, \tag{13.13}$$
$$T_{r,\max} = \mu_x F_{z,r} R_w, \tag{13.14}$$
$$T_{r,\min} = -\mu_x F_{z,r} R_w, \tag{13.15}$$
$$F_{x,i,\max} = \mu_x F_{z,i}, \tag{13.16}$$
$$F_{y,i,\max} = \mu_y F_{z,i}, \ i \in \{f, r\}. \tag{13.17}$$

With this choice of the maximum driving and braking torques, a dependency on the studied surface is introduced *via* the friction coefficients. This is justified by the fact that the surface models adopted in this study are only identified and validated for a certain region in the plane spanned by the slip quantities κ and α. Hence, allowing excessive input torques may result in inconsistent behavior of the tire-force model. The steering angle and steering-angle rate were constrained according to

$$\delta_{\max} = 30 \text{ deg}, \quad \dot{\delta}_{\max} = 60 \text{ deg/s}, \tag{13.18}$$

similarly as for the maneuvers investigated in Section 13.2. Also in these optimizations, the wheel angular velocities ω_f and ω_r were constrained to be nonnegative—*i.e.*, the wheels were not allowed to roll backwards or back-spin in the maneuver.

With an initial velocity of $v_0 = 25$ km/h for the vehicle, the results displayed in Figures 13.13–13.14 were obtained. For comparison of the different road surfaces, the model variables are visualized as function of the driven distance s instead of time. With this plotting layout, the time dependency of the variables is implicit, but the relations of the control inputs the different parts of the maneuver is more accentuated. Moreover, the path trajectories in the XY-plane corresponding to the time-optimal control strategies are presented in Figure 13.15. The FS-diagrams with the optimal slip trajectories are presented in Figures 13.16–13.19. The computed times for completing the maneuver are 8.48 s, 8.79 s, 13.83 s, and 19.18 s for dry asphalt, wet asphalt, snow, and smooth ice, respectively.

Discussion on Characteristics under Different Road Conditions

Considering the path trajectories in Figure 13.15 for the optimal maneuvers on the different surfaces, it can be concluded that the overall behavior is similar. As expected, the time for completing the maneuver is longer on the snow and ice surfaces than on asphalt. This is a result of that the tire forces that can be realized on these surfaces are significantly lower in magnitude

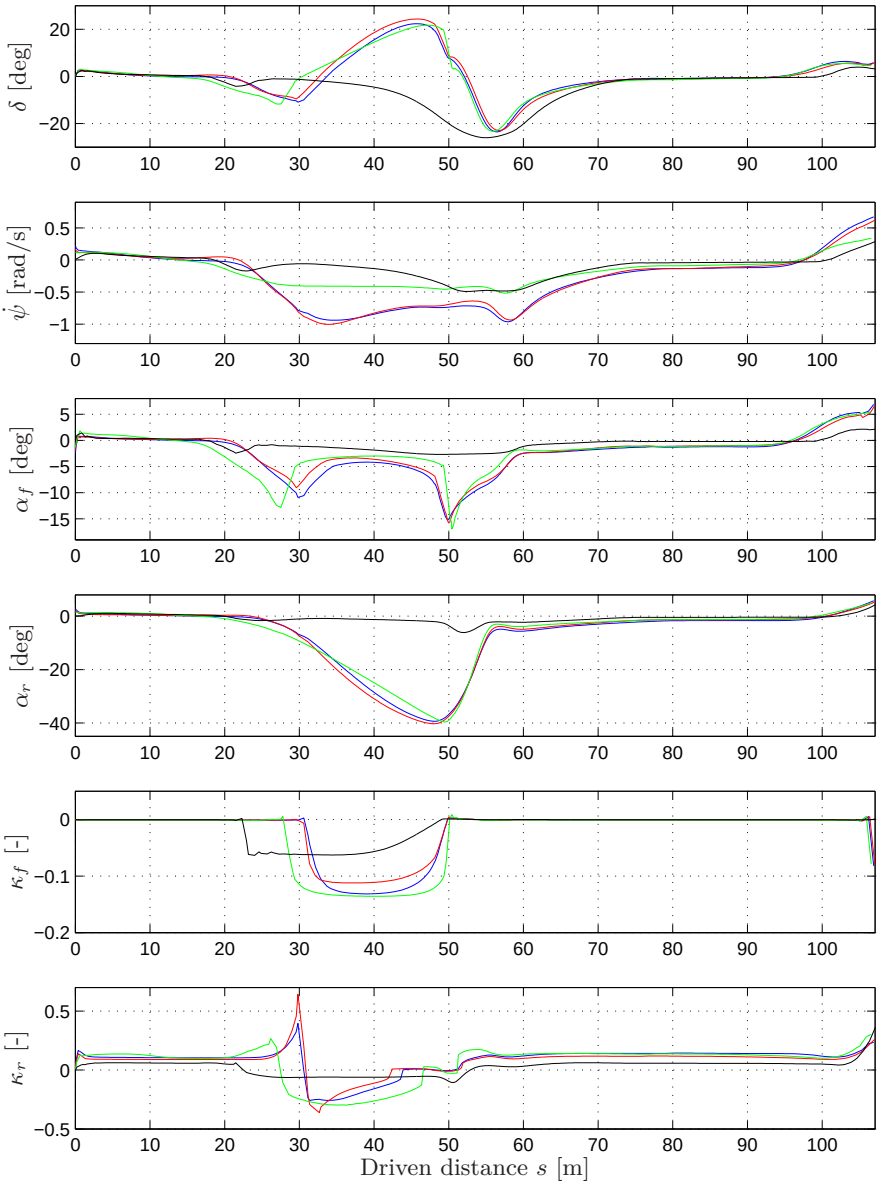

Figure 13.13 A subset of the model variables during the time-optimal hairpin maneuver on the different surfaces, plotted as a function of the driven distance s. The color scheme is as follows: dry asphalt–blue, wet asphalt–red, snow–green, and smooth ice–black.

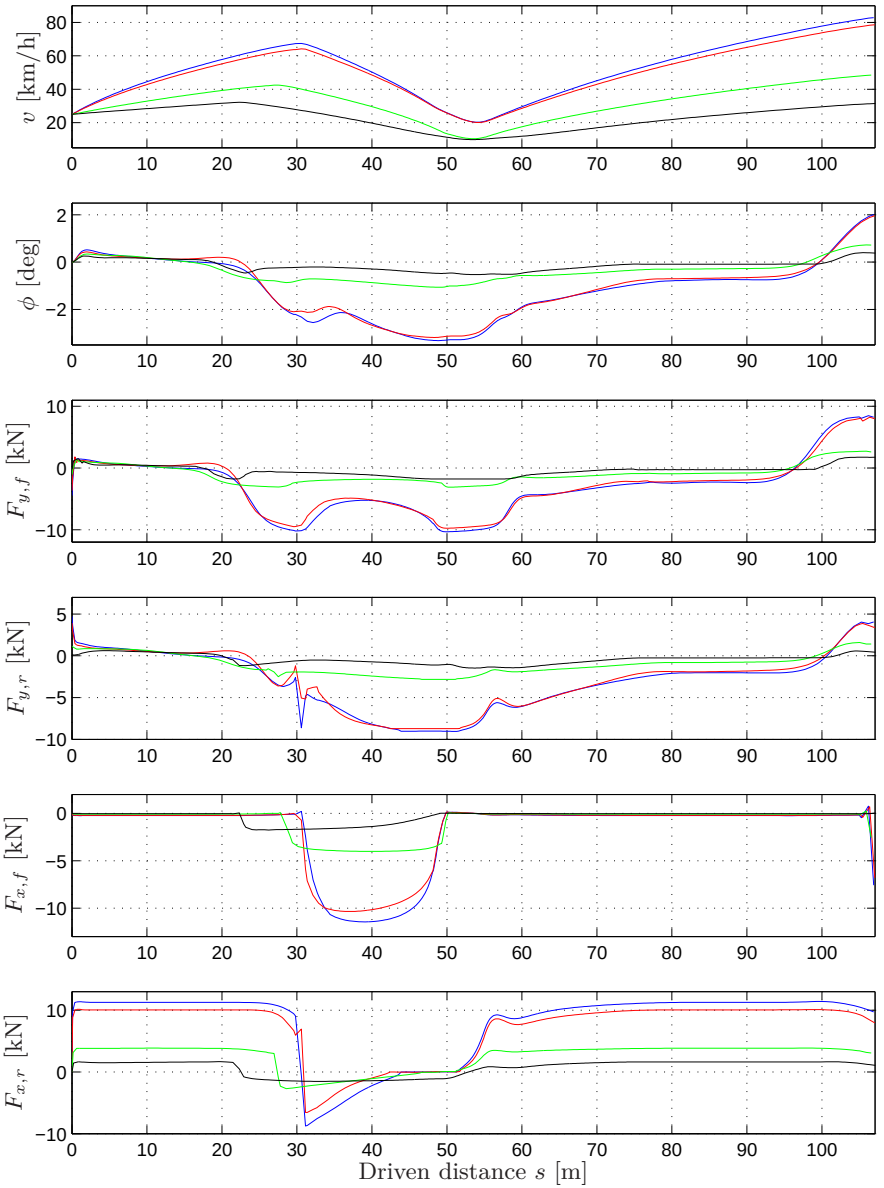

Figure 13.14 A subset of the model variables during the time-optimal hairpin maneuver on the different surfaces, plotted as a function of the driven distance s. Same color scheme as in Figure 13.13.

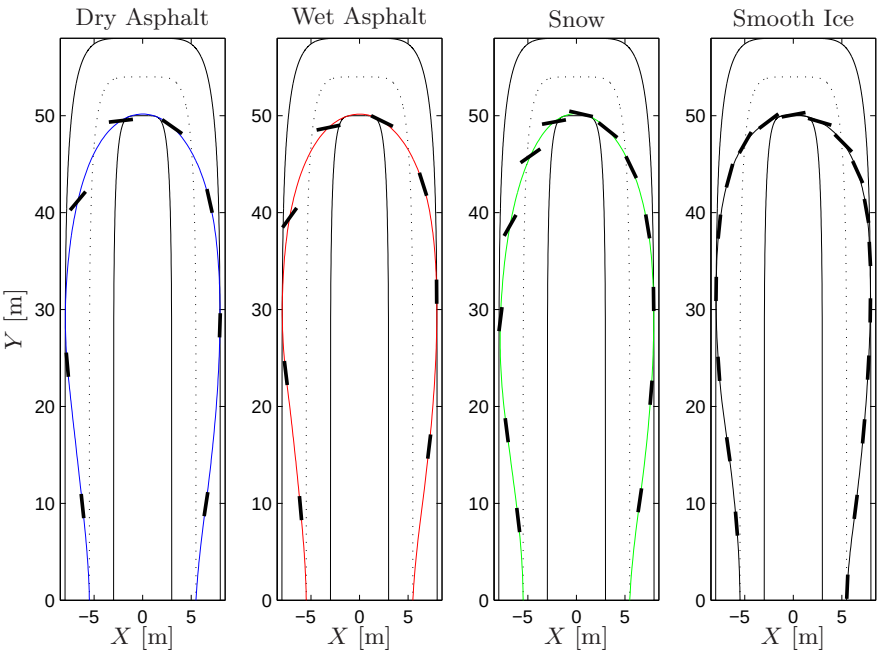

Figure 13.15 Trajectories in the XY-plane for the different road surfaces. The black rectangles indicate the position and direction of the vehicle each second of the maneuver. The increase in time for completing the maneuver on low-friction surfaces is clearly visible.

than those on asphalt, *cf.* the friction coefficients for the different models in Table A.3 in Appendix A. Further, the vehicle exhibits high slip in the critical part of the maneuver on all surfaces except smooth ice. The explanation for this can be derived by examining the force characteristics of the model for smooth ice compared to, *e.g.*, the model for dry asphalt. In Figures 13.20 and 13.21 the longitudinal and lateral tire forces are shown for these surfaces, *cf.* the resulting tire forces in Figures 13.16 and 13.19, respectively. It is observed that the tire forces for smooth ice exhibit a considerably sharper peak and thus decay faster, with respect to combined longitudinal and lateral slip, than for dry asphalt. This means that combined slip yields a significantly smaller resulting tire force. Consequently, to achieve the desired time-optimality on the ice surface, it is natural that a small-slip control strategy is chosen. This is also natural from an intuitive argument; on the low-friction surfaces, maneuvering of the vehicle is particularly critical and both the velocity and body slip must be reduced in order to be able to fulfill the geometric path constraint.

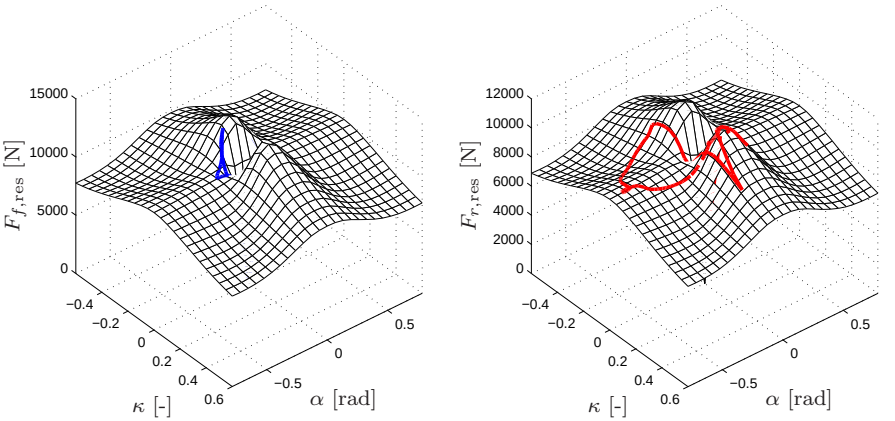

Figure 13.16 The resulting tire forces for the model corresponding to dry asphalt in the time-optimal hairpin maneuver. The front tire force is shown in blue and the rear tire force is shown in red. The rear tire force exhibits more variation, which is a result of that the vehicle is rear-wheel driven.

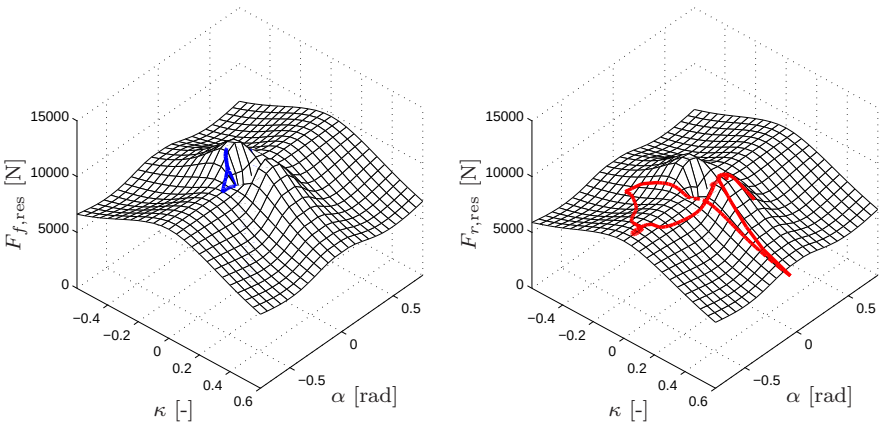

Figure 13.17 The resulting tire forces for the model corresponding to wet asphalt in the time-optimal hairpin maneuver. The front tire force is shown in blue and the rear tire force is shown in red.

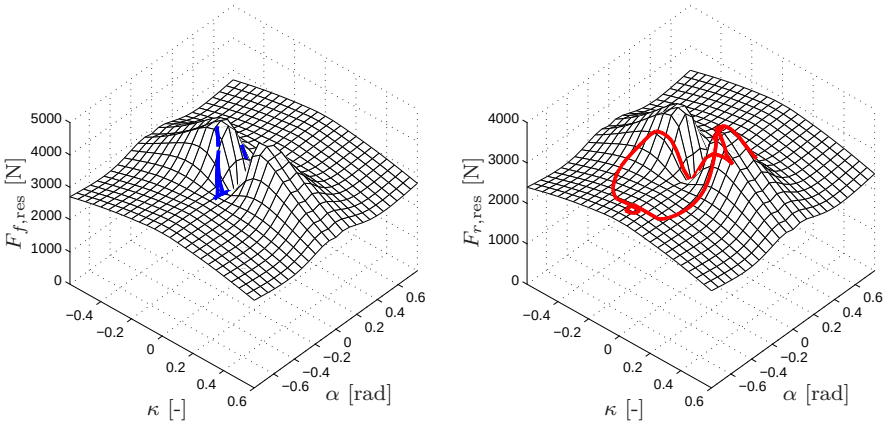

Figure 13.18 The resulting tire forces for the model corresponding to snow in the time-optimal hairpin maneuver. The front tire force is shown in blue and the rear tire force is shown in red.

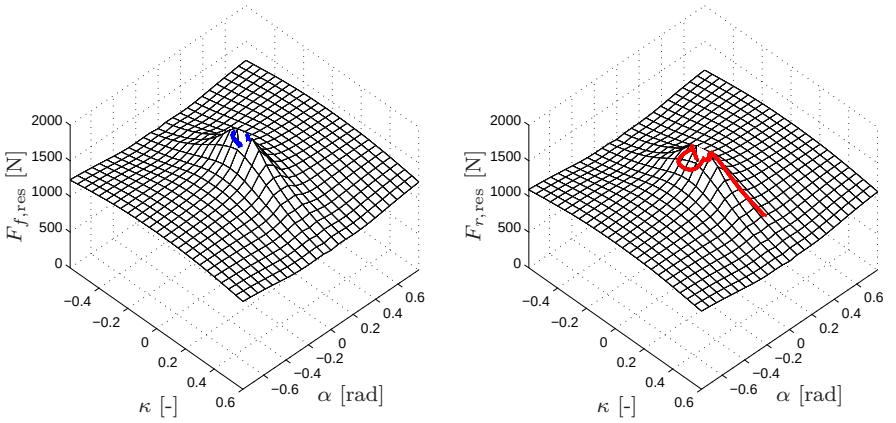

Figure 13.19 The resulting tire forces for the model corresponding to smooth ice in the time-optimal hairpin maneuver. The front tire force is shown in blue and the rear tire force is shown in red.

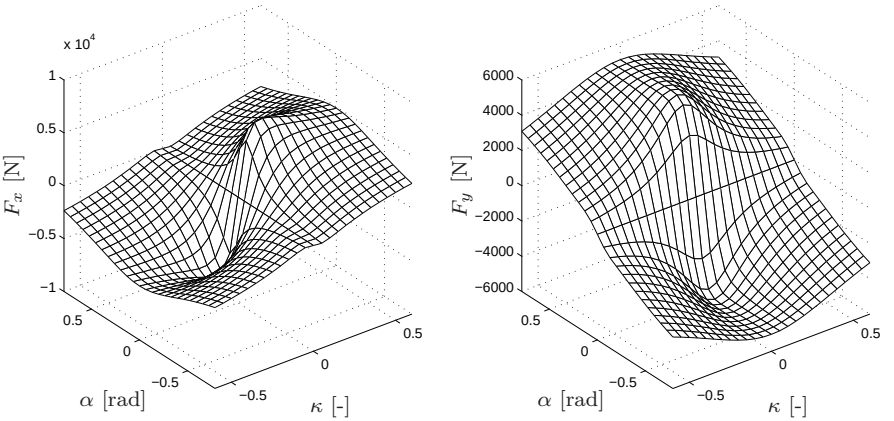

Figure 13.20 Front tire forces in the longitudinal and lateral wheel directions for dry asphalt, corresponding to Figure 13.16.

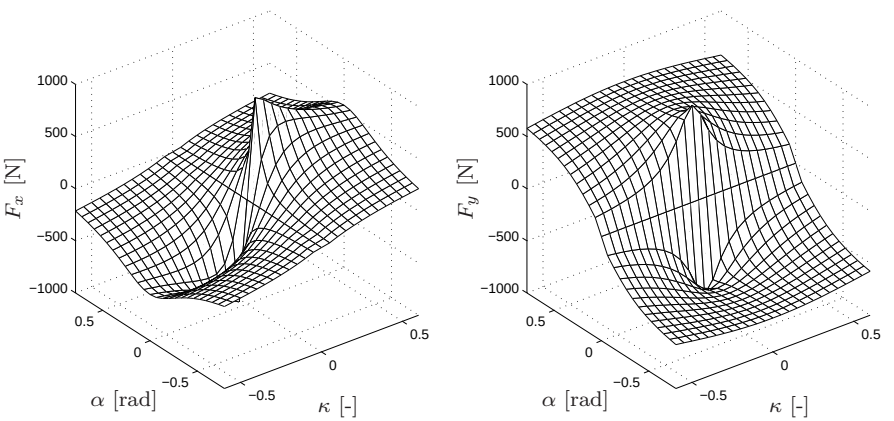

Figure 13.21 Front tire forces in the longitudinal and lateral wheel directions for ice, corresponding to Figure 13.19.

Comparison of Control Strategies Investigating the internal variables of the vehicle chassis and tire models during the maneuver closer, see Figures 13.13–13.14, it is clear that the behavior for dry and wet asphalt is similar. The similarity between these surfaces is expected, considering the tire-force characteristics, see Figure 13.16 and Figure 13.17, which exhibit related behavior in the two cases. The major difference between them is the time for completion of the maneuver, which is slightly longer for the wet asphalt surface. This is expected since the friction coefficients—and thus the maximum tire forces that can be realized—are lower than for dry asphalt.

The discrepancies between asphalt, snow, and ice when considering the fundamental control strategy are more pronounced. First, it can be concluded that the optimal maneuver on the snow and ice surfaces are more proactive in the sense that both the steering angle δ and braking forces are applied considerably earlier when approaching the hairpin. This is most certainly an effect of the significantly reduced tire forces that can be realized on these surfaces compared to asphalt. Hence, the deceleration phase must be initiated earlier on the low-friction surfaces. The steering angle also differs between ice and the other surfaces. The reason for this is that the vehicle employs counter-steering when it starts to slip on asphalt and snow as it approaches the hairpin. This is not the case on the ice surface, where the slip is much smaller and the steering angle is thus not utilized to the same extent. Moreover, the roll angle is smaller for the low-friction surfaces, which is a consequence of that the torque about the roll axis arising as a result of the contact between the tire and road is smaller. Further, even on dry asphalt, the roll angle is kept below approximately 3.2 deg, verifying that the chassis motion is not too aggressive during the turn part of the maneuver. The slip ratio κ differs in amplitude between the surfaces, which is related to the differences in the strategies during the acceleration and deceleration phases of the maneuver. More specifically, the reason becomes clear when investigating the tire forces in the FS-diagrams and the corresponding tire utilization, displayed in Figures 13.16–13.19. The peak of the resulting tire force in the plane spanned by the slip quantities κ and α occurs at smaller slip values when changing from surfaces with high friction to surfaces with low friction. Consequently, in order to perform the maneuver in minimum time, it is natural that a control solution with smaller slip angles is chosen.

Influence of Tire Model on Optimal Maneuver Different approaches to tire-model calibration were investigated prior to performing the study presented in this section. One approach would be to only scale the friction coefficients μ_x and μ_y, as mentioned in Section 13.1 and previously investigated in the literature. However, studying the force characteristics in Figures 13.16–13.19 closer, it can be observed that the maxima and minima in the resulting tire-force surfaces occur at different combinations of lateral and

longitudinal slip. Also, the sharpness and width of the maxima and minima change for the different surface models. Consequently, only changing the friction coefficients will result in different force characteristics compared to when changing the complete set of tire-model parameters. This implies that it is plausible that the resulting optimal maneuver will be different if the latter tire-modeling approach is chosen. This was verified by constructing a tire-force model where the model corresponding to dry asphalt was used together with the friction coefficients corresponding to ice—*i.e.*, all parameters in the constructed model were equal to those in the model for dry asphalt, except the friction coefficients. Solving the time-optimal control problem for the constructed model, see Figure 13.22, shows that the optimal solution has significant slip, on the contrary to the results obtained for the empirical model corresponding to smooth ice.

13.4 Comparison of Optimal Maneuvers for ESC Systems

New possibilities for ESC [Isermann, 2006; Bosch, 2011; Rajamani, 2006] for vehicles are now being developed, and the reason is increased situation awareness as a result of additional information from external sensors. Examples here include cameras, radar, and satellite positioning systems. This fact, in combination with the availability of individual braking of each wheel in personal vehicles, constitute the foundation for a spectrum of new systems, *e.g.*, collision avoidance or collision mitigation [Chakraborty et al., 2013].

One highly critical situation is if a vehicle leaves its lane, *i.e.*, enters the lane of opposing traffic or exits the road. Today, lane-departure warning systems that are aware of such situations and alert the driver in case of danger are available. This information is also deployed in lane-keeping systems, which during normal driving actively corrects a driver drifting out of the road with superimposed control [Ali, 2012]. A next step is to utilize this information in more severe situations, such as previously performed in, *e.g.*, [Ali et al., 2013], leading to systems here referred to as lane-keeping ESC. A number of interesting questions immediately arise, both around what the resulting vehicle behavior would be and how significant it would be in terms of increased safety for the driver and the passengers. In this section, the situation of over-speeding when entering a curve with a vehicle is considered. The specific aim is to quantify the possible performance gains in terms of the highest possible entry speed while staying in lane, when also trying to maintain vehicle controllability.

A study closely related to the one performed in this section was presented in [Ali et al., 2013], where a lane-keeping ESC is compared to traditional ESC. The system proposed in that reference consists of an MPC controller, essentially preventing the vehicle from entering the critical operating regions

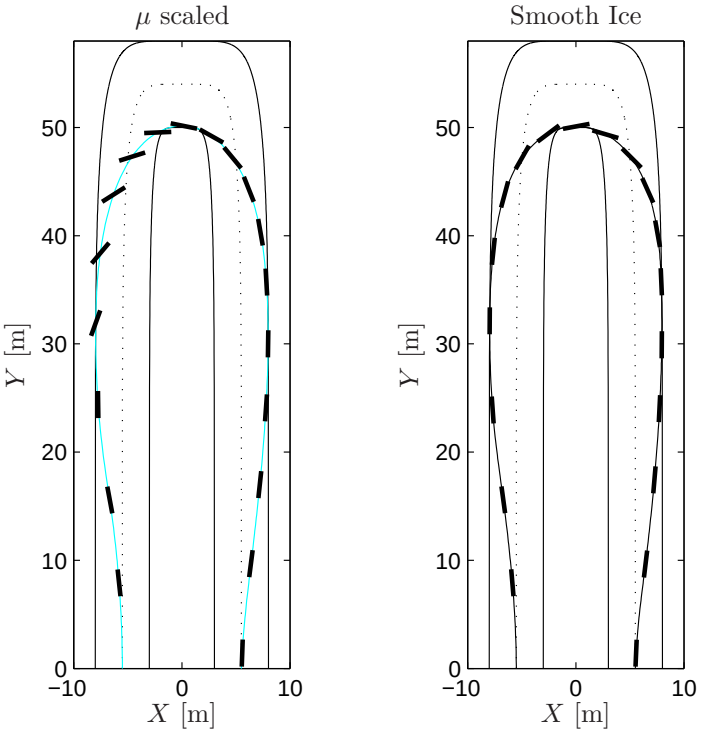

Figure 13.22 Results obtained by using the tire parameters for dry asphalt with the friction coefficients for the model corresponding to ice (left plot). Comparing with scaling all tire parameters as in the empirical model for ice (right plot), the time-optimal solution has significant slip for the constructed model (left plot).

when initiating a corner with excessive speed. In a similar context, optimal behavior in a curve have been investigated using a similar setup as used in the study in this section, see [Sundström et al., 2010] and [Andreasson, 2009]. These references compared optimal solutions for different actuator configurations of the vehicle. Compared to their studies, the modeling employed in this section includes wheel dynamics and slightly more advanced tire-force modeling. The results presented in this section relate to the investigations in [Sundström et al., 2010] and [Andreasson, 2009], but focus on the quantification of the significance of different behaviors compared to traditional control strategies such as yaw-moment regulation. The problem of lane keeping in this section is formulated as an optimization problem using the method outlined in Chapter 12 for each of the considered ESC configurations, with the

road borders as boundary conditions in the optimization. The major questions to be analyzed in the results are whether braking all wheels and then steer should be prioritized, or if braking wheels on one side to create a turning moment inducing early rotation of the vehicle is more beneficial.

The two cases of vehicles equipped with different braking-control systems in the lane-keeping scenario considered here are:

- Case 1: A vehicle with wheel braking on only the left side in a left-hand turn;
- Case 2: A vehicle with wheel braking on all wheels in a left-hand turn.

Case 1 is to resemble the characteristics of the traditional ESC, with focus on creating a yaw-moment around the vertical axis of the vehicle for stabilization. Given a desired moment, the optimal braking strategy is to apply braking on both wheels at one side, distributed between front and rear depending on the current circumstances [Tøndel and Johansen, 2005]. To investigate the full potential of the control strategy in Case 1, the best possible steering input together with an optimal braking distribution of the wheels on one side is computed by solving an optimal control problem. The left-hand turn is initiated with excessive speed. In this situation, the vehicle is subject to under-steering, and a counterclockwise yaw moment is therefore desired by the control system. To achieve this with an optimal braking distribution, the wheels on the left side are used in Case 1. It is clear that such a strategy would tend to overestimate the best behavior achievable with pure yaw-moment control, since the driver influence is neglected and the braking and steering typically are coupled. The second case considered in the comparison, Case 2, is a vehicle able to brake all wheels individually and to steer optimally. It is expected that such a vehicle should perform better, but the interesting question to investigate is if there are any principle differences in behavior, and the maximum velocities that can be handled when over-speeding in the curve. Regarding behavior, the question is whether the vehicle makes a different trade-off between yaw rate, side slip, and lane keeping. The fundamental aspect here to investigate is if the different strategies exhibit any significant improvements with respect to safety.

Optimal Control Problem

The DT chassis model and the WF tire model from Chapter 12 were used in the present study of optimal maneuvers for the vehicles considered in Case 1 and Case 2. The dynamics of the vehicle is given by $\dot{x}(t) = G(x(t), y(t), u(t))$, where the wheel-torque control signals, $T_u = \begin{pmatrix} T_{u,1} & T_{u,2} & T_{u,3} & T_{u,4} \end{pmatrix}$, as well as the steer angle δ of the front wheels are considered as inputs in the optimal control problem. The wheel-torque control signals T_u are related to the wheel torques T *via* a first-order dynamic filter, to model the motor

dynamics. For simplicity, it is assumed that the front wheels have the same steer angle. In Case 1, it is required that $T_{u,2} = T_{u,4} = 0$. Further, the WF tire-force model is formulated as $h(x, y, u) = 0$. The optimization problem is defined over the time horizon $t \in [0, t_f]$, where the upper limit t_f on the time interval is free in the optimization. Similarly to [Sundström et al., 2010], the optimization objective is to maximize the initial velocity v_0 when entering the curve. However, depending on the road geometry and the surroundings, it is sometimes not enough just to stay on the road—*e.g.*, when it leads to highly reduced vehicle controllability potentially resulting in subsequent dangerous situations. Thus, the aim is also to avoid large body slip β. This trade-off is parametrized by using a weight η in the objective function. Accordingly, the dynamic optimization problem to be solved is written as:

$$\begin{aligned}
\text{minimize} \quad & -v_0 + \eta \int_0^{t_f} \beta^2 \mathrm{d}t \\
\text{subject to} \quad & T_{u,i,\min} \leq T_{u,i} \leq T_{u,i,\max}, \ i \in \{1,2,3,4\}, \\
& |\delta| \leq \delta_{\max}, \quad |\dot{\delta}| \leq \dot{\delta}_{\max}, \\
& x(0) = x_0, \quad x(t_f) = x_{t_f}, \\
& f(X_p, Y_p) \leq 0, \\
& \dot{x} = G(x, y, u), \quad h(x, y, u) = 0,
\end{aligned} \tag{13.19}$$

The function $f(X_p, Y_p)$ defining the road constraint for the center-of-mass (X_p, Y_p) of the vehicle in the curve maneuver, is formulated as two circles with different radii in the XY-plane. Since the primary objective of a safety system is to capture the first and most critical part of the situation, the terminal constraint is formulated as

$$\dot{e}(t_f) = 0,$$

where e is the lateral deviation from the middle of the road, defined as

$$e = \sqrt{X_p^2 + Y_p^2} - R, \tag{13.20}$$

and R is the mean radius of the curve. This terminal constraint implies that the vehicle has succeeded in staying on the road. The continuous-time optimal control problem (13.19) is solved using the method outlined in Chapter 12.

Results

The optimization problem (13.19) was solved for both Case 1 and Case 2, in the circular-shaped curve with the lateral deviation (13.20) limited to $-1 \leq e \leq 1$ m. The maximum steer angle and maximum steer rate were set to $\delta_{\max} = 30$ deg and $\dot{\delta}_{\max} = 60$ deg/s, corresponding to reasonable physical

and driver limitations. All applied torques were constrained to be negative, because only braking was considered. The wheel-torque constraints were thus set to $T_{u,i,\max} = 0$ kNm and $T_{u,i,\min} = -7.4$ kNm. The numerical value of the constraint $T_{u,i,\min}$ was chosen sufficiently large, such that the tire force F_X is the main limiting factor for braking in the maneuver.

To evaluate the influence of the weighting factor η, the optimization problem was solved for different values of this parameter. In addition, to quantify the differences between Case 1 and Case 2 for different road conditions, the friction coefficients μ and the road-curvature radius R were also altered separately. The scaling of μ was introduced as

$$\mu_{x,\text{scaled}} = \gamma_\mu \mu_x,$$
$$\mu_{y,\text{scaled}} = \gamma_\mu \mu_y,$$

where γ_μ is the scaling factor. This is not an ideal representation for low-friction surfaces, see Section 13.3, but gives at least an indication of possible differences for various road surfaces. Solutions for all parametrization configurations of the optimization problem were typically found in 100–300 iterations.

In Table 13.3, the initial velocity and the maximum body-slip, corresponding to different η-values, are summarized for the optimal solutions obtained in Case 1 and Case 2 for $\gamma_\mu = 1$ and curve radius $R = 30$ m. In Case 2, *i.e.*, with all wheels braking, it is obvious that v_0 does not decrease to a large extent when penalizing body-slip more. For example, there is only a decrease of 1.7% in v_0 between $\eta = 0$ and $\eta = 1000$ in Case 2. On the contrary, in Case 1 the vehicle exhibits a 11.2% decrease of v_0 for the corresponding difference in the weight on the body slip. Also, it is to be noted that the vehicle in Case 1 results in considerably larger maximum values of the body slip, $|\beta|_{\max}$, for all values of the weight. Table 13.4 presents the maximum initial velocities in Case 1 and Case 2, when decreasing the friction coefficient μ, using a weighting parameter of $\eta = 100$ and a road-curvature radius of $R = 30$ m. The table shows that the advantage of the all-wheel braking in Case 2 can be seen to persist for the different friction levels, with very small variations, allowing an initial velocity approximately 7–8% higher than for the vehicle in Case 1. The initial velocities resulting from different road-curvature radii are shown in Table 13.5. Similar to the results obtained when scaling the friction coefficients, the vehicle in Case 2 results in a larger v_0 for all radii. Considering the initial velocity, the relative advantage of braking with all wheels and optimal control inputs is also consistent over the different road curvatures, with the increase being approximately 7–9% compared to the vehicle in Case 1.

The resulting optimal trajectories obtained in Case 1 and Case 2, using $\eta = 100$, $\gamma_\mu = 1$, and $R = 30$ m, were also compared from a qualitative perspective. In Figure 13.23, the geometric vehicle trajectories are presented.

Table 13.3 Initial velocity and maximum body-slip for different η, using $\gamma_\mu = 1$ and $R = 30$ m.

	Case 1		Case 2	
η	v_0 [km/h]	$\lvert\beta\rvert_{\max}$ [deg]	v_0 [km/h]	$\lvert\beta\rvert_{\max}$ [deg]
0	62.6	15	65.6	7.0
10	62.4	11	65.6	5.8
100	60.3	5.0	65.2	2.6
1000	55.6	2.1	64.5	1.0

Table 13.4 Initial velocity for different friction coefficients, using $\eta = 100$ and $R = 30$ m.

γ_μ	Case 1, v_0 [km/h]	Case 2, v_0 [km/h]	Diff. in v_0 [%]
1.00	60	65	8.2
0.90	57	62	8.1
0.80	54	58	7.9
0.70	51	55	7.4
0.60	47	51	7.3
0.50	43	46	7.3
0.40	38	41	7.4
0.30	33	36	7.7
0.20	27	29	8.4

Table 13.5 Initial velocity for different radii of the road curvature, using $\eta = 100$ and $\gamma_\mu = 1$.

R [m]	Case 1, v_0 [km/h]	Case 2, v_0 [km/h]	Diff. in v_0 [%]
10	38	42	9.3
15	46	49	8.4
20	52	55	7.6
25	56	61	7.7
30	60	65	8.2
40	68	74	8.2
50	75	81	7.8

The start position was set to $(X_{p,0}, Y_{p,0}) = (30, 0)$ m—*i.e.*, in the lower right corner in the figure—with the vehicle heading in the tangential direction of the road. Figures 13.24–13.25 show the optimal trajectories for the wheel variables and the tire forces for the respective case. It is to be noted that it is the wheel torques T_i in (12.12) in Chapter 12 that are shown in the figure. Moreover, Figure 13.26 shows a comparison of the trajectories for a subset of the chassis variables in Case 1 and Case 2. Further, Figure 13.27 shows the total tire forces F_X and F_Y, resolved in the road-surface plane. In addition, the yaw moment M_Z generated from the tire forces—*i.e.*, the moment about an axis orthogonal to the road—is visualized. These quantities are displayed as functions of the driven distance s to allow comparison of the results from the vehicles in Case 1 and Case 2. In Figure 13.28, the component of the yaw moment that is a result of the applied braking torques is shown. This moment is denoted ΔM and is the sum of the longitudinal braking forces F_x and the decrease of the lateral forces resulting from the increased slip ratio κ during braking. The mathematical definition is

$$\begin{aligned}\Delta M = &- F_{x,1}(w\cos\delta - l_f\sin\delta) + F_{x,2}(w\cos\delta + l_f\sin\delta)\\ &- F_{x,3}w + F_{x,4}w - (F_{y0,1} - F_{y,1})(l_f\cos\delta + w\sin\delta)\\ &- (F_{y0,2} - F_{y,2})(l_f\cos\delta - w\sin\delta) + (F_{y0,3} - F_{y,3})l_r\\ &+ (F_{y0,4} - F_{y,4})l_r.\end{aligned}$$

From Figures 13.23–13.28 it can be seen that the vehicle in Case 2 completes the maneuver both in shorter time and requires a shorter driven distance. In Case 2, the time for completing the maneuver is $t_f = 1.0$ s and for Case 1 the corresponding time is $t_f = 1.2$ s, and the total driven distance is $s_f = 14.8$ m and $s_f = 17.7$ m, respectively. When analyzing the control strategies for the considered cases, see Figures 13.24–13.25, the most apparent difference is the emphasis on braking in contrast to creating a yaw moment for stabilization of the vehicle. In Case 1, the vehicle initially applies significant braking at the left wheels, which subsequently is reduced as the normal forces F_z for these wheels decrease as a result of lateral load transfer. With this strategy, a positive yaw-moment contribution is always achieved, see ΔM in Figure 13.28.

For the vehicle in Case 2, braking is applied during the complete maneuver (see the trajectory for T in Figure 13.25). Initially, a large braking effort is applied on all wheels, followed by reduced braking on all wheels except wheel 4. Because of the longitudinal and lateral load transfer, the normal load $F_{z,3}$ is significantly reduced. Hence, only small tire forces can be realized, and the braking is therefore rapidly decreased for this wheel, *cf.* the trajectory for T_3. Considering the lateral load transfer, larger tire forces can be utilized at the outer wheels (wheel 2 and 4), and a higher braking effort can thus be employed at these wheels. The differences between the braking

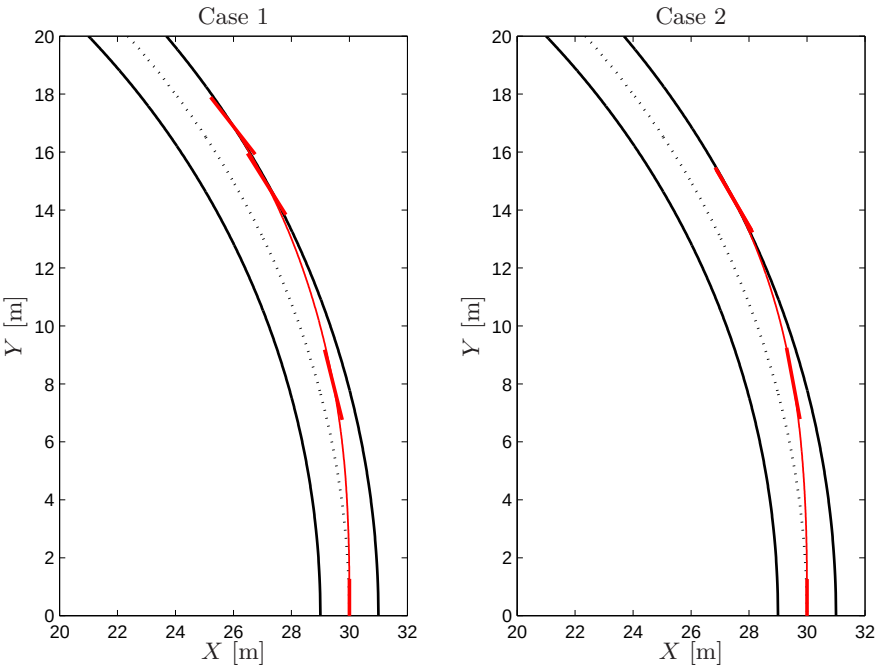

Figure 13.23 Geometric trajectories obtained in Case 1 and Case 2, with $\eta = 100$, $\gamma_\mu = 1$, and $R = 30$ m. The red bars represent the vehicle position and heading direction every half second and at the final time.

strategy throughout the maneuver in the two cases is also clearly seen when investigating F_X in Figure 13.27. Hence, as expected, the vehicle in Case 2 results in a control action with more emphasis on braking. Simultaneously, the cornering possibilities are partly decreased in terms of lateral forces F_Y and yaw moment M_Z.

Summary of the Results from Case 1 and Case 2

The obtained results from quantitative and qualitative comparisons of the different control principles considered in Cases 1–2 exhibited several interesting differences. First, the control strategy observed in Case 2 differs considerably from the approach of more traditional stability control incorporating yaw rotations. Concerning the method, the most obvious is the significant braking leading to decreased velocity, and thus kinetic energy, instead of focusing on applying an asymmetric braking behavior resulting in a yaw moment such that the vehicle starts to rotate in the initial phase. It is plausible that a physical interpretation of this is that reduced kinetic energy makes it easier to stay on the road.

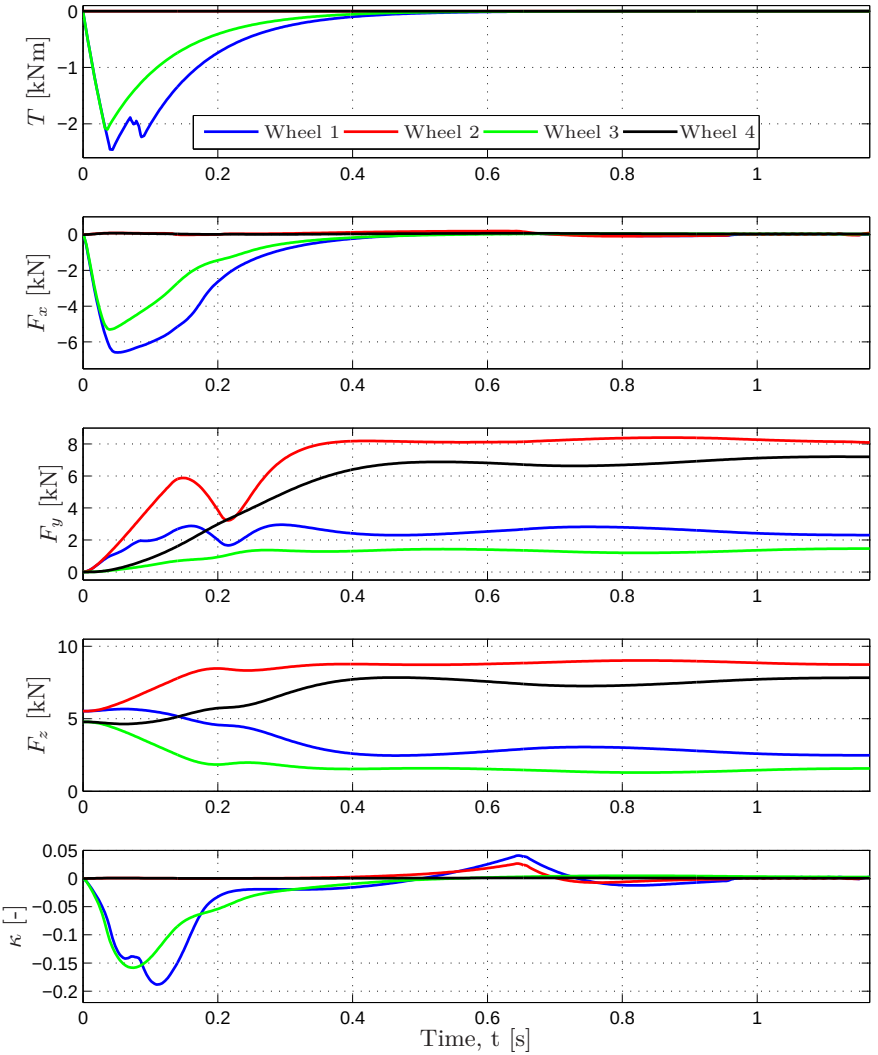

Figure 13.24 Optimal control solutions obtained in Case 1, with $\eta = 100$, $\gamma_\mu = 1$, and $R = 30$ m. Note that $T_2 = T_4 = 0$ in the upper plot.

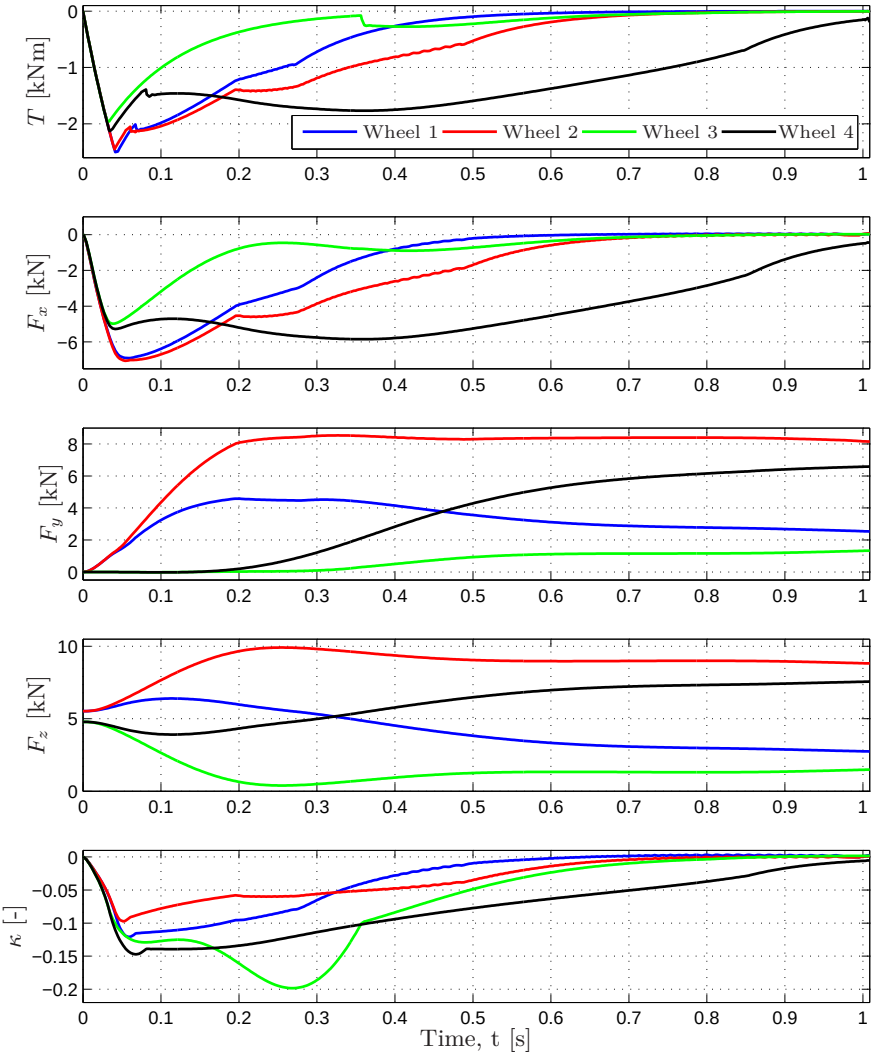

Figure 13.25 Optimal control solutions obtained for Case 2, with $\eta = 100$, $\gamma_\mu = 1$, and $R = 30$ m.

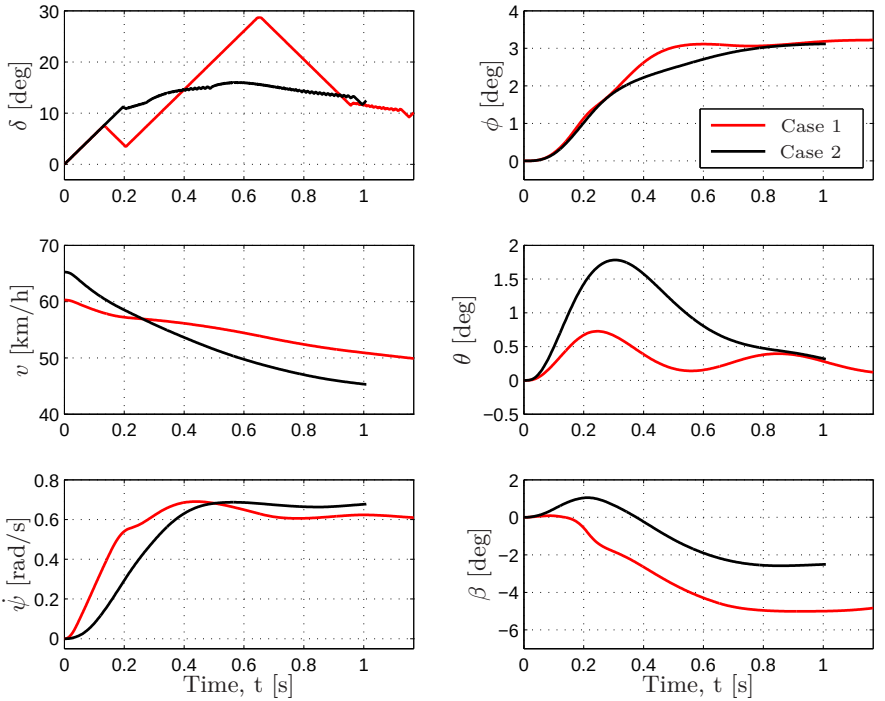

Figure 13.26 Comparison of different safety-related variables in Case 1 and Case 2, with $\eta = 100$, $\gamma_\mu = 1$, and $R = 30$ m.

A series of optimal control problems was solved, where the maximum possible initial velocity was determined for different friction coefficients and curve radii (see Tables 13.4–13.5). It is clear that the vehicle in Case 2 performs significantly better than the vehicle in Case 1 for all situations considered. The performance improvement that the strategy with braking of all wheels can provide in terms of increased initial velocity is consistent, independent of the friction coefficient and the road-curvature radius. Quantitatively, the gain achieved is approximately 8%, when measured as increased initial velocity that can be handled. It is further to be noted that the vehicle in Case 1 determined the optimal combination of steering input and yaw-moment generation with full vehicle state and road information, thus corresponding to the best achievable performance for this control principle in the given situation. This clearly tends to overestimate the capabilities of traditional ESC approaches. Consequently, the increase in safety performance with complete state information, optimal steering, and braking with all wheels in a concrete implementation should be significant.

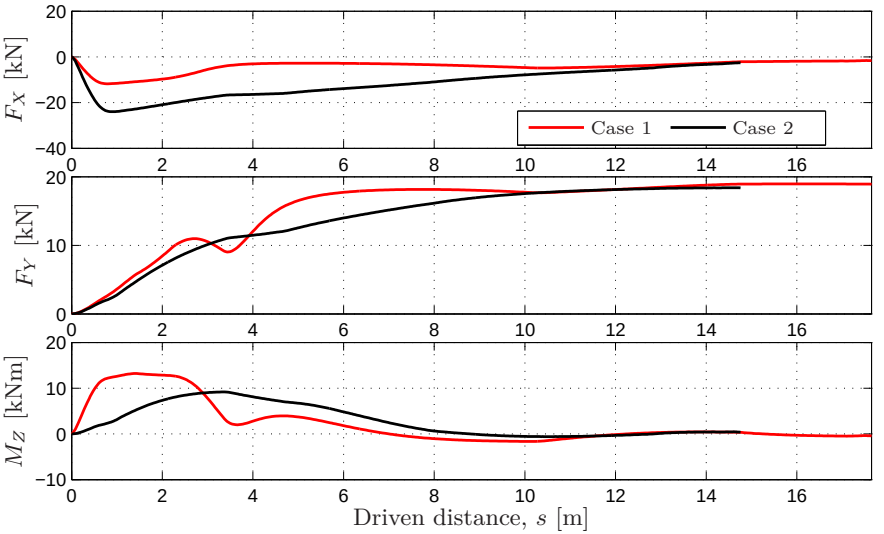

Figure 13.27 Longitudinal force F_X, lateral force F_Y, and yaw moment M_Z developed by the tires. The forces and moment are illustrated as functions of the driven distance s.

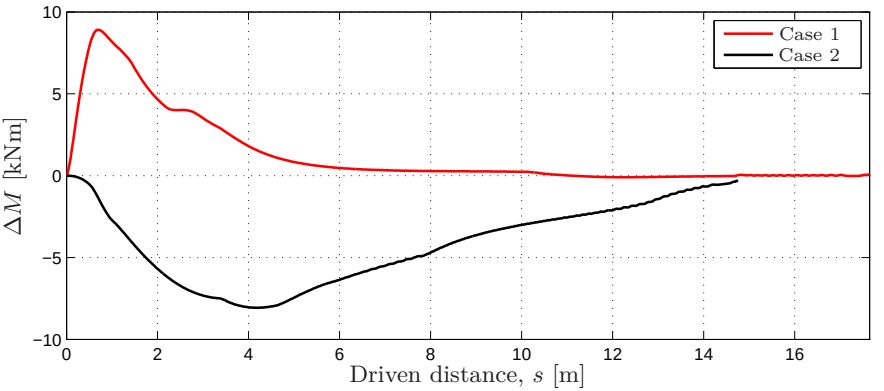

Figure 13.28 The contribution ΔM of the braking forces to the yaw moment.

13.5 Conclusions

This chapter investigated optimal road-vehicle maneuvers using the method developed in Chapter 12. The overall objective of the different applications of the method was to gain insight into the dynamics when the vehicle and the tires operate at their physical performance limits and analyze the obtained optimization results from a control perspective. Three different applications were considered; investigations and comparison of the computed optimal maneuvers for different combinations of chassis and tire models, a study of the resulting optimal maneuvers under different road conditions, and a numerical quantification of the performance increase possible to achieve with new vehicle-safety systems. All of the resulting optimal maneuvers were analyzed from a control perspective and the characteristic behaviors were identified. The overall aim of the presented results is that the observed behaviors in the analysis should contribute to the development of future driver-assistance systems.

14

Conclusions and Future Research

In this part of the thesis, optimal motion control applied to robot manipulators and road vehicles has been investigated. The research regarding optimal path tracking for robots targeted online control for high-accuracy and high-performance applications, where the nominal trajectories have been determined by minimizing a problem-specific cost function. The execution time of the task was considered as a major objective, but the control architecture is not limited to this case. An interesting aspect of future research is experimental validation of the path-tracking accuracy on a manipulator in complex scenarios with significant model uncertainties. The results in Chapter 11 also relate to the sensor-fusion algorithms considered in Chapter 8, where the effects on the path-tracking accuracy of uncertainties in, *e.g.*, the tool mass can be decreased by using the computed estimates of the arm-side pose. Further, although the nominal trajectories were assumed to be given *a priori* in Chapter 11, the control architecture can be combined with online trajectory-generation approaches such as the method for robot manipulators proposed in [Verscheure et al., 2009a], which also was employed in [Berntorp et al., 2014a; Berntorp, 2014] for pseudo-omnidirectional mobile platforms. Another method for online trajectory generation for fixed-time point-to-point motions with robots was discussed in [Ghazaei Ardakani et al., 2015], where MPC was used. That method was based on kinematic robot models and feedforward control in the online execution. Developing online feedback algorithms, in the same spirit as the architecture discussed in this thesis, is an interesting aspect of future research in order to make the method more robust in cases where the path itself is of explicit interest.

Optimal control was also investigated to the purpose of trajectory generation for road vehicles. The solutions to the optimal control problems were computed offline, and three different applications of the considered approach were investigated based on simulations. The solutions obtained in the inves-

tigated situations are in themselves interesting for performance evaluation and analysis. However, from a practical implementation perspective, online optimization and subsequent employment of the solutions are still difficult because of the nonlinear and nonconvex nature of the corresponding problems. In this research, the real-time execution was identified as a key aspect and the trade-off between model granularity and computational complexity was extensively evaluated. For the future, the optimization results presented in this thesis can, *e.g.*, be used for creating a library of optimal vehicle maneuvers to be applied online in a vehicle using real-time scheduling. In this context, also other optimization criteria than the final time, such as control-signal utilization and maximum body slip, when computing the maneuvers are of interest. Another aspect of future research is to combine the computed optimal trajectories (then considered as feedforward signals) with feedback for online execution, thus increasing the robustness to model uncertainties and disturbances. This is similar to what was investigated for robot manipulators in Chapter 11. One such approach for road vehicles was proposed in [Berntorp and Magnusson, 2015], where nonlinear and linear predictive control were combined in a hierarchical control structure. Moreover, online trajectory generation for road vehicles would be of interest in, *e.g.*, ESC and related safety systems, where it is essential that the vehicle stays within its current road lane even with significant body and side slip. The results presented in this part of the thesis are considered as contributions toward the goal of having online optimization-based motion controllers for robot manipulators and safety and driver-assistance technologies for road vehicles.

Bibliography

ABB Robotics (2012). *ABB IRB2400 Industrial Robot Data sheet*. Data sheet nr. PR10034 EN_R7.

ABB Robotics (2014). *ABB IRB140 Industrial Robot Data sheet*. Data sheet nr. PR10031 EN_R1.

Abele, E, M Weigold, and S Rothenbücher (2007). "Modeling and identification of an industrial robot for machining applications". *CIRP Annals* **56**:1, pp. 387–390.

Abele, E., J. Bauer, T. Hemker, R. Laurischkat, H. Meier, S. Reese, and O. von Stryk (2011). "Comparison and validation of implementations of a flexible joint multibody dynamics system model for an industrial robot". *CIRP J. Manufacturing Science and Technology* **4**:1, pp. 38–43.

Abele, E., S. Rothenbücher, and M. Weigold (2008). "Cartesian compliance model for industrial robots using virtual joints". *Prod. Eng. Res. Devel.* **2**, pp. 339–343.

Åkesson, J. (2007). *Languages and Tools for Optimization of Large-Scale Systems*. PhD Thesis ISRN LUTFD2/TFRT--1081--SE. Department of Automatic Control, Lund University, Sweden.

Åkesson, J. (2008). "Optimica—an extension of Modelica supporting dynamic optimization". In: *6th Int. Modelica Conf. 2008*. Modelica Association.

Åkesson, J. and P. Hagander (2003). "Integral action – A disturbance observer approach". In: *Proc. European Control Conf. (ECC)*. Cambridge, United Kingdom.

Åkesson, J., K.-E. Årzén, M. Gäfvert, T. Bergdahl, and H. Tummescheit (2010). "Modeling and optimization with Optimica and JModelica.org—Languages and tools for solving large-scale dynamic optimization problems". *Computers and Chemical Engineering* **34**:11, pp. 1737–1749.

Al Janaideh, M., Y. Feng, S. Rakheja, C.-Y. Su, and C. Alain Rabbath (2009). "Hysteresis compensation for smart actuators using inverse generalized Prandtl-Ishlinskii model". In: *Proc. American Control Conf. (ACC)*. St. Louis, MO, pp. 307–312.

Al Janaideh, M., J. Mao, S. Rakheja, W. Xie, and C.-Y. Su (2008). "Generalized prandtl-ishlinskii hysteresis model: hysteresis modeling and its inverse for compensation in smart actuators". In: *Proc. IEEE Conf. Decision and Control (CDC)*. Cancun, Mexico, pp. 5182–5187.

Alfred Jäger GmbH (2013). *Chopper 3300 H S5 Spindle*. Ober-Mörlen, Germany.

Ali, M. (2012). *Decision Making and Control for Automotive Safety*. Thesis No. 3413. PhD thesis. Chalmers University of Technology, Sweden.

Ali, M., P. Falcone, C. Olsson, and J. Sjöberg (2013). "Predictive prevention of loss of vehicle control for roadway departure avoidance". *IEEE Trans. Intell. Transp. Syst.* **14**:1, pp. 56–68.

Allison, B. and A. Isaksson (1998). "Design and performance of mid-ranging controllers". *J. Process Control* **8**:5, pp. 469–474.

Allison, B. and S. Ogawa (2003). "Design and tuning of valve position controllers with industrial applications". *Trans. Institute of Measurement and Control* **25**:1, pp. 3–16.

Anderson, S., S. Peters, T. Pilutti, and K. Iagnemma (2010). "An optimal-control-based framework for trajectory planning, threat assessment, and semi-autonomous control of passenger vehicles in hazard avoidance scenarios". *Int. J. Vehicle Autonomous Systems* **8**:2/3/4, pp. 190–216.

Anderson, S., S. Karumanchi, and K. Iagnemma (2012). "Constraint-based planning and control for safe, semi-autonomous operation of vehicles". In: *Proc. IEEE Intelligent Vehicles Symp.* Alcalá de Henares, Spain, pp. 383–388.

Andreasson, J. (2009). "Enhancing active safety by extending controllability — How much can be gained?" In: *Proc. IEEE Intelligent Vehicles Symp.* Xi'an, Shaanxi, China, pp. 658–662.

Antonelli, G., S. Chiaverini, and G. Fusco (2003). "A new on-line algorithm for inverse kinematics of robot manipulators ensuring path tracking capability under joint limits". *IEEE Trans. Robot. Autom.* **19**:1, pp. 162–167.

Antonelli, G., C. Curatella, and A. Marino (2011). "Constrained motion planning for open-chain industrial robots". *Robotica* **29** (3), pp. 403–420.

Arai, H., K. Tanie, and S. Tachi (1994). "Path tracking control of a manipulator considering torque saturation". *IEEE Trans. Ind. Electron.* **41**:1, pp. 25–31.

Arimoto, S., S. Kawamura, and F. Miyazaki (1984). "Bettering operation of robots by learning". *J. Robotic Systems* **1**:2, pp. 123–140.

Assimulo (2015). Accessed: 2015-06-30. URL: `http://www.jmodelica.org/assimulo`.

Åström, K. J. and T. Bohlin (1966). "Numerical identification of linear dynamic systems from normal operating records". In: *Theory of Self-Adaptive Control Systems*. Plenum Press, pp. 96–111.

Åström, K. J. and T. Hägglund (2005). *Advanced PID Control*. ISA—The Instrumentation, Systems, and Automation Society, Research Triangle Park, NC.

Åström, K. J. and B. Wittenmark (1997). *Computer-Controlled Systems*. Prentice Hall, Englewood Cliffs, NJ.

Axehill, J. W., I. Dressler, S. Gunnarsson, A. Robertsson, and M. Norrlöf (2014). "Estimation-based ILC applied to a parallel kinematic robot". *Control Engineering Practice* **33**, pp. 1–9.

Axelsson, P., R. Karlsson, and M. Norrlöf (2013). "Estimation-based ILC using particle filter with application to industrial manipulators". In: *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*. Tokyo, Japan, pp. 1740–1745.

Axelsson, P. and M. Norrlöf (2012). "Method to estimate the position and orientation of a triaxial accelerometer mounted to an industrial manipulator". In: *Proc. 10th IFAC Symp. Robot Control (SYROCO)*. Dubrovnik, Croatia, pp. 283–288.

Axelsson, P., R. Karlsson, and M. Norrlöf (2012). "Bayesian state estimation of a flexible industrial robot". *Control Engineering Practice* **20**:11, pp. 1220–1228.

Bauer, H., K.-H. Dietsche, J. Crepin, and F. Dinkler, (Eds.) (2000). *Automotive Handbook*. 5th edition. Robert Bosch GmbH.

Bayer, F. and J. Hauser (2012). "Trajectory optimization for vehicles in a constrained environment". In: *Proc. IEEE Conf. Decision and Control (CDC)*. Maui, Hawaii, pp. 5625–5630.

Beckhoff Automation GmbH (2013). *Beckhoff information system, TwinCAT 2*. Verl, Germany.

Bemporad, A., T. Tarn, and N. Xi (1999). "Predictive path parameterization for constrained robot control". *IEEE Trans. Control Syst. Technol.* **7**:6, pp. 648–656.

Bennett, D., J. Hollerbach, and P. Henri (1992). "Kinematic calibration by direct estimation of the Jacobian matrix". In: *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*. Nice, France, pp. 351–357.

Berglund, E. and G. E. Hovland (2000). "Automatic elasticity tuning of industrial robot manipulators". In: *Proc. IEEE Conf. Decision and Control (CDC)*. Sydney, Australia, pp. 5091–5096.

Berntorp, K., B. Olofsson, and A. Robertsson (2014a). "Path tracking with obstacle avoidance for pseudo-omnidirectional mobile robots using convex optimization". In: *Proc. American Control Conf. (ACC)*. Portland, OR, pp. 517–524.

Berntorp, K. (2008). *ESP for Suppression of Jackknifing in an Articulated Bus*. Master's Thesis ISRN LUTFD2/TFRT--5831--SE. Department of Automatic Control, Lund University, Sweden.

Berntorp, K. (2013). *Derivation of a Six Degrees-of-Freedom Ground-Vehicle Model for Automotive Applications*. Technical Report ISRN LUTFD2/TFRT--7627--SE. Department of Automatic Control, Lund University, Sweden.

Berntorp, K. (2014). *Particle Filtering and Optimal Control for Vehicles and Robots*. PhD thesis ISRN LUTFD2/TFRT--1101--SE. Department of Automatic Control, Lund University, Sweden.

Berntorp, K. and F. Magnusson (2015). "Hierarchical predictive control for ground-vehicle maneuvering". In: *Proc. American Control Conf. (ACC)*. Chicago, IL, pp. 2771–2776.

Berntorp, K., B. Olofsson, K. Lundahl, and L. Nielsen (2014b). "Models and methodology for optimal trajectory generation in safety-critical road–vehicle manoeuvres". *Vehicle System Dynamics* **52**:10, pp. 1304–1332.

Biegler, L. T., A. M. Cervantes, and A. Wächter (2002). "Advances in simultaneous strategies for dynamic process optimization". *Chemical Engineering Science* **57**, pp. 575–593.

Birbach, O. and B. Bauml (2014). "Calibrating a pair of inertial sensors at opposite ends of an imperfect kinematic chain". In: *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*. Chicago, IL, pp. 422–428.

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer Verlag, Berlin Heidelberg.

Bittencourt, A. C., E. Wernholt, S. Sander-Tavallaey, and T. Brogårdh (2010). "An extended friction model to capture load and temperature effects in robot joints". In: *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*. Taipei, Taiwan, pp. 6161–6167.

Blomdell, A., G. Bolmsjö, T. Brogårdh, P. Cederberg, M. Isaksson, R. Johansson, M. Haage, K. Nilsson, M. Olsson, T. Olsson, A. Robertsson, and J. Wang (2005). "Extending an industrial robot controller—Implementation and applications of a fast open sensor interface". *IEEE Robot. Autom. Mag.* **12**:3, pp. 85–94.

Blomdell, A., I. Dressler, K. Nilsson, and A. Robertsson (2010). "Flexible application development and high-performance motion control based on external sensing and reconfiguration of ABB industrial robot controllers". In: *Proc. Workshop "Innovative Robot Control Architectures for Demanding (Research) Applications—How to Modify and Enhance Commercial Controllers", IEEE Int. Conf. Robotics and Automation (ICRA)*. Anchorage, AK, pp. 62–66.

Bobrow, J. E., S Dubowsky, and J. S. Gibson (1983). "On the optimal control of robotic manipulators with actuator constraints". In: *Proc. American Control Conf. (ACC)*. San Francisco, CA, pp. 782–787.

Bobrow, J. E., S Dubowsky, and J. S. Gibson (1985). "Time-optimal control of robotic manipulators along specified paths". *Int. J. Robotics Research* **4**:3, pp. 3–17.

Bohlin, T. (1994). "A case study of grey box identification". *Automatica* **30**:2, pp. 307–318.

Bosch, R., (Ed.) (2011). *Bosch Automotive Handbook*. 8th. Wiley, Hoboken, NJ.

Boyd, S. and L. Vandenberghe (2004). *Convex Optimization*. 6th ed. Cambridge Univ. Press, Cambridge, United Kingdom.

Braghin, F., F. Cheli, and E Sabbioni (2006). "Environmental effects on Pacejka's scaling factors". *Vehicle System Dynamics* **44**:7, pp. 547–568.

Calinon, S., F. Guenter, and A. Billard (2007). "On learning, representing, and generalizing a task in a humanoid robot". *IEEE Trans. Syst. Man, Cybern. B, Cybern.* **37**:2, pp. 286–298.

Cano Marchal, P., O. Sörnmo, B. Olofsson, A. Robertsson, J. Gómez Ortega, and R. Johansson (2014). "Iterative learning control for machining with industrial robots". In: *Proc. 19th IFAC World Congress*. Cape Town, South Africa, pp. 9327–9333.

Carlson, C. and J. Gerdes (2005). "Consistent nonlinear estimation of longitudinal tire stiffness and effective radius". *IEEE Trans. Control Syst. Technol.* **13**:6, pp. 1010–1020.

Casanova, D (2000). *On minimum time vehicle manoeuvring: The theoretical optimal lap*. PhD Thesis. Cranfield University, United Kingdom.

Cavallo, A., G. De Maria, C. Natale, and S. Pirozzi (2007). "Gray-box identification of continuous-time models of flexible structures". *IEEE Trans. Control Syst. Technol.* **15**:5, pp. 967–981.

Cescon, M., I. Dressler, R. Johansson, and A. Robertsson (2009). "Subspace-based identification of compliance dynamics of parallel kinematic manipulator". In: *Proc. IEEE/ASME Int. Conf. Advanced Intelligent Mechatronics (AIM)*. Singapore, pp. 1028–1033.

Chakraborty, I., P. Tsiotras, and J. Lu (2011). "Vehicle posture control through aggressive maneuvering for mitigation of T-bone collisions". In: *Proc. IEEE Conf. Decision and Control (CDC)*. Orlando, FL, pp. 3264–3269.

Chakraborty, I., P. Tsiotras, and R. S. Diaz (2013). "Time-optimal vehicle posture control to mitigate unavoidable collisions using conventional control inputs". In: *Proc. American Control Conf. (ACC)*. Washington, DC, pp. 2165–2170.

Chen, W. and M. Tomizuka (2014). "Direct joint space state estimation in robots with multiple elastic joints". *IEEE/ASME Trans. Mechatronics* **19**:2, pp. 697–706.

Chen, Y. and A. A. Desrochers (1989). "Structure of minimum-time control law for robotic manipulators with constrained paths". In: *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*. Scottsdale, AZ, pp. 971–976.

COMET (2015). EU/FP7-project: *Plug-and-produce COmponents and METhods for adaptive control of industrial robots enabling cost effective, high precision manufacturing in factories of the future*, Ref. #258769, Accessed: 2015-06-30. URL: `http://www.cometproject.eu`.

Conte, G., S. Duranti, and T. Merz (2004). "Dynamic 3D path following for an autonomous helicopter". In: *Proc. IFAC Symp. Intelligent Autonomous Vehicles*. Lisbon, Portugal.

Crassidis, J. L., F. L. Markley, and Y. Cheng (2007). "Survey of nonlinear attitude estimation methods". *J. Guidance, Control, and Dynamics* **30**:1, pp. 12–28.

Cuypers, W., N. Van Gestel, A. Voet, J. Kruth, J. Mingneau, and P. Bleys (2009). "Optical measurement techniques for mobile and large-scale dimensional metrology". *Optics and Lasers in Engineering* **47**:3, pp. 292–300.

CVX Research Inc. (2015). *CVX: Matlab software for disciplined convex programming, version 2.0 beta*. Accessed: 2015-06-30. URL: `http://cvxr.com/cvx`.

Dahl, O. and L. Nielsen (1990a). "Stability analysis of an online algorithm for torque limited path following". In: *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*. Cincinnati, OH, pp. 1216–1222.

Dahl, O. (1992). *Path Constrained Robot Control*. ISRN LUTFD2/TFRT--1038--SE. PhD thesis. Department of Automatic Control, Lund University, Sweden.

Dahl, O. (1993). "Path constrained motion optimization for rigid and flexible joint robots". In: *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*. Atlanta, GA, pp. 223–229.

Dahl, O. (1994). "Path constrained robot control—experimental evaluation". *Mechatronics* **4**:2, pp. 173–198.

Dahl, O. and L. Nielsen (1990b). "Torque limited path following by on-line trajectory time scaling". *IEEE Trans. Robot. Autom.* **6**:5, pp. 554–561.

De Luca, A. and R. Farina (2002). "Dynamic scaling of trajectories for robots with elastic joints". In: *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*. Washington, DC, pp. 2436–2442.

De Luca, A., D Schröder, and M. Thümmel (2007). "An acceleration-based state observer for robot manipulators with elastic joints". In: *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*. Roma, Italy, pp. 3817–3823.

Debrouwere, F., W. Van Loock, G. Pipeleers, Q. Tran Dinh, M. Diehl, J. De Schutter, and J. Swevers (2013). "Time-optimal path following for robots with convex-concave constraints using sequential convex programming". *IEEE Trans. Robot.* **29**:6, pp. 1485–1495.

Denavit, J. and R. S. Hartenberg (1955). "A kinematic notation for lower-pair mechanisms based on matrices". *Trans. ASME J. Appl. Mech.* **23**, pp. 215–221.

Dennis, J. and R. Schnabel (1983). *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. Classics in Applied Mathematics, Society for Industrial and Applied Mathematics (SIAM).

Dept. of Computer Science, Lund University (2015). *LabComm protocol*. Accessed: 2015-06-30. URL: `http://wiki.cs.lth.se/moin/LabComm`.

Dingle, P. and L. Guzzella (2010). "Optimal emergency maneuvers on highways for passenger vehicles with two- and four-wheel active steering". In: *Proc. American Control Conf. (ACC)*. Baltimore, MD, pp. 5374–5381.

dSPACE GmbH (2007). *DS1103 PPC Controller Board—Hardware Installation and Configuration*. dSPACE GmbH, Paderborn, Germany.

Dumas, C., S. Caro, S. Garnier, and B. Furet (2011). "Joint stiffness identification of six-revolute industrial serial robots". *Robotics and Computer-Integrated Manufacturing* **27**:4, pp. 881–888.

Egeland, O. and J. T. Gravdahl (2002). *Modeling and Simulation for Automatic Control*. Vol. 76. Marine Cybernetics, Trondheim, Norway.

Egerstedt, M., X. Hu, and A. Stotsky (2001). "Control of mobile platforms using a virtual vehicle approach". *IEEE Trans. Autom. Control* **46**:11, pp. 1777–1782.

Eielsen, A. A., J. T. Gravdahl, and K. K. Leang (2015). "Low-order continuous-time robust repetitive control: Application in nanopositioning". *Mechatronics*. In press. DOI: `http://dx.doi.org/10.1016/j.mechatronics.2015.07.006`.

Esmailzadeh, E., A. Goodarzi, and G. Vossoughi (2003). "Optimal yaw moment control law for improved vehicle handling". *Mechatronics* **13**:7, pp. 659–675.

Fasse, E. and N. Hogan (1995). "Control of physical contact and dynamic interaction". In: *Proc. 7th Int. Symp. Robotics Research*. Munich, Germany, pp. 28–38.

Freidovich, L. B. and A. S. Shiriaev (2012). "Transverse linearization for underactuated nonholonomic mechanical systems with application to orbital stabilization". In: Johansson, R. et al. (Eds.). *Distributed Decision Making and Control*. Springer Verlag, London, pp. 245–258.

Funke, J., P. Theodosis, R. Hindiyeh, G. Stanek, K. Kritatakirana, C. Gerdes, D. Langer, M. Hernandez, B. Muller-Bessler, and B. Huhnke (2012). "Up to the limits: Autonomous Audi TTS". In: *Proc. IEEE Intelligent Vehicles Symp*. Alcalá de Henares, Spain, pp. 541–547.

Gautier, M and W Khalil (1992). "Exciting trajectories for the identification of base inertial parameters of robots". *Int. J. Robotics Research* **11**:4, pp. 362–375.

Gautier, M. (1997). "Dynamic identification of robots with power model". In: *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*. Vol. 3. Albuquerque, New Mexico, pp. 1922–1927.

Ge, P. and M. Jouaneh (1996). "Tracking control of a piezoceramic actuator". *IEEE Trans. Control Syst. Technol.* **4**:3, pp. 209–216.

Gerelli, O. and C. Guarino Lo Bianco (2008). "Real-time path-tracking control of robotic manipulators with bounded torques and torque-derivatives". In: *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*. Nice, France, pp. 532–537.

Ghazaei Ardakani, M. M., B. Olofsson, A. Robertsson, and R. Johansson (2015). "Real-time trajectory generation using model predictive control". In: *Proc. IEEE Int. Conf. Automation Science and Engineering (CASE)*. Göteborg, Sweden. *Accepted for Publication*.

Golub, G. H. and C. F. Van Loan (1996). *Matrix Computations*. 3rd edition. The Johns Hopkins Univ. Press, Baltimore, MD.

Gordon, N. J., D. J. Salmond, and A. F. Smith (1993). "Novel approach to nonlinear/non-Gaussian Bayesian state estimation". In: *IEE Proc. Radar and Signal Process.* Vol. 140. 2, pp. 107–113.

Grant, M. and S. Boyd (2008). "Graph implementations for nonsmooth convex programs". In: Blondel, V. et al. (Eds.). *Recent Advances in Learning and Control*. Springer Verlag, Berlin Heidelberg, pp. 95–110.

Griewank, A. (2000). "Evaluating derivatives: principles and techniques of algorithmic differentiation". *Frontiers in Applied Mathematics, SIAM* **19**.

Grundelius, M. (2001). *Methods for Control of Liquid Slosh*. ISRN LUTFD2/TFRT--1062--SE. PhD thesis. Department of Automatic Control, Lund University, Sweden.

Guarino Lo Bianco, C. and O. Gerelli (2011). "Online trajectory scaling for manipulators subject to high-order kinematic and dynamic constraints". *IEEE Trans. Robot.* **27**:6, pp. 1144–1152.

Guarino Lo Bianco, C. and F. Ghilardelli (2014). "Real-time planner in the operational space for the automatic handling of kinematic constraints". *IEEE Trans. Autom. Sci. Eng.* **11**:3, pp. 730–739.

Guarino Lo Bianco, C. and F. Wahl (2011). "A novel second order filter for the real-time trajectory scaling". In: *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*. Shanghai, China, pp. 5813–5818.

Gustafsson, F. (2010). *Statistical Sensor Fusion*. Studentlitteratur, Lund, Sweden.

Hastie, T., R. Tibshirani, and J. Friedman (2008). *The Elements of Statistical Learning*. Springer Verlag, New York.

Hauser, J. and A. Banaszuk (1997). "Approximate feedback linearization around a trajectory: application to trajectory planning". In: *Proc. IEEE Conf. Decision and Control (CDC)*. Tampa, FL, pp. 7–11.

Haverkamp, B. and M. Verhaegen (1997). *SMI Toolbox: State Space Model Identification Software for Multivariable Dynamical Systems*. Delft University of Technology, Delft, The Netherlands.

Haverkamp, B. R. J., M. Verhaegen, C. T. Chou, and R. Johansson (1997). "Continuous-time identification of MIMO state-space models from sampled data". In: *Proc. IFAC Symp. System Identification (SYSID)*. Kitakyushu, Fukuoka, Japan.

Hehn, M., R. Ritz, and R. D'Andrea (2012). "Performance benchmarking of quadrotor systems using time-optimal control". *Autonomous Robots* **33**:1–2, pp. 69–88.

Henriksson, R, M Norrlöf, S Moberg, E Wernholt, and T. Schön (2009). "Experimental comparison of observers for tool position estimation of industrial robots". In: *Proc. IEEE Conf. Decision and Control (CDC)*. Shanghai, China, pp. 8065–8070.

Ho, B. L. and R. E. Kalman (1966). "Effective construction of linear state-variable models from input/output functions". *Regelungstechnik* **14**:12, pp. 545–592.

Hoffmann, G. M., S. L. Waslander, and C. J. Tomlin (2008). "Quadrotor helicopter trajectory tracking control". In: *Proc. AIAA Guidance, Navigation and Control Conf. and Exhibit*. Honolulu, HI, pp. 1–14.

Hogan, N. (1985). "Impedance control: An approach to manipulation: parts I-III". *ASME J. Dynamic Systems, Measurement, and Control* **107**, pp. 1–24.

Hol, J. D. (2011). *Sensor Fusion and Calibration of Inertial Sensors, Vision, Ultra-Wideband and GPS*. PhD Thesis No. 1368. Linköping University, Linköping, Sweden.

Hollerbach, J. M. (1984). "Dynamic scaling of manipulator trajectories". *ASME J. Dynamic Systems, Measurement, and Control* **106**:1, pp. 102–106.

Hovland, G., S. Hanssen, S. Moberg, T Brogårdh, S. Gunnarson, and M. Isaksson (2002). "Nonlinear identification of backlash in robot transmissions". In: *Proc. Int. Symp. Robotics (ISR), Stockholm, Sweden*.

Hovland, G. E., E. Berglund, and S Hanssen (2001). "Identification of coupled elastic dynamics using inverse eigenvalue theory". In: *Proc. Int. Symp. Robotics (ISR)*. Seoul, Korea.

IFR Statistical Department (2012). *World Robotics: Industrial Robots 2012*. International Federation of Robotics.

Isermann, R. (2006). *Fahrdynamik-Regelung: Modellbildung, Fahrerassistenzsysteme, Mechatronik*. Vieweg-Verlag, Wiesbaden, Germany.

ISO (1985). *ISO DIN 1101: Geometrical Tolerances*. Beuth, Berlin, Germany.

ISO 3888-2:2011 (2011). *Passenger cars—Test track for a severe lane-change manoeuvre—Part 2: Obstacle avoidance*. International Organization for Standardization, Geneva, Switzerland.

Janssens, P., W. Van Loock, G. Pipeleers, F. Debrouwere, and J. Swevers (2013). "Iterative learning control for optimal path following problems". In: *Proc. IEEE Conf. Decision and Control (CDC)*. Florence, Italy, pp. 6670–6675.

Jeon, S., M. Tomizuka, and T. Katou (2009). "Kinematic Kalman filter (KKF) for robot end-effector sensing". *ASME J. Dynamic Systems, Measurement, and Control* **131**:2, pp. 021010–1–8.

JModelica.org (2015). Accessed: 2015-06-30. URL: `http://www.jmodelica.org`.

Johansson, R. (1993). *System Modeling and Identification*. Prentice Hall, Englewood Cliffs, NJ.

Johansson, R., M. Verhaegen, and C. T. Chou (1999). "Stochastic theory of continuous-time state-space identification". *IEEE Trans. Signal Process.* **47**, pp. 41–51.

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

Jonsson, M., A. Stolt, A. Robertsson, S. von Gegerfelt, and K. Nilsson (2013). "On force control for assembly and deburring of castings". *Production Engineering* **7**:4, pp. 351–360.

Joubair, A., A. Nubiola, and I. Bonev (2013). "Calibration efficiency analysis based on five observability indices and two calibration models for a six-axis industrial robot". *Int. J. Aerospace* **6**:1, pp. 161–168.

Juang, J.-N. and R. S. Pappa (1985). "An eigensystem realization algorithm for modal parameter identification and model reduction". *J. Guidance, Control, and Dynamics* **8**:5, pp. 620–627.

Kailath, T., A. Sayed, and B. Hassibi (2000). *Linear Estimation*. Prentice Hall, Upper Saddle River, NJ.

Kalman, R. E. (1960). "A new approach to linear filtering and prediction problems". *Trans. of the ASME—J. Basic Engineering* **82**:D, pp. 35–45.

Kant, K. and S. W. Zucker (1986). "Toward efficient trajectory planning: the path-velocity decomposition". *Int. J. Robotics Research* **5**:3, pp. 72–89.

Karlsson, R. and M Norrlöf (2004). "Bayesian position estimation of an industrial robot using multiple sensors". In: *Proc. Int. Conf. Control Applications (CCA)*. Vol. 1. Taipei, Taiwan, pp. 303–308.

Kelly, D. P. and R. S. Sharp (2010). "Time-optimal control of the race car: a numerical method to emulate the ideal driver". *Vehicle System Dynamics* **48**:12, pp. 1461–1474.

Keyence Corp. (2006). *LK-G Series User Manual*. Keyence Corp., Osaka, Japan.

Kieffer, J., A. Cahill, and M. James (1997). "Robust and accurate time-optimal path-tracking control for robot manipulators". *IEEE Trans. Robot. Autom.* **13**:6, pp. 880–890.

Kiencke, U. and L. Nielsen (2005). *Automotive Control Systems—For Engine, Driveline and Vehicle*. 2nd edition. Springer Verlag, Berlin Heidelberg, Germany.

Krasnosel'skii, M., A. Pokrovskii, and M. Niezgódka (1989). *Systems with Hysteresis*. Springer Verlag, Berlin Heidelberg, Germany.

Krejci, P. and K. Kuhnen (2001). "Inverse control of systems with hysteresis and creep". In: *IEE Proc. Control Theory and Applications*. Vol. 148. 3, pp. 185–192.

KUKA Roboter GmbH (2013). *KR 125 Serie 2000 Data Sheet*. RoMeDBKR125-04.99.02.

Larimore, W. E. (1990). "Canonical variate analysis in identification, filtering, and adaptive control". In: *Proc. IEEE Conf. Decision and Control (CDC)*. Honolulu, HI, pp. 596–604.

LaValle, S. M. (2006). *Planning Algorithms*. Cambridge Univ. Press, Cambridge, United Kingdom.

Lee, S., I. Suh, S. Calinon, and R. Johansson (2014). "Autonomous framework for segmenting robot trajectories of manipulation task". *Autonomous Robots* **38**:2, pp. 107–141.

Lehmann, C., B. Olofsson, K. Nilsson, M. Halbauer, M. Haage, A. Robertsson, O. Sörnmo, and U. Berger (2013). "Robot joint modeling and parameter identification using the clamping method". In: *Proc. IFAC Conf. Manufacturing Modelling, Management, and Control (MIM)*. St. Petersburg, Russia, pp. 843–848.

Lei, L., K. K. Tan, S. Huang, and T. H. Lee (2011). "Online parameter estimation and compensation of Preisach hysteresis by SVD updating". In: *Proc. 18th IFAC World Congress*. Milano, Italy, pp. 5249–5254.

Lertpiriyasuwat, V., M. C. Berg, and K. W. Buffinton (2000). "Extended Kalman filtering applied to a two-axis robotic arm with flexible links". *Int. J. Robotics Research* **19**:3, pp. 254–270.

Levenberg, K. (1944). "A method for the solution of certain problems in least squares". *Quarterly of Applied Mathematics* **2**, pp. 164–168.

Li, W.-L., R. Iltis, and M. Win (2013). "Integrated IMU and radiolocation-based navigation using a Rao-Blackwellized particle filter". In: *Proc. IEEE Conf. Acoustics, Speech and Signal Processing (ICASSP)*. Vancouver, Canada, pp. 5165–5169.

Li, X., J.-X. Xu, and D. Huang (2014). "An iterative learning control approach for linear systems with randomly varying trial lengths". *IEEE Trans. Autom. Control* **59**:7, pp. 1954–1960.

Li, Y. and Q. Xu (2011). "A totally decoupled piezo-driven XYZ flexure parallel micropositioning stage for micro/nanomanipulation". *IEEE Trans. Autom. Sci. Eng.* **8**:20, pp. 265–279.

Liaw, H. C. and B. Shirinzadeh (2010). "Constrained motion tracking control of piezo-actuated flexure-based four-bar mechanisms for micro/nano manipulation". *IEEE Trans. Autom. Sci. Eng.* **7**:3, pp. 699–705.

Liebemann, K., K. Meder, J. Schuh, and G. Nenninger (2005). "Safety and performance enhancement: the Bosch electronic stability control". Paper Number 05-0471, Robert Bosch GmbH. Germany.

Limebeer, D. and A. Rao (2015). "Faster, higher, and greener: vehicular optimal control". *IEEE Control Syst. Mag.* **35**:2, pp. 36–56.

Lindsten, F., T. Schön, and J. Olsson (2011). "An explicit variance reduction expression for the Rao-Blackwellised particle filter". In: *Proc. 18th IFAC World Congress*. Milan, Italy, pp. 11979–11984.

Lipp, T. and S. Boyd (2014). "Minimum-time speed optimization over a fixed path". *Int. J. Control* **87**:6, pp. 1297–1311.

Ljung, L. (2010). *System Identification Toolbox 7: Users's Guide*. The MathWorks, Inc., Natick, MA.

Lundahl, K, C. Lee, E. Frisk, and L. Nielsen (2015). "Analyzing rollover indices for critical truck maneuvers". *SAE Int. J. Commer. Veh.* **8**:1, pp. 189–196.

Lundahl, K., J. Åslund, and L. Nielsen (2013). *Vehicle Dynamics Platform, Experiments, and Modeling Aiming at Critical Maneuver Handling*. Technical Report LiTH-R-3064. Department of Electrical Engineering, Linköping University, Linköping, Sweden.

Lundahl, K., B. Olofsson, K. Berntorp, J. Åslund, and L. Nielsen (2014). "Towards lane-keeping electronic stability control for road-vehicles". In: *Proc. 19th IFAC World Congress*. Cape Town, South Africa, pp. 6319–6325.

Lyzell, C. (2012). *Structural Reformulations in System Identification*. Linköping Studies in Science and Technology. PhD Thesis No. 1475. Linköping University, Sweden.

Lyzell, C., M. Enqvist, and L. Ljung (2009). "Handling certain structure information in subspace identification". In: *Proc. IFAC Symp. System Identification (SYSID)*. Saint-Malo, France, pp. 90–95.

Madsen, H. (2008). *Time Series Analysis*. Chapman & Hall/CRC, Boca Raton, FL.

Magnussen, Ø., M. Ottestad, and G. Hovland (2015). "Calibration procedure for an inertial measurement unit using a 6-degree-of-freedom hexapod". Accessed: 2015-06-30. URL: `http://home.uia.no/geirh/PDF/C52.pdf`.

Mahr GmbH (2011). *MarSurf M400 Flyer*. Mahr GmbH, Göttingen, Germany.

Marquardt, D. W. (1963). "An algorithm for least-squares estimation of nonlinear parameters". *J. Society for Industrial & Applied Mathematics (SIAM)* **11**:2, pp. 431–441.

MathWorks Inc. (2010). *Real-Time Workshop 7: Users's Guide*. The MathWorks, Inc., Natick, MA.

McKelvey, T., H. Akçay, and L. Ljung (1996). "Subspace-based multivariable system identification from frequency response data". *IEEE Trans. Autom. Control* **41**:7, pp. 960–979.

Meriam, J. L. and L. G. Kraige (2012). *Engineering mechanics: Dynamics*. Vol. 2. John Wiley & Sons, Hoboken, NJ.

Miyazaki, F, S Kawamura, M Matsumori, and S Arimoto (1986). "Learning control scheme for a class of robot systems with elasticity". In: *Proc. IEEE Conf. Decision and Control (CDC)*. Vol. 25. Athens, Greece, pp. 74–79.

Moberg, S., E. Wernholt, S. Hanssen, and T. Brogårdh (2014). "Modeling and parameter estimation of robot manipulators using extended flexible joint models". *ASME J. Dynamic Systems, Measurement, and Control* **136**:3, pp. 031005–1–13.

Modelica Association (2015). Accessed: 2015-06-30. URL: `http://www.modelica.org`.

Morawiec, A. (1998). "A note on mean orientation". *J. Applied Crystallography* **31**:5, pp. 818–819.

Nikon Metrology (2010). *K-series optical CMM solutions*. Data sheet Optical_CMM_EN_0311.

Nilsson, K. and R. Johansson (1999). "Integrated architecture for industrial robot programming and control". *J. Robotics and Autonomous Systems* **29**:4, pp. 205–226.

Niu, W. and M. Tomizuka (2001). "A new approach of coordinated motion control subjected to actuator saturation". *ASME J. Dynamic Systems, Measurement, and Control* **123**:3, pp. 496–504.

Norén, J. (2014). *IMU-based Robot Tool Pose Estimation*. MSc Thesis LiTH-ISY-EX–14/4787–SE. Linköping University, Sweden.

Norrlöf, M. (2002). "An adaptive iterative learning control algorithm with experiments on an industrial robot". *IEEE Trans. Robot. Autom.* **18**:2, pp. 245–251.

Norrlöf, M. (2000). *Iterative Learning Control: Analysis, Design, and Experiments*. Thesis No. 653. PhD thesis. Linköping University, Sweden.

Olofsson, B. and L. Nielsen (2015). "Path-tracking velocity control for robot manipulators with actuator constraints". *Mechatronics. Submitted*.

Olofsson, B., O. Sörnmo, U. Schneider, A. Robertsson, A. Puzik, and R. Johansson (2011a). "Modeling and control of a piezo-actuated high-dynamic compensation mechanism for industrial robots". In: *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*. San Francisco, CA, pp. 4704–4709.

Olofsson, B., H. Nilsson, A. Robertsson, and J. Åkesson (2011b). "Optimal tracking and identification of paths for industrial robots". In: *Proc. 18th IFAC World Congress*. Milano, Italy, pp. 1126–1132.

Olofsson, B., O. Sörnmo, U. Schneider, M. Barho, A. Robertsson, and R. Johansson (2012). "Increasing the accuracy for a piezo-actuated micro manipulator for industrial robots using model-based nonlinear control".

In: *Proc. 10th Int. IFAC Symp. Robot Control (SYROCO)*. Dubrovnik, Croatia, pp. 277–282.

Olofsson, B., K. Lundahl, K. Berntorp, and L. Nielsen (2013). "An investigation of optimal vehicle maneuvers for different road conditions". In: *Proc. 7th IFAC Symp. Advances in Automotive Control (AAC)*. Tokyo, Japan, pp. 66–71.

Olofsson, B., O. Sörnmo, A. Robertsson, and R. Johansson (2014). "Continuous-time gray-box identification of mechanical systems using subspace-based identification methods". In: *Proc. IEEE/ASME Int. Conf. Advanced Intelligent Mechatronics (AIM)*. Besançon, France, pp. 327–333.

Olofsson, B., J. Antonsson, H. Kortier, B. Bernhardsson, A. Robertsson, and R. Johansson (2015). "Sensor fusion for robotic workspace state estimation". *IEEE/ASME Trans. Mechatronics. Submitted*.

Olsson, H. (1996). *Control Systems with Friction*. PhD thesis ISRN LUTFD2/TFRT–1045–SE. Department of Automatic Control, Lund University, Sweden.

Olsson, T., R. Johansson, and A. Robertsson (2004). "Flexible force-vision control for surface following using multiple cameras". In: *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*. Sendai, Japan, pp. 798–803.

Olsson, T., M. Haage, H. Kihlman, R. Johansson, K. Nilsson, A. Robertsson, M. Björkman, R. Isaksson, G. Ossbahr, and T. Brogårdh (2010). "Cost-efficient drilling using industrial robots with high-bandwidth force feedback". *Robotics and Computer-Integrated Manufacturing* **26**, pp. 24–38.

Pacejka, H. B. (2006). *Tyre and Vehicle Dynamics*. 2nd edition. Butterworth-Heinemann, Oxford, United Kingdom.

Pan, Z. and H. Zhang (2009). "Improving robotic machining accuracy by real-time compensation". In: *Proc. ICROS-SICE Int. Joint Conf.* Fukuoka, Japan, pp. 4289–4294.

Parsa, K, J Angeles, and A. K. Misra (2004). "Rigid-body pose and twist estimation using an accelerometer array". *Arch. Applied Mechanics* **74**:3–4, pp. 223–236.

Pfeiffer, F. and R. Johanni (1987). "A concept for manipulator trajectory planning". *IEEE J. Robot. Autom.* **3**:2, pp. 115–123.

Preisach, F. (1935). "Über die magnetische Nachwirkung". *Zeitschrift für Physik* **94** (5–6), pp. 277–302.

Puzik, A., A. Pott, C. Meyer, and A. Verl (2009). "Industrial robots for machining processes in combination with an additional actuation mechanism for error compensation". In: *Proc. 7th Int. Conf. Manufacturing Research (ICMR)*. University of Warwick, United Kingdom.

Puzik, A., C. Meyer, and A. Verl (2010). "Results of robot machining with additional 3D-piezo-actuation-mechanism for error compensation". In: *Proc. 7th CIRP Int. Conf., Intelligent Computation in Manufacturing Eng.: Innovative and Cognitive Production Technology and Systems*. Capri, Italy.

Puzik, A. (2011). *Genauigkeitssteigerung bei der spanenden Bearbeitung mit Industrierobotern durch Fehlerkompensation mit 3D-Ausgleichsaktorik*. PhD Dissertation. University of Stuttgart, Stuttgart, Germany.

Rajamani, R. (2006). *Vehicle Dynamics and Control*. Springer Verlag, Berlin Heidelberg, Germany.

Reinl, C., M. Friedmann, J. Bauer, M. Pischan, E. Abele, and O. von Stryk (2011). "Model-based off-line compensation of path deviation for industrial robots in milling applications". In: *Proc. Int. Conf. Advanced Intelligent Mechatronics (AIM)*. Budapest, Hungary, pp. 367–372.

Reis GmbH (2011). *Reis RV40 Fact Sheet*. Reis GmbH, Obernburg, Germany.

Rigatos, G. G. (2009). "Particle filtering for state estimation in nonlinear industrial systems". *IEEE Trans. Instrum. Meas.* **58**:11, pp. 3885–3900.

Roth, Z., B. Mooring, and B. Ravani (1987). "An overview of robot calibration". *IEEE J. Robot. and Autom.* **5**:3, pp. 377–385.

Ruderman, M., F. Hoffmann, and T. Bertram (2009). "Modeling and identification of elastic robot joints with hysteresis and backlash". *IEEE Trans. Ind. Electron.* **56**:10, pp. 3840–3847.

Sarkar, N., X. Yun, and V. Kumar (1993). "Dynamic path following: a new control algorithm for mobile robots". In: *Proc. IEEE Conf. Decision and Control (CDC)*, pp. 2670–2675.

Särkkä, S. (2013). *Bayesian filtering and smoothing*. Cambridge Univ. Press, Cambridge, United Kingdom.

Schindler, E. (2007). *Fahrdynamik: Grundlagen Des Lenkverhaltens Und Ihre Anwendung Für Fahrzeugregelsysteme*. Expert-Verlag, Renningen, Germany.

Schneider, U. (2010). *Regelung einer 3D-Ausgleichsaktorik für Fräsanwendungen mit Industrierobotern*. Diploma Thesis. Fraunhofer Institute for Manufacturing and Engineering, Stuttgart, Germany.

Schneider, U., M. Drust, A. Puzik, and A. Verl (2013a). "Compensation of errors in robot machining with a parallel 3D-piezo compensation mechanism". In: *Proc. Forty Sixth CIRP Conf. Manufacturing Systems*. Sesimbra, Portugal, pp. 305–310.

Schneider, U., M. Ansaloni, M. Drust, F. Leali, and A. Verl (2013b). "Experimental investigation of error sources in robot machining". In: *Proc. Int. Conf. Flexible Automation and Intelligent Manufacturing (FAIM)*. Porto, Portugal, pp. 14–26.

Schneider, U., J. R. Diaz Posada, M. Drust, and A. Verl (2013c). "Position control of an industrial robot using an optical measurement system for machining purposes". In: *Proc. 11th Int. Conf. Manufacturing Research (ICMR)*. Cranfield University, United Kingdom, pp. 307–312.

Schneider, U., B. Olofsson, O. Sörnmo, M. Drust, A. Robertsson, M. Hägele, and R. Johansson (2014). "Integrated approach to robotic machining with macro/micro-actuation". *Robotics and Computer-Integrated Manufacturing* **30**:6, pp. 636–647.

Schofield, B. (2008). *Model-Based Vehicle Dynamics Control for Active Safety*. PhD Thesis ISRN LUTFD2/TFRT--1083--SE. Department of Automatic Control, Lund University, Sweden.

Schön, T., F. Gustafsson, and P.-J. Nordlund (2005). "Marginalized particle filters for mixed/linear nonlinear state-space models". *IEEE Trans. Signal Process.* **53**:7, pp. 2279–2289.

Schön, T., A. Wills, and B. Ninness (2011). "System Identification of Nonlinear State-Space Models". *Automatica* **47**:1, pp. 39–49.

Schröer, K., S. Albright, and M. Grethlein (1997). "Complete, minimal and model-continuous kinematic models for robot calibration". *Robotics and Computer-Integrated Manufacturing* **13**:1, pp. 73–85.

Shan, Y. and K. K. Leang (2009). "Repetitive control with Prandtl-Ishlinskii hysteresis inverse for piezo-based nanopositioning". In: *Proc. American Control Conf. (ACC)*. St. Louis, MO, pp. 301–306.

Sharon, A., N. Hogan, and D. Hardt (1993). "The macro/micro manipulator: an improved architecture for robot control". *Robotics and Computer-Integrated Manufacturing* **10**:3, pp. 209–222.

Sharp, R. S. and H. Peng (2011). "Vehicle dynamics applications of optimal control theory". *Vehicle System Dynamics* **49**:7, pp. 1073–1111.

Shiller, Z. (1994a). "On singular time-optimal control along specified paths". *IEEE Trans. Robot. Autom.* **10**:4, pp. 561–566.

Shiller, Z. (1994b). "Time-energy optimal control of articulated systems with geometric path constraints". In: *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*. Vol. 4, pp. 2680–2685.

Shiller, Z. and S. Sundar (1998). "Emergency lane-change maneuvers of autonomous vehicles". *ASME J. Dynamic Systems, Measurement, and Control* **120**:1, pp. 37–44.

Shin, K. G. and N. D. McKay (1985). "Minimum-time control of robotic manipulators with geometric path constraints". *IEEE Trans. Autom. Control* **30**:6, pp. 531–541.

Shin, K. G. and N. D. McKay (1987). "Robust trajectory planning for robotic manipulators under payload uncertainties". *IEEE Trans. Autom. Control* **32**:12, pp. 1044–1054.

Shuster, M. D. (1993). "Survey of attitude representations". *J. Astronautical Sciences* **41**, pp. 439–517.

Siciliano, B. and L. Villani (1999). *Robot Force Control*. Kluwer Academic Publishers, Dordrecht, The Netherlands.

Siciliano, B., L. Sciavicco, L. Villani, and G. Oriolo (2009). *Robotics: Modelling, Planning and Control*. Springer Verlag, London.

SMErobotics (2015). EU/FP7-project: *The European Robotics Initiative for Strengthening the Competitiveness of SMEs in Manufacturing*, Ref. #287787, Accessed: 2015-06-30. URL: `http://www.smerobotics.org`.

Sørdalen, O. and C. Canudas de Wit (1993). "Exponential control law for a mobile robot: extension to path following". *IEEE Trans. Robot. Autom.* **9**:6, pp. 837–842.

Sörnmo, O., B. Olofsson, U. Schneider, A. Robertsson, and R. Johansson (2012a). "Increasing the milling accuracy for industrial robots using a piezo-actuated high-dynamic micro manipulator". In: *Proc. IEEE/ASME Int. Conf. Advanced Intelligent Mechatronics (AIM)*. Kaohsiung, Taiwan, pp. 104–110.

Sörnmo, O., B. Olofsson, A. Robertsson, and R. Johansson (2012b). "Increasing time-efficiency and accuracy of robotic machining processes using model-based adaptive force control". In: *Proc. 10th Int. IFAC Symp. Robot Control (SYROCO)*. Dubrovnik, Croatia, pp. 543–548.

Sörnmo, O., B. Olofsson, A. Robertsson, and R. Johansson (2013). "Adaptive internal model control for mid-ranging of closed-loop systems with internal saturation". In: *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*. Tokyo, Japan, pp. 4893–4899.

Sörnmo, O., B. Olofsson, A. Robertsson, and R. Johansson (2015). "Learning approach to cycle-time minimization of wood milling using adaptive force control". *ASME J. Manufacturing Science and Engineering*. DOI: `http://dx.doi.org/10.1115/1.4030751`. *Accepted for Publication*.

Spong, M. W., S. Hutchinson, and M. Vidyasagar (2006). *Robot Modeling and Control*. John Wiley & Sons, Hoboken, NJ.

Stolt, A., M. Linderoth, A. Robertsson, and R. Johansson (2011). "Force controlled assembly of emergency stop button". In: *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*. Shanghai, China, pp. 3751–3756.

Sun, Y. and B. Yang (2009). "Compensation of hysteresis nonlinearity for the piezoelectric actuators". In: *Proc. 3rd IEEE Int. Conf. Computer Science and Information Technol. (ICCSIT)*. St. Louis, MO, pp. 307–312.

Sundström, P., M. Jonasson, J. Andreasson, A. Stensson Trigell, and B. Jacobsson (2010). "Path and control optimisation for over-actuated vehicles in two safety-critical maneuvers". In: *Proc. 10th Int. Symp. on Advanced Vehicle Control (AVEC)*. Loughborough, United Kingdom.

Tan, X., H. Tanaka, and Y. Ohta (2012). "Grey-box modeling of rotary type pendulum system with position-variable load". In: *Proc. IFAC Symp. System Identification (SYSID)*. Brussels, Belgium, pp. 1263–1268.

Tanaka, H., Y. Ohta, and Y. Okimura (2011). "A local approach to LPV-identification of a twin rotor MIMO system". In: *Proc. 18th IFAC World Congress*. Milano, Italy, pp. 7749–7754.

Tavernini, D., M. Massaro, E. Velenis, D. I. Katzourakis, and R. Lot (2013). "Minimum time cornering: the effect of road surface and car transmission layout". *Vehicle System Dynamics* **51**:10, pp. 1533–1547.

Thorpe, J. A. (1979). *Elementary topics in differential geometry*. Springer Verlag, Berlin Heidelberg.

Timings, J. P. and D. J. Cole (2013). "Minimum maneuver time calculation using convex optimization". *ASME J. Dynamic Systems, Measurement, and Control* **135**:3, pp. 031015–1–031015–9.

Tisseur, F. and K. Meerbergen (2001). "The quadratic eigenvalue problem". *SIAM Rev.* **43**:10, pp. 235–286.

Tjønnås, J. and T. Johansen (2010). "Stabilization of automotive vehicles using active steering and adaptive brake control allocation". *IEEE Trans. Control Syst. Technol.* **18**:3, pp. 545–558.

Tøndel, P. and T. Johansen (2005). "Control allocation for yaw stabilization in automotive vehicles using multiparametric nonlinear programming". In: *Proc. American Control Conf. (ACC)*. Portland, OR, pp. 453–458.

Törnqvist, D. (2008). *Estimation and Detection with Applications to Navigation*. PhD Thesis No. 1216. Linköping University, Linköping, Sweden.

Tyapin, I., G. Hovland, and T. Brogårdh (2014). "Method for estimating combined controller, joint and link stiffnesses of an industrial robot". In: *Proc. IEEE Int. Symp. Robotic and Sensors Environments (ROSE)*. Timisoara, Romania.

Van Loock, W., G. Pipeleers, and J. Swevers (2013). "Time-optimal path planning for flat systems with application to a wheeled mobile robot". In: *Proc. Workshop Robot Motion and Control (RoMoCo)*. Wasowo, Poland, pp. 192–196.

Van Overschee, P. and B. De Moor (1994). "N4SID: Subspace algorithms for the identification of combined deterministic-stochastic systems". *Automatica* **30**:1, pp. 75–93.

Van Overschee, P. and B. De Moor (1996). *Subspace Identification for Linear Systems—Theory, Implementation, Applications*. Kluwer Academic Publishers, Boston, London, Dordrecht.

Velenis, E. and P. Tsiotras (2005). "Minimum time vs. maximum exit velocity path optimization during cornering". In: *Proc. IEEE Int. Symp. on Industrial Electronics (ISIE)*. Dubrovnik, Croatia, pp. 355–360.

Velenis, E. (2011). "FWD vehicle drifting control: the handbrake-cornering technique". In: *Proc. IEEE Conf. Decision and Control (CDC)*. Orlando, FL, pp. 3258–3263.

Verhaegen, M. and P. Dewilde (1992). "Subspace model identification—The output-error state-space model identification class of algorithms". *Int. J. Control* **56**, pp. 1187–1210.

Vernaza, P. and D. Lee (2006). "Rao-Blackwellized particle filtering for 6-DOF estimation of attitude and position via GPS and inertial sensors". In: *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*. Orlando, FL, pp. 1571–1578.

Verscheure, D., B. Demeulenaere, J. Swevers, J. De Schutter, and M. Diehl (2008). "Time-energy optimal path tracking for robots: a numerically efficient optimization approach". In: *Proc. 10th Int. Workshop Advanced Motion Control*. Trento, Italy.

Verscheure, D., M. Diehl, J. De Schutter, and J. Swevers (2009a). "On-line time-optimal path tracking for robots". In: *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*. Kobe, Japan, pp. 599–605.

Verscheure, D., M. Diehl, J. De Schutter, and J. Swevers (2009b). "Recursive log-barrier method for on-line time-optimal robot path tracking". In: *Proc. American Control Conf. (ACC)*. St. Louis, MI, pp. 4134–4140.

Verscheure, D., B. Demeulenaere, J. Swevers, J. De Schutter, and M. Diehl (2009c). "Time-optimal path tracking for robots: a convex optimization approach". *IEEE Trans. Autom. Control* **54**:10, pp. 2318–2327.

Vuong, N. D., M. H. Ang Jr., T. M. Lim, and S. Y. Lim (2009). "Multi-rate operational space control of compliant motion in robotic manipulators". In: *Proc. IEEE Int. Conf. Systems, Man and Cybernetics*, pp. 3175–3180.

Wächter, A. and L. T. Biegler (2006). "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming". *Mathematical Programming* **106**:1, pp. 25–57.

Wang, J., H. Zhang, and T. Fuhlbrigge (2009). "Improving machining accuracy with robot deformation compensation". In: *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*. St. Louis, MO, pp. 3826–3831.

Wernholt, E. and S. Moberg (2011). "Nonlinear Gray-Box Identification Using Local Models Applied to Industrial Robots". *Automatica* **47**:4, pp. 650–660.

Werth GmbH (2013). *CMM VideoCheck HA400*. Gießen, Germany.

Wigström, O., B. Lennartson, A. Vergnano, and C. Breitholtz (2013). "High-level scheduling of energy optimal trajectories". *IEEE Trans. Autom. Sci. Eng.* **10**:1, pp. 57–64.

Wong, J. (2008). *Theory of Ground Vehicles*. John Wiley & Sons, Hoboken, NJ.

Xsens Inc. (2014). *MTi 100-series*. Data sheet MT0502P.D.

Xu, J.-H. (1993). "Neural network control of a piezo tool positioner". In: *Proc. Canadian Conf. Electrical and Computer Engineering*. Vancouver, Canada, pp. 333–336.

Xu, J.-X., S. K. Panda, and T. H. Lee (2010). *Advances in Industrial Control—Real-time Iterative Learning Control: Design and Applications*. Springer Verlag, London.

Xu, J.-X. (1998). "Direct learning of control efforts for trajectories with different time scales". *IEEE Trans. Autom. Control* **43**:7, pp. 1027–1030.

Yi, J., J. Li, J. Lu, and Z. Liu (2012). "On the stability and agility of aggressive vehicle maneuvers: a pendulum-turn maneuver example". *IEEE Trans. Control Syst. Technol.* **20**:3, pp. 663–676.

Zhang, H., J. Wang, G. Zhang, Z. Gan, Z. Pan, H. Cui, and Z. Zhu (2005). "Machining with flexible manipulator: Toward improving robotic machining performance". In: *Proc. IEEE/ASME Int. Conf. Advanced Intelligent Mechatronics (AIM)*. Monterey, CA, pp. 1127–1132.

A

Vehicle- and Tire-Model Parameters

In this appendix, the model parameters for the vehicle and tire configurations investigated in Chapters 12–13 are provided.

Table A.1 Vehicle-model parameters used in (12.1)–(12.12) for the results presented in Chapter 13.

Notation	Value	Unit
l_f	1.3	m
l_r	1.5	m
w	0.8	m
m	2 100	kg
I_{xx}	765	kgm^2
I_{yy}	3 477	kgm^2
I_{zz}	3 900	kgm^2
R_w	0.3	m
I_w	2.0	kgm^2
σ	0.3	m
g	9.82	ms^{-2}
h	0.5	m
$K_{\phi,f}, K_{\phi,r}$	89 000	Nm(rad)^{-1}
$D_{\phi,f}, D_{\phi,r}$	8 000	Nms(rad)^{-1}
K_θ	363 540	Nm(rad)^{-1}
D_θ	30 960	Nms(rad)^{-1}

Table A.2 Model parameters in (12.13)–(12.21) employed in Chapters 12–13 (originating from [Pacejka, 2006]). The models represent the tire forces on dry asphalt with friction ellipse (FE) and weighting functions (WF), respectively. For DT, the same parameters are used for both the left and the right wheels.

FE	Front	Rear
μ_x	1.20	1.20
B_x	11.7	11.1
C_x	1.69	1.69
E_x	0.377	0.362
μ_y	0.935	0.961
B_y	8.86	9.30
C_y	1.19	1.19
E_y	-1.21	-1.11

WF	Front	Rear
μ_x	1.20	1.20
B_x	11.7	11.1
C_x	1.69	1.69
E_x	0.377	0.362
μ_y	0.935	0.961
B_y	8.86	9.30
C_y	1.19	1.19
E_y	-1.21	-1.11
B_{x1}	12.4	12.4
B_{x2}	-10.8	-10.8
$C_{x\alpha}$	1.09	1.09
B_{y1}	6.46	6.46
B_{y2}	4.20	4.20
$C_{y\kappa}$	1.08	1.08

Table A.3 Tire-model parameters used to represent dry asphalt, wet asphalt, snow, and smooth ice in Chapter 13, using the scaling factors presented in [Braghin et al., 2006].

Parameter	Dry	Wet	Snow	Ice
$\mu_{x,f}$	1.20	1.06	0.407	0.172
$\mu_{x,r}$	1.20	1.07	0.409	0.173
$B_{x,f}$	11.7	12.0	10.2	31.1
$B_{x,r}$	11.1	11.5	9.71	29.5
$C_{x,f}, C_{x,r}$	1.69	1.80	1.96	1.77
$E_{x,f}$	0.377	0.313	0.651	0.710
$E_{x,r}$	0.362	0.300	0.624	0.681
$\mu_{y,f}$	0.935	0.885	0.383	0.162
$\mu_{y,r}$	0.961	0.911	0.394	0.167
$B_{y,f}$	8.86	10.7	19.1	28.4
$B_{y,r}$	9.30	11.3	20.0	30.0
$C_{y,f}, C_{y,r}$	1.19	1.07	0.550	1.48
$E_{y,f}$	-1.21	-2.14	-2.10	-1.18
$E_{y,r}$	-1.11	-1.97	-1.93	-1.08
$C_{x\alpha,f}, C_{x\alpha,r}$	1.09	1.09	1.09	1.02
$B_{x1,f}, B_{x1,r}$	12.4	13.0	15.4	75.4
$B_{x2,f}, B_{x2,r}$	-10.8	-10.8	-10.8	-43.1
$C_{y\kappa,f}, C_{y\kappa,r}$	1.08	1.08	1.08	0.984
$B_{y1,f}, B_{y1,r}$	6.46	6.78	4.19	33.8
$B_{y2,f}, B_{y2,r}$	4.20	4.20	4.20	42.0