



TECHNISCHE
UNIVERSITÄT
WIEN

DISSERTATION

Path and Surface Following Control for Industrial Robotic Applications

Ausgeführt zum Zwecke der Erlangung des akademischen Grades eines
Doktors der technischen Wissenschaften (Dr.techn.)

unter der Leitung von
Univ.-Prof. Dipl.-Ing. Dr.techn. Andreas Kugi
E376
Institut für Automatisierungs- und Regelungstechnik

eingereicht an der
Technischen Universität Wien
Fakultät für Elektrotechnik und Informationstechnik

von
Bernhard Bischof
Matrikelnummer: 1126016

Wien, im Oktober 2018

Studiendekan: Univ.-Prof. Mag.rer.nat. Dr.rer.nat. Gottfried Strasser

Tag des Rigorosums: 23.11.2018

Betreuer: Univ.-Prof. Dipl.-Ing. Dr.techn. Andreas Kugi

Erster Gutachter: Prof. Dr.-Ing. Alin Albu-Schäffer

Zweiter Gutachter: Ao.Univ.Prof. Dipl.-Ing. Dr.techn. Markus Vincze

Vorwort

Die vorliegende Arbeit entstand während meiner Tätigkeit als wissenschaftlicher Mitarbeiter im Rahmen einer Forschungsk Kooperation zwischen dem Institut für Automatisierungs- und Regelungstechnik (ACIN) der Technischen Universität Wien und der KEBA AG. Das Projekt wurde von der Österreichischen Forschungsförderungsgesellschaft (Fördervertrag Nr. 850952) und aus Mitteln des Projektpartners finanziert.

Mein besonderer Dank gilt meinem Doktorvater Univ.-Prof. Andreas Kugi. Du hast mir trotz meines eher ungewöhnlichen Lebenslaufes die Möglichkeit geben eine wissenschaftliche Karriere zu starten und mich stets unterstützt. Außerdem möchte ich mich bei Univ.-Prof. Alin Albu-Schäffer und bei Ao. Univ.-Prof. Markus Vincze für die Erstellung der Gutachten bedanken.

Ich möchte mich auch bei meinen Kolleginnen und Kollegen am ACIN für die Hilfsbereitschaft und anregenden Gespräche bedanken. Ganz besonders möchte ich hier Tobias Glück für seine große Unterstützung vor allem in den ersten beiden Jahren und bei den ersten Publikationen hervorheben. Ohne dich wäre mein Einstieg in die Wissenschaft deutlich schwieriger gewesen. Zusätzlich möchte ich mich bei Stefan Flixeder, Christian Hartl-Nesić, Martin Meiringer und Andreas Deutschmann für die Unterstützung beim Aufbau von Prüfständen, beim Lösen von kniffligen Fragen und bei der Erstellung dieser Arbeit sowie bei Martin Müller für seine geduldige Hilfe bei Problemen mit \LaTeX bedanken.

Ein großer Dank gilt meiner Familie, meinen Freunden und besonders meiner Lebensgefährtin, die mein Leben jeden Tag bereichern und mir viel Kraft geben.

Wien, im Oktober 2018

Bernhard Bischof

Abstract

This work discusses the application of manifold stabilization to industrial robots with a focus on human-robot collaboration. Manifold stabilization aims at stabilizing submanifolds defined in the output space of a dynamical system without any a priori time parametrization. A robot is typically operating in a three-dimensional Euclidean space and thus the stabilization of the end-effector on a path (one-dimensional submanifold) or a surface (two-dimensional submanifold) together with the regulation of the orientation are of particular interest. These special types of manifold stabilization are denoted as path following control (PFC) and surface following control (SFC), respectively.

Novel PFC and SFC approaches for fully actuated manipulators and elastic joint robots in three-dimensional space are proposed. The presented approaches can handle open, closed, and intersecting manifolds parametrized as regular paths or surfaces (e.g., splines) and are based on input-output linearization. The controllers transform the nonlinear robot dynamics into a linear system with decoupled dynamics for the orientation and in tangential and transversal direction with respect to a path or surface. A feasible neighborhood of the path or surface is defined for which a diffeomorphism can be found that maps the generalized coordinates (joint coordinates) to tangential, transversal, and rotational coordinates. A parallel transport frame is used for the design of the PFC, which not only allows to directly cope with paths having zero curvature, but also drastically simplifies the PFC law compared to existing approaches known from literature, which typically rely on the Frenet-Serret frame. In SFC, the special choice of the coordinate transformation ensures that the two tangential states locally represent physically interpretable lengths in orthogonal directions. These properties of the presented PFC and SFC approaches make them highly suitable for industrial robotic applications. In particular a combination of PFC and SFC strategies with compliance control opens up new possibilities for the systematic design of robot operation in contact with the environment and for

human-robot collaboration.

Moreover, it is shown that a large number of virtual fixtures can be systematically generated with the proposed PFC and SFC approaches. Virtual fixtures denote control algorithms that restrict the workspace of a manipulator in physical human-robot interaction tasks, e.g., hand-guiding operation. The presented PFC and SFC approaches allow to independently define the dynamics in tangential and transversal direction to a path or surface in a physically interpretable manner. This feature enables to systematically generate numerous different constraint types like guidance and forbidden region virtual fixtures, hard and soft constraints as well as static and dynamic virtual fixtures and their combinations. This is in contrast to the existing approaches known from literature, which usually cover only few different virtual fixture types. The paths and the surfaces can be defined by splines allowing for a high flexibility to represent different geometries.

Apart from a number of simulation studies, experimental results on the DELTA robot FESTO EXPT-45 and on the 6-axis industrial robot COMAU Racer 1.4 demonstrate the feasibility of the proposed concepts. Amongst others, a semi-automation production use case is shown, where the mounting pins of a heavy plate that is carried by the robot have to be inserted into tight-fitting boreholes by a human operator.

Kurzzusammenfassung

Diese Arbeit beschäftigt sich mit der Stabilisierung von Mannigfaltigkeiten für Industrieroboteranwendungen, wobei der Schwerpunkt auf die Mensch-Roboter Kooperation gelegt wird. Bei der Stabilisierung von Mannigfaltigkeiten werden Submannigfaltigkeiten, welche im Ausgangsraum eines dynamischen Systems definiert sind, ohne festgelegte Zeitparametrierung stabilisiert. Ein Roboter operiert typischerweise in einem dreidimensionalen Euklidischen Raum weshalb die Stabilisierung des Endeffektors entlang eines Pfades (eindimensionale Submannigfaltigkeit) und auf Flächen (zweidimensionale Submannigfaltigkeit) gemeinsam mit der Regelung der Orientierung von besonderem Interesse sind. Diese speziellen Formen der Stabilisierung von Mannigfaltigkeiten werden als Pfadfolgeregelung (Englisch Path Following Control, PFC) und Flächenfolgeregelung (Englisch Surface Following Control, SFC) bezeichnet.

In dieser Arbeit werden neuartige Pfad- und Flächenfolgeregelungskonzepte für vollaktuierte Manipulatoren und Roboter mit elastischen Gelenken vorgestellt. Die Regelungskonzepte basieren auf der Eingangs-/Ausgangslinearisierung und verwenden reguläre Pfade oder Flächen in parametrierter Darstellung (z.B. Splines), welche offen, geschlossen oder selbst schneidend sein können. Die Pfad- und Flächenfolgeregelungen transformieren die nichtlineare Dynamik eines Roboters in ein lineares System mit entkoppelter Dynamik für die Orientierung und in tangentialer und transversaler Richtung in Bezug auf einen Pfad oder eine Fläche. Es wird eine zulässige Umgebung um den Pfad bzw. die Fläche angegeben, in welcher ein Diffeomorphismus gefunden werden kann, der die generalisierten Koordinaten (Gelenkkordinaten) auf die tangentialen und transversalen Koordinaten sowie die Orientierungskordinaten abbildet. Bei der Pfadfolgeregelung wird ein mitbewegtes Koordinatensystem verwendet, welches auf dem Prinzip des Paralleltransportes beruht. Dadurch ist es im Gegensatz zu den in der Literatur üblicherweise verwendeten Koordinatensystemen, welche auf den Frenet-Serret Gleichungen beruhen, kein Problem, Kurven mit Krümmung null zu verwenden

und zusätzlich vereinfacht sich das resultierende Regelgesetz deutlich. Die spezielle Wahl der Koordinatentransformation bei der Flächenfolgeregelung gewährleistet, dass die beiden tangentialen Zustände lokal physikalisch interpretierbare Längen in orthogonalen Richtungen darstellen. Aufgrund dieser Eigenschaften sind die vorgestellten Pfad- und Flächenfolgeregelungskonzepte sehr gut für industrielle Roboteranwendungen geeignet. Insbesondere eine Kombination der Pfad- und Flächenfolgeregelungen mit einer Nachgiebigkeitsregelung eröffnet neue Möglichkeiten für den systematischen Entwurf von Roboteraufgaben im Kontakt mit der Umgebung und für die Mensch-Roboter Kooperation.

Ein weiterer Teil der Arbeit ist der virtuellen Beschränkung des Arbeitsraumes eines Roboters bei einer Mensch-Roboter Kooperation unter Verwendung der präsentierten Pfad- und Flächenfolgeregelungskonzepte gewidmet. Diese Regelungskonzepte erlauben es, die Dynamik entkoppelt in tangentialer und transversaler Richtung bezüglich eines Pfades oder einer Fläche und in einer physikalisch interpretierbaren Art und Weise festzulegen. Dadurch können viele unterschiedliche Arten von virtuellen Arbeitsraumbeschränkungen systematisch implementiert werden. Dies inkludiert Beschränkungen entlang einer Mannigfaltigkeit oder Beschränkung innerhalb eines zulässigen Raumes, harte oder nachgiebige Beschränkungen, sowie statische oder zeitvariante Beschränkungen und deren Kombinationen. Das ist auch der Unterschied zu den bestehenden Arbeiten in der Literatur, die typischerweise nur die Implementierung von wenigen unterschiedlichen Arten von virtuellen Beschränkungen ermöglichen. Die Pfade und Flächen der vorgestellten Regelungskonzepte können mit Splines definiert werden, wodurch eine hohe Flexibilität gewährleistet ist und beliebige Geometrien approximiert werden können.

Neben einer Vielzahl von Simulationsstudien wird die Brauchbarkeit der vorgestellten Konzepte anhand von experimentellen Ergebnissen an dem DELTA Roboter FESTO EXPT-45 und an dem 6-achsigen Industrieroboter COMAU Racer 1.4 demonstriert. Unter anderem wird eine Roboteranwendung gezeigt, bei der die Befestigungsstifte einer schweren, vom Roboter gehaltenen Platte durch einen menschlichen Bediener in eng sitzende Bohrlöcher eingefügt werden.

Contents

Nomenclature	xi
1 Introduction	1
1.1 Manifold stabilization	1
1.2 Physical human-robot interaction	3
1.3 Goal and overview of this work	6
2 Basics of the Mathematical Modeling of Industrial Robots	9
2.1 Kinematics of serial manipulators	10
2.1.1 Direct kinematics	10
2.1.2 Inverse kinematics	12
2.1.3 Manipulator Jacobian	14
2.2 Kinematics of a DELTA robot	16
2.2.1 Direct and inverse kinematics	17
2.2.2 Manipulator Jacobian	19
2.3 Friction effects	20
2.4 Dynamics	21
2.4.1 Kinetic energy	22
2.4.2 Potential energy	24
2.4.3 Equations of motion of serial manipulators with rigid trans- missions	24
2.4.4 Equations of motion of serial manipulators with elastic joints	26
2.4.5 Equations of motion of the Delta robot	28
2.4.6 Base parameter set	29
2.5 Dynamic parameter identification	30
2.5.1 Excitation trajectory	31
2.5.1.1 Periodic function	31

2.5.1.2	Optimization	32
2.5.2	Signal processing	32
2.5.3	Parameter estimation	33
3	Path Following Control	35
3.1	Path assumptions	36
3.2	Control objectives	36
3.3	Orthonormal frame	37
3.4	Projection operator and feasible neighborhood	38
3.5	Path following control for fully actuated manipulators	39
3.5.1	Coordinate transformation	40
3.5.1.1	Tangential subsystem	40
3.5.1.2	Transversal subsystem	40
3.5.1.3	Rotational subsystem	41
3.5.1.4	Diffeomorphism	41
3.5.2	Feedback linearization	42
3.5.3	Stabilization of the linearized system	43
3.5.3.1	Position control	44
3.5.3.2	Compliance control	45
3.5.4	Simplification using joint velocity controllers	48
3.5.4.1	Coordinate transformation	48
3.5.4.2	Feedback linearization	49
3.5.4.3	Stabilization of the linearized system	49
3.6	Path following control for elastic joint robots	50
3.6.1	Robots with visco-elastic joints	50
3.6.1.1	Coordinate transformation	50
3.6.1.2	Feedback linearization	51
3.6.1.3	Stabilization of the linearized system	52
3.6.2	Robots with pure elastic joints	53
3.6.2.1	Coordinate transformation	53
3.6.2.2	Feedback linearization	54
3.6.2.3	Stabilization of the linearized system	55
3.6.3	Simplification using singular perturbation theory	55
3.6.3.1	Singular perturbation approach	55
3.6.3.2	Controller	56
3.7	Implementation	56
3.7.1	Numerical solution of the optimization problem	57
3.7.2	Discretization of the parallel transport frame	57
3.7.3	Path parametrization	59
3.8	Applications	60
3.8.1	Path following control of a Delta robot	61
3.8.2	Compliant path following control of a Delta Robot	62
3.8.3	Path following control of a 3R elastic joint robot	64

3.8.4	Comparison between path following and trajectory tracking control	68
4	Surface Following Control	73
4.1	Surface assumptions	73
4.2	Control objectives	74
4.3	Moving frame	75
4.4	Projection operator and feasible neighborhood	76
4.5	Surface following control for fully actuated manipulators	77
4.5.1	Coordinate transformation	77
4.5.1.1	Tangential subsystem	77
4.5.1.2	Transversal subsystem	78
4.5.1.3	Diffeomorphism	78
4.5.2	Feedback linearization	79
4.5.3	Stabilization of the linearized system	80
4.5.3.1	Position Control	80
4.5.3.2	Compliance control	81
4.5.4	Simplification using joint velocity controllers	82
4.5.4.1	Coordinate transformation	82
4.5.4.2	Feedback linearization	82
4.5.4.3	Stabilization of the linearized system	83
4.6	Surface following control for elastic joint robots	83
4.6.1	Robots with visco-elastic joints	83
4.6.1.1	Coordinate transformation	83
4.6.1.2	Feedback linearization	84
4.6.1.3	Stabilization of the linearized system	84
4.6.2	Robots with pure elastic joints	85
4.6.2.1	Coordinate transformation	85
4.6.2.2	Feedback linearization	85
4.6.2.3	Stabilization of the linearized system	86
4.7	Implementation	86
4.8	Applications	87
4.8.1	Surface following control of a Delta robot	87
4.8.2	Compliant surface following control of an industrial robot	90
4.8.2.1	Measurement results of the compliant SFC	92
4.8.3	Surface following control of a 3R visco-elastic joint robot	93
5	Path and Surface Following Control for Human-Robot Interaction	97
5.1	Generating virtual fixtures with path and surface following control	98
5.1.1	Guidance and forbidden region virtual fixtures	98
5.1.2	Hard and soft constraints	100

5.1.3	Static and dynamic virtual fixtures	101
5.1.4	Summary of the virtual fixture generation with PFC and SFC	101
5.2	Virtual fixture implementation examples	103
5.2.1	Example 1: Static guidance virtual fixtures on a path with soft constraints	103
5.2.2	Example 2: Static guidance virtual fixtures on a path with hard constraints	105
5.2.3	Example 3: Dynamic guidance virtual fixtures on a path with hard constraints	106
5.2.4	Example 4: Static forbidden region virtual fixtures with hard constraints	108
5.3	Semi-automation production use case	111
6	Conclusions and Outlook	117
6.1	Conclusions	118
6.2	Outlook	119
A	Friction Observer	121
B	Base Parameters of the Comau Racer	125
C	Parameter Estimation of the Comau Racer	129
C.1	Excitation trajectory	129
C.2	Estimated parameters	130
	References	135

Nomenclature

The following list presents acronyms, notation, and symbols used in this work.

Acronyms

AdmC	admittance control (position based impedance control)
CAD	computer aided design
DOF	degree of freedom
KF	Kalman filter
PFC	path following control
pHRI	physical human-robot interaction
SFC	surface following control
s.t.	subject to
TFL	transversal feedback linearization
TTC	trajectory tracking control
WLSE	weighted least squares estimation

General Notation

a, ξ, A, Ξ, \dots	scalars
$\mathbf{a}, \boldsymbol{\xi}, \mathbf{A}, \boldsymbol{\Xi}, \dots$	vectors and matrices
$\mathbf{a}^T, \mathbf{A}^T$	transpose of vectors and matrices
\dot{a}	total derivative with respect to time
$\ddot{a}, a^{(3)}, a^{(4)}, \dots$	higher-order total derivatives with respect to time
$\mathbf{a}'(\theta)$	partial derivative of \mathbf{a} with respect to θ
$a_i, a_{c,i}$	i th components of the vectors \mathbf{a} and \mathbf{a}_c with $i \in \mathbb{N}$
$\mathbf{A}[k, j]$	element in the k th row and j th column of the matrix \mathbf{A} with $k, j \in \mathbb{N}$
$\ \cdot\ _2$	Euclidean norm

General Symbols

$\mathbf{0}$	vector or matrix exclusively containing zeros
\mathbf{I}	identity matrix
\mathbf{R}	rotation matrix
\mathbf{H}	homogeneous transformation
\mathbf{q}	vector of generalized coordinates
\mathbf{q}_J	vector of generalized joint coordinates
\mathbf{q}_M	vector of actuator coordinates
$\mathbf{h}(\mathbf{q})$	output function of a dynamic system
\mathbf{y}	system output vector containing position and orientation
\mathbf{y}_t	position part of the output vector
\mathbf{y}_r	orientation part of the output vector
\mathbf{J}	Jacobian
\mathbf{J}_g	manipulator (geometric) Jacobian
\mathbf{J}_a	analytical Jacobian
\mathbf{T}	transformation matrix
ϕ	Euler angle vector
T	kinetic energy
V	potential energy
m	mass
$m_{L,i}$	mass of link i
\mathbf{p}_c	vector to the center of mass
$\bar{\mathbf{I}}$	inertia tensor relative to the center of mass
\mathbf{D}	mass matrix
\mathbf{C}	Coriolis matrix
\mathbf{g}	vector of potential forces
$\boldsymbol{\tau}$	generalized force vector
$\boldsymbol{\tau}_d$	drive force vector
$\boldsymbol{\tau}_{ext}$	contact force vector with the environment
$\boldsymbol{\tau}_f$	friction force vector
f_c	Coulomb friction parameter
f_v	viscous friction parameter
\mathbf{K}_s	joint stiffness matrix
\mathbf{K}_d	joint damping matrix
\mathbf{Y}	regression matrix of the dynamic system
$\boldsymbol{\varrho}$	dynamic system parameter vector
\mathbf{Y}_p	regression matrix of the dynamic system linear in the base parameters
$\boldsymbol{\varrho}_b$	dynamic system base parameter vector
\mathbf{X}	information matrix for parameter identification
$\boldsymbol{\Upsilon}$	covariance matrix of the actuator torque measurements

ω	angular frequency
I	imaginary unit
γ	path
γ_t	position part of a path
\mathcal{S}	surface
\mathcal{S}_t	position part of a surface
θ	path parameter
$\boldsymbol{\theta}$	surface parameter vector
$\boldsymbol{\sigma}(\theta)$	regular path parametrization
$\boldsymbol{\sigma}_t(\theta)$	position part of path parametrization
$\boldsymbol{\sigma}_r(\theta)$	orientation part of path parametrization
$\boldsymbol{\sigma}(\boldsymbol{\theta})$	regular surface parametrization
$\boldsymbol{\sigma}_t(\boldsymbol{\theta})$	position part of surface parametrization
$\boldsymbol{\sigma}_r(\boldsymbol{\theta})$	orientation part of surface parametrization
$\mathbf{e}_{ }$	tangential unit vector onto a path
\mathbf{e}_{\perp}	normal unit vector onto a path or surface
\mathbf{e}_{∇}	second normal unit vector onto a path
$\hat{\mathbf{y}}_p$	virtual PFC output
$\hat{\mathbf{y}}_s$	virtual SFC output
η	tangential coordinate
ξ	transversal coordinate
ζ	rotational coordinate
$\hat{\mathbf{y}}_p$	virtual PFC output
$\hat{\mathbf{y}}_s$	virtual SFC output
$\hat{\mathbf{J}}_p$	PFC Jacobian
$\hat{\mathbf{J}}_s$	SFC Jacobian
\mathbf{v}_p	new control input PFC
\mathbf{v}_s	new control input SFC
m^d	mass of a virtual spring-mass-damper system
d^d	damping of a virtual spring-mass-damper system
k^d	stiffness of a virtual spring-mass-damper system
t	time
t_0	initial time
T_s	sampling time

CHAPTER 1

Introduction

Industrial robots are re-programmable and multi-functional manipulators. They are widespread across different industries with a worldwide operational stock of over 1.8 million industrial robots in 2016 and they aim at increasing productivity and precision as well as minimizing costs [1]. The vast majority of industrial robots perform repetitive tasks in known environments like welding, painting, car body or electronics assembly and material handling [2, 3]. In these repetitive tasks, the motion of the robot is usually preprogrammed (or planned at the beginning of the motion considering input of vision sensors) with a fixed time parametrization. Classical trajectory tracking control is well suited for this type of motions [3–9]. However, classical trajectory tracking control is not suitable for applications in a changing environment or where the robot collaborates with humans. Manifold stabilization is one approach to avoid a fixed preprogrammed time parametrization of the robot’s motion and, therefore, expands the possible types of applications.

1.1 Manifold stabilization

Manifold stabilization, which is a generalization of set point stabilization, aims at stabilizing submanifolds such as paths and surfaces defined in the output space of a dynamical system without any a priori time parametrization. Early work in this field was done by Samson [10] and Banaszuk and Hauser [11]. Based on the work of Banaszuk and Hauser and using input-output feedback linearization, Nielsen and Maggiore introduced the so called Transverse Feedback Linearization (TFL) for input-affine systems in [12, 13]. In TFL, a controlled invariant submanifold of the state space is stabilized. If the system fulfills some sufficient conditions formulated in [13], the dynamics transversal to the submanifold can be linearized

via static state feedback. A manifold stabilization approach for quadratic flat systems was introduced in [14], where not only the dynamics transversal to the submanifold are linearized, but also the dynamics in tangential direction, which simplifies the tangential control design. The approaches presented in [12–14] require an implicit representation of the submanifold to be stabilized.

A robot is typically operating in a three-dimensional Euclidean space and thus the stabilization of the end-effector on a path (one-dimensional submanifold) or a surface (two-dimensional submanifold) together with the regulation of the orientation are of particular interest. Manifold stabilization is beneficial compared to trajectory tracking control when the desired motion of the robot is not known in advance, e.g., because it is instantaneously defined by an operator or it depends on environmental changes.

TFL was used in [15] to solve the path following problem for a five degrees-of-freedom magnetically levitated positioning system. References [16–19] are concerned with the design of path following controllers for mechanical systems that can be either under- or fully actuated. It was shown that by applying the TFL to fully actuated rigid body systems, the tangential subsystem is linearized as well. The path following controller design method presented in [16–19] requires both the parametrized and the (closed form) implicit representation of the path.

To find a closed form of the implicit representation of a path or surface can be a laborious task or even impossible. In robotic applications, the paths and surfaces can be of arbitrary geometric shape and it is therefore advantageous to use parametrized representations.

A concept for the design of a path following controller for planar problems using TFL, which only depends on the parametrized path representation, was proposed in [20]. In this work, an orthonormal frame with respect to a parametrized curve is constructed and the first transversal state is chosen as the projection of the shortest distance to the path onto the normal unit vector.

In three-dimensional Euclidean space, the orthonormal frame with respect to a path is not unique. A common way to construct orthonormal unit vectors is given by the Frenet-Serret frame, see, e.g., [21]. The work [22] extends the path following controller design using the parametric path representation presented in [20] to the three-dimensional Euclidean space. TFL and the Frenet-Serret frame are used to handle paths parametrized by splines.

The Frenet-Serret frame can only be uniquely defined at points on the path where its curvature is nonzero. Moreover, the normal vectors become discontinuous when passing points with zero curvature, see, e.g., [21]. In [23], the Frenet-Serret frame was improved by introducing a signed curvature to overcome these drawbacks. However, in practical applications one might be interested in following a path with a curvature close to zero. In this case, the Frenet-Serret frame used in [22] and also the frame proposed in [23] give rise to extremely high changing rates of the normal vectors leading to a chattering in the control law.

An orthonormal frame with respect to the path, which depends on the idea of

relatively parallel fields, was proposed by Bishop in [24]. A normal vector field is said to be relatively parallel along a curve, if its derivative is tangential. This frame, which is often referred to as parallel transport frame [25], is of class \mathcal{C}^1 for regular \mathcal{C}^2 curves even at points with zero curvature.

A stiff stabilization of the submanifold is unfeasible in applications where the robot dynamically interacts with the environment by exchanging mechanical work, e.g. in physical human-robot interaction (pHRI). In this case, the control laws that stabilize the transformed dynamics in tangential and transversal direction to the submanifold have to provide a compliant behavior because pure position control would lead to damages of the robot and/or its environment. A compliant control law for robots was introduced by Hogan with impedance control in the 1980s, where the robot in the closed-loop system behaves like a virtual mass-spring-damper system [26]. Pure force control is another approach to establish compliance developed in the 1980s, which aims at following a predefined contact force reference. However, damping is difficult to implement in pure force control due to the typically high measurement noise of force sensors, see, e.g., [27]. Since then, much research has been conducted regarding contact stability and the combination of force and position control [28–34].

In position-based impedance control (admittance control), the compliance is realized by tracking the trajectory of the exponentially stable reference impedance model using position control in an inner control loop and an impedance control in the outer loop [29]. As shown in [32], this method is well suited for accurate positioning in free space as well as for contact situations with rigid environments. Due to the possibility of using high gains in the inner loop, the position-based impedance control is rather insensitive to model uncertainties.

1.2 Physical human-robot interaction

Full industrial automation can become very complex and expensive, in particular in changing environments. This is why a large number of complex production steps are still performed fully manually by human workers or the manual work is partly assisted by specially designed manipulators, e.g., when heavy workpieces have to be handled. The manipulators are usually custom made for one specific task and, hence, expensive and hardly adaptable to production changes. Semi-automation can be achieved by collaboration of a robot with a human operator by means of physical human-robot interaction (pHRI). Thereby, the manipulator is replaced by a robot, which is programmable and therefore usable in various tasks. In pHRI, the human and the robot share the same workspace and come in contact with each other. In industrial semi-automation, pHRI is usually realized by a so called hand-guiding collaborative operation, which is defined in the technical specification ISO/TS 15066. Thereby, an operator utilizes a hand-operated device, e.g., including a force sensor, to guide the motion of a robot without effort and the

robot carries all the payload. This type of human-robot collaboration combines the advantages of robots, e.g., strength and endurance, with those of humans, e.g., sensitivity, using special control algorithms. This not only helps to speed up the production process but also reduces the worker's risk for repeated trauma disorders [35, 36].

Safety is a critical issue in pHRI especially when heavy and stiff industrial robots are involved. Injuries with very high severity can occur in the case of a collision between the robot and the human even with active collision detection and reaction strategies [37, 38]. The safety of the operator can be increased by limiting the workspace of the robot to a so called restricted space using control methods like virtual fixtures [39, 40].

Virtual fixtures restrict the workspace of a manipulator by means of control algorithms. They have been used in hand-guiding cooperative human-robot tasks in the automobile industry since the late 1990s [41]. Virtual fixtures are also common in teleoperation and hand-guiding operation in robotically assisted surgery [40], [42].

Six principal methods to generate virtual fixtures (active constraints) can be identified in literature [43]; these are (i) simple functions of constraint proximity, (ii) potential fields, (iii) non-energy storing constraints, (iv) constrained joint optimization, (v) reference direction fixtures, and (vi) passive constraint enforcing mechanisms. The virtual fixtures can either be guidance constraints, where the motion is restricted to a specific manifold like a path, or forbidden-region constraints, where the motion is free unless a forbidden region is entered [40]. In both cases, the constraints can be soft or hard. Soft constraints allow for some deviation while hard constraints limit the motion to the virtual fixture.

- (i.) Guidance constraints [44] and forbidden-region constraints [45] can be generated by simple functions of constraint proximity. Only soft constraints can be achieved within this approach, because the constraint force vector is a linear function of the closest distance to the constraint manifold effectively emulating a spring. In [45], Abbot and Okamura investigated the stability of the control law for a linear system, where the human operator is modeled as a linear and time-invariant mass-spring-damper system. They concluded that the stability of the closed-loop system decreases with an increasing stiffness of the constraint and the constraint cannot be made arbitrarily stiff.
- (ii.) Potential fields can also be used to establish virtual fixtures, where areas in the workspace with low potential are attractive and areas with high potential are repulsive. In [46], the potential field approach was employed to generate forbidden-region constraints for collision avoidance. At each point in the workspace, the gradient of the potential field of all sources has to be calculated to determine the resulting force that pulls the robot away from the forbidden regions. Also guidance constraints can be generated using attractive fields resulting in a control law very similar to the method

of simple functions described above.

- (iii.) A non-energy storing constraint was introduced in [47] by using simulated plasticity, which is modeled as Coulomb friction. The initial collision with the constraint is thereby stiff until a certain force into the restricted area is applied. When penetrating the constraint, energy is only dissipated and no energy is stored. Hence, no force is applied by the control law to recover the penetration. According to the authors, the non-energy storing feature can increase the safety for various applications. Some effort was made to deal with the discontinuity of the plasticity. A virtual proxy is introduced on which the plasticity takes effect. The proxy is then coupled to the haptic device or manipulator via a spring and a damper. This reduces the discontinuity problem but adds some (small) stored potential energy. Bowyer and Rodriguez y Baena improved this approach significantly in [48–50]. Friction redirection was introduced to assist the operator in recovering from penetrations of the constraint. Additionally, their approach allows for time-variant constraints and they showed that their control law is dissipative even for combined translational and rotational constraints.
- (iv.) Constrained joint optimization is used since the early 1990s to establish virtual fixtures for surgical robots, which can also be redundant [42]. A constrained optimization problem is solved to compute the new reference velocities of the joints at each sampling instance. A cost function is minimized that represents the difference between reference velocities given from the operator and new reference velocities satisfying the constraints. The constraints can include the virtual fixtures as well as mechanical and dynamic limits of the joints. Linear constraints for point fixtures are given in [42], which are extended to line and plane fixtures in [51]. With this method, the constraints are probably not satisfied in between the sampling instants. Therefore, the sampling intervals have to be relatively short compared to the maximum velocity of the manipulator. To find optimal solutions that fulfill nonlinear constraints can be a challenging task. Hence, the numerical implementation has to be carried out very carefully for each application to ensure an appropriate and stable behavior.
- (v.) In [52], reference direction fixtures were introduced to establish constrained hand-guided operation. The input force of an operator is thereby projected onto the tangential direction or onto the tangential plane of the constraint manifold and is used as velocity reference for the servo controllers. This restricts the motion of the robot parallel to the manifold. The constraint can be made soft by adding a fraction of the operators force orthogonal to the manifold in the control law. When the robot is off the manifold, the direction of the force projection is modified to guide the operator towards the manifold. A stability analysis is carried out for the simple case of a linear two-dimensional manipulator. However, even in this case, stability can only be proven when the robot is exactly on the manifold. Castillo-Cruces and

Wahrburg [53] added a proportional error term in the control law to make the manifold attractive and also extended the algorithm to six degrees of freedom.

- (vi.) In passive constraint enforcing mechanisms, the actuation force is applied by a human operator and the control architecture is only able to limit or redirect the motion. Therefore, these mechanisms are naturally safer than actively driven methods but their applicability is very limited and they are not suitable for teleoperation. An algorithm to achieve hard guidance constraints on a curve for wheeled passive robots was introduced in [54]. This concept was extended to active manipulators in [39] using continuously variable transmissions (CVT). With these CVTs the ratios of the angular velocities of the manipulator's joints are controlled such that only one degree of freedom is left for the end-effector that satisfies the guidance constraints. However, the control law is not defined for zero velocity and becomes ill-conditioned when the velocity is orthogonal to the curve.

The methods described above to establish virtual fixtures typically neglect the dynamics of the manipulators and consider only the kinematics. This is the main reason why, in general, closed-loop stability cannot be proven. One exception is the work in [45], where a linear dynamic system is considered and a proof of stability is given.

1.3 Goal and overview of this work

The purpose of this work is to present a novel, simple and efficient manifold stabilization approach for industrial robots and to combine it with compliance control. Thereby, arbitrary parametrized paths (one-dimensional manifolds) and surfaces (two-dimensional manifolds) can be handled. The combination of the manifold stabilization approach with compliance control allows to separately define the impedance in tangential and orthogonal direction to the manifold. With this approach, a large number of different virtual fixtures for pHRI can be systematically generated.

The proposed path following control approach (PFC) is based on input-output linearization and can handle open, closed, and intersecting paths in the three-dimensional space parametrized as regular smooth curves, which is in contrast to most of the existing works, cf. Section 1.1. Using a parallel transport frame for the design of the path following controller not only allows to directly cope with paths having zero curvature, it also drastically simplifies the path following control law compared to, e.g., [22]. The PFC approach is adapted to design controllers that stabilize parametrized, regular, and smooth surfaces, which extends the works [13, 14], where an implicit representation is required. This so called surface following control (SFC) linearizes and transforms the dynamics of a (nonlinear) system into two orthogonal directions in the tangential plane and the orthogonal direction

onto the surface.

The systematic generation of a large number of virtual fixtures by a combination of PFC/SFC with compliance control is presented; i.e., guidance or forbidden-region virtual fixtures that can either be soft or hard and the constraints can also be time variant. The behavior along and away from the virtual fixtures can, thereby, be defined in a physically interpretable manner. Additionally, in contrast to the approaches mentioned in Section 1.2, the PFC/SFC approach allows to systematically prove the closed-loop stability.

The remainder of this work is organized as follows: Chapter 2 summarizes the basics for the mathematical modeling of the industrial robots considered in this work. This comprises, the kinematics, the dynamics and the dynamic parameter estimation for serial and parallel robots.

A novel PFC approach for fully actuated manipulators as well as for elastic joint robots is introduced in Chapter 3. This includes the path assumptions, the control objectives, the orthonormal frame, and the projection operator, which are presented in Section 3.1 - 3.4. The coordinate transformation and the stabilization of the linearized system for fully actuated manipulators as well as for elastic joint robots are presented in Section 3.5 and Section 3.6. Section 3.7 gives insights into the implementation of the PFC on a digital computer and Section 3.8 presents the application of the proposed PFC approach to a DELTA robot with three linear drives and to a six-axis industrial robot.

Chapter 4 is devoted to the SFC approach for fully actuated manipulators as well as for elastic joint robots. The control objectives, the moving frame, and the projection operator are introduced in Section 4.1 - 4.4. The coordinate transformation and the stabilization of the linearized system for fully actuated manipulators are presented in Section 4.5 and for elastic joint robots in Section 4.6. In Section 4.7, some implementation issues are considered. Applications of the proposed SFC approach to a DELTA robot and to a six-axis industrial robot are given in Section 4.8.

The systematic generation of virtual fixtures for pHRI using PFC and SFC is introduced in Chapter 5. In Section 5.1, it is described in detail how guidance and forbidden region virtual fixtures with either soft or hard constraints and static or dynamic virtual fixtures are generated and which control laws are required for each case. Four experiments on a six-axis industrial robot are presented in Section 5.2 for a validation of the approach. Section 5.3 shows the application of the virtual fixture approach to a semi-automated production use-case, where a plate is picked up from a magazine, handled to a frame and assembled into the frame using a human-robot collaboration.

Finally, Chapter 6 concludes this work and gives an outlook on future research topics.

Basics of the Mathematical Modeling of Industrial Robots

Modeling of industrial robots has been an important research area in the last 30 years. A lot of specialized scientific journals, conferences and textbooks deal with this topic. This chapter summarizes some relevant basics of the vast literature regarding kinematics [3–5, 55, 56], dynamics [3–5, 55–59], and dynamic parameter identification [5, 60–65]

Mathematical models are crucial to understand and control the behavior of physical systems like robots. Robots consist of links that are connected by joints, which are usually of revolute or prismatic type and driven by actuators. In this work, it is assumed that the robot's links are rigid bodies and that the actuators are realized as electric motors with rotating rotors. The mathematical modeling of manipulators can be subdivided into kinematics and dynamics.

Kinematics describes the motion of bodies and ignores the forces and torques that cause the motion. In robotics, the direct kinematics describes the geometric relation between the joint space (i.e. coordinates of the joints \mathbf{q}_J) and the operational space (i.e. position \mathbf{y}_t and orientation \mathbf{y}_r of the end-effector)

$$\mathbf{y} = \begin{bmatrix} \mathbf{y}_t \\ \mathbf{y}_r \end{bmatrix} = \mathbf{h}(\mathbf{q}_J) . \quad (2.1)$$

Literature regarding kinematics differentiate between serial and parallel manipulators. The links of serial manipulators form an open kinematic chain and universal approaches to derive the direct kinematics as well as the manipulator Jacobian, which describes the differential kinematics, are available [3, 4]. Parallel manipulators consist of a closed kinematic chain. With this type of manipulators, the direct kinematics is in general complex to obtain, and it is not guaranteed that an algebraic expression even exists [55, 56].

Dynamics describe the relation between (generalized) forces and the resulting motion and are in robotics usually formulated in the joint space. Two different methods to derive the dynamic equations are common, i.e., the Euler-Lagrange and the Newton-Euler formulation. The Euler-Lagrange formulation offers a conceptually simple and systematic approach to derive the model based on the kinetic and potential energy and is utilized in this work. The Newton-Euler approach, which is based on the balance of all forces acting on a rigid body, yields the model in recursive form and is often used for numerical calculation.

Two different types of robots are used for the experiments throughout this work, i.e., a DELTA robot consisting of a closed kinematic chain with three translational degrees of freedom of the type FESTO EXPT-45 and a six-axis serial robot with revolute joints of the type COMAU Racer 1.4. Their mathematical models are derived and explained in detail in this chapter.

2.1 Kinematics of serial manipulators

This section is concerned with the derivation of the direct and inverse kinematics as well as the manipulator Jacobian of the industrial robot COMAU Racer 1.4.

2.1.1 Direct kinematics

The direct kinematics determines the position and orientation (pose) of the end-effector as a function of the joint coordinates. This can be done in a systematic and simple way by using the homogeneous transformation

$$\mathbf{H}_i^j = \begin{bmatrix} \mathbf{R}_i^j & \mathbf{d}_i^j \\ \mathbf{0} & 1 \end{bmatrix}. \quad (2.2)$$

The transformation \mathbf{H}_i^j combines rotations and translations from the coordinate system j to the coordinate system i , with the orthogonal rotation matrix \mathbf{R}_i^j to transform a vector expressed in j to the coordinate system i and \mathbf{d}_i^j is the vector from the origin of i to the origin of j expressed in i . The transformation from a coordinate system k to the coordinate system i is given by

$$\mathbf{H}_i^k = \mathbf{H}_i^j \mathbf{H}_j^k. \quad (2.3)$$

Pure rotations by the angle ϕ about the local axis $i \in \{x, y, z\}$ are denoted by $\mathbf{H}_{Ri,\phi}$ and pure translations in direction of the local axis i by length s are denoted by $\mathbf{H}_{Ti,s}$.

The Denavit-Hartenberg (DH) convention has become the standard for the description of the geometry of industrial robots. Thereby, two rotations and two translations represent the four homogeneous transformations for each joint i and can be written as

$$\mathbf{H}_{i-1}^i = \mathbf{H}_{Rz,\theta_i} \mathbf{H}_{Tz,d_i} \mathbf{H}_{Tx,a_i} \mathbf{H}_{Rx,\alpha_i}, \quad (2.4)$$

with the parameters θ_i , a_i , d_i , and α_i . The z_i -axis coincides with the axis of joint $i + 1$. Hence, θ_i is the variable (coordinate) for rotational joints and d_i for prismatic joints.

The direct kinematics of a rigid robot manipulator with N joints can then easily be calculated in the form

$$\mathbf{H}_{dk} = \mathbf{H}_0^N = \mathbf{H}_0^1 \mathbf{H}_1^2 \dots \mathbf{H}_{N-1}^N \begin{bmatrix} \mathbf{I} & [x_{N,T}, y_{N,T}, z_{N,T}]^T \\ \mathbf{0} & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R}_e & \mathbf{y}_t \\ \mathbf{0} & 1 \end{bmatrix}, \quad (2.5)$$

with the end-effector position $\mathbf{y}_t = \mathbf{d}_0^N = \mathbf{H}_{dk}[1 \dots 3, 4]$ expressed in the inertial frame $(0, x_0, y_0, z_0)$ and the orientation $\mathbf{R}_e = \mathbf{R}_0^N = \mathbf{H}_{dk}[1 \dots 3, 1 \dots 3]$, where $[x_{N,T}, y_{N,T}, z_{N,T}]^T$ is the tool offset defined in the end-effector frame (N, x_N, y_N, z_N) .

In the case of three rotational degrees of freedom, a minimal representation of the orientation can be obtained by three parameters like the Euler angles $\boldsymbol{\phi} = [\varphi, \vartheta, \psi]^T$. All minimal representations suffer from representation singularities, which have to be taken into account. The ZYZ Euler angle definition

$$\mathbf{R}_{ZYZ}(\boldsymbol{\phi}) = \mathbf{R}_{z,\varphi} \mathbf{R}_{y,\vartheta} \mathbf{R}_{z,\psi} = \begin{bmatrix} c_\varphi c_\vartheta c_\psi - s_\varphi s_\psi & -c_\varphi c_\vartheta s_\psi - s_\varphi c_\psi & c_\varphi s_\vartheta \\ s_\varphi c_\vartheta c_\psi + c_\varphi s_\psi & -s_\varphi c_\vartheta s_\psi + c_\varphi c_\psi & s_\varphi s_\vartheta \\ -s_\vartheta c_\psi & s_\vartheta s_\psi & c_\vartheta \end{bmatrix} \quad (2.6)$$

is widely used in robotics with representation singularities at $\vartheta = i\pi$ and $i \in \mathbb{Z}$, where s_α and c_α are abbreviations for $\sin \alpha$ and $\cos \alpha$, respectively and $\mathbf{R}_{j,\alpha}$, $j \in \{x, y, z\}$, is the rotation matrix for rotations through the angle α with respect to the (local) j -axis. Another common convention is the ZYX Euler angle definition

$$\mathbf{R}_{ZYX}(\boldsymbol{\phi}) = \mathbf{R}_{z,\varphi} \mathbf{R}_{y,\vartheta} \mathbf{R}_{x,\psi} = \begin{bmatrix} c_\varphi c_\vartheta & c_\varphi s_\vartheta s_\psi - s_\varphi c_\psi & c_\varphi s_\vartheta c_\psi + s_\varphi s_\psi \\ s_\varphi c_\vartheta & s_\varphi s_\vartheta s_\psi + c_\varphi c_\psi & s_\varphi s_\vartheta c_\psi - c_\varphi s_\psi \\ -s_\vartheta & c_\vartheta s_\psi & c_\vartheta c_\psi \end{bmatrix}, \quad (2.7)$$

with representation singularities at $\vartheta = \pi/2 + i\pi$ and $i \in \mathbb{Z}$. The inverse problem is to determine the Euler angles $\boldsymbol{\phi}(\mathbf{R})$ corresponding to a given rotation matrix

$$\mathbf{R} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}.$$

For example $\varphi = \text{atan2}(r_{23}, r_{13})$ holds for the ZYZ Euler angle definition under the assumption that $|r_{13}| + |r_{23}| > 0$, cf. (2.6). See, e.g., [3] for the calculation of the remaining Euler angles.

The direct kinematics $\mathbf{h}(\cdot)$ can then be defined as

$$\mathbf{y} = \begin{bmatrix} \mathbf{y}_t \\ \boldsymbol{\phi}_e(\mathbf{R}_e) \end{bmatrix} = \mathbf{h}(\mathbf{q}_J), \quad (2.8)$$

with \mathbf{y}_t and \mathbf{R}_e according to (2.5), the Euler angles of the end-effector ϕ_e , and the vector of the joint coordinates $\mathbf{q}_J \in \mathbb{R}^N$. Note that the orientation in the three dimensional space can also be represented by four parameters, e.g. quaternions, that are free from representation singularities, but are not considered in this work.

The COMAU Racer 1.4 consists of six rotational joints, hence $N = 6$. Its schematic and the coordinate systems in DH convention are depicted in Fig. 2.1 and Tab. 2.1, respectively. Inserting the robot's DH parameters of Tab. 2.1 into (2.5) with the joint coordinates $\mathbf{q}_J^T = [\theta_1, \dots, \theta_6]$ and using (2.8) immediately yields the direct kinematics $\mathbf{y} = \mathbf{h}_{CR}(\mathbf{q}_J) \in \mathbb{R}^6$.

Table 2.1: DH parameters of the COMAU Racer 1.4.

i	d_i	a_i	α_i	i	d_i	a_i	α_i
1	0.43m	0.15m	$\frac{\pi}{2}$	4	0.684m	0	$-\frac{\pi}{2}$
2	0	0.59m	0	5	0	0	$\frac{\pi}{2}$
3	0	0.13m	$\frac{\pi}{2}$	6	0.1m	0	0

2.1.2 Inverse kinematics

The inverse kinematics deals with the determination of the joint coordinates \mathbf{q}_J as a function of the end-effector position and orientation (pose). It is more complex than the direct kinematics problem, because the equations to be solved are nonlinear,

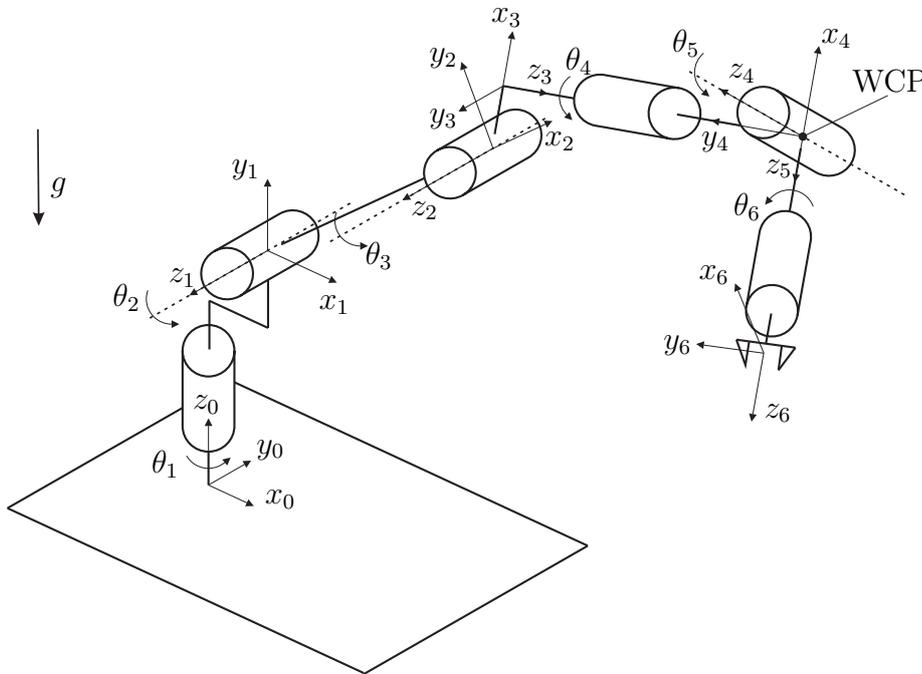


Figure 2.1: Schematic of the COMAU Racer 1.4.

multiple solutions might exist, and there might be no admissible solutions for some poses. No general approach exists to derive the inverse kinematics for serial manipulators. Thus in this work, only the specific class of six-axis serial manipulators with rotational joints and a spherical wrist will be considered. The COMAU Racer 1.4 is part of this specific manipulator class. A spherical wrist implies that the rotation axes of the last three joints intersect at a single point (wrist center point). Therefore, the position and orientation problem can be considered separately and solved in three steps.

First, the position of the wrist center point (WCP) is calculated based on the orientation and geometry of the last link with

$$\mathbf{y}_{WCP} = \mathbf{y}_t - \mathbf{R}_e \begin{bmatrix} x_{6T} \\ y_{6T} \\ z_{6T} + d_6 \end{bmatrix} \quad (2.9)$$

and the tool offset $[x_{6,T}, y_{6,T}, z_{6,T}]^T$. Second, the geometric approach in [4] is used to determine the first three joint angles, which is based on the position of the WCP. In this work, the inverse kinematics is only given for one configuration of the robot, which is used throughout the experiments. For more details on the derivation and the inverse kinematics for different robot configurations, the reader is referred to [4]. The coordinate of the first joint θ_1 can be determined by the x - and y -component of $\mathbf{y}_{WCP}^T = [y_{WCP,x}, y_{WCP,y}, y_{WCP,z}]$ with

$$\theta_1 = \text{atan2}(y_{WCP,y}, y_{WCP,x}) , \quad (2.10a)$$

because the axes of the second and third joint are horizontal. Equation (2.10a) is undefined for $y_{WCP,x} = y_{WCP,y} = 0$ and the manipulator is then in a singular configuration. The determination of θ_2 and θ_3 is reduced to a planar problem that can be solved with the law of cosines. For the elbow up configuration, the angles are given by

$$\theta_2 = \arctan\left(\frac{s_{ik}}{r_{ik}}\right) + \text{atan2}\left(b_{ik} \cos(\theta_3 + \gamma_{ik}), a_2 + b_{ik} \sin(\theta_3 + \gamma_{ik})\right) \quad (2.10b)$$

and

$$\theta_3 = \text{atan2}\left(\sqrt{1 - c_{ik}^2}, c_{ik}\right) - \gamma_{ik} - \frac{\pi}{2} , \quad (2.10c)$$

with $r_{ik} = \sqrt{y_{WCP,x}^2 + y_{WCP,y}^2} - a_1$, $s_{ik} = y_{WCP,z} - d_1$, $b_{ik} = \sqrt{a_3^2 + d_4^2}$, $\gamma_{ik} = \arctan(a_3/d_4)$, and $c_{ik} = (a_2^2 + b_3^2 - r_{ik}^2 - s_{ik}^2)/(2a_2b_3)$.

Third, the last three joint angles result from the orientation between the third link and the end-effector \mathbf{R}_3^6 , which is the upper left 3×3 -matrix of the homogeneous transformation $\mathbf{H}_3^6 = \mathbf{H}_3^4 \mathbf{H}_4^5 \mathbf{H}_5^6$. Hence, solving the equation $\mathbf{R}_3^6 = (\mathbf{R}_0^3)^T \mathbf{R}_e$ yields for $0 < \theta_5 < \pi$, see, e.g., [4]

$$\theta_4 = \text{atan2}(\mathbf{R}_3^6[2, 3], \mathbf{R}_3^6[1, 3]) , \quad (2.10d)$$

$$\theta_5 = \arctan \left(\sqrt{\mathbf{R}_3^6[1, 3]^2 + \mathbf{R}_3^6[2, 3]^2}, \mathbf{R}_3^6[3, 3] \right), \quad (2.10e)$$

$$\theta_6 = \text{atan2}(\mathbf{R}_3^6[3, 2], -\mathbf{R}_3^6[3, 1]), \quad (2.10f)$$

where $\mathbf{R}_3^6[i, j]$ denotes the element in i -th row and j -th column of the matrix $\mathbf{R}_3^6 = (\mathbf{R}_0^3)^T \mathbf{R}_e$. If $\theta_5 = k\pi$, $k \in \mathbb{Z}$, the axes of the joints four and six (z_3 and z_5) are collinear, the matrix elements $\mathbf{R}_3^6[1, 3]$, $\mathbf{R}_3^6[2, 3]$, $\mathbf{R}_3^6[3, 1]$, and $\mathbf{R}_3^6[3, 2]$ equal to zero, and θ_4 as well as θ_6 are undefined. Hence, $\theta_5 = k\pi$, $k \in \mathbb{Z}$, is another singular configuration.

2.1.3 Manipulator Jacobian

The manipulator (geometric) Jacobian \mathbf{J}_g describes the relationship between the joint space velocities $\dot{\mathbf{q}}_J \in \mathbb{R}^N$ and the velocities in the operational space with

$$\begin{bmatrix} \dot{\mathbf{y}}_t \\ \boldsymbol{\omega}_e \end{bmatrix} = \begin{bmatrix} \mathbf{J}_t \\ \mathbf{J}_\omega \end{bmatrix} \dot{\mathbf{q}}_J = \mathbf{J}_g \dot{\mathbf{q}}_J. \quad (2.11)$$

In (2.11), $\dot{\mathbf{q}}_J$ contains the time derivatives of the joint coordinates, i.e. $\dot{\theta}_i$ or \dot{d}_i , respectively, for $i = 1, \dots, N$, and $\boldsymbol{\omega}_e$ is the angular velocity vector of the end-effector expressed in the inertial frame $(0, x_0, y_0, z_0)$. The cross product of the unit vector of one rotation axis with the vector to the end-effector describes the translational velocity of the end-effector that is caused by the movement of the joint rotating about this axis. Therefore, the translational Jacobian can be calculated as, see, e.g., [3]

$$\mathbf{J}_t = [\mathbf{j}_{t,1} \quad \mathbf{j}_{t,2} \quad \dots \quad \mathbf{j}_{t,N}], \quad (2.12)$$

with the 3×1 vectors

$$\mathbf{j}_{t,i} = \mathbf{e}_{z,i-1}^0 \times (\mathbf{d}_0^N - \mathbf{d}_0^{i-1}), \quad (2.13)$$

where $\mathbf{e}_{z,k}^0$ is the unit vector of the z -axis of the frame k expressed in the inertial frame (last column of \mathbf{R}_0^k).

The rotational Jacobian can simply be defined as

$$\mathbf{J}_\omega = [\mathbf{e}_{z,0}^0 \quad \mathbf{e}_{z,1}^0 \quad \dots \quad \mathbf{e}_{z,N-1}^0], \quad (2.14)$$

with

$$\boldsymbol{\omega}_e = \mathbf{J}_\omega \dot{\mathbf{q}}_J.$$

The 6×6 manipulator Jacobian \mathbf{J}_g for the COMAU Racer 1.4 can then be calculated by (2.12) and (2.14) for $N = 6$ and $\mathbf{q}_J^T = [\theta_1, \theta_2, \dots, \theta_6]$ using (2.5) with the DH parameters of Tab. 2.1.

The orientation of the end-effector is often represented in Euler angles $\boldsymbol{\phi}_e$. With the transformation

$$\boldsymbol{\omega}_e = \mathbf{T}_i \dot{\boldsymbol{\phi}}_e, \quad (2.15)$$

a relationship between the angular velocities $\boldsymbol{\omega}_e$ and the time derivative of the Euler angles $\dot{\boldsymbol{\phi}}_e$ is given, where i represents the Euler angle definition, e.g., $i = ZYZ$ or $i = ZYX$ according to (2.6) or (2.7), respectively. Hence, the velocities of the operational space with minimal orientation representation is given by

$$\dot{\mathbf{y}} = \begin{bmatrix} \dot{\mathbf{y}}_t \\ \dot{\boldsymbol{\phi}}_e \end{bmatrix} = \mathbf{J}_a \dot{\mathbf{q}}_J, \quad (2.16)$$

with the so called analytical Jacobian

$$\mathbf{J}_a = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{T}_i^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{J}_t \\ \mathbf{J}_\omega \end{bmatrix}. \quad (2.17)$$

For the ZYZ representation (2.6), the angular velocities $\boldsymbol{\omega}_e^T = [\omega_{e,x}, \omega_{e,y}, \omega_{e,z}]$ resulting from each Euler angle derivative are given by, see [3]

$$\begin{aligned} \boldsymbol{\omega}_e(\dot{\varphi}) &= \dot{\varphi}[0, 0, 1]^T, \\ \boldsymbol{\omega}_e(\dot{\vartheta}) &= \dot{\vartheta}[-\sin \varphi, \cos \varphi, 0]^T, \\ \boldsymbol{\omega}_e(\dot{\psi}) &= \dot{\psi}[\cos \varphi \sin \vartheta, \sin \varphi \sin \vartheta, \cos \vartheta]^T, \end{aligned} \quad (2.18)$$

and, therefore, the matrix \mathbf{T}_{ZYZ} reads as

$$\mathbf{T}_{ZYZ} = \begin{bmatrix} 0 & -\sin \varphi & \cos \varphi \sin \vartheta \\ 0 & \cos \varphi & \sin \varphi \sin \vartheta \\ 1 & 0 & \cos \vartheta \end{bmatrix}, \quad (2.19)$$

with the determinant $\det(\mathbf{T}_{ZYZ}) = -\sin(\vartheta)$. Hence, matrix \mathbf{T}_{ZYZ} is singular at the representation singularities $\vartheta = k\pi$, $k \in \mathbb{Z}$. The angular velocities resulting from each ZYX Euler angle derivative are given by

$$\begin{aligned} \boldsymbol{\omega}_e(\dot{\varphi}) &= \dot{\varphi}[0, 0, 1]^T, \\ \boldsymbol{\omega}_e(\dot{\vartheta}) &= \dot{\vartheta}[-\sin \varphi, \cos \varphi, 0]^T, \\ \boldsymbol{\omega}_e(\dot{\psi}) &= \dot{\psi}[\cos \varphi \cos \vartheta, \sin \varphi \cos \vartheta, -\sin \vartheta]^T. \end{aligned} \quad (2.20)$$

Hence, matrix \mathbf{T}_{ZYX} is given by

$$\mathbf{T}_{ZYX} = \begin{bmatrix} 0 & -\sin \varphi & \cos \varphi \cos \vartheta \\ 0 & \cos \varphi & \sin \varphi \cos \vartheta \\ 1 & 0 & -\sin \vartheta \end{bmatrix}, \quad (2.21)$$

with $\det(\mathbf{T}_{ZYX}) = -\cos(\vartheta)$ and singular points at the representation singularities $\vartheta = \pi/2 + k\pi$, $k \in \mathbb{Z}$. The time derivative of a rotation matrix \mathbf{R} reads as, see, e.g., [3]

$$\dot{\mathbf{R}} = \mathbf{S}(\boldsymbol{\omega})\mathbf{R}, \quad (2.22)$$

with the angular velocity $\boldsymbol{\omega}$ of the frame \mathbf{R} with respect to the reference frame and the skew-symmetric matrix operator $\mathbf{S}(\cdot)$, which performs the cross product $\mathbf{S}(\mathbf{a})\mathbf{b} = \mathbf{a} \times \mathbf{b}$.

2.2 Kinematics of a DELTA robot

In this work, the robot FESTO EXPT-45 is used for experiments, which is a DELTA robot with three translational degrees of freedom introduced in [66]. No universal approach exists to solve the direct kinematics problem for parallel manipulators and analytic expressions only exist for special cases [56]. Hence, this section is concerned only with this special type of parallel robot.

The kinematics of a DELTA robot with rotary drives are discussed in, e.g., [55, 67–69]. However, the FESTO EXPT-45 features linear drives (prismatic joints) and its kinematics is derived in the following. Fig. 2.2 shows a schematic diagram of the considered DELTA robot with linear drives. A DELTA robot has three translational and no rotational degrees of freedom. The robot basically consists

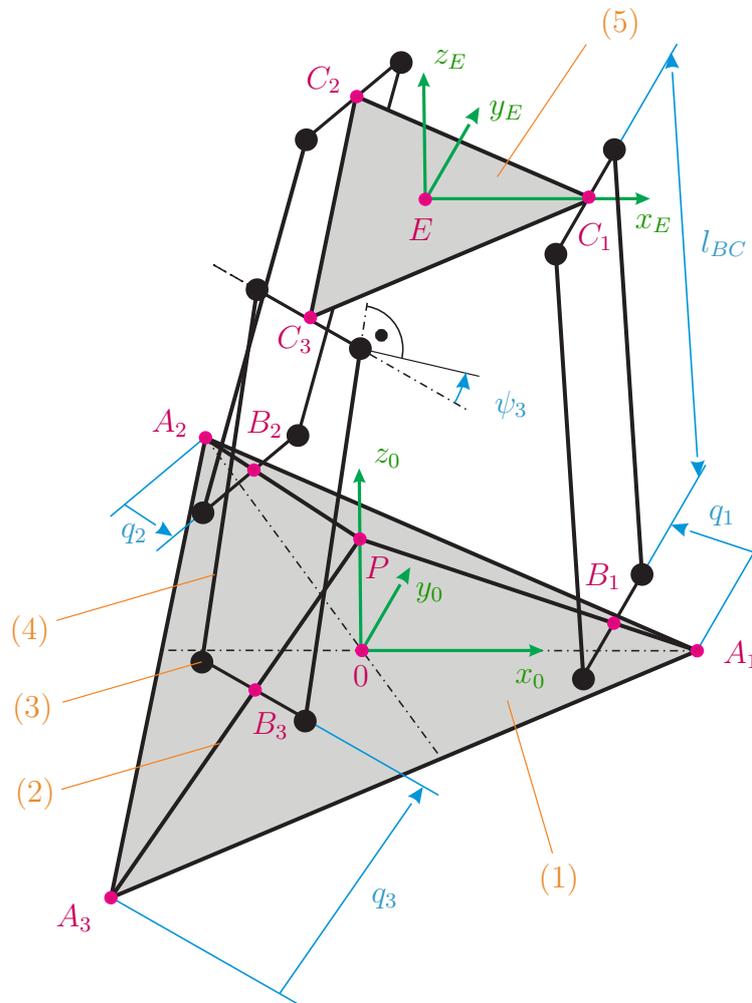


Figure 2.2: Schematic diagram of the DELTA robot FESTO EXPT-45 with linear drives. [70] © 2017 IEEE

of a base plate (1), the end-effector plate (5), three parallelogram arms (4), and three electric linear drives (2). The linear drives are symmetrically arranged. They are mounted at the points A_i , $i = 1, 2, 3$, at the base plate and at a common joint point P . The three parallelogram arms are fixed to the slides of the linear drives at B_i , $i = 1, 2, 3$, and the end-effector plate at the points C_i , $i = 1, 2, 3$. The rods of the parallelogram (4), with length l_{BC} , are on both sides attached via ball joints (3). The angles ψ_i , $i = 1, 2, 3$, denote the orthogonal shears in each parallelogram plane, i.e., the reduction of the initially right angle of the parallelograms. The inertial coordinate system is given by $(0, x_0, y_0, z_0)$ with the center of area of the base plate 0 as origin and the x_0 -axis pointing from 0 to A_1 . A second coordinate system is defined by (E, x_E, y_E, z_E) with the center of area of the end-effector plate E as origin. The coordinate systems are not rotated against each other. The position of the slides B_i with respect to A_i along the segment $\overline{A_i P}$ serve as generalized coordinates q_i , $i = 1, 2, 3$. Hence, the vector of generalized coordinates reads as $\mathbf{q}^T = \mathbf{q}_J^T = [q_1, q_2, q_3] \in \mathbb{R}^3$. The end-effector position $\mathbf{y}_t^T = [y_x, y_y, y_z] \in \mathbb{R}^3$ is defined as the vector from the origin 0 to E expressed in the inertial coordinate system. In Fig. 2.3, the dimensions of one single robot arm are shown. The coordinate systems $(0, x_{0i}, y_{0i}, z_{0i})$, $i = 1, 2, 3$, are introduced, which are equal to the inertial frame $(0, x_0, y_0, z_0)$ rotated by an angle α_{1i} with respect to the z_0 -axis, where $\alpha_{11} = 0$ rad, $\alpha_{12} = 2\pi/3$ rad, and $\alpha_{13} = 4\pi/3$ rad. Moreover, the distances $\overline{0P} = h_P$, $\overline{EC_i} = l_{EC}$, and $\overline{0A_i} = l_A$ and the angles ϕ_i , $i = 1, 2, 3$ are introduced. The parameters of the FESTO EXPT-45 are listed in Tab. 2.2, where $m_{s,i}$, $i = 1, 2, 3$, denotes the mass of a single slide, m_p the mass of a pair of rods, m_e the mass of the end-effector plate, and m_l the load mass.

Table 2.2: Nominal model parameters of the FESTO EXPT-45.

Symbol	Value	Unit	Symbol	Value	Unit
l_{BC}	0.5	m	$m_{s,i}$	1.0	kg
l_A	0.416	m	m_e	1.0	kg
l_{EC}	0.06	m	m_p	0.15	kg
h_P	0.325	m	m_l	0.7	kg

2.2.1 Direct and inverse kinematics

Unique solutions for the direct and inverse kinematics are derived based on the approach of [67]. According to Fig. 2.2 and Fig. 2.3, the end-effector position in the coordinate systems $(0, x_{0i}, y_{0i}, z_{0i})$ is given by

$$\overrightarrow{(0E)}_{0i} = \overrightarrow{(0A_i)}_{0i} + \overrightarrow{(A_i B_i)}_{0i} + \overrightarrow{(B_i C_i)}_{0i} + \overrightarrow{(C_i E)}_{0i}, \quad (2.23)$$

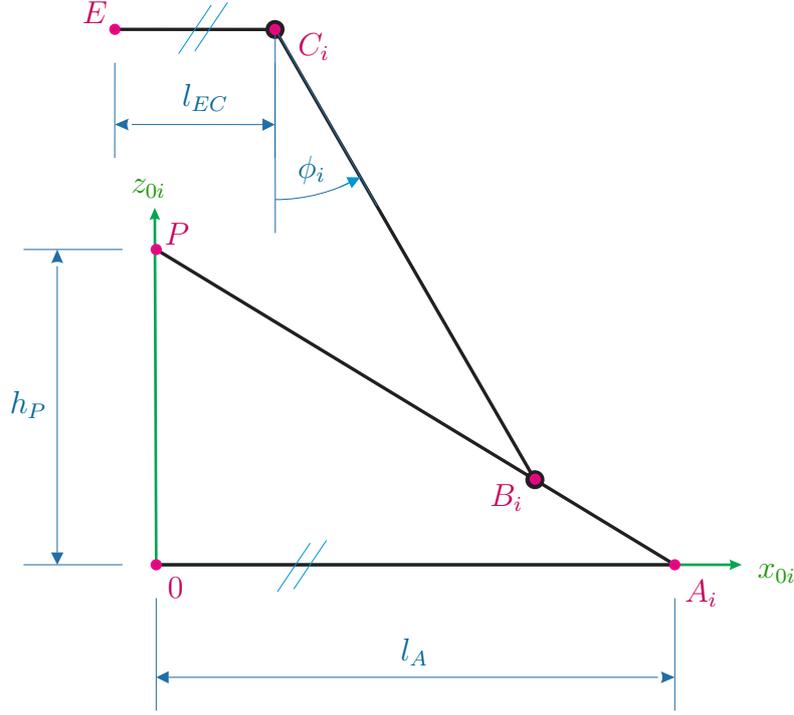


Figure 2.3: Single arm of the DELTA robot FESTO EXPT-45 with linear drives.

with $\overrightarrow{(0E)}_{0i} = \mathbf{R}_{z,\alpha_{1i}}^T \mathbf{y}_t$, $\overrightarrow{(0A_i)}_{0i} = l_A \mathbf{e}_{xi}$, $\overrightarrow{(A_i B_i)}_{0i} = q_i k_D (h_P \mathbf{e}_{zi} - l_A \mathbf{e}_{xi})$, $\overrightarrow{(B_i C_i)}_{0i} = l_{BC} \mathbf{R}_{y,\phi_i}^T \mathbf{R}_{x,\psi_i}^T \mathbf{e}_{zi}$, and $\overrightarrow{(C_i E)}_{0i} = l_{EC} \mathbf{e}_{xi}$. Herein, $k_D = 1/\sqrt{l_A^2 + h_P^2}$ and \mathbf{e}_{ji} , $j \in \{x, y, z\}$, $i \in \{1, 2, 3\}$, denote the Cartesian orthonormal unit vectors in the frame $(0, x_{0i}, y_{0i}, z_{0i})$. Solving (2.23) for $\overrightarrow{(B_i C_i)}_{0i}$, squaring, and adding the rows allows to eliminate the angles ϕ_i and ψ_i and to derive the quadratic equation

$$y_x^2 + y_y^2 + y_z^2 + d_i y_x + e_i y_y + f_i y_z + g_i = 0, \quad (2.24)$$

where

$$\begin{aligned} d_i &= 2(q_i k_D l_A - l) \cos(\alpha_{1i}), & e_i &= 2(q_i k_D l_A - l) \sin(\alpha_{1i}), \\ f_i &= -2q_i k_D h_P, & g_i &= q_i^2 - l_{BC}^2 + l^2 - 2q_i k_D l_A l, \end{aligned}$$

with $l = l_A - l_{EC}$. By subtracting (2.24) for $i = 1$ from (2.24) for $i = 2$ and $i = 3$, respectively, the quadratic parts of the components of \mathbf{y}_t are eliminated and the components y_x and y_y can be expressed as

$$y_x = \frac{y_z a_1 + b_1}{d} \quad \text{and} \quad y_y = \frac{y_z a_2 + b_2}{d}, \quad (2.25)$$

with

$$\begin{aligned} a_1 &= e_1(f_2 - f_3) + e_2(f_3 - f_1) + e_3(f_1 - f_2) \\ b_1 &= e_1(g_2 - g_3) + e_2(g_3 - g_1) + e_3(g_1 - g_2) \\ a_2 &= d_1(f_3 - f_2) + d_2(f_1 - f_3) + d_3(f_2 - f_1) \\ b_2 &= d_1(g_3 - g_2) + d_2(g_1 - g_3) + d_3(g_2 - g_1) \\ d &= d_1(e_2 - e_3) + d_2(e_3 - e_1) + d_3(e_1 - e_2). \end{aligned}$$

By inserting (2.25) into (2.24) for $i = 1$, we get a quadratic equation for y_z . Because of the construction of the robot $y_z > h_P$ holds and the unique solution of this quadratic equation is given by

$$y_z = \frac{-c_1 + \sqrt{c_1^2 - 4c_2c_0}}{2c_2}, \quad (2.26)$$

with

$$\begin{aligned} c_2 &= \frac{a_1^2 + a_2^2}{d^2} + 1 \\ c_1 &= 2\frac{a_1b_1 + a_2b_2}{d^2} + \frac{a_1d_1 + a_2e_1}{d} + f_1 \\ c_0 &= \frac{b_1^2 + b_2^2}{d^2} - \frac{b_1d_1 + b_2e_1}{d} + g_1. \end{aligned}$$

The equations (2.25) and (2.26) represent the unique solution of the direct kinematics $\mathbf{y}_t = \mathbf{h}_D(\mathbf{q})$. Moreover, since (2.24) is a quadratic function of q_i and $0 \leq q_i < \sqrt{l_A^2 + h_P^2}$ holds, solving for q_i yields

$$q_i = -\frac{c_{1,i}}{2} - \sqrt{\left(\frac{c_{1,i}}{2}\right)^2 - c_{0,i}}, \quad (2.27)$$

with

$$\begin{aligned} c_{1,i} &= 2k_D(l_A(y_{x,0i} - l) - h_P y_z) \\ c_{0,i} &= y_x^2 + y_y^2 + y_z^2 - 2ly_{x,0i} + l^2 - l_{BC}^2, \end{aligned} \quad (2.28)$$

$l = l_A - l_{EC}$, and $y_{x,0i} = \cos(\alpha_{1i})y_x + \sin(\alpha_{1i})y_y$. Hence, the inverse kinematics $\mathbf{q} = \mathbf{h}_D^{-1}(\mathbf{y}_t)$ is given by (2.27).

2.2.2 Manipulator Jacobian

The vector $\overrightarrow{B_i C_i}$ in the frame $(0, x_0, y_0, z_0)$ reads as

$$\mathbf{w}_i = \overrightarrow{(B_i C_i)_0} = q_i \mathbf{c}_i - l \mathbf{R}_{z_0, \alpha_{1i}} \mathbf{e}_{xi} + \mathbf{y}_t, \quad (2.29)$$

with unit vector $\mathbf{c}_i = k_D \overrightarrow{PA_i}_0 = k_D \mathbf{R}_{z_0, \alpha_{1i}} (l_A \mathbf{e}_{xi} - h_P \mathbf{e}_{zi})$. The vector (2.29) has constant length l_{BC} and, thus, has to fulfill the constraint

$$\mathbf{w}_i^T \mathbf{w}_i - l_{BC}^2 = 0. \quad (2.30)$$

Taking the time derivative of (2.30) yields

$$\mathbf{w}_i^T \dot{\mathbf{w}}_i = 0, \quad \text{where} \quad \dot{\mathbf{w}}_i = \dot{q}_i \mathbf{c}_i + \dot{\mathbf{y}}_t. \quad (2.31)$$

In matrix notation, (2.31) reads as

$$\mathbf{W}(\mathbf{q}, \mathbf{y}_t) \dot{\mathbf{y}}_t + \mathbf{P}(\mathbf{q}, \mathbf{y}_t) \dot{\mathbf{q}} = \mathbf{0}, \quad (2.32)$$

with matrices $\mathbf{W}^T(\mathbf{q}, \mathbf{y}_t) = [\mathbf{w}_1, \mathbf{w}_2, \mathbf{w}_3]$ and $\mathbf{P}(\mathbf{q}, \mathbf{y}_t) = \text{diag}([\mathbf{w}_1^T \mathbf{c}_1, \mathbf{w}_2^T \mathbf{c}_2, \mathbf{w}_3^T \mathbf{c}_3])$. The matrices \mathbf{W} and \mathbf{P} are nonsingular as long as not at least two rods, or vectors $\overrightarrow{B_i C_i}$, are parallel. This is normally prevented due to limitations of the movement range of the slides. The nonsingular manipulator Jacobian of the DELTA robot with linear drives is, thus, obtained from (2.32) as

$$\mathbf{J}_D(\mathbf{q}) = \frac{\partial \mathbf{h}_D}{\partial \mathbf{q}} = - \left(\mathbf{W}^{-1}(\mathbf{q}, \mathbf{y}_t) \mathbf{P}(\mathbf{q}, \mathbf{y}_t) \right) \Big|_{\mathbf{y}_t = \mathbf{h}_D(\mathbf{q})}. \quad (2.33)$$

2.3 Friction effects

Friction occurs between two surfaces in contact. Numerous different more or less complex methods exist to model these effects. The friction effects can play a dominant role in mechanical systems like robots that are driven by electric motors with gears or hydraulic and pneumatic cylinders and the compensation of friction may be required for high precision and high performance motion control [71].

In control applications, two different types of friction models are typically used, namely static and dynamic models. For static models, a static relationship between the actual velocity and the friction force is employed. Coulomb and viscous friction as well as stick-slip effects can be covered [72]. Dynamic models contain one or more states and can additionally model the elastic/plastic pre-sliding motion, which occurs at small displacements, as well as effects at varying velocities [72]. Generally, dynamic models require more parameters than static ones and the dynamic parameters are often hard to identify. One of the most common dynamic friction model is the so called LuGre model, which includes the Stribeck effect as well as rate dependent friction phenomena like varying break-away force and frictional lag and can be parametrized by six variables [73]. At constant velocities the static and dynamic friction models are equivalent.

In this work, a simple static friction model of the form

$$\tau_f(v) = f_c \text{sign}(v) + f_v v \quad (2.34)$$

is used, with the relative (angular) velocity v between the two surfaces in contact, the Coulomb friction parameter f_c , the viscous friction parameter f_v , and the signum function $\text{sign}(v)$.

Friction parameters depend on normal forces, temperature, lubrication condition, wear of the bearings and gear, etc. and are, thus, changing during operation. Hence, on-line parameter adaptation is advantageous to improve the quality of friction compensation. In Appendix A, an observer for the Coulomb and viscous friction parameters based on the generalized momentum is presented.

2.4 Dynamics

The Euler-Lagrange equations for a system consisting of m generalized coordinates are given by

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{q}_i} - \frac{\partial L}{\partial q_i} = \tau_i, \quad i = 1, \dots, m, \quad (2.35)$$

with the Lagrangian $L = T - V$, where T denotes the kinetic energy, V the potential energy, and τ_i is the generalized force acting on the generalized coordinate q_i . Provided that the potential energy V is only a function of $\mathbf{q}^T = [q_1, q_2, \dots, q_m]$ and the kinetic energy can be expressed as $T = \frac{1}{2} \dot{\mathbf{q}}^T \mathbf{D}(\mathbf{q}) \dot{\mathbf{q}}$, which is the case for manipulators with rigid links, the Euler-Lagrange equations can be written in vector form as

$$\mathbf{D}(\mathbf{q}) \ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau}, \quad (2.36)$$

with the generalized forces $\boldsymbol{\tau}^T = [\tau_1, \tau_2, \dots, \tau_m]$. In (2.36), $\mathbf{D}(\mathbf{q}) \in \mathbb{R}^{m \times m}$ denotes the symmetric positive definite generalized mass matrix, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{q}} \in \mathbb{R}^m$ represents the centrifugal and Coriolis forces, and $\mathbf{g}(\mathbf{q}) \in \mathbb{R}^m$ is the vector of potential forces.

The Christoffel symbols of the first kind are defined as

$$c_{ijk} = \frac{1}{2} \left(\frac{\partial d_{kj}}{\partial q_i} + \frac{\partial d_{ki}}{\partial q_j} - \frac{\partial d_{ij}}{\partial q_k} \right), \quad (2.37)$$

where d_{ij} is the $(i, j)^{\text{th}}$ element of the mass matrix $\mathbf{D}(\mathbf{q})$. The $(k, j)^{\text{th}}$ element of the Coriolis matrix $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ is then given by

$$\mathbf{C}[k, j] = \sum_{i=1}^m c_{ijk} \dot{q}_i. \quad (2.38)$$

Due to the symmetry of $\mathbf{D}(\mathbf{q})$ and the definition of $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ with (2.38), the matrix

$$\mathbf{N}(\mathbf{q}, \dot{\mathbf{q}}) = \dot{\mathbf{D}}(\mathbf{q}) - 2\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \quad (2.39)$$

is skew-symmetric, cf., e.g., [3]. Hence, the notable relationship

$$\dot{\mathbf{D}}(\mathbf{q}) = \mathbf{C}^T(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \quad (2.40)$$

holds.

In robotics, it is beneficial to subdivide the generalized forces into drive forces $\boldsymbol{\tau}_d \in \mathbb{R}^m$, friction forces $\boldsymbol{\tau}_f(\dot{\mathbf{q}}) \in \mathbb{R}^m$, and contact forces with the environment $\boldsymbol{\tau}_{ext} \in \mathbb{R}^m$, hence, $\boldsymbol{\tau} = \boldsymbol{\tau}_d - \boldsymbol{\tau}_f(\dot{\mathbf{q}}) + \boldsymbol{\tau}_{ext}$. Using the static friction model (2.34), the generalized friction force reads as

$$\boldsymbol{\tau}_f(\dot{\mathbf{q}}) = \mathbf{F}_c \text{sign}(\dot{\mathbf{q}}) + \mathbf{F}_v \dot{\mathbf{q}} , \quad (2.41)$$

where $\mathbf{F}_c = \text{diag}(\mathbf{f}_c)$, $\mathbf{f}_c^T = [f_{c,1}, f_{c,2}, \dots, f_{c,m}]$, $\mathbf{F}_v = \text{diag}(\mathbf{f}_v)$, $\mathbf{f}_v^T = [f_{v,1}, f_{v,2}, \dots, f_{v,m}]$. The dynamics (2.36) can then be rewritten as

$$\mathbf{D}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{n}(\mathbf{q}, \dot{\mathbf{q}}) = \boldsymbol{\tau}_d + \boldsymbol{\tau}_{ext} , \quad (2.42)$$

with $\mathbf{n}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) + \boldsymbol{\tau}_f(\dot{\mathbf{q}})$. Although $\boldsymbol{\tau}_d \in \mathbb{R}^m$ holds, note that the degree of actuation of the manipulators considered in this work is equal to the number of joints N . Hence, $\boldsymbol{\tau}_d$ contains N independent applied generalized forces.

Parallel manipulators are subject to holonomic constraints due to the closed kinematic chain. The equations of motion of a system with holonomic constraints $\mathbf{f}_h(\mathbf{q}) = \mathbf{0}$, see, e.g., (2.30) for the DELTA robot, are given by, cf. [57],

$$\mathbf{D}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) + \boldsymbol{\Gamma}_h^T(\mathbf{q})\boldsymbol{\lambda} = \boldsymbol{\tau} , \quad (2.43)$$

with $\boldsymbol{\Gamma}_h(\mathbf{q}) = \frac{\partial \mathbf{f}_h(\mathbf{q})}{\partial \mathbf{q}}$ and the Lagrange multipliers $\boldsymbol{\lambda}$.

2.4.1 Kinetic energy

The kinetic energy stored in each body is a function of its translational and angular velocity. In robotics, the moving bodies are the links and the actuators and the total kinetic energy is given by

$$T = \sum_{i=1}^N T_{L,i} + T_{M,i} , \quad (2.44)$$

where $T_{L,i}$ denotes the kinetic energy of link i and $T_{M,i}$ of actuator i , $i = 1, \dots, N$.

The translational component of the kinetic energy for a single link i is given by

$$T_{L,t,i} = \frac{1}{2} m_{L,i} \dot{\mathbf{p}}_{L,c,i}^T \dot{\mathbf{p}}_{L,c,i} , \quad (2.45)$$

with the link mass $m_{L,i}$ and the vector to the center of mass $\mathbf{p}_{L,c,i}$ expressed in the inertial frame. The rotational part is calculated by

$$T_{L,r,i} = \frac{1}{2} \boldsymbol{\omega}_i^T \mathbf{R}_0^i \bar{\mathbf{I}}_{L,i} (\mathbf{R}_0^i)^T \boldsymbol{\omega}_i , \quad (2.46)$$

where $\boldsymbol{\omega}_i$ is the angular velocity of link i expressed in the inertial frame and $\bar{\mathbf{I}}_{L,i}$ represents the inertia tensor of link i with respect to its center of mass expressed

in the body-fixed coordinate frame of link i . Hence, $\mathbf{R}_0^i \bar{\mathbf{I}}_{L,i} (\mathbf{R}_0^i)^\top$ is the inertia tensor of link i transformed to the inertial frame by the rotation matrix \mathbf{R}_0^i . Since the kinetic energy of the links is expressed in the form $T_L = \frac{1}{2} \dot{\mathbf{q}}_J^\top \mathbf{D}_L(\mathbf{q}_J) \dot{\mathbf{q}}_J$, with the joint coordinates \mathbf{q}_J as a subset of the generalized coordinates \mathbf{q} , $\dot{\mathbf{p}}_{L,c,i}$ and $\boldsymbol{\omega}_i$ have to be replaced by the joint velocities $\dot{\mathbf{q}}_J$. With the translational Jacobian $\mathbf{J}_{t,c,i} = \partial \mathbf{p}_{L,c,i} / \partial \mathbf{q}_J$ of the center of mass of link i , we obtain

$$T_{L,t,i} = \frac{1}{2} m_{L,i} \dot{\mathbf{q}}_J^\top \mathbf{J}_{t,c,i}^\top \mathbf{J}_{t,c,i} \dot{\mathbf{q}}_J \quad (2.47)$$

and with the rotational Jacobian of link i , see also (2.14),

$$\mathbf{J}_{\omega,i} = [\mathbf{e}_{z,0}^0 \ \cdots \ \mathbf{e}_{z,i-1}^0 \ \mathbf{0}_{3 \times N-i}] \quad (2.48)$$

the rotational kinetic energy follows as

$$T_{L,r,i} = \frac{1}{2} \dot{\mathbf{q}}_J^\top \mathbf{J}_{\omega,i}^\top \mathbf{R}_0^i \bar{\mathbf{I}}_{L,i} (\mathbf{R}_0^i)^\top \mathbf{J}_{\omega,i} \dot{\mathbf{q}}_J . \quad (2.49)$$

The kinetic energy of all links can then be written as

$$T_L = \frac{1}{2} \dot{\mathbf{q}}_J^\top \mathbf{D}_L(\mathbf{q}_J) \dot{\mathbf{q}}_J , \quad (2.50)$$

with the positive definite mass matrix

$$\mathbf{D}_L(\mathbf{q}_J) = \sum_{i=1}^N m_{L,i} \mathbf{J}_{t,c,i}^\top \mathbf{J}_{t,c,i} + \mathbf{J}_{\omega,i}^\top \mathbf{R}_0^i \bar{\mathbf{I}}_{L,i} (\mathbf{R}_0^i)^\top \mathbf{J}_{\omega,i} . \quad (2.51)$$

In this work, it is assumed that the contribution of the kinetic energy of the stator of the actuator i (the fixed part) is included in the kinetic energy $T_{L,j}$ of the link j on which actuator i is located. Thus, only the kinetic energy of the rotor of the actuators is considered in the following.

The kinetic energy of rotor i , $i = 1, \dots, N$, is given by

$$T_{M,i} = \frac{1}{2} m_{M,i} \dot{\mathbf{p}}_{M,c,i}^\top \dot{\mathbf{p}}_{M,c,i} + \frac{1}{2} \boldsymbol{\omega}_{M,i}^\top \mathbf{R}_0^{M,i} \bar{\mathbf{I}}_{M,i} (\mathbf{R}_0^{M,i})^\top \boldsymbol{\omega}_{M,i} , \quad (2.52)$$

with the rotor mass $m_{M,i}$, the vector to the center of mass $\mathbf{p}_{M,c,i}$ expressed in the inertial frame, the angular velocity $\boldsymbol{\omega}_{M,i}$, the inertia tensor of the rotor $\bar{\mathbf{I}}_{M,i}$, and the rotation matrix from the motor frame to the inertial frame $\mathbf{R}_0^{M,i}$.

For the actuators it is assumed that the center of mass of the rotor is located on the rotation axis, the z -axis of the rotor frame corresponds to the rotation axis, and the rotor is axially symmetric. Hence, the position of the center of mass $\mathbf{p}_{M,c,i}$ of rotor i is independent of its rotation angle and the inertia tensor of the rotor i is given by the diagonal matrix $\bar{\mathbf{I}}_{M,i} = \text{diag}([I_{M,xx,i}, I_{M,yy,i}, I_{M,zz,i}])$. The vector $\mathbf{p}_{M,c,i}$ then only depends on the joint coordinates \mathbf{q}_J , the relation $\dot{\mathbf{p}}_{M,c,i} = \mathbf{J}_{t,M,i} \dot{\mathbf{q}}_J$,

with the Jacobian $\mathbf{J}_{t,M,i} = \partial \mathbf{p}_{M,c,i} / \partial \mathbf{q}_J$ holds, and the translational kinetic energy of of rotor i follows as

$$T_{M,t,i} = \frac{1}{2} m_{M,i} \dot{\mathbf{q}}_J^T \mathbf{J}_{t,M,i}^T \mathbf{J}_{t,M,i} \dot{\mathbf{q}}_J . \quad (2.53)$$

The calculation of the rotational kinetic energy of the rotors as a function of \mathbf{q} and $\dot{\mathbf{q}}$ depends on which link j the motor i is connected to and if the transmissions (gears) are rigid or elastic. This will be discussed in more detail in the following sections.

2.4.2 Potential energy

In this work, the potential energy caused by gravity is given by

$$V_g = \sum_{i=1}^N m_{L,i} \mathbf{g}_e^T \mathbf{p}_{L,c,i} + m_{M,i} \mathbf{g}_e^T \mathbf{p}_{M,c,i} , \quad (2.54)$$

with $\mathbf{g}_e = [0, 0, g]^T$ and the gravitational acceleration g always pointing in negative z_0 -direction. Potential energy can also be stored in springs, e.g., elastic joints, and reads as

$$V_{s,ni} = \int_{s_0}^s f_s(\tilde{s}) d\tilde{s} , \quad (2.55)$$

with the relaxed spring length s_0 and the elastic force $f_s(s)$ as a function of the length s . For elastic joint robots, the elasticity between the motors and the links is usually modeled as a linear spring using $f_s(s) = k_s(s - s_0)$, with the spring constant k_s [7, 9]. The displacement of the spring for joint i is then given by the difference between the joint and the motor coordinate $q_{J,i} - q_{M,i}$. Hence, the stored energy in the elastic joints follows as

$$V_s = \frac{1}{2} (\mathbf{q}_J - \mathbf{q}_M)^T \mathbf{K}_s (\mathbf{q}_J - \mathbf{q}_M) , \quad (2.56)$$

with the vector of the link coordinates \mathbf{q}_J , the vector of the motor coordinates at the output side \mathbf{q}_M , and the diagonal, positive definite stiffness matrix $\mathbf{K}_s = \text{diag}([k_{s,1}, k_{s,2}, \dots, k_{s,N}])$. The sum of the gravitational energy V_g and elastic energy V_s gives the total potential energy $V = V_g + V_s$. The vector of the generalized forces caused by the potential energy is then computed as

$$\mathbf{g}^T = \frac{\partial V}{\partial \mathbf{q}} . \quad (2.57)$$

2.4.3 Equations of motion of serial manipulators with rigid transmissions

Rigid transmissions have a constant velocity relationship between link i and rotor i , which reads as $\dot{q}_{J,i} = k_{r,i} \dot{\gamma}_{M,i}$, with the gear reduction ratio $k_{r,i}$ and $\dot{\gamma}_{M,i}$ as

the angular velocity of rotor i . For fully actuated serial manipulators with rigid transmissions, $m = N$ holds and the generalized coordinates are usually chosen as the joint coordinates, i.e., $\mathbf{q} = \mathbf{q}_J$. Since the angular velocity of rotor j connected to link i is given by $\boldsymbol{\omega}_{M,j} = \boldsymbol{\omega}_i + \frac{1}{k_{r,j}} \dot{q}_j \mathbf{e}_{z,M,j}^0 = \mathbf{J}_{\omega,i} \dot{\mathbf{q}} + \frac{1}{k_{r,j}} \dot{q}_j \mathbf{R}_0^i \mathbf{e}_{z,M,j}^i$, the rotational kinetic energy follows as

$$\begin{aligned} T_{M,r,j} &= \frac{1}{2} \dot{\mathbf{q}}^T \mathbf{J}_{\omega,i}^T \mathbf{R}_0^{M,j} \bar{\mathbf{I}}_{M,j} \left(\mathbf{R}_0^{M,j} \right)^T \mathbf{J}_{\omega,i} \dot{\mathbf{q}} + \frac{1}{2k_{r,j}^2} \dot{q}_j^2 I_{M,zz,j} \\ &\quad + \frac{1}{k_{r,j}} \dot{q}_j I_{M,zz,j} \left(\mathbf{R}_0^i \mathbf{e}_{z,M,j}^i \right)^T \mathbf{J}_{\omega,i} \dot{\mathbf{q}} , \end{aligned} \quad (2.58)$$

where $\mathbf{R}_0^{M,j} \bar{\mathbf{I}}_{M,j} \left(\mathbf{R}_0^{M,j} \right)^T$ is the inertia tensor of rotor j transformed to the inertial frame by the rotation matrix $\mathbf{R}_0^{M,j} = \mathbf{R}_0^i \mathbf{R}_i^{M,j}$, $\mathbf{R}_i^{M,j}$ is the constant rotation matrix from the frame of rotor j to the frame of link i , the unit vector $\mathbf{e}_{z,M,j}^i$ is the last column of $\mathbf{R}_i^{M,j}$, and $\mathbf{J}_{\omega,i}$ is given by (2.48). The last term in (2.58) represents coupling effects between the rotor and the link motion.

The rotor-link configuration of the COMAU Racer 1.4 is given in Tab. 2.3 and its total kinetic energy can be written as, see (2.47), (2.49), (2.53), and (2.58)

$$\begin{aligned} T_C &= \frac{1}{2} \dot{\mathbf{q}}^T \left(\sum_{i=1}^6 m_{L,i} \mathbf{J}_{t,c,i}^T \mathbf{J}_{t,c,i} + m_{M,i} \mathbf{J}_{t,M,i}^T \mathbf{J}_{t,M,i} + \mathbf{J}_{\omega,i}^T \mathbf{R}_0^i \bar{\mathbf{I}}_{L,i} \left(\mathbf{R}_0^i \right)^T \mathbf{J}_{\omega,i} \right) \dot{\mathbf{q}} \\ &\quad + \sum_{i=1}^6 T_{M,r,i} = \frac{1}{2} \dot{\mathbf{q}}^T \mathbf{D}(\mathbf{q}) \dot{\mathbf{q}} . \end{aligned} \quad (2.59)$$

Hence, the equations of motion for the COMAU Racer 1.4 are given by (2.42), with \mathbf{D} from (2.59), \mathbf{C} from (2.38), $\boldsymbol{\tau}_f$ from (2.41), and \mathbf{g} according to (2.57), where $V_s = 0$.

The output $\mathbf{y} \in \mathbb{R}^n$ is a function of the joint coordinates $\mathbf{q}_J = \mathbf{q}$ and is given by

$$\mathbf{y} = \begin{bmatrix} \mathbf{y}_t \\ \mathbf{y}_r \end{bmatrix} = \begin{bmatrix} \mathbf{h}_t(\mathbf{q}) \\ \mathbf{h}_r(\mathbf{q}) \end{bmatrix} = \mathbf{h}(\mathbf{q}) , \quad (2.60)$$

with the end-effector position $\mathbf{y}_t \in \mathbb{R}^{n_t}$ and orientation $\mathbf{y}_r \in \mathbb{R}^{n_r}$ in some minimal representation of the rotation matrix \mathbf{R}_e of the direct kinematics (2.5). In the case of $\dim(\mathbf{y}_r) = n_r = 3$, \mathbf{y}_r is represented by the Euler angles $\boldsymbol{\phi}_e$ of \mathbf{R}_e according to (2.6) or (2.7), respectively, and the output \mathbf{y} is given by the direct kinematics (2.8). Thus, the output function $\mathbf{h}(\mathbf{q})$ is smooth as long as no representation singularities occur. The output dimension is given by $\dim(\mathbf{y}) = n = n_t + n_r$, where $n = N$ holds for fully actuated manipulators like the COMAU Racer 1.4, with $n_t = n_r = 3$.

Table 2.3: Rotor-link configuration of the COMAU Racer 1.4.

rotor j	fixed to link i	$\mathbf{R}_i^{M,j}$
1	1	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix}$
2	1	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix}$
3	3	$\begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$
4	3	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix}$
5	4	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix}$
6	5	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix}$

2.4.4 Equations of motion of serial manipulators with elastic joints

The model for stiff joints is not accurate enough when the rotors and links of a manipulator are elastically coupled. In this case, the rotors and the links have to be modeled as separate rigid bodies that are coupled via an elastic element, typically modeled as a linear spring. Hence, the $m = 2N$ generalized coordinates can be chosen as $\mathbf{q}^T = [\mathbf{q}_J^T, \mathbf{q}_M^T]$, with the joint coordinates $\mathbf{q}_J \in \mathbb{R}^N$ and the motor coordinates at the output side $\mathbf{q}_M \in \mathbb{R}^N$, where $\dot{q}_{M,i} = k_{r,i} \hat{\gamma}_{M,i}$. The energy stored in the elastic joints is given by (2.56) and the joint viscosity is modeled by $\boldsymbol{\tau}_{jv} = \mathbf{K}_d(\dot{\mathbf{q}}_J - \dot{\mathbf{q}}_M)$ with the constant, diagonal, and positive definite matrix $\mathbf{K}_d = \text{diag}([k_{d,1}, k_{d,2}, \dots, k_{d,N}])$.

For an elastic joint robot, the angular velocity of rotor j connected to link i is

given by

$$\boldsymbol{\omega}_{M,j} = \boldsymbol{\omega}_i + \frac{1}{k_{r,j}} \dot{q}_{M,j} \mathbf{e}_{z,M,j}^0 = \mathbf{J}_{\omega,i} \dot{\mathbf{q}}_J + \frac{1}{k_{r,j}} \dot{q}_{M,j} \mathbf{R}_0^i \mathbf{e}_{z,M,j}^i \quad (2.61)$$

and the rotational kinetic energy of rotor j follows as

$$\begin{aligned} T_{eM,r,j}(\mathbf{q}_J, \dot{\mathbf{q}}_J, \dot{\mathbf{q}}_M) &= \frac{1}{2} \dot{\mathbf{q}}_J^T \mathbf{J}_{\omega,i}^T \mathbf{R}_0^{M,j} \bar{\mathbf{I}}_{M,j} (\mathbf{R}_0^{M,j})^T \mathbf{J}_{\omega,i} \dot{\mathbf{q}}_J + \frac{1}{2k_{r,j}^2} \dot{q}_{M,j}^2 I_{M,zz,j} \\ &+ \frac{1}{k_{r,j}} \dot{q}_{M,j} I_{M,zz,j} (\mathbf{R}_0^i \mathbf{e}_{z,M,j}^i)^T \mathbf{J}_{\omega,i} \dot{\mathbf{q}}_J . \end{aligned} \quad (2.62)$$

The total kinetic energy of an elastic joint robot can be written as, see (2.47), (2.49), (2.53), and (2.62)

$$\begin{aligned} T_E &= \frac{1}{2} \dot{\mathbf{q}}_J^T \left(\sum_{i=1}^6 m_{L,i} \mathbf{J}_{t,c,i}^T \mathbf{J}_{t,c,i} + m_{M,i} \mathbf{J}_{t,M,i}^T \mathbf{J}_{t,M,i} + \mathbf{J}_{\omega,i}^T \mathbf{R}_0^i \bar{\mathbf{I}}_{L,i} (\mathbf{R}_0^i)^T \mathbf{J}_{\omega,i} \right) \dot{\mathbf{q}}_J \\ &+ \sum_{i=1}^6 T_{eM,r,i} = \begin{bmatrix} \dot{\mathbf{q}}_J \\ \dot{\mathbf{q}}_M \end{bmatrix}^T \begin{bmatrix} \mathbf{D}_E(\mathbf{q}_J) & \mathbf{U}(\mathbf{q}_J) \\ \mathbf{U}^T(\mathbf{q}_J) & \mathbf{D}_M \end{bmatrix} \begin{bmatrix} \dot{\mathbf{q}}_J \\ \dot{\mathbf{q}}_M \end{bmatrix} = \frac{1}{2} \dot{\mathbf{q}}^T \mathbf{D}(\mathbf{q}_J) \dot{\mathbf{q}} , \end{aligned} \quad (2.63)$$

where $\mathbf{D}_M = \text{diag}([k_{r,1}^{-2} I_{M,zz,1}, k_{r,2}^{-2} I_{M,zz,2}, \dots, k_{r,N}^{-2} I_{M,zz,N}]) \in \mathbb{R}^{N \times N}$ denotes the motor inertia diagonal matrix and the upper triangle matrix $\mathbf{U}(\mathbf{q}_J) \in \mathbb{R}^{N \times N}$ takes into account the motor/link inertia couplings, cf. [74], [75]. Using (2.63), (2.38), and (2.57), the dynamics of the elastic joint robot in matrix form (2.36) follow as, see [74], [75],

$$\begin{aligned} &\underbrace{\begin{bmatrix} \mathbf{D}_E(\mathbf{q}_J) & \mathbf{U}(\mathbf{q}_J) \\ \mathbf{U}^T(\mathbf{q}_J) & \mathbf{D}_M \end{bmatrix}}_{\mathbf{D}(\mathbf{q}_J)} \begin{bmatrix} \ddot{\mathbf{q}}_J \\ \ddot{\mathbf{q}}_M \end{bmatrix} + \underbrace{\begin{bmatrix} \mathbf{C}_E(\mathbf{q}_J, \dot{\mathbf{q}}_J) & \mathbf{C}_1(\mathbf{q}_J, \dot{\mathbf{q}}_J) \\ \mathbf{C}_2(\mathbf{q}_J, \dot{\mathbf{q}}_J) & \mathbf{0} \end{bmatrix}}_{\mathbf{C}(\mathbf{q}_J, \dot{\mathbf{q}}_J)} \begin{bmatrix} \dot{\mathbf{q}}_J \\ \dot{\mathbf{q}}_M \end{bmatrix} \\ &+ \underbrace{\begin{bmatrix} \mathbf{g}_L(\mathbf{q}_J) + \mathbf{K}_s(\mathbf{q}_J - \mathbf{q}_M) \\ -\mathbf{K}_s(\mathbf{q}_J - \mathbf{q}_M) \end{bmatrix}}_{\mathbf{g}(\mathbf{q})} + \begin{bmatrix} \mathbf{K}_d(\dot{\mathbf{q}}_J - \dot{\mathbf{q}}_M) \\ -\mathbf{K}_d(\dot{\mathbf{q}}_J - \dot{\mathbf{q}}_M) \end{bmatrix} + \underbrace{\begin{bmatrix} \boldsymbol{\tau}_{f,J}(\dot{\mathbf{q}}_J) \\ \boldsymbol{\tau}_{f,M}(\dot{\mathbf{q}}_M) \end{bmatrix}}_{\boldsymbol{\tau}_f(\dot{\mathbf{q}})} = \begin{bmatrix} \boldsymbol{\tau}_{ext} \\ \boldsymbol{\tau}_d \end{bmatrix} , \end{aligned} \quad (2.64)$$

with the generalized friction forces $\boldsymbol{\tau}_{f,J} \in \mathbb{R}^N$ and $\boldsymbol{\tau}_{f,M} \in \mathbb{R}^N$ modeled by (2.41), the generalized forces of the motors $\boldsymbol{\tau}_d \in \mathbb{R}^N$, and the generalized external forces $\boldsymbol{\tau}_{ext} \in \mathbb{R}^N$. The gravitation forces $\mathbf{g}_L(\mathbf{q}_J) \in \mathbb{R}^N$ are the same as for the rigid body model. The Coriolis matrices $\mathbf{C}_1(\mathbf{q}_J, \dot{\mathbf{q}}_J) \in \mathbb{R}^{N \times N}$ and $\mathbf{C}_2(\mathbf{q}_J, \dot{\mathbf{q}}_J) \in \mathbb{R}^{N \times N}$ also result from the motor/link inertia couplings, see [74], [75].

System (2.64) can be simplified by neglecting the motor/link inertia couplings given by the last term in (2.62), which implies $\mathbf{U} = \mathbf{0}$ and $\mathbf{C}_1 = \mathbf{C}_2 = \mathbf{0}$, cf. [7]. The resulting simplified system is given by

$$\mathbf{D}_E(\mathbf{q}_J) \ddot{\mathbf{q}}_J + \mathbf{n}_E(\mathbf{q}_J, \dot{\mathbf{q}}_J) = \boldsymbol{\tau}_J + \boldsymbol{\tau}_{ext} \quad (2.65a)$$

$$\mathbf{D}_M \ddot{\mathbf{q}}_M + \boldsymbol{\tau}_{f,M}(\dot{\mathbf{q}}_M) = \boldsymbol{\tau}_d - \boldsymbol{\tau}_J, \quad (2.65b)$$

with

$$\mathbf{n}_E(\mathbf{q}_J, \dot{\mathbf{q}}_J) = \mathbf{C}_E(\mathbf{q}_J, \dot{\mathbf{q}}_J)\dot{\mathbf{q}}_J + \mathbf{g}_L(\mathbf{q}_J) + \boldsymbol{\tau}_{f,J}(\dot{\mathbf{q}}_J)$$

and the joint forces

$$\boldsymbol{\tau}_J = \mathbf{K}_s(\mathbf{q}_M - \mathbf{q}_J) + \mathbf{K}_d(\dot{\mathbf{q}}_M - \dot{\mathbf{q}}_J) , \quad (2.66)$$

where $\mathbf{K}_s > 0$ and $\mathbf{K}_d > 0$. If the viscosity of the joints is negligibly small, $\mathbf{K}_d = \mathbf{0}$ holds, and the joint forces are given by

$$\boldsymbol{\tau}_J = \mathbf{K}_s(\mathbf{q}_M - \mathbf{q}_J) . \quad (2.67)$$

2.4.5 Equations of motion of the Delta robot

The mass of the rods is small in comparison to the other moving parts. As proposed in [69], the rods are modeled as two point masses, where 2/3 of the mass is added to the slide and 1/3 to the end-effector. Thus, the effective mass of each slide is given by $m_{q,i} = m_{s,i} + 2/3m_p$, $i = 1, 2, 3$. The effective mass of the end-effector reads as $m_y = m_e + m_l + m_p$. The nominal mass parameters of the FESTO EXPT-45 are listed in Tab. 2.2, where $m_{q,1} = m_{q,2} = m_{q,3} = m_{q,i}$. Hence, the constant mass matrices are given by $\mathbf{D}_q = \text{diag}([m_{q,1}, m_{q,2}, m_{q,3}])$ and $\mathbf{D}_y = m_y \mathbf{I}$. The kinetic energy reads as

$$T = \frac{1}{2} \left(\dot{\mathbf{q}}^T \mathbf{D}_q \dot{\mathbf{q}} + \dot{\mathbf{y}}_t^T \mathbf{D}_y \dot{\mathbf{y}}_t \right) \quad (2.68)$$

and the potential energy of the system takes the form

$$V = - \left(m_y y_z + h_p k_D (m_{q,1} q_1 + m_{q,2} q_2 + m_{q,3} q_3) \right) g , \quad (2.69)$$

with the gravitational acceleration g . The forward kinematics are rearranged and the holonomic constraint, see also (2.43),

$$\mathbf{f}_{h,c}(\mathbf{q}, \mathbf{y}_t) = \mathbf{y}_t - \mathbf{h}_D(\mathbf{q}) = \mathbf{0} \quad (2.70)$$

is introduced. Since there are no mixed terms in T and V , application of the Lagrange formalism with holonomic constraints (2.43) results in the equations of motion, which can be written as

$$\begin{aligned} \mathbf{D}_q \ddot{\mathbf{q}} + \mathbf{g}_q(\mathbf{q}) + \boldsymbol{\Gamma}_{h,q}^T(\mathbf{q}) \boldsymbol{\lambda} &= \boldsymbol{\tau}_d - \boldsymbol{\tau}_f(\dot{\mathbf{q}}) \\ \mathbf{D}_y \ddot{\mathbf{y}}_t + \mathbf{g}_y(\mathbf{y}_t) + \boldsymbol{\Gamma}_{h,y}^T(\mathbf{y}_t) \boldsymbol{\lambda} &= \boldsymbol{\tau}_y , \end{aligned} \quad (2.71)$$

with the Lagrange multipliers $\boldsymbol{\lambda}^T = [\lambda_1, \lambda_2, \lambda_3]$, the gravitational forces $\mathbf{g}_q^T(\mathbf{q}) = \partial V / \partial \mathbf{q}$ and $\mathbf{g}_y^T(\mathbf{y}_t) = \partial V / \partial \mathbf{y}_t$, linear drive forces $\boldsymbol{\tau}_d^T = [\tau_{d,1}, \tau_{d,2}, \tau_{d,3}]$, the friction forces $\boldsymbol{\tau}_f(\dot{\mathbf{q}})$, and external end-effector forces $\boldsymbol{\tau}_y^T = [\tau_{y,1}, \tau_{y,2}, \tau_{y,3}]$. The equations (2.71) are coupled by the holonomic constraint (2.70), $\boldsymbol{\Gamma}_{h,q}(\mathbf{q}) = \partial \mathbf{f}_{h,c} / \partial \mathbf{q} =$

$-\mathbf{J}_D(\mathbf{q})$, see also (2.33), and $\mathbf{\Gamma}_{h,y}(\mathbf{y}_t) = \partial \mathbf{f}_{h,c} / \partial \mathbf{y}_t = \mathbf{I}_{3 \times 3}$. Eliminating the Lagrange multipliers $\boldsymbol{\lambda}$ and using the time derivative of (2.33) yields the rigid body dynamics (2.42) with the symmetric positive definite generalized mass matrix

$$\mathbf{D}(\mathbf{q}) = \mathbf{D}_q + \mathbf{J}_D^T(\mathbf{q})\mathbf{D}_y\mathbf{J}_D(\mathbf{q}) , \quad (2.72a)$$

the Coriolis matrix

$$\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{J}_D^T(\mathbf{q})\mathbf{D}_y\dot{\mathbf{J}}_D(\mathbf{q}, \dot{\mathbf{q}}) , \quad (2.72b)$$

the vector of potential forces

$$\mathbf{g}(\mathbf{q}) = \mathbf{g}_q(\mathbf{q}) + \mathbf{J}_D^T(\mathbf{q})\mathbf{g}_y \circ \mathbf{h}_D(\mathbf{q}) , \quad (2.72c)$$

and the vector of generalized external forces

$$\boldsymbol{\tau}_{ext} = \mathbf{J}_D^T(\mathbf{q})\boldsymbol{\tau}_y . \quad (2.72d)$$

Hence, this model of the DELTA robot constitutes a fully actuated manipulator.

2.4.6 Base parameter set

The system dynamics (2.36) for manipulators with rigid transmissions are linear in the dynamic parameter vector $\boldsymbol{\rho}$, cf. [3, 4], i.e.

$$\mathbf{Y}(\ddot{\mathbf{q}}, \dot{\mathbf{q}}, \mathbf{q})\boldsymbol{\rho} = \frac{\partial}{\partial \boldsymbol{\rho}} (\mathbf{D}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) + \boldsymbol{\tau}_f(\dot{\mathbf{q}})) \boldsymbol{\rho} = \boldsymbol{\tau}_d \quad (2.73)$$

holds, where $\boldsymbol{\rho}$ contains the inertial and friction parameters and $\boldsymbol{\tau}_{ext} = \mathbf{0}$ is assumed. The regression matrix $\mathbf{Y}(\ddot{\mathbf{q}}, \dot{\mathbf{q}}, \mathbf{q})$ depends on the manipulator's geometry, which is considered to be known. Ten inertial parameters exist for each rigid link i . These are the six entries of the overall inertia matrix $\bar{\mathbf{I}}_i$ of link i including all rotors connected to link i and transformed into the origin of the coordinate system $(0_i, x_i, y_i, z_i)$, the first moments $m_i \mathbf{p}_{c,i}^i$, and the mass m_i of the link i including all masses of the rotors connected to link i , where $\mathbf{p}_{c,i}^i$ denotes the constant vector to the center of mass defined in the coordinate system $(0_i, x_i, y_i, z_i)$, see [3]. Additionally, each drive j depends on three parameters, namely the rotor inertia $I_{M,zz,j}$ and the two friction parameters $f_{c,j}$ and $f_{v,j}$, $j = 1, \dots, N$.

However, not all of these parameters appear in the system equations or are linear independent [3, 59, 65, 76]. A minimum set of linear independent inertial parameters that determines the dynamic model completely is called a base parameter set. This set is not unique but has a well defined dimension. The system dynamics in the base parameters $\boldsymbol{\rho}_b \in \mathbb{R}^{n_b}$ can be expressed as

$$\mathbf{Y}_b(\ddot{\mathbf{q}}, \dot{\mathbf{q}}, \mathbf{q})\boldsymbol{\rho}_b = \boldsymbol{\tau}_d , \quad (2.74)$$

with the regression matrix $\mathbf{Y}_b(\ddot{\mathbf{q}}, \dot{\mathbf{q}}, \mathbf{q}) \in \mathbb{R}^{N \times n_b}$.

Since the DELTA robot features simple dynamics, its base parameter set can directly be given by $\boldsymbol{\rho}_{b,D}^T = [m_{q,1}, m_{q,2}, m_{q,3}, m_y, f_{v,1}, f_{c,1}, f_{v,2}, f_{c,2}, f_{v,3}, f_{c,3}]$, with the viscous and Coulomb friction parameters $f_{v,i}, f_{c,i}, i = 1, 2, 3$. The base parameter set of a six-axis serial manipulator like the COMAU Racer 1.4 cannot be determined just by guessing [77]. Mayeda et al. developed a method to determine the base parameter set of serial manipulators with rotary joints in an analytic form, where adjacent joints are either parallel or perpendicular [59]. A short summary of this method as well as the resulting base parameters for the COMAU Racer 1.4 are given in Appendix B.

2.5 Dynamic parameter identification

For model-based control, a suitable mathematical model of the system has to be known. The model typically depends on specific parameters, which may have an extensive influence on the behavior of the system. Therefore, good knowledge of the parameters is an important aspect.

Robot manufacturers usually not only provide the kinematics but also some inertial parameters. However, in most cases, not all parameters contained in the dynamic model, e.g., friction parameters, are included in the data sheets. Additionally, the given parameters are often derived from CAD drawings and are not containing certain components of the complete robot, e.g., cables, screws or custom-made add-ons. Therefore, experimental identification has to be applied to obtain the required inertial and friction parameters of the robot. The references [5, 60, 61] give a good overview of the extensive literature concerning this topic.

The system dynamics are linear in the base parameters $\boldsymbol{\rho}_b$, see (2.74), and are therefore well suited for the identification procedure, which can be divided into the following five steps:

1. finding an excitation trajectory that leads to well identifiable parameters;
2. running an experiment and sampling a sufficient number of data points;
3. signal filtering and computation of velocity and acceleration;
4. construction of an overdetermined linear system using the information matrix;
5. applying a linear regression technique to estimate the base parameters.

The overdetermined linear system is given by

$$\mathbf{X}\boldsymbol{\rho}_b = \mathbf{b}_\tau, \quad (2.75)$$

with the information matrix

$$\mathbf{X} = \begin{bmatrix} \mathbf{Y}_b(\ddot{\mathbf{q}}(t_1), \dot{\mathbf{q}}(t_1), \mathbf{q}(t_1)) \\ \vdots \\ \mathbf{Y}_b(\ddot{\mathbf{q}}(t_K), \dot{\mathbf{q}}(t_K), \mathbf{q}(t_K)) \end{bmatrix} \quad (2.76)$$

and

$$\mathbf{b}_\tau = \begin{bmatrix} \tau_d(t_1) \\ \vdots \\ \tau_d(t_K) \end{bmatrix}, \quad (2.77)$$

where K is the number of sampling points and $t_i, i = 1, \dots, K$, are the corresponding sampling instances. For a reliable estimation of $\boldsymbol{\rho}_b$, the number of sampling points K has to be much larger than the number n_b of base parameters.

The parameter estimation approach of Swevers et al. [62, 63] is used to identify the base parameters of the COMAU Racer 1.4 and is summarized in the following. The application to the COMAU Racer 1.4 is presented in Appendix C.

2.5.1 Excitation trajectory

The quality of the model identification strongly depends on the chosen excitation trajectory. This trajectory must ensure that the information matrix \mathbf{X} has full column rank and that all motion constraints of the joint angles, velocities, accelerations, and the end-effector position are met. Various approaches to parametrize the excitation trajectories and to find an optimal set of parameters can be found in literature. A few methods are highlighted in the following.

Armstrong suggested to minimize the condition number or to maximize the smallest singular value of the information matrix \mathbf{X} [64]. Gautier minimized a linear combination of the condition number and a parameter equilibrating the values of the elements of the information matrix, with the joint angles and velocities at discrete time steps as optimization variables [65]. The trajectory is calculated by interpolating a polynomial between the optimal points. Therefore, it is not guaranteed that the resulting trajectory satisfies all constraints or minimizes the condition number.

Swevers et al. use a finite sum of harmonic sine and cosine functions as excitation trajectory [62, 63]. This method allows to use time-domain data averaging and analytic calculation of the joint velocities and accelerations in the frequency domain. They suggest to maximize the determinant of $\mathbf{X}^T \mathbf{X}$, the so called d-optimality, which gives slightly better results than the minimization of the condition number. In the next subsections, this approach is explained in more detail.

2.5.1.1 Periodic function

The finite Fourier series for each joint $i = 1, \dots, N$ is given by, cf. [62],

$$q_i(t) = q_{i,0} + \sum_{k=1}^M (a_{i,k} \sin(k\omega_f t) + b_{i,k} \cos(k\omega_f t)) , \quad (2.78)$$

with the fundamental angular frequency ω_f , the offset $q_{i,0}$ and the Fourier coefficients $a_{i,k}$ and $b_{i,k}$. The fundamental angular frequency ω_f is the same for all

joints and, therefore, the positions, velocities, and accelerations are all periodic with the period $T_f = 2\pi/\omega_f$. The period T_f has to be an integer multiple of the sampling time T_s .

When choosing ω_f and M some aspects have to be taken into account. With a lower fundamental frequency ω_f a larger part of the workspace can be covered at the price of a longer measurement time. A larger number of harmonics M leads to higher accelerations, which are necessary to identify the inertia matrix. However, the highest frequency of the excitation trajectory should be much smaller than the resonance frequencies of the robot.

2.5.1.2 Optimization

The Fourier coefficients $a_{i,k}$ and $b_{i,k}$ and the offsets $q_{i,0}$ are the $k_\nu = N(2M + 1)$ independent variables for the optimization of the excitation trajectory. The determinant of the matrix $\mathbf{X}^T \mathbf{X}$ is used as optimization criterion [62]. Several constraints like the limitations of the positions, velocities, and accelerations of the joints and the end-effector position of the robot have to be taken into account. The constrained optimization problem can be written as

$$\begin{aligned} \boldsymbol{\nu}^* = \arg \min_{\boldsymbol{\nu} \in \mathbb{R}^{k_\nu}} & -\log \det \left(\mathbf{X}^T(\boldsymbol{\nu}) \mathbf{X}(\boldsymbol{\nu}) \right) \\ \text{s.t. } & \mathbf{g}_1(\boldsymbol{\nu}) \leq \mathbf{0} \\ & \mathbf{g}_2(\boldsymbol{\nu}) \leq \mathbf{0} , \end{aligned} \quad (2.79)$$

where $\boldsymbol{\nu} = [q_{1,0}, a_{1,1}, b_{1,1}, \dots, q_{N,0}, \dots, a_{N,M}, b_{N,M}]^T \in \mathbb{R}^{k_\nu}$, \mathbf{g}_1 summarizes the joint constraints, and \mathbf{g}_2 represents the constraints in the robot's workspace. The logarithm in (2.79) is used to avoid numerical issues, as the value of the determinant can get extremely large.

2.5.2 Signal processing

The excitation trajectory is a finite Fourier series. Due to the nonlinearity of the system, the measured positions may contain different and even more harmonics than the excitation trajectory. However, the resulting motion is still periodic with the same period as the exciting trajectory and also band limited because of the band-limited motion controller. With these properties, an exact differentiation of the position is possible [62]. Using the discrete Fourier transform, the measured positions are transformed into the frequency domain and then filtered by selecting the relevant frequencies. The velocity and acceleration are calculated by multiplying the spectrum with $I\omega$ and $-\omega^2$, respectively. Finally, the spectrum of the filtered position, velocity, and acceleration are transformed back into the time domain using the inverse discrete Fourier transform.

The measured torques are also periodic with T_f and the signal-to-noise ratio can be improved by data averaging.

2.5.3 Parameter estimation

With the signal processing described in Section 2.5.2, the joint positions, velocities, and accelerations can be considered as nearly noise free. Hence, the information matrix \mathbf{X} is also free of noise and a weighted least-squares estimation (WLSE) yields the same bias free results as the more complex maximum likelihood estimation [62]. The difference between the WLSE and a standard least-squares approach is that the noise of the torque measurement is taken into account, which also includes unmodeled periodic effects in the gear transmissions. The regression matrix and the torques are weighted with the covariance matrix $\mathbf{\Upsilon}$ of the actuator torque measurements. The estimation of the base parameters $\boldsymbol{\rho}_b$ using WLSE is given by

$$\hat{\boldsymbol{\rho}}_b = \left(\mathbf{X}^T \mathbf{\Upsilon}^{-1} \mathbf{X} \right)^{-1} \mathbf{X}^T \mathbf{\Upsilon}^{-1} \mathbf{b}_\tau . \quad (2.80)$$

CHAPTER 3

Path Following Control

In this chapter, path following control (PFC) in three-dimensional space for fully actuated rigid manipulators and elastic joint robots is introduced. The presented approach can handle open, closed, and intersecting paths parametrized as regular curves, e.g., splines, and is based on input-output linearization, cf. [78]. In contrast to [22], the parametrization is split into the position and orientation parametrization and transversal feedback linearization (TFL) is only applied to the position parametrization. A parallel transport frame is introduced for the design of the path following controller, which not only allows to directly cope with paths having zero curvature, but also drastically simplifies the path following control law compared to the works known from literature, e.g., [22]. Moreover, a feasible neighborhood of the path is defined for which a diffeomorphism can be found that maps the generalized coordinates to tangential, transversal, and rotational coordinates. These properties of the presented PFC approach make it highly suitable for industrial robotic applications. Following the ideas of [79], where pure planar problems are considered, the method is combined with compliance control.

The proposed PFC approach for fully actuated manipulators is applied to a DELTA Robot for a proof of concept. This manipulator, with parallel kinematics, has three translational degrees of freedoms. In a first experiment, a specific motion on a complex path defined by quartic splines in free space is performed. In a second experiment, the combination of the presented path following control approach with compliance control is demonstrated. To validate the compliant path following control strategy, the end-effector is operated in the notch of a rigid object. The PFC approach for elastic joint robots is verified by simulation studies in MATLAB[®]/Simulink[®]. In the first simulation example, the pure PFC for elastic joint robots, the singular perturbation PFC approach, and the PFC for fully actuated manipulators are compared by applying these approaches to an elastic

joint robot with relatively soft transmissions. The PFC approach is compared to a classical trajectory tracking controller (TTC) in the second simulation example to point out the differences between the two control concepts.

Large parts of this chapter are published in similar form in [70, 80, 81].

3.1 Path assumptions

Suppose the path γ is given as a regular parametrized curve $\boldsymbol{\sigma}^T(\theta) = [\boldsymbol{\sigma}_t^T(\theta), \boldsymbol{\sigma}_r^T(\theta)] : \mathcal{T} \mapsto \mathbb{R}^n$, with reference position $\boldsymbol{\sigma}_t(\theta) \in \mathbb{R}^{n_t}$, reference orientation $\boldsymbol{\sigma}_r(\theta) \in \mathbb{R}^{n_r}$, $n = n_t + n_r$, and path parameter θ , which is element of a nonempty set $\mathcal{T} \subseteq \mathbb{R}$. In the three-dimensional space, $n_t = 3$ holds. For a given parametrization $\boldsymbol{\sigma}(\theta)$, the path γ is defined as the subset of the output \mathbf{y} of (2.1) $\gamma = \{\bar{\mathbf{y}} \in \mathbb{R}^n : \bar{\mathbf{y}} = \boldsymbol{\sigma}(\bar{\theta}), \bar{\theta} \in \mathcal{T}\}$, where $\boldsymbol{\sigma}_t(\theta)$ defines the position part γ_t , see Fig. 3.1, and $\boldsymbol{\sigma}_r(\theta)$ the orientation part γ_r . The parametrization $\boldsymbol{\sigma}(\theta)$ of the path γ is regular, if $\boldsymbol{\sigma}'_t(\bar{\theta}) = (\partial\boldsymbol{\sigma}_t/\partial\theta)(\bar{\theta}) \neq \mathbf{0}$ for $\bar{\theta} \in \mathcal{T}$. Another path requirement is that the parametrization has to be k -fold continuously differentiable, where $k = 3$ for fully actuated manipulators and $k = 5$ for elastic joint robots. The path γ is open or closed depending on the choice of $\boldsymbol{\sigma}(\theta)$ and \mathcal{T} . For more information on curves, see, e.g., [21].

3.2 Control objectives

To formulate the objectives of path following control, the mapping $\|\mathbf{y}_t\|_{\gamma_t} : \mathbb{R}^{n_t} \mapsto \mathbb{R}_0^+$ is introduced, which assigns each position \mathbf{y}_t in the output space a nonnegative

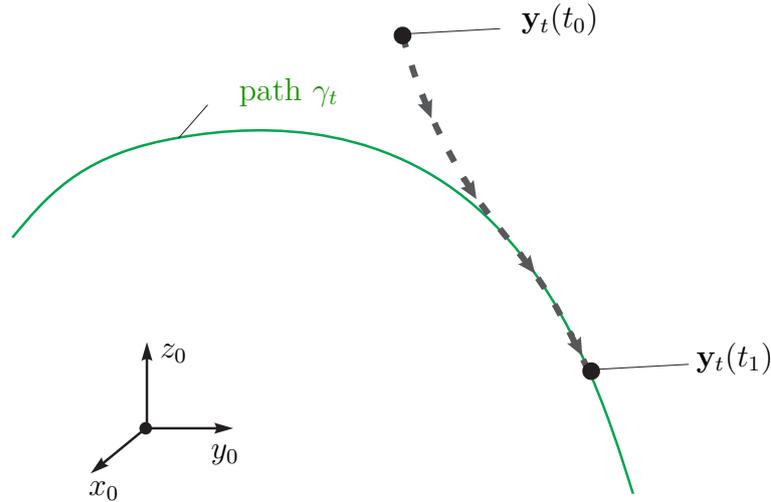


Figure 3.1: Path γ_t and convergence of the output \mathbf{y}_t to the path.

real number that is given by the shortest distance to the path γ_t , i.e., $\|\mathbf{y}_t\|_{\gamma_t} = \inf_{\bar{\mathbf{y}}_t \in \gamma_t} \|\mathbf{y}_t - \bar{\mathbf{y}}_t\|_2$. The control objectives of path following control are defined as follows, see [15] and Fig. 3.1.

- (O1) *Asymptotic convergence to $\sigma_t(\cdot)$* : The position output \mathbf{y}_t of (2.1) converges asymptotically to the path γ_t , i.e., $\|\mathbf{y}_t(t)\|_{\gamma_t} \rightarrow 0$ for $t \rightarrow \infty$.
- (O2) *Invariance property*: If the generalized coordinates and velocities of the dynamic system at time t_0 , i.e., $[\bar{\mathbf{q}}_J^T(t_0), \dot{\bar{\mathbf{q}}}_J^T(t_0)]^T$, are elements of the controlled invariant subset Γ^* of $\Gamma = \{[\bar{\mathbf{q}}_J^T, \dot{\bar{\mathbf{q}}}_J^T]^T \in \mathbb{R}^{2m} : \mathbf{h}(\bar{\mathbf{q}}_J) \in \gamma\}$, then $\|\mathbf{y}_t(t)\|_{\gamma_t} = 0, \forall t \geq t_0$.
- (O3) *Tangential motion*: The motion on the path γ_t meets application-specific requirements.

3.3 Orthonormal frame

An orthonormal frame with respect to a parametrized curve is constructed as depicted in Fig. 3.2. For this, one possibility is given by the Frenet-Serret frame, see, e.g., [21], which is used in [22] for PFC. The direct application of the Frenet-Serret frame requires a given position parametrization $\sigma_t(\theta) : \mathcal{T} \mapsto \mathbb{R}^{n_t}$ to be regular of order $n_t - 1 = 2$, i.e., that the first two derivatives of $\sigma_t(\theta)$ with respect

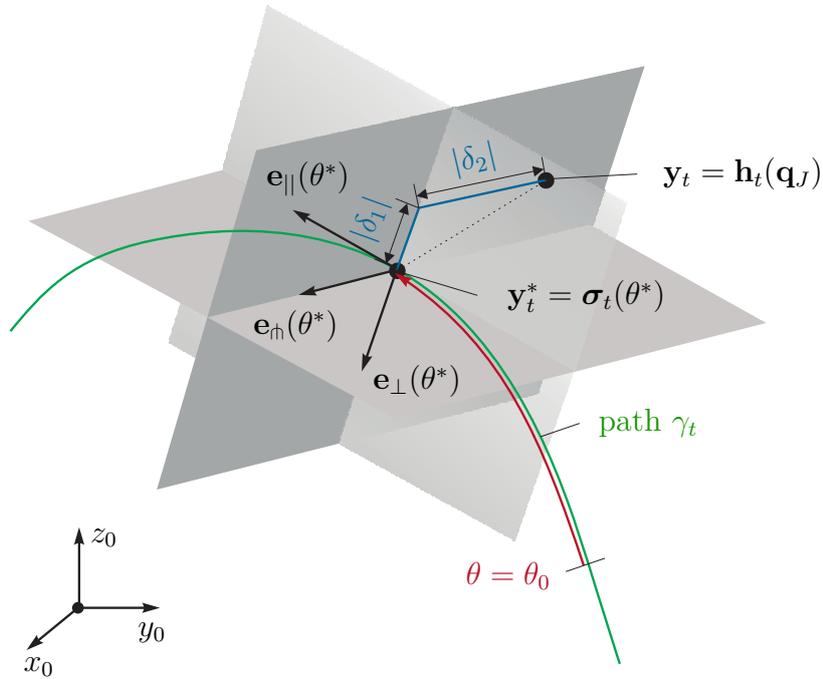


Figure 3.2: Orthonormal frame for PFC. [70] © 2017 IEEE

to θ are linearly independent. Note that second-order regularity of a path implies a nonzero curvature of a parametrization $\boldsymbol{\sigma}(\theta)$ for all $\theta \in \mathcal{T}$, see, e.g., [21]. Thus, a straight line path segment has zero curvature, which shows that the requirement of second-order regularity of a path is rather restrictive. Therefore, in contrast to the works known from literature, e.g., [22], the parallel transport frame [24, 25] is used in this work, which can handle curves with zero curvature and is introduced in the following.

Recalling that $\boldsymbol{\sigma}'_t(\theta)$ is the tangential vector to γ_t at θ , the tangential unit vector is given by

$$\mathbf{e}_{\parallel}(\theta) = \frac{\boldsymbol{\sigma}'_t(\theta)}{\|\boldsymbol{\sigma}'_t(\theta)\|_2}. \quad (3.1)$$

To construct the parallel transport frame, the derivative of the normal vectors $\mathbf{e}_{\perp}(\theta)$ and $\mathbf{e}_{\natural}(\theta)$ has to point in the direction of the tangential unit vector $\mathbf{e}_{\parallel}(\theta)$, see [24]. Together with the requirement of orthonormality, the overdetermined differential-algebraic equations

$$\mathbf{e}'_i(\theta) = \gamma_i(\theta)\mathbf{e}_{\parallel}(\theta), \quad \mathbf{e}_i(\theta_0) = \mathbf{e}_{i,0} \quad (3.2a)$$

$$0 = \mathbf{1} - \mathbf{e}_i^{\text{T}}(\theta)\mathbf{e}_i(\theta) \quad (3.2b)$$

$$0 = \mathbf{e}_{\parallel}^{\text{T}}(\theta)\mathbf{e}_i(\theta), \quad (3.2c)$$

for $i \in \{\perp, \natural\}$ are obtained. Therein, $\gamma_i(\theta)$ is a scalar and $\mathbf{e}_{i,0}$ is the initial condition, which complies with the algebraic equations (3.2b) and (3.2c). Note that the equations (3.2a) and (3.2b) are of Hessenberg index-2 form with hidden constraint (3.2c), see, e.g., [82] for more information on differential-algebraic systems and hidden constraints. Differentiating (3.2b) with respect to θ and substituting (3.2a) gives (3.2c). Differentiating (3.2c) with respect to θ and multiplying (3.2a) by $\mathbf{e}_{\parallel}^{\text{T}}(\theta)$ results in

$$\gamma_i(\theta) = -\left(\mathbf{e}'_{\parallel}(\theta)\right)^{\text{T}} \mathbf{e}_i(\theta). \quad (3.3)$$

Hence, inserting (3.3) into (3.2a) yields the ordinary differential equations for $i \in \{\perp, \natural\}$

$$\mathbf{e}'_i(\theta) = -\left(\mathbf{e}'_{\parallel}(\theta)\right)^{\text{T}} \mathbf{e}_i(\theta)\mathbf{e}_{\parallel}(\theta), \quad \mathbf{e}_i(\theta_0) = \mathbf{e}_{i,0}. \quad (3.4)$$

Note that the second normal vector $\mathbf{e}_{\natural}(\theta)$ can also be obtained using the cross product, i.e.,

$$\mathbf{e}_{\natural}(\theta) = \mathbf{e}_{\parallel}(\theta) \times \mathbf{e}_{\perp}(\theta). \quad (3.5)$$

3.4 Projection operator and feasible neighborhood

The closest point $\mathbf{y}_t^* = \boldsymbol{\sigma}_t(\theta^*)$ on the path γ_t to \mathbf{y}_t is determined by the orthogonal projection $\text{P}_{\mathcal{T}}(\mathbf{y}_t)$, cf. [20]. Given a parametrized curve $\boldsymbol{\sigma}(\theta)$ with $\theta \in \mathcal{T} \subseteq \mathbb{R}$, the

orthogonal projection of \mathbf{y}_t onto $\boldsymbol{\sigma}_t(\theta)$ requires the solution of an optimization problem, i.e.,

$$\theta^* = P_{\mathcal{T}}(\mathbf{y}_t) = \arg \min_{\theta \in \mathcal{T}} \|\mathbf{y}_t - \boldsymbol{\sigma}_t(\theta)\|_2^2 \in \mathcal{T} . \quad (3.6)$$

If θ^* is an interior point of \mathcal{T} , (3.6) features a strict minimum in a feasible neighborhood, which fulfills the first-order necessary condition for optimality

$$\left(\mathbf{y}_t - \boldsymbol{\sigma}_t(\theta^*)\right)^{\text{T}} \boldsymbol{\sigma}'_t(\theta^*) = 0 \quad (3.7)$$

and the second-order sufficient condition for optimality

$$\|\boldsymbol{\sigma}'_t(\theta^*)\|_2^2 - \left(\mathbf{y}_t - \boldsymbol{\sigma}_t(\theta^*)\right)^{\text{T}} \boldsymbol{\sigma}''_t(\theta^*) > 0 , \quad (3.8)$$

see, e.g., [83]. Because $\boldsymbol{\sigma}'_t(\theta^*)$ is tangential to the path at $\boldsymbol{\sigma}_t(\theta^*)$, (3.7) implies that the vector $\mathbf{y}_t - \boldsymbol{\sigma}_t(\theta^*)$ is orthogonal to the path, cf. Fig. 3.2. Rearranging (3.8) and introducing

$$\alpha(\mathbf{y}_t) = \frac{\left(\mathbf{y}_t - \boldsymbol{\sigma}_t(\theta^*)\right)^{\text{T}} \boldsymbol{\sigma}''_t(\theta^*)}{\|\boldsymbol{\sigma}'_t(\theta^*)\|_2^2} \quad (3.9)$$

allows to define the feasible neighborhood of a path $\mathcal{Y}_t = \{\bar{\mathbf{y}}_t \in \mathbb{R}^{n_t} : \alpha(\bar{\mathbf{y}}_t) < 1\}$. Differentiating the first-order condition for optimality (3.7) with respect to the time and using (3.1) and (3.9) yields the time derivative of the optimal path parameter

$$\dot{\theta}^* = \frac{\beta(\mathbf{y}_t) \mathbf{e}_{\parallel}^{\text{T}}(\theta^*)}{\|\boldsymbol{\sigma}'_t(\theta^*)\|_2} \dot{\mathbf{y}}_t , \quad (3.10)$$

where

$$\beta(\mathbf{y}_t) = \frac{1}{1 - \alpha(\mathbf{y}_t)} . \quad (3.11)$$

If the position output \mathbf{y}_t is on the path, $\alpha(\mathbf{y}_t = \boldsymbol{\sigma}_t(\theta^*)) = 0$ and $\beta(\mathbf{y}_t = \boldsymbol{\sigma}_t(\theta^*)) = 1$ holds true.

3.5 Path following control for fully actuated manipulators

A fully actuated manipulator can be described by the dynamics (2.42) with the output function (2.1) and the number of generalized coordinates m equals the number of joint coordinates N , cf. Section 2.4.3. Here, the generalized coordinates \mathbf{q} are identical to the joint coordinates \mathbf{q}_J . This type of manipulator features a (vector) relative degree of $\{2, 2, \dots, 2\}$ and is full state exact linearizable via static state feedback. See [78] for a definition of the (vector) relative degree and the exact linearization via state feedback. In the following, a path following controller for this type of manipulators is presented. Additionally, a drastically simplified path following concept using joint velocity controllers is introduced.

3.5.1 Coordinate transformation

A coordinate transformation is deduced that maps the generalized coordinates and velocities, \mathbf{q} and $\dot{\mathbf{q}}$, of the system (2.42) to tangential coordinates $\boldsymbol{\eta}^T = [\eta_1, \eta_2]$, transversal coordinates $\boldsymbol{\xi}^T = [\xi_1, \xi_2, \xi_3, \xi_4]$, and rotational coordinates $\boldsymbol{\zeta}^T = [\mathbf{y}_r^T, \dot{\mathbf{y}}_r^T]$ with respect to a path γ parametrized as a regular \mathcal{C}^3 curve $\boldsymbol{\sigma}(\theta)$. It will be shown that the coordinate transformation is a diffeomorphism onto its image in a feasible neighborhood \mathcal{Y}_t of the path γ_t .

3.5.1.1 Tangential subsystem

The first tangential coordinate is, as proposed in [20], chosen by the arc length

$$\eta_1 = g(\mathbf{y}_t) = \int_{\theta_0}^{\theta^*} \|\boldsymbol{\sigma}'_t(\tau)\|_2 d\tau . \quad (3.12)$$

See Theorem 1.4.1 in [21] for a definition of the arc length. Calculating the time derivative of (3.12) and using (3.10), the second tangential coordinate follows as

$$\eta_2 = \dot{\eta}_1 = \|\boldsymbol{\sigma}'_t(\theta^*)\|_2 \dot{\theta}^* = \underbrace{\beta(\mathbf{y}_t) \mathbf{e}_{\parallel}^T(\theta^*)}_{(\nabla g)^T} \underbrace{\dot{\mathbf{y}}_t}_{\nabla \mathbf{h}_t \dot{\mathbf{q}}} , \quad (3.13)$$

with gradient $(\nabla g)^T = \partial g / \partial \mathbf{y}_t$ and Jacobian $\nabla \mathbf{h}_t = \partial \mathbf{h}_t / \partial \mathbf{q} = \mathbf{J}_t$. The limit case $\alpha(\mathbf{y}_t) \rightarrow 1$ implies $\beta(\mathbf{y}_t) \rightarrow \infty$. Hence, small values of $\dot{\mathbf{y}}_t$ result in large derivatives of the arc length.

3.5.1.2 Transversal subsystem

The transversal coordinates, ξ_1 and ξ_3 , are defined as the projections of $\mathbf{y}_t - \boldsymbol{\sigma}_t(\theta^*)$ onto the normal vectors \mathbf{e}_{\perp} and \mathbf{e}_{\parallel} , see also Fig. 3.2,

$$\xi_1 = \delta_1(\mathbf{y}_t) = \mathbf{e}_{\perp}^T(\theta^*) (\mathbf{y}_t - \boldsymbol{\sigma}_t(\theta^*)) \quad (3.14)$$

and

$$\xi_3 = \delta_2(\mathbf{y}_t) = \mathbf{e}_{\parallel}^T(\theta^*) (\mathbf{y}_t - \boldsymbol{\sigma}_t(\theta^*)) . \quad (3.15)$$

If the output \mathbf{y}_t is on the path, $\xi_1 = \xi_3 = 0$ holds. Differentiating (3.14) and (3.15) with respect to the time yields

$$\begin{aligned} \dot{\xi}_2 = \dot{\xi}_1 &= \underbrace{\left(\mathbf{e}'_{\perp}(\theta^*) \right)^T \dot{\theta}^* (\mathbf{y}_t - \boldsymbol{\sigma}_t(\theta^*))}_{(3.2a), (3.7)_0} - \underbrace{\mathbf{e}_{\perp}^T(\theta^*) \boldsymbol{\sigma}'_t(\theta^*) \dot{\theta}^*}_{(3.2c)_0} \\ &\quad + \underbrace{\mathbf{e}_{\perp}^T(\theta^*)}_{(\nabla \delta_1)^T} \underbrace{\dot{\mathbf{y}}_t}_{\nabla \mathbf{h}_t \dot{\mathbf{q}}} \end{aligned} \quad (3.16)$$

and

$$\begin{aligned} \xi_4 = \dot{\xi}_3 = & \underbrace{\left(\mathbf{e}'_{\hat{\mathbf{h}}}(\theta^*) \right)^T \dot{\theta}^* \left(\mathbf{y}_t - \boldsymbol{\sigma}_t(\theta^*) \right)}_{(3.2a),(3.7)_0} - \underbrace{\mathbf{e}_{\hat{\mathbf{h}}}^T(\theta^*) \boldsymbol{\sigma}'_t(\theta^*) \dot{\theta}^*}_{(3.2c)_0} \\ & + \underbrace{\mathbf{e}_{\hat{\mathbf{h}}}^T(\theta^*)}_{(\nabla \delta_2)^T} \underbrace{\dot{\mathbf{y}}_t}_{\nabla \mathbf{h}_t \dot{\mathbf{q}}}, \end{aligned} \quad (3.17)$$

with gradients $(\nabla \delta_1)^T = \partial \delta_1 / \partial \mathbf{y}_t$ and $(\nabla \delta_2)^T = \partial \delta_2 / \partial \mathbf{y}_t$. Note that the first parts in (3.16) and (3.17) are zero due to the usage of the parallel transport frame.

3.5.1.3 Rotational subsystem

The first and second rotational coordinates are chosen as

$$\boldsymbol{\zeta}_1 = \mathbf{y}_r = \mathbf{h}_r(\mathbf{q}) \quad (3.18)$$

and

$$\boldsymbol{\zeta}_2 = \dot{\boldsymbol{\zeta}}_1 = \dot{\mathbf{y}}_r = \nabla \mathbf{h}_r \dot{\mathbf{q}}, \quad (3.19)$$

with Jacobian $\nabla \mathbf{h}_r = \partial \mathbf{h}_r / \partial \mathbf{q} = \mathbf{J}_r$.

3.5.1.4 Diffeomorphism

The tangential, transversal, and rotational maps, (3.12)-(3.19), are used to construct a \mathcal{C}^1 -diffeomorphism, see, e.g., [84, p.147] for a definition of a diffeomorphism. The virtual PFC output $\hat{\mathbf{y}}_p^T = \hat{\mathbf{h}}_p^T(\mathbf{q}) = [\eta_1, \xi_1, \xi_3, \boldsymbol{\zeta}_1^T]$ is introduced, which allows to define the mapping

$$\begin{bmatrix} \hat{\mathbf{y}}_p \\ \hat{\mathbf{y}}_p \end{bmatrix} = \begin{bmatrix} \eta_1 \\ \xi_1 \\ \xi_3 \\ \boldsymbol{\zeta}_1 \\ \eta_2 \\ \xi_2 \\ \xi_4 \\ \boldsymbol{\zeta}_2 \end{bmatrix} = \begin{bmatrix} g \circ \mathbf{h}_t(\mathbf{q}) \\ \delta_1 \circ \mathbf{h}_t(\mathbf{q}) \\ \delta_2 \circ \mathbf{h}_t(\mathbf{q}) \\ \mathbf{h}_r(\mathbf{q}) \\ (\nabla g)^T \nabla \mathbf{h}_t \dot{\mathbf{q}} \\ (\nabla \delta_1)^T \nabla \mathbf{h}_t \dot{\mathbf{q}} \\ (\nabla \delta_2)^T \nabla \mathbf{h}_t \dot{\mathbf{q}} \\ \nabla \mathbf{h}_r \dot{\mathbf{q}} \end{bmatrix} = \boldsymbol{\Phi}(\mathbf{q}, \dot{\mathbf{q}}). \quad (3.20)$$

Throughout this work, the subscript p in $\hat{\mathbf{y}}_p$, $\hat{\mathbf{h}}_p$, $\hat{\mathbf{J}}_p, \dots$ refers to PFC, whereas the subscript s is used for surface following control introduced in the next chapter.

Lemma 1. *The mapping $\boldsymbol{\Phi} : \mathcal{X} \mapsto \mathcal{Z}$ with $\mathcal{X} = \mathcal{Q} \times \mathcal{T}_{\mathbf{q}}\mathcal{Q}$, $\mathcal{Q} = \{\bar{\mathbf{q}} \in \mathbb{R}^N : \alpha \circ \mathbf{h}_t(\bar{\mathbf{q}}) < 1\}$, and tangential space $\mathcal{T}_{\mathbf{q}}\mathcal{Q}$ is a \mathcal{C}^1 -diffeomorphism, if $\mathbf{J}^T(\mathbf{q}) = [\mathbf{J}_t^T(\mathbf{q}), \mathbf{J}_r^T(\mathbf{q})]$ is nonsingular.*

Proof. By the inverse function theorem, see, e.g., Theorem 8.2 in [84], the following properties have to be shown:

- (i.) \mathcal{X} and \mathcal{Z} are open in \mathbb{R}^{2N} ,
- (ii.) $\Phi \in \mathcal{C}^1(\mathcal{X}, \mathcal{Z})$, and
- (iii.) $\nabla \Phi = [\partial \Phi / \partial \mathbf{q}, \partial \Phi / \partial \dot{\mathbf{q}}]$ is nonsingular for all $[\mathbf{q}^T, \dot{\mathbf{q}}^T]^T \in \mathcal{X}$.

Since \mathcal{Q} is an open subset of \mathbb{R}^N , \mathcal{X} and \mathcal{Z} are open in \mathbb{R}^{2N} . Since the output $\mathbf{y} = \mathbf{h}(\mathbf{q})$ is assumed to be sufficiently smooth and $\sigma(\theta) \in \mathcal{C}^3(\mathcal{T}, \mathbb{R}^N)$, $\Phi \in \mathcal{C}^1(\mathcal{X}, \mathcal{Z})$ holds. The Jacobian of Φ reads as

$$\nabla \Phi = \begin{bmatrix} \hat{\mathbf{J}}_p(\mathbf{q}) & \mathbf{0} \\ * & \hat{\mathbf{J}}_p(\mathbf{q}) \end{bmatrix}, \quad (3.21)$$

with the PFC Jacobian

$$\hat{\mathbf{J}}_p(\mathbf{q}) = \mathbf{L}_p(\mathbf{q})\mathbf{J}(\mathbf{q}) \quad (3.22)$$

and matrices

$$\mathbf{L}_p(\mathbf{q}) = \begin{bmatrix} \mathbf{E}_p(\mathbf{q}) & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \quad \text{and} \quad \mathbf{E}_p(\mathbf{q}) = \begin{bmatrix} \beta(\mathbf{y}_t)\mathbf{e}_{\parallel}^T \\ \mathbf{e}_{\perp}^T \\ \mathbf{e}_{\dot{\parallel}}^T \end{bmatrix},$$

where \mathbf{I} is the 3×3 identity matrix. If $\mathbf{J}(\mathbf{q})$ is nonsingular and $\alpha(\mathbf{y}_t) < 1$, then, $\mathbf{E}_p(\mathbf{q})$ and $\mathbf{L}_p(\mathbf{q})$ are nonsingular, $\beta(\mathbf{y}_t) < \infty$, and thus, $\nabla \Phi$ is nonsingular for all $[\mathbf{q}^T, \dot{\mathbf{q}}^T]^T \in \mathcal{X}$. \square

From (3.20) and (3.22) it follows that the relation

$$\dot{\mathbf{y}}_p = \mathbf{L}_p(\mathbf{q})\dot{\mathbf{y}} = \hat{\mathbf{J}}_p(\mathbf{q})\dot{\mathbf{q}} \quad (3.23)$$

holds.

3.5.2 Feedback linearization

Differentiating the tangential state $\eta_2 = \dot{\eta}_1$ from (3.13) with respect to the time yields

$$\begin{aligned} \ddot{\eta}_1 &= \left((\nabla \beta)^T \dot{\mathbf{y}}_t \mathbf{e}_{\parallel}^T(\theta^*) + \beta(\mathbf{y}_t) \left(\mathbf{e}'_{\parallel}(\theta^*) \right)^T \dot{\theta}^* \right) \dot{\mathbf{y}}_t \\ &\quad + \beta(\mathbf{y}_t) \mathbf{e}_{\parallel}^T(\theta^*) \ddot{\mathbf{y}}_t, \end{aligned} \quad (3.24)$$

where $(\nabla \beta)^T = \partial \beta / \partial \mathbf{y}_t$. The time derivatives of the transversal states $\xi_2 = \dot{\xi}_1$ from (3.16) and $\xi_4 = \dot{\xi}_3$ from (3.17) take the form

$$\ddot{\xi}_1 = \left(\mathbf{e}'_{\perp}(\theta^*) \right)^T \dot{\theta}^* \dot{\mathbf{y}}_t + \mathbf{e}_{\perp}^T(\theta^*) \ddot{\mathbf{y}}_t \quad (3.25)$$

and

$$\ddot{\xi}_3 = (\mathbf{e}'_{\mathfrak{n}}(\theta^*))^T \dot{\theta}^* \dot{\mathbf{y}}_t + \mathbf{e}_{\mathfrak{n}}^T(\theta^*) \ddot{\mathbf{y}}_t . \quad (3.26)$$

The second-order time derivative of ζ_1 from (3.18) gives

$$\ddot{\zeta}_1 = \ddot{\mathbf{y}}_r . \quad (3.27)$$

The equations (3.24)-(3.27) can be written in matrix form as

$$\ddot{\hat{\mathbf{y}}}_p = \hat{\mathbf{J}}_p(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{q}} + \hat{\mathbf{J}}_p(\mathbf{q}) \mathbf{D}^{-1}(\mathbf{q}) (\boldsymbol{\tau}_d + \boldsymbol{\tau}_{ext} - \mathbf{n}(\mathbf{q}, \dot{\mathbf{q}})) . \quad (3.28)$$

Hence, application of the feedback transformation

$$\boldsymbol{\tau}_d = \mathbf{n}(\mathbf{q}, \dot{\mathbf{q}}) - \boldsymbol{\tau}_{ext} + \mathbf{D}(\mathbf{q}) \hat{\mathbf{J}}_p^{-1}(\mathbf{q}) (\mathbf{v}_p - \hat{\mathbf{J}}_p(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{q}}) \quad (3.29)$$

to the system (2.42) and (2.1), with new control input $\mathbf{v}_p^T = [\mathbf{v}_{p,t}^T, \mathbf{v}_{p,r}^T]$, where $\mathbf{v}_{p,t}^T = [v_{||}, v_{\perp}, v_{\mathfrak{n}}]$, results in a linear input-output relation from the new input \mathbf{v}_p to the virtual output $\hat{\mathbf{y}}_p$ in the form of N integrator chains of length two

$$\ddot{\hat{\mathbf{y}}}_p = \mathbf{v}_p . \quad (3.30)$$

Note that the dynamics of the transformed system are linear with respect to a nonlinear plant and a nonlinear path γ .

The virtual inputs in the direction of the normal vectors, v_{\perp} and $v_{\mathfrak{n}}$, can effectively be used to fulfill objective (O1), i.e., to stabilize the transversal ξ -subsystem and to guarantee asymptotic convergence to $\boldsymbol{\sigma}_t(\cdot)$. The controlled invariant subset is given by

$$\Gamma^* = \left\{ [\bar{\mathbf{q}}^T, \dot{\bar{\mathbf{q}}}^T]^T \in \mathcal{X} : \Phi_{\xi}(\bar{\mathbf{q}}, \dot{\bar{\mathbf{q}}}) = \mathbf{0} \right\} , \quad (3.31)$$

with

$$\Phi_{\xi}(\mathbf{q}, \dot{\mathbf{q}}) = \begin{bmatrix} \delta_1 \circ \mathbf{h}_t(\mathbf{q}) \\ \delta_2 \circ \mathbf{h}_t(\mathbf{q}) \\ (\nabla \delta_1)^T \nabla \mathbf{h}_t \dot{\mathbf{q}} \\ (\nabla \delta_2)^T \nabla \mathbf{h}_t \dot{\mathbf{q}} \end{bmatrix} . \quad (3.32)$$

Thus, objective (O2) is met, because if $[\bar{\mathbf{q}}^T(t_0) \quad \dot{\bar{\mathbf{q}}}^T(t_0)]^T \in \Gamma^*$, then, by choosing $v_{\perp} = 0$ and $v_{\mathfrak{n}} = 0$, $\xi_i = 0$, with $i = 1, 2, 3, 4$ and $\|\mathbf{y}_t(t)\|_{\gamma_t} = 0$, $\forall t > t_0$. Moreover, the virtual input in tangential direction $v_{||}$ allows to control the motion along the path, thus, objective (O3) can be satisfied.

3.5.3 Stabilization of the linearized system

Any controller that stabilizes the (exactly) linearized systems (3.30) can be used to compute the new input \mathbf{v}_p . In the following, a position controller and a compliance controller for the PFC to compute \mathbf{v}_p are presented.

3.5.3.1 Position control

The simple position control law

$$\mathbf{v}_p = \begin{bmatrix} \mathbf{v}_{p,t} \\ \mathbf{v}_{p,r} \end{bmatrix} = \begin{bmatrix} \ddot{\eta}_1^p - a_{\eta,2}\dot{e}_\eta^p - a_{\eta,1}e_\eta^p \\ \ddot{\xi}_1^p - a_{\xi,2}\dot{e}_{\xi_1}^p - a_{\xi,1}e_{\xi_1}^p \\ \ddot{\xi}_3^p - a_{\xi,2}\dot{e}_{\xi_3}^p - a_{\xi,1}e_{\xi_3}^p \\ \ddot{\mathbf{y}}_r^p - a_{r,2}\dot{\mathbf{e}}_r^p - a_{r,1}\mathbf{e}_r^p \end{bmatrix} \quad (3.33)$$

is utilized, where $e_\eta^p = \eta_1 - \eta_1^p$, $e_{\xi_j}^p = \xi_j - \xi_j^p$, $\mathbf{e}_r^p = \mathbf{y}_r - \mathbf{y}_r^p$, $a_{i,j} > 0$ with $i \in \{\eta, \xi, r\}$ and $j = 1, 2$. The \mathcal{C}^2 reference position on the path is denoted by η_1^p , where the superscript p designates references for position controllers throughout this work. The \mathcal{C}^2 references for the orthogonal states and the orientation are given by ξ_1^p , ξ_3^p , and $\mathbf{y}_r^p = \boldsymbol{\sigma}_r(\theta^*)$, respectively. Note that integral parts can also be added to the control law (3.33) to eliminate the control error in stationary conditions. Inserting the control law (3.33) into the linear system (3.30) yields the exponentially stable error dynamics

$$\begin{bmatrix} \ddot{e}_\eta^p + a_{\eta,2}\dot{e}_\eta^p + a_{\eta,1}e_\eta^p \\ \ddot{e}_{\xi_1}^p + a_{\xi,2}\dot{e}_{\xi_1}^p + a_{\xi,1}e_{\xi_1}^p \\ \ddot{e}_{\xi_3}^p + a_{\xi,2}\dot{e}_{\xi_3}^p + a_{\xi,1}e_{\xi_3}^p \\ \ddot{\mathbf{e}}_r^p + a_{r,2}\dot{\mathbf{e}}_r^p + a_{r,1}\mathbf{e}_r^p \end{bmatrix} = \mathbf{0} . \quad (3.34)$$

The components two and three of (3.34) show that ξ_1 and ξ_3 exponentially converge to the references ξ_1^p and ξ_3^p , respectively, and, thus, objective (O1) is fulfilled for $\xi_1^p = \xi_3^p = 0$. Additionally, if $[\mathbf{q}^T(t_0), \dot{\mathbf{q}}^T(t_0)]^T \in \Gamma^*$, then $v_\perp = 0$ and $v_\parallel = 0$ holds, the output \mathbf{y}_t stays on the path γ_t for all $t \geq t_0$, and objective (O2) is fulfilled. Component one of (3.34) shows that the tangential coordinate η_1 exponentially converges to the reference η_1^p . Hence, objective (O3) can also be satisfied. A schematic drawing of the PFC with position control is depicted in Fig. 3.3.

As introduced in Chapter 2, the orientation is always represented by Euler angles in the case of $n_r = 3$. In this case, the control law (3.33) demands the use of the analytical Jacobian \mathbf{J}_a of (2.17) for the calculation of the PFC Jacobian $\hat{\mathbf{J}}_p$ in (3.22), i.e. $\mathbf{J} = \mathbf{J}_a$. It should always be possible to find an appropriate Euler angle definition (e.g. ZYZ, ZYX) that avoids representation singularities for the reference $\boldsymbol{\phi}_d^T = [\varphi_d, \vartheta_d, \psi_d] = \boldsymbol{\sigma}_r^T(\theta^*)$. However, due to control errors, the actual orientation $\mathbf{y}_r = \boldsymbol{\phi}_e$ can get close to representation singularities leading to an ill-conditioned control law. To avoid this problem, the geometric Jacobian \mathbf{J}_g of (2.11) is used in the calculation of the PFC Jacobian $\hat{\mathbf{J}}_p$, i.e. $\mathbf{J} = \mathbf{J}_g$, resulting in the linear system

$$\ddot{\mathbf{y}}_p^T = [\ddot{\eta}_1, \ddot{\xi}_1, \ddot{\xi}_3, \ddot{\boldsymbol{\omega}}_e^T]^T = [\mathbf{v}_{p,t}^T, \mathbf{v}_{p,o}^T]^T = \mathbf{v}_p^T , \quad (3.35)$$

with the new rotational control input $\mathbf{v}_{p,o}$ and the angular velocities of the end-effector $\boldsymbol{\omega}_e$ expressed in the inertial frame. The orientation error can then be

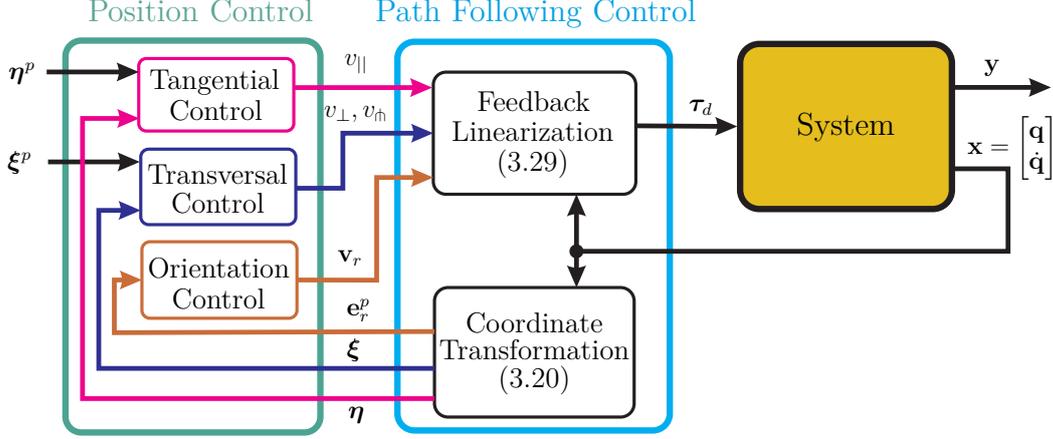


Figure 3.3: Path following control scheme.

defined locally as, see [6],

$$\mathbf{R}_d^e = \mathbf{R}_e^T \mathbf{R}_d, \quad (3.36)$$

where \mathbf{R}_d is the rotation matrix of the reference Euler angles ϕ_d , \mathbf{R}_e is the rotation matrix of the end-effector orientation ϕ_e , and ϕ_{de} are the ZYX Euler angles of \mathbf{R}_d^e . The time derivative of (3.36) yields, see (2.22),

$$\dot{\mathbf{R}}_d^e = \mathbf{R}_e^T (\mathbf{S}^T(\boldsymbol{\omega}_e) + \mathbf{S}(\boldsymbol{\omega}_d)) \mathbf{R}_d = \mathbf{S}(\mathbf{R}_e^T(\boldsymbol{\omega}_d - \boldsymbol{\omega}_e)) \mathbf{R}_d^e \quad (3.37)$$

and, therefore, the relation $\mathbf{R}_e^T(\boldsymbol{\omega}_d - \boldsymbol{\omega}_e) = \mathbf{T}_{ZYX}(\phi_{de}) \dot{\phi}_{de}$ holds, see (2.15). Following along the lines of [6], the new rotational control input is chosen as

$$\mathbf{v}_{p,o} = \dot{\boldsymbol{\omega}}_d - \dot{\mathbf{B}}_e(\phi_{de}) \dot{\phi}_{de} + \mathbf{B}_e(\phi_{de}) (a_{r,2} \dot{\phi}_{de} + a_{r,1} \phi_{de}), \quad (3.38)$$

with $\boldsymbol{\omega}_d = \mathbf{T}_{ZYX}(\boldsymbol{\sigma}_r(\theta^*)) \boldsymbol{\sigma}'_r(\theta^*) \dot{\theta}^*$, $\mathbf{B}_e(\phi_{de}) = \mathbf{R}_e \mathbf{T}_{ZYX}(\phi_{de})$, and $\mathbf{T}_{ZYX}(\cdot)$ according to (2.21). Hence, the control law (3.38) is well defined for angular control errors of less than $\pi/2$. Replacing the control law for $\mathbf{v}_{p,r}$ in (3.33) with $\mathbf{v}_{p,o}$ of (3.38) and using the relation $\boldsymbol{\omega}_d - \boldsymbol{\omega}_e = \mathbf{B}_e(\phi_{de}) \dot{\phi}_{de}$ yields the exponentially stable error dynamics for the orientation

$$\ddot{\phi}_{de} + a_{r,2} \dot{\phi}_{de} + a_{r,1} \phi_{de} = \mathbf{0}, \quad (3.39)$$

with $a_{r,1} > 0$ and $a_{r,2} > 0$.

3.5.3.2 Compliance control

Compliance control addresses a classical problem in robotics of simultaneously controlling the position and the interaction forces with the environment [26]. In the following, it is shown how to combine the presented path following control concept with compliance control. For this, a position-based impedance control is

employed, see [29]. In position-based impedance control, the compliance is realized by tracking the trajectory of the exponentially stable reference impedance model using position control in an inner control loop and an impedance control in the outer loop. As shown in [32], this method is well suited for accurate positioning in free space as well as for scenarios where the robot is in contact with rigid environments. Due to the possibility of using high gains in the inner loop, the position-based impedance control is rather insensitive to model uncertainties.

The fundamental idea of compliance control is to design a controller which imposes a reference dynamics (impedance) between external forces and the position. In the case of the PFC approach presented in this section, the reference impedance of the motion along and orthogonal to the path γ_t as well as the reference impedance of the orientation can be separately defined as

$$\begin{bmatrix} \tau_{\parallel} \\ \tau_{\perp} \\ \tau_{\text{th}} \\ \boldsymbol{\tau}_r \end{bmatrix} = \begin{bmatrix} m_{\parallel}^d \ddot{e}_{\eta}^d + d_{\parallel}^d \dot{e}_{\eta}^d + k_{\parallel}^d e_{\eta}^d \\ m_{\perp}^d \ddot{e}_{\xi_1}^d + d_{\perp}^d \dot{e}_{\xi_1}^d + k_{\perp}^d e_{\xi_1}^d \\ m_{\perp}^d \ddot{e}_{\xi_3}^d + d_{\perp}^d \dot{e}_{\xi_3}^d + k_{\perp}^d e_{\xi_3}^d \\ m_r^d \ddot{\mathbf{e}}_r^d + d_r^d \dot{\mathbf{e}}_r^d + k_r^d \mathbf{e}_r^d \end{bmatrix}, \quad (3.40)$$

where $e_{\eta}^d = \eta_1 - \eta_1^d$, $e_{\xi_1}^d = \xi_1 - \xi_1^d$, and $e_{\xi_3}^d = \xi_3 - \xi_3^d$ denote the errors between the coordinates η_1 , ξ_1 , and ξ_3 and the references η_1^d , ξ_1^d , and ξ_3^d and $\mathbf{e}_r^d = \mathbf{y}_r - \boldsymbol{\sigma}_r(\theta^*)$. Moreover, m_i^d , d_i^d and k_i^d for $i = \{\parallel, \perp, r\}$ are design parameters representing the mass, damping, and stiffness and

$$\begin{bmatrix} \tau_{\parallel} \\ \tau_{\perp} \\ \tau_{\text{th}} \\ \boldsymbol{\tau}_r \end{bmatrix} = \hat{\mathbf{J}}_p^{-\text{T}} \boldsymbol{\tau}_{ext} = \begin{bmatrix} \frac{1}{\beta} \mathbf{e}_{\parallel}^{\text{T}} & \mathbf{0} \\ \mathbf{e}_{\perp}^{\text{T}} & \mathbf{0} \\ \mathbf{e}_{\text{th}}^{\text{T}} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \mathbf{J}_g^{-\text{T}} \boldsymbol{\tau}_{ext} \quad (3.41)$$

are the external (projected) generalized forces.

Assuming perfect tracking of the inner position loop (3.33), the actual tangential and transversal coordinates η_1 , ξ_1 , and ξ_3 as well as the orientation \mathbf{y}_r in (3.40) can be replaced by the position controller references η_1^p , ξ_1^p , ξ_3^p , and \mathbf{y}_r^p . Introducing the errors $e_{\eta}^{pd} = \eta_1^p - \eta_1^d$, $e_{\xi_1}^{pd} = \xi_1^p - \xi_1^d$, $e_{\xi_3}^{pd} = \xi_3^p - \xi_3^d$, and $\mathbf{e}_r^{pd} = \mathbf{y}_r^p - \boldsymbol{\sigma}_r(\theta^*)$ allows to deduce the impedance control law

$$\ddot{\eta}_1^p = \ddot{\eta}_1^d + \frac{\tau_{\parallel}}{m_{\parallel}^d} - \frac{d_{\parallel}^d}{m_{\parallel}^d} \dot{e}_{\eta}^{pd} - \frac{k_{\parallel}^d}{m_{\parallel}^d} e_{\eta}^{pd}, \quad (3.42a)$$

$$\dot{\eta}_1^p = \int_0^t \ddot{\eta}_1^p d\tau, \quad \eta_1^p = \int_0^t \dot{\eta}_1^p d\tau,$$

$$\ddot{\xi}_1^p = \ddot{\xi}_1^d + \frac{\tau_{\perp}}{m_{\perp}^d} - \frac{d_{\perp}^d}{m_{\perp}^d} \dot{e}_{\xi_1}^{pd} - \frac{k_{\perp}^d}{m_{\perp}^d} e_{\xi_1}^{pd}, \quad (3.42b)$$

$$\dot{\xi}_1^p = \int_0^t \ddot{\xi}_1^p d\tau, \quad \xi_1^p = \int_0^t \dot{\xi}_1^p d\tau,$$

$$\begin{aligned}\ddot{\xi}_3^p &= \ddot{\xi}_3^d + \frac{\tau_{\hat{n}}}{m_{\perp}^d} - \frac{d_{\perp}^d}{m_{\perp}^d} \dot{e}_{\xi_3}^{pd} - \frac{k_{\perp}^d}{m_{\perp}^d} e_{\xi_3}^{pd}, \\ \dot{\xi}_3^p &= \int_0^t \ddot{\xi}_3^p d\tau, \quad \xi_3^p = \int_0^t \dot{\xi}_3^p d\tau,\end{aligned}\tag{3.42c}$$

$$\begin{aligned}\ddot{\mathbf{y}}_r^p &= \ddot{\boldsymbol{\sigma}}_r(\theta^*) + \frac{\boldsymbol{\tau}_r}{m_r^d} - \frac{d_r^d}{m_r^d} \dot{\mathbf{e}}_r^{pd} - \frac{k_r^d}{m_r^d} \mathbf{e}_r^{pd}, \\ \dot{\mathbf{y}}_r^p &= \int_0^t \ddot{\mathbf{y}}_r^p d\tau, \quad \mathbf{y}_r^p = \int_0^t \dot{\mathbf{y}}_r^p d\tau.\end{aligned}\tag{3.42d}$$

Hence, in the combination of PFC with compliance control, the (external) references in tangential and orthogonal direction to the path γ are denoted by the superscript d and the references for the inner position control loop are denoted by the superscript p .

If the Euler angles are used in the case of $\mathbf{y}_r \in \mathbb{R}^3$, the impedance depends on the orientation of the compliant frame with respect to the inertial frame when using the control law (3.33) to compute $\mathbf{v}_{p,r}$ together with the compliance control (3.42d), cf. [33]. To avoid this disadvantage and to cope with the representation singularities, the geometric Jacobian \mathbf{J}_g of (2.11) is used in the calculation of the PFC Jacobian $\hat{\mathbf{J}}_p$ leading to the linearized system (3.35) and the orientation between the compliant frame p and the desired frame d is defined as, cf. [33],

$$\mathbf{R}_p^d = \mathbf{R}_d^T \mathbf{R}_p, \tag{3.43}$$

where \mathbf{R}_d is the rotation matrix of the reference Euler angles $\boldsymbol{\phi}_d = \boldsymbol{\sigma}_r(\theta^*)$ and \mathbf{R}_p is the rotation matrix of the compliant frame. The impedance is defined as

$$m_r^d \ddot{\boldsymbol{\phi}}_{pd} + d_r^d \dot{\boldsymbol{\phi}}_{pd} + k_r^d \boldsymbol{\phi}_{pd} = \mathbf{T}_{ZYX}^T(\boldsymbol{\phi}_{pd}) \boldsymbol{\mu}^d, \tag{3.44}$$

where $\boldsymbol{\phi}_{pd} = [\varphi_{pd}, \vartheta_{pd}, \psi_{pd}]$ are the ZYX Euler angles of \mathbf{R}_p^d , and $(\boldsymbol{\mu}^d)^T = [\mu_{\varphi}^d, \mu_{\vartheta}^d, \mu_{\psi}^d]$ is the measured torque vector expressed in frame d . Since $\boldsymbol{\omega}^T \boldsymbol{\mu} = \boldsymbol{\omega}^T [\mathbf{0} \ \mathbf{I}] \mathbf{J}_g^{-T} \boldsymbol{\tau}_{ext} = \boldsymbol{\omega}^T \mathbf{R}_d \mathbf{R}_d^T [\mathbf{0} \ \mathbf{I}] \mathbf{J}_g^{-T} \boldsymbol{\tau}_{ext} = (\boldsymbol{\omega}^d)^T [\mathbf{0} \ \mathbf{R}_d^T] \mathbf{J}_g^{-T} \boldsymbol{\tau}_{ext}$, with the angular velocity $\boldsymbol{\omega}^d$ expressed in frame d holds, the measured torque vector is given by

$$\boldsymbol{\mu}^d = [\mathbf{0} \ \mathbf{R}_d^T] \mathbf{J}_g^{-T} \boldsymbol{\tau}_{ext}. \tag{3.45}$$

Note that representation singularities do not appear in (3.44) for $|\vartheta_{pd}| < \pi/2$. The control law

$$\begin{aligned}\mathbf{v}_{p,o} &= \dot{\boldsymbol{\omega}}_d + \dot{\mathbf{B}}_d(\boldsymbol{\phi}_{ed}) \dot{\boldsymbol{\phi}}_{ed} \\ &+ \mathbf{B}_d(\boldsymbol{\phi}_{ed}) \left(\ddot{\boldsymbol{\phi}}_{pd} + a_{r,2} (\dot{\boldsymbol{\phi}}_{pd} - \dot{\boldsymbol{\phi}}_{ed}) + a_{r,1} (\boldsymbol{\phi}_{pd} - \boldsymbol{\phi}_{ed}) \right),\end{aligned}\tag{3.46}$$

with $\mathbf{B}_d(\boldsymbol{\phi}_{ed}) = \mathbf{R}_d \mathbf{T}_{ZYX}(\boldsymbol{\phi}_{ed})$, $\boldsymbol{\omega}_d = \mathbf{T}_i(\boldsymbol{\sigma}_r(\theta^*)) \boldsymbol{\sigma}'_r(\theta^*) \dot{\theta}^*$, for $i \in \{ZYX, ZYZ\}$, and $\boldsymbol{\phi}_{ed}$ as the ZYX Euler angles of \mathbf{R}_e^d leads to the exponentially stable error dynamics for the orientation

$$\ddot{\boldsymbol{\phi}}_{ed} - \ddot{\boldsymbol{\phi}}_{pd} + a_{r,2} (\dot{\boldsymbol{\phi}}_{ed} - \dot{\boldsymbol{\phi}}_{pd}) + a_{r,1} (\boldsymbol{\phi}_{ed} - \boldsymbol{\phi}_{pd}) = \mathbf{0} , \quad (3.47)$$

cf. [33]. Hence, the orientation of the end-effector \mathbf{R}_e converges to the desired compliant orientation \mathbf{R}_p .

3.5.4 Simplification using joint velocity controllers

In robotics, the nonlinear system dynamics (2.42) are often neglected and the joints are independently controlled using fast high bandwidth linear position and/or velocity controllers. Feedforward of, e.g., the gravitational forces can be used to improve the performance of the subordinate velocity controllers. If the joint velocity controllers are assumed to be ideal, the system dynamics (2.42) simplifies to

$$\dot{\mathbf{q}} = \dot{\mathbf{q}}_{\text{ref}} , \quad (3.48)$$

with the reference velocity input $\dot{\mathbf{q}}_{\text{ref}}$.

In the following, a path following control approach based on (3.48) is presented and is denoted as kinematic PFC. In contrast to the PFC based on the dynamics for fully actuated systems (2.42), $\boldsymbol{\sigma}(\theta)$ has only to be \mathcal{C}^2 for the kinematic PFC because the (vector) relative degree of the system reduces to $\{1, 1, \dots, 1\}$.

3.5.4.1 Coordinate transformation

The coordinate transformation $\boldsymbol{\Phi} : \mathbb{R}^N \rightarrow \mathbb{R}^N$ is derived, which maps the generalized coordinates \mathbf{q} to the tangential, transversal, and rotational coordinates $\hat{\mathbf{y}}_p$ and reads as

$$\hat{\mathbf{y}}_p = \begin{bmatrix} g \circ \mathbf{h}_t(\mathbf{q}) \\ \delta_1 \circ \mathbf{h}_t(\mathbf{q}) \\ \delta_2 \circ \mathbf{h}_t(\mathbf{q}) \\ \mathbf{h}_r(\mathbf{q}) \end{bmatrix} = \boldsymbol{\Phi}(\mathbf{q}) , \quad (3.49)$$

with g , δ_1 , and δ_2 from (3.12), (3.14), and (3.15), respectively.

Lemma 2. *The mapping $\boldsymbol{\Phi} : \mathcal{Q} \mapsto \mathcal{Z} \subset \mathbb{R}^N$, with $\mathcal{Q} = \{\bar{\mathbf{q}} \in \mathbb{R}^N : \alpha \circ \mathbf{h}_t(\bar{\mathbf{q}}) < 1\}$, is a \mathcal{C}^1 -diffeomorphism, if $\mathbf{J}(\mathbf{q})$ is nonsingular.*

Proof. Based on the inverse function theorem it has to be shown that

- (i.) \mathcal{Q} and \mathcal{Z} are open in \mathbb{R}^N ,
- (ii.) $\boldsymbol{\Phi} \in \mathcal{C}^1(\mathcal{Q}, \mathcal{Z})$, and
- (iii.) $\nabla \boldsymbol{\Phi} = \partial \boldsymbol{\Phi} / \partial \mathbf{q} = \hat{\mathbf{J}}_p(\mathbf{q})$ is nonsingular for all $\mathbf{q} \in \mathcal{Q}$.

Since \mathcal{Q} is an open subset of \mathbb{R}^N , \mathcal{Z} is open in \mathbb{R}^N . As the output $\mathbf{y} = \mathbf{h}(\mathbf{q})$ is assumed to be sufficiently smooth and $\boldsymbol{\sigma}(\theta) \in \mathcal{C}^2(\mathcal{T}, \mathbb{R}^N)$, $\boldsymbol{\Phi} \in \mathcal{C}^1(\mathcal{Q}, \mathcal{Z})$ holds. If $\mathbf{J}(\mathbf{q})$ is nonsingular and $\alpha(\mathbf{y}_t) < 1$, then, $\mathbf{E}_p(\mathbf{q})$ and $\mathbf{L}_p(\mathbf{q})$ are nonsingular, $\beta(\mathbf{y}_t) < \infty$, thus, $\nabla \boldsymbol{\Phi} = \hat{\mathbf{J}}_p(\mathbf{q})$ is nonsingular for all $\mathbf{q} \in \mathcal{Q}$. \square

3.5.4.2 Feedback linearization

Because of the relation (3.23), application of the feedback transformation

$$\dot{\mathbf{q}}_{\text{ref}} = \hat{\mathbf{J}}_p^{-1} \mathbf{v}_{p,k} \quad (3.50)$$

to the system (3.48) and (2.1), with the new control input $\mathbf{v}_{p,k}^T = [\mathbf{v}_{p,t,k}^T, \mathbf{v}_{p,r,k}^T]$, where $\mathbf{v}_{p,t,k}^T = [v_{||,k}, v_{\perp,k}, v_{\dot{\theta},k}]$, results in a linear input-output relation from the new control input $\mathbf{v}_{p,k}$ to the virtual PFC output $\hat{\mathbf{y}}_p$ in the form of N decoupled integrators

$$\dot{\hat{\mathbf{y}}}_p = \mathbf{v}_{p,k} . \quad (3.51)$$

3.5.4.3 Stabilization of the linearized system

The position control law

$$\mathbf{v}_{p,k} = \begin{bmatrix} \mathbf{v}_{p,t,k} \\ \mathbf{v}_{p,r,k} \end{bmatrix} = \begin{bmatrix} \dot{\eta}_1^p - a_\eta e_\eta^p \\ \dot{\xi}_1^p - a_\xi e_{\xi_1}^p \\ \dot{\xi}_3^p - a_\xi e_{\xi_3}^p \\ \dot{\mathbf{y}}_r^p - a_r \mathbf{e}_r^p \end{bmatrix}, \quad (3.52)$$

where $e_\eta^p = \eta_1 - \eta_1^p$, $e_{\xi_1}^p = \xi_1 - \xi_1^p$, $e_{\xi_3}^p = \xi_3 - \xi_3^p$, and $\mathbf{e}_r^p = \mathbf{y}_r - \mathbf{y}_r^p$, yields an exponentially stable error dynamics, if $a_i > 0$ with $i \in \{\eta, \xi, r\}$. The reference position on the path is denoted by η_1^p and the references for the transversal states by ξ_1^p and ξ_3^p .

A compliant behavior is achieved by using (3.42) to compute the references η_1^p , ξ_1^p , ξ_3^p , and \mathbf{y}_r^p together with the position controller (3.52).

A similar approach as in Section 3.5.3.2 is used to deal with representation singularities in the case of $n_r = 3$. Using the geometric Jacobian to compute the PFC Jacobian $\hat{\mathbf{J}}_p$, i.e., $\mathbf{J} = \mathbf{J}_g$, for the PFC feedback transformation (3.50) yields the linear system $\dot{\hat{\mathbf{y}}}_p^T = [\dot{\eta}_1^T, \dot{\xi}_1^T, \dot{\boldsymbol{\omega}}_e^T] = [\mathbf{v}_{p,t,k}^T, \mathbf{v}_{p,o,k}^T]$, where $\mathbf{v}_{p,o,k}$ denotes the new rotational control input. The Euler angles $\boldsymbol{\phi}_{pd}$ are determined by the impedance law (3.44) and the rotation matrix of the compliant frame is given by $\mathbf{R}_p = \mathbf{R}_d(\boldsymbol{\phi}_d) \mathbf{R}_p^d(\boldsymbol{\phi}_{pd})$, see (3.43). The control law

$$\mathbf{v}_{p,o,k} = \boldsymbol{\omega}_p - \mathbf{B}_p(\boldsymbol{\phi}_{ep}) a_r \boldsymbol{\phi}_{ep}, \quad (3.53)$$

with $\mathbf{B}_p(\boldsymbol{\phi}_{ep}) = \mathbf{R}_p \mathbf{T}_{ZYX}(\boldsymbol{\phi}_{ep})$ and $\boldsymbol{\phi}_{ep}$ as the ZYX Euler angles of $\mathbf{R}_e^p = \mathbf{R}_p^T \mathbf{R}_e$, leads to the exponentially stable error dynamics

$$\dot{\boldsymbol{\phi}}_{ep} + a_r \boldsymbol{\phi}_{ep} = \mathbf{0}, \quad (3.54)$$

with $a_r > 0$. For this derivation, the relations $\boldsymbol{\omega}_{pd}^d = \mathbf{T}_{ZYX}(\boldsymbol{\phi}_{pd})\dot{\boldsymbol{\phi}}_{pd}$, $\boldsymbol{\omega}_p = \boldsymbol{\omega}_d + \mathbf{R}_d\boldsymbol{\omega}_{pd}^d$, $\boldsymbol{\omega}_d = \mathbf{T}_i(\boldsymbol{\sigma}_r(\theta^*))\boldsymbol{\sigma}'_r(\theta^*)\dot{\theta}^*$ with $i \in \{ZYX, ZYZ\}$, and $\boldsymbol{\omega}_{ep} = \boldsymbol{\omega}_e - \boldsymbol{\omega}_p = \mathbf{R}_p\boldsymbol{\omega}_{ep}^p = \mathbf{B}_p(\boldsymbol{\phi}_{ep})\dot{\boldsymbol{\phi}}_{ep}$ are used.

3.6 Path following control for elastic joint robots

The dynamic system (2.64) with output function (2.1) is not input-output linearizable by applying static state feedback [7, 9]. De Luca showed that linearization is possible using dynamic state feedback [9]. However, this feedback law becomes quite complicated for manipulators with more than two degrees of freedom.

This section presents two path following controllers for the simplified elastic joint manipulator dynamics (2.65) with joint forces $\boldsymbol{\tau}_J$ including viscosity (2.66) and without viscosity (2.67), respectively, which are both input-output linearizable via static state feedback.

3.6.1 Robots with visco-elastic joints

The simplified elastic joint manipulator dynamics (2.65) with joint forces $\boldsymbol{\tau}_J$ including viscosity (2.66) (visco-elastic joint) and output function (2.1) has a (vector) relative degree of $\{3, 3, \dots, 3\}$ and is input-output linearizable via static state feedback [74].

3.6.1.1 Coordinate transformation

A mapping Φ is introduced that transforms the joint coordinates \mathbf{q}_J and the motor coordinates \mathbf{q}_M as well as their first time derivatives into tangential η_1 , transversal ξ_1 and ξ_3 , and rotational ζ_1 coordinates of (3.12), (3.14), (3.15), and (3.18) with respect to a \mathcal{C}^4 path γ . Since the (vector) relative degree is given by $\{3, 3, \dots, 3\}$, the first $3N$ transformed coordinates are chosen as the virtual output $\hat{\mathbf{y}}_p^T = [\eta_1, \xi_1, \xi_3, \zeta_1^T]$ as well as their first and second time derivatives. Following the idea of [74], the velocity of the motor coordinates $\dot{\mathbf{q}}_M$ is chosen as the N remaining coordinates. The mapping Φ then reads as

$$\begin{bmatrix} \hat{\mathbf{y}}_p \\ \dot{\hat{\mathbf{y}}}_p \\ \ddot{\hat{\mathbf{y}}}_p \\ \dot{\mathbf{q}}_M \end{bmatrix} = \begin{bmatrix} g \circ \mathbf{h}_t(\mathbf{q}_J) \\ \delta_1 \circ \mathbf{h}_t(\mathbf{q}_J) \\ \delta_2 \circ \mathbf{h}_t(\mathbf{q}_J) \\ \mathbf{h}_r(\mathbf{q}_J) \\ \hat{\mathbf{J}}_p \dot{\mathbf{q}}_J \\ \dot{\hat{\mathbf{J}}}_p \dot{\mathbf{q}}_J + \hat{\mathbf{J}}_p \mathbf{D}_E^{-1}(\boldsymbol{\tau}_J + \boldsymbol{\tau}_{ext} - \mathbf{n}_E) \\ \dot{\mathbf{q}}_M \end{bmatrix} = \Phi(\mathbf{q}_J, \dot{\mathbf{q}}_J, \mathbf{q}_M, \dot{\mathbf{q}}_M), \quad (3.55)$$

with the PFC Jacobian $\hat{\mathbf{J}}_p$ of (3.22).

Lemma 3. *The mapping $\Phi : \mathcal{X} \mapsto \mathcal{Z}$ with $\mathcal{X} = \mathcal{Q} \times \mathcal{T}_q \mathcal{Q}$, $\mathcal{Q} = \{[\bar{\mathbf{q}}_J^T, \bar{\mathbf{q}}_M^T]^T \in \mathbb{R}^{2N} : \alpha \circ \mathbf{h}_t(\bar{\mathbf{q}}_J) < 1\}$ and tangential space $\mathcal{T}_q \mathcal{Q}$, is a \mathcal{C}^1 -diffeomorphism, if $\mathbf{J}(\mathbf{q}_J)$ is nonsingular.*

Proof. As in the previous section, the inverse function theorem is employed. Hence, the following conditions have to be satisfied:

- (i.) \mathcal{X} and \mathcal{Z} are open in \mathbb{R}^{4N} ,
- (ii.) $\Phi \in \mathcal{C}^1(\mathcal{X}, \mathcal{Z})$, and
- (iii.) $\nabla \Phi = \left[\partial \Phi / \partial \mathbf{q}_J, \partial \Phi / \partial \dot{\mathbf{q}}_J, \partial \Phi / \partial \mathbf{q}_M, \partial \Phi / \partial \dot{\mathbf{q}}_M \right]$ is nonsingular for all $[\mathbf{q}_J^T, \dot{\mathbf{q}}_J^T, \mathbf{q}_M^T, \dot{\mathbf{q}}_M^T]^T \in \mathcal{X}$.

Since \mathcal{Q} is an open subset of \mathbb{R}^{2N} , \mathcal{X} and \mathcal{Z} are open in \mathbb{R}^{4N} . The output $\mathbf{y} = \mathbf{h}(\mathbf{q}_J)$ is assumed to be sufficiently smooth and $\sigma(\theta) \in \mathcal{C}^4(\mathcal{T}, \mathbb{R}^N)$, hence, $\Phi \in \mathcal{C}^1(\mathcal{X}, \mathcal{Z})$ holds. Since \mathbf{D}_E , \mathbf{K}_s , and \mathbf{K}_d are positive definite matrices, the Jacobian of the mapping Φ

$$\nabla \Phi = \begin{bmatrix} \hat{\mathbf{J}}_p & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ * & \hat{\mathbf{J}}_p & \mathbf{0} & \mathbf{0} \\ * & * & \hat{\mathbf{J}}_p \mathbf{D}_E^{-1} \mathbf{K}_s & \hat{\mathbf{J}}_p \mathbf{D}_E^{-1} \mathbf{K}_d \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} \end{bmatrix} \quad (3.56)$$

is nonsingular, if $\mathbf{J}(\mathbf{q}_J)$ is nonsingular, and $[\mathbf{q}_J^T, \mathbf{q}_M^T]^T \in \mathcal{Q}$, which implies that $\hat{\mathbf{J}}_p$ is nonsingular, see also Lemma 1. \square

3.6.1.2 Feedback linearization

The third time derivative of the virtual output $\hat{\mathbf{y}}_p$ is given by

$$\begin{aligned} \hat{\mathbf{y}}_p^{(3)} = & \ddot{\hat{\mathbf{J}}}_p \dot{\mathbf{q}}_J + 2\dot{\hat{\mathbf{J}}}_p \ddot{\mathbf{q}}_J - \hat{\mathbf{J}}_p \mathbf{D}_E^{-1} \left((\dot{\mathbf{D}}_E + \mathbf{K}_d) \ddot{\mathbf{q}}_J + \dot{\mathbf{n}}_E - \dot{\boldsymbol{\tau}}_{ext} \right. \\ & \left. - \mathbf{K}_s (\dot{\mathbf{q}}_M - \dot{\mathbf{q}}_J) + \mathbf{K}_d \mathbf{D}_M^{-1} (\boldsymbol{\tau}_J + \boldsymbol{\tau}_{f,M}) \right) + \hat{\mathbf{J}}_p \mathbf{D}_E^{-1} \mathbf{K}_d \mathbf{D}_M^{-1} \boldsymbol{\tau}_d . \end{aligned} \quad (3.57)$$

Hence, application of the feedback transformation

$$\boldsymbol{\tau}_d = \mathbf{D}_{p,v}^{-1} (\mathbf{v}_{p,v} - \mathbf{b}_{p,v}(\mathbf{q}_J, \dot{\mathbf{q}}_J, \mathbf{q}_M, \dot{\mathbf{q}}_M)) \quad (3.58a)$$

to the system (2.65), joint forces $\boldsymbol{\tau}_J$ of (2.66) and output function (2.1), with the new control input $\mathbf{v}_{p,v}^T = [\mathbf{v}_{p,t,v}^T, \mathbf{v}_{p,r,v}^T] \in \mathbb{R}^N$, the decoupling matrix

$$\mathbf{D}_{p,v}(\mathbf{q}_J) = \hat{\mathbf{J}}_p \mathbf{D}_E^{-1} \mathbf{K}_d \mathbf{D}_M^{-1} , \quad (3.58b)$$

and the vector

$$\begin{aligned} \mathbf{b}_{p,v}(\mathbf{q}_J, \dot{\mathbf{q}}_J, \mathbf{q}_M, \dot{\mathbf{q}}_M) = & \ddot{\hat{\mathbf{J}}}_p \dot{\mathbf{q}}_J + 2\dot{\hat{\mathbf{J}}}_p \ddot{\mathbf{q}}_J - \hat{\mathbf{J}}_p \mathbf{D}_E^{-1} \left((\dot{\mathbf{D}}_E + \mathbf{K}_d) \ddot{\mathbf{q}}_J + \dot{\mathbf{n}}_E \right. \\ & \left. - \dot{\boldsymbol{\tau}}_{ext} - \mathbf{K}_s(\dot{\mathbf{q}}_M - \dot{\mathbf{q}}_J) + \mathbf{K}_d \mathbf{D}_M^{-1}(\boldsymbol{\tau}_J + \boldsymbol{\tau}_{f,M}) \right) \end{aligned} \quad (3.58c)$$

results in the closed-loop dynamics

$$\hat{\mathbf{y}}_p^{(3)} = \mathbf{v}_{p,v} \quad (3.59a)$$

and

$$\ddot{\mathbf{q}}_M = \mathbf{D}_M^{-1} \left(\mathbf{D}_{p,v}^{-1} (\mathbf{v}_{p,v} - \mathbf{b}_{p,v}(\mathbf{q}_J, \dot{\mathbf{q}}_J, \mathbf{q}_M, \dot{\mathbf{q}}_M)) - \boldsymbol{\tau}_J - \boldsymbol{\tau}_{f,M} \right) . \quad (3.59b)$$

Thereby, equation (3.59b) represents the non-observable internal dynamics of the system (3.59) with the output $\mathbf{y} = [\mathbf{y}_t^T, \mathbf{y}_r^T]^T = \mathbf{h}(\mathbf{q}_J)$ according to (2.1). Note that the time derivative of the external forces $\boldsymbol{\tau}_{ext}$ is required for the feedback transformation (3.58), which can be hard to obtain in real applications due to the sensor noise.

The zero dynamics is defined as the internal dynamics of the input-output linearized system (3.59) for $\hat{\mathbf{y}}_p = \mathbf{0} \forall t \geq t_0$ and has to be stable [78]. The condition $\hat{\mathbf{y}}_p \equiv \mathbf{0}$ implies that $\mathbf{v}_{p,v} \equiv \mathbf{0}$ as well as $\mathbf{q}_{J,v} \equiv \mathbf{q}_{J,0} = \text{const}$. Hence, the zero dynamics of system (3.59), where $\boldsymbol{\tau}_{ext}$ is assumed to be zero, is given by

$$\ddot{\mathbf{q}}_M(\hat{\mathbf{y}}_p \equiv \mathbf{0}) = -\mathbf{K}_d^{-1} \mathbf{K}_s \dot{\mathbf{q}}_M . \quad (3.60)$$

Since \mathbf{K}_d and \mathbf{K}_s are positive definite matrices, the zero dynamics are exponentially stable.

The new inputs $v_{\perp,v}$ and $v_{\hat{n},v}$ of $\mathbf{v}_{p,t,v}^T = [v_{\parallel,v}, v_{\perp,v}, v_{\hat{n},v}]$ can be used to fulfill the control objectives (O1) and (O2) and with $v_{\parallel,v}$ an application specific motion along the path can be achieved. Note that the decoupling matrix $\mathbf{D}_{p,v}(\mathbf{q}_J)$ is nonsingular for $[\mathbf{q}_J^T, \mathbf{q}_M^T]^T \in \mathcal{Q}$.

3.6.1.3 Stabilization of the linearized system

Similar to the PFC for fully actuated systems, the position control law

$$\mathbf{v}_{p,v} = \begin{bmatrix} (\eta_1^p)^{(3)} - a_{\eta,3,v} \ddot{e}_\eta - a_{\eta,2,v} \dot{e}_\eta - a_{\eta,1,v} e_\eta \\ (\xi_1^p)^{(3)} - a_{\xi,3,v} \ddot{e}_{\xi_1} - a_{\xi,2,v} \dot{e}_{\xi_1} - a_{\xi,1,v} e_{\xi_1} \\ (\xi_3^p)^{(3)} - a_{\xi,3,v} \ddot{e}_{\xi_3} - a_{\xi,2,v} \dot{e}_{\xi_3} - a_{\xi,1,v} e_{\xi_3} \\ (\mathbf{y}_r^p)^{(3)} - a_{r,3,v} \ddot{\mathbf{e}}_r - a_{r,2,v} \dot{\mathbf{e}}_r - a_{r,1,v} \mathbf{e}_r \end{bmatrix} \quad (3.61)$$

asymptotically stabilizes the linear system (3.59a) if $p_i(s) = s^3 + a_{i,3,v} s^2 + a_{i,2,v} s + a_{i,1,v}$ constitutes a Hurwitz polynomial for $i \in \{\eta, \xi, r\}$. In (3.61), η_1^p , ξ_1^p , and ξ_3^p denote \mathcal{C}^3 -references for the position on the path and the deviation from the path.

The control errors are given by $e_\eta = \eta_1 - \eta_1^p$, $e_{\xi_1} = \xi_1 - \xi_1^p$, $e_{\xi_3} = \xi_3 - \xi_3^p$, and $\mathbf{e}_r = \mathbf{y}_r - \mathbf{y}_r^p$, where $\mathbf{y}_r^p = \boldsymbol{\sigma}_r(\theta^*)$.

In principle, the visco-elastic PFC approach can also be combined with compliance control, but the references $\ddot{\eta}_1^p$, $\ddot{\xi}_1^p$, $\ddot{\xi}_3^p$, and $\ddot{\mathbf{y}}_r^p$ of the impedance control law (3.42) have to be differentiated with respect to the time to obtain the references $(\eta_1^p)^{(3)}$, $(\xi_1^p)^{(3)}$, $(\xi_3^p)^{(3)}$, and $(\mathbf{y}_r^p)^{(3)}$ for the position control law (3.61). This further implies that the time derivative of the measured external forces or torques $\boldsymbol{\tau}_{ext}$ is required.

3.6.2 Robots with pure elastic joints

If the viscosity inside the joints is negligibly small, the joint forces $\boldsymbol{\tau}_J$ of the dynamic system (2.65) are given by (2.67) (pure elastic joint), the system has a (vector) relative degree of $\{4, 4, \dots, 4\}$, and is full state exact linearizable via static state feedback [7].

3.6.2.1 Coordinate transformation

A coordinate transformation is derived that maps the joint coordinates \mathbf{q}_J and the motor coordinates \mathbf{q}_M as well as their first time derivative onto the virtual output $\hat{\mathbf{y}}_p^T = [\eta_1, \xi_1, \xi_3, \boldsymbol{\zeta}_1^T]$ as well as their first, second, and third time derivatives, with tangential η_1 , transversal ξ_1 and ξ_3 , and rotational $\boldsymbol{\zeta}_1$ coordinates of (3.12), (3.14), (3.15), and (3.18) with respect to a \mathcal{C}^5 path γ . The coordinate transformation Φ reads as

$$\begin{bmatrix} \hat{\mathbf{y}}_p \\ \dot{\hat{\mathbf{y}}}_p \\ \ddot{\hat{\mathbf{y}}}_p \\ \hat{\mathbf{y}}_p^{(3)} \end{bmatrix} = \begin{bmatrix} g \circ \mathbf{h}_t(\mathbf{q}_J) \\ \delta_1 \circ \mathbf{h}_t(\mathbf{q}_J) \\ \delta_2 \circ \mathbf{h}_t(\mathbf{q}_J) \\ \mathbf{h}_r(\mathbf{q}_J) \\ \hat{\mathbf{J}}_p \dot{\mathbf{q}}_J \\ \hat{\mathbf{J}}_p \dot{\mathbf{q}}_J + \hat{\mathbf{J}}_p \mathbf{D}_E^{-1}(\boldsymbol{\tau}_J + \boldsymbol{\tau}_{ext} - \mathbf{n}_E) \\ \mathbf{r} + \hat{\mathbf{J}}_p \mathbf{D}_E^{-1}(\dot{\boldsymbol{\tau}}_J + \dot{\boldsymbol{\tau}}_{ext} - \dot{\mathbf{n}}_E - \dot{\mathbf{D}}_E \ddot{\mathbf{q}}_J) \end{bmatrix} = \Phi(\mathbf{q}_J, \dot{\mathbf{q}}_J, \mathbf{q}_M, \dot{\mathbf{q}}_M), \quad (3.62)$$

with the PFC Jacobian $\hat{\mathbf{J}}_p$ of (3.22) and the vector $\mathbf{r} = \ddot{\hat{\mathbf{J}}}_p(\mathbf{q}_J, \dot{\mathbf{q}}_J, \ddot{\mathbf{q}}_J) \dot{\mathbf{q}}_J + 2\dot{\hat{\mathbf{J}}}_p(\mathbf{q}_J, \dot{\mathbf{q}}_J) \ddot{\mathbf{q}}_J$.

Lemma 4. *The mapping $\Phi : \mathcal{X} \mapsto \mathcal{Z}$ with $\mathcal{X} = \mathcal{Q} \times \mathcal{T}_q \mathcal{Q}$, $\mathcal{Q} = \{[\bar{\mathbf{q}}_J^T, \bar{\mathbf{q}}_M^T]^T \in \mathbb{R}^{2N} : \alpha \circ \mathbf{h}_t(\bar{\mathbf{q}}_J) < 1\}$ and tangential space $\mathcal{T}_q \mathcal{Q}$, is a \mathcal{C}^1 -diffeomorphism, if $\mathbf{J}(\mathbf{q}_J)$ is nonsingular.*

Proof. Based on the inverse function theorem it has to be shown that

- (i.) \mathcal{X} and \mathcal{Z} are open in \mathbb{R}^{4N} ,

(ii.) $\Phi \in \mathcal{C}^1(\mathcal{X}, \mathcal{Z})$, and

(iii.) $\nabla \Phi = \left[\partial \Phi / \partial \mathbf{q}_J, \partial \Phi / \partial \dot{\mathbf{q}}_J, \partial \Phi / \partial \mathbf{q}_M, \partial \Phi / \partial \dot{\mathbf{q}}_M \right]$ is nonsingular for all $\left[\mathbf{q}_J^T, \dot{\mathbf{q}}_J^T, \mathbf{q}_M^T, \dot{\mathbf{q}}_M^T \right]^T \in \mathcal{X}$.

Since \mathcal{Q} is an open subset of \mathbb{R}^{2N} , \mathcal{X} and \mathcal{Z} are open in \mathbb{R}^{4N} . The output $\mathbf{y} = \mathbf{h}(\mathbf{q}_J)$ is assumed to be sufficiently smooth and $\boldsymbol{\sigma}(\theta) \in \mathcal{C}^5(\mathcal{T}, \mathbb{R}^N)$, hence, $\Phi \in \mathcal{C}^1(\mathcal{X}, \mathcal{Z})$ holds. The Jacobian of the mapping Φ

$$\nabla \Phi = \begin{bmatrix} \hat{\mathbf{J}}_p & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ * & \hat{\mathbf{J}}_p & \mathbf{0} & \mathbf{0} \\ * & * & \hat{\mathbf{J}}_p \mathbf{D}_E^{-1} \mathbf{K}_s & \mathbf{0} \\ * & * & * & \hat{\mathbf{J}}_p \mathbf{D}_E^{-1} \mathbf{K}_s \end{bmatrix} \quad (3.63)$$

is nonsingular, if $\mathbf{J}(\mathbf{q}_J)$ is nonsingular, and $[\mathbf{q}_J^T, \mathbf{q}_M^T]^T \in \mathcal{Q}$, which implies that $\hat{\mathbf{J}}_p$ is nonsingular. \square

3.6.2.2 Feedback linearization

From the fourth time derivative of the virtual output $\hat{\mathbf{y}}_p$

$$\begin{aligned} \hat{\mathbf{y}}_p^{(4)} = & \hat{\mathbf{J}}_p^{(3)} \dot{\mathbf{q}}_J + 3\ddot{\hat{\mathbf{J}}}_p \ddot{\mathbf{q}}_J + 3\dot{\hat{\mathbf{J}}}_p \mathbf{q}_J^{(3)} - \hat{\mathbf{J}}_p \mathbf{D}_E^{-1} (2\dot{\mathbf{D}}_E \mathbf{q}_J^{(3)} + \ddot{\mathbf{D}}_E \ddot{\mathbf{q}}_J + \ddot{\mathbf{n}}_E) \\ & - \ddot{\boldsymbol{\tau}}_{ext} + \mathbf{K}_s \ddot{\mathbf{q}}_J + \mathbf{K}_s \mathbf{D}_M^{-1} (\boldsymbol{\tau}_J - \boldsymbol{\tau}_{f,M}) + \hat{\mathbf{J}}_p \mathbf{D}_E^{-1} \mathbf{K}_s \mathbf{D}_M^{-1} \boldsymbol{\tau}_d, \end{aligned} \quad (3.64)$$

the feedback transformation

$$\boldsymbol{\tau}_d = \mathbf{D}_{p,e}^{-1} (\mathbf{v}_{p,e} - \mathbf{b}_{p,e}(\mathbf{q}_J, \dot{\mathbf{q}}_J, \mathbf{q}_M, \dot{\mathbf{q}}_M)) , \quad (3.65a)$$

with the decoupling matrix

$$\mathbf{D}_{p,e}(\mathbf{q}_J) = \hat{\mathbf{J}}_p \mathbf{D}_E^{-1} \mathbf{K}_s \mathbf{D}_M^{-1} , \quad (3.65b)$$

and the vector

$$\begin{aligned} \mathbf{b}_{p,e}(\mathbf{q}_J, \dot{\mathbf{q}}_J, \mathbf{q}_M, \dot{\mathbf{q}}_M) = & \hat{\mathbf{J}}_p^{(3)} \dot{\mathbf{q}}_J + 3\ddot{\hat{\mathbf{J}}}_p \ddot{\mathbf{q}}_J + 3\dot{\hat{\mathbf{J}}}_p \mathbf{q}_J^{(3)} - \mathbf{D}_{p,e} (\boldsymbol{\tau}_J - \boldsymbol{\tau}_{f,M}) \\ & - \hat{\mathbf{J}}_p \mathbf{D}_E^{-1} (2\dot{\mathbf{D}}_E \mathbf{q}_J^{(3)} + \ddot{\mathbf{D}}_E \ddot{\mathbf{q}}_J + \ddot{\mathbf{n}}_E - \ddot{\boldsymbol{\tau}}_{ext} + \mathbf{K}_s \ddot{\mathbf{q}}_J) \end{aligned} \quad (3.65c)$$

can be derived. Note that the second time derivative of the external forces $\boldsymbol{\tau}_{ext}$ appears in the feedback transformation (3.65), which can be unfeasible to obtain in real applications due to the sensor noise. Application of (3.65) to the system (2.65), joint forces $\boldsymbol{\tau}_J$ of (2.67), and output function (2.1), with the new control input $\mathbf{v}_{p,e} \in \mathbb{R}^N$, results in a linear input-output relation from the new input $\mathbf{v}_{p,e}^T = [\mathbf{v}_{p,t,e}^T, \mathbf{v}_{p,r,e}^T]$, with $\mathbf{v}_{p,t,e}^T = [v_{||,e}, v_{\perp,e}, v_{\hat{n},e}]$, to the virtual output $\hat{\mathbf{y}}_p$ in the form of N integrator chains of length four

$$\hat{\mathbf{y}}_p^{(4)} = \mathbf{v}_{p,e} . \quad (3.66)$$

The new inputs $v_{\perp,e}$ and $v_{\hat{n},e}$ can be used to fulfill the control objectives (O1) and (O2) and with $v_{||,e}$ an application specific motion along the path can be achieved.

3.6.2.3 Stabilization of the linearized system

If $p_i(s) = s^4 + a_{i,4,e}s^3 + a_{i,3,e}s^2 + a_{i,2,e}s + a_{i,1,e}$ constitutes a Hurwitz polynomial for $i \in \{\eta, \xi, r\}$, the position control law

$$\mathbf{v}_{p,e} = \begin{bmatrix} (\eta_1^p)^{(4)} - a_{\eta,4,e}e_\eta^{(3)} - a_{\eta,3,e}\ddot{e}_\eta - a_{\eta,2,e}\dot{e}_\eta - a_{\eta,1,e}e_\eta \\ (\xi_1^p)^{(4)} - a_{\xi,4,e}e_{\xi_1}^{(3)} - a_{\xi,3,e}\ddot{e}_{\xi_1} - a_{\xi,2,e}\dot{e}_{\xi_1} - a_{\xi,1,e}e_{\xi_1} \\ (\xi_3^p)^{(4)} - a_{\xi,4,e}e_{\xi_3}^{(3)} - a_{\xi,3,e}\ddot{e}_{\xi_3} - a_{\xi,2,e}\dot{e}_{\xi_3} - a_{\xi,1,e}e_{\xi_3} \\ (\mathbf{y}_r^p)^{(4)} - a_{r,4,e}\mathbf{e}_r^{(3)} - a_{r,3,e}\ddot{\mathbf{e}}_r - a_{r,2,e}\dot{\mathbf{e}}_r - a_{r,1,e}\mathbf{e}_r \end{bmatrix} \quad (3.67)$$

asymptotically stabilizes the linear system (3.66). The references for the position on the path and deviation from the path in (3.67) have to be of class \mathcal{C}^4 .

The pure elastic PFC approach can also be combined with compliance control, but, in the present case, the references $\ddot{\eta}_1^p$, $\ddot{\xi}_1^p$, $\ddot{\xi}_3^p$, and $\ddot{\mathbf{y}}_r^p$ of the impedance control law (3.42) have to be differentiated twice with respect to the time to obtain the references $(\eta_1^p)^{(4)}$, $(\xi_1^p)^{(4)}$, $(\xi_3^p)^{(4)}$, and $(\mathbf{y}_r^p)^{(4)}$ for the position control law (3.67). This entails that the second time derivative of the measured external forces or torques $\boldsymbol{\tau}_{ext}$ is required.

3.6.3 Simplification using singular perturbation theory

The feedback transformation (3.65) for the full state model of a pure elastic joint robot contains the second order time derivative of the mass matrix $\mathbf{D}_E(\mathbf{q}_J)$ and the Coriolis matrix $\mathbf{C}_E(\mathbf{q}_J, \dot{\mathbf{q}}_J)$, contained in $\mathbf{n}_E(\mathbf{q}_J, \dot{\mathbf{q}}_J)$. Hence, the control law is computationally demanding for robots with several degrees of freedom. Therefore, a singular perturbation approach is proposed in the following, which allows to resort to a simpler control law.

3.6.3.1 Singular perturbation approach

Singular perturbation theory, see [76], has been widely used to simplify the controller design for robots with elastic joints, see, e.g., [7] and [85]. Within this approach, the system dynamics are split up into a fast and a slow subsystem. For the problem at hand, the slow subsystem is related to the link dynamics and the fast subsystem to the elastic joint dynamics. In order to apply the singular perturbation theory to the elastic joint robot, the dynamics of the joint torques $\boldsymbol{\tau}_J$ have to be fast in comparison to the dynamics of the joint coordinates \mathbf{q}_J . Hence, the stiffness \mathbf{K}_s needs to be sufficiently large, which is why \mathbf{K}_s is formally replaced by $\mathbf{K}_s = \mathbf{K}_\epsilon/\epsilon^2$, with $0 < \epsilon \ll 1$, see [85]. The slow quasi-steady state model is then given by, see [85],

$$(\mathbf{D}_E(\bar{\mathbf{q}}_J) + \mathbf{D}_M) \ddot{\bar{\mathbf{q}}}_J + \mathbf{n}_E(\bar{\mathbf{q}}_J, \dot{\bar{\mathbf{q}}}_J) = \bar{\boldsymbol{\tau}}_d + \bar{\boldsymbol{\tau}}_{ext}, \quad (3.68)$$

where a bar refers to a quasi-steady state and the generalized friction forces on the motor side $\boldsymbol{\tau}_{f,M}$ are assumed to be zero. This model corresponds to the rigid

body model with neglected coupling effects between the rotor and the link motion. The boundary layer model in the new coordinates $\mathbf{z} = \boldsymbol{\tau}_J - \bar{\boldsymbol{\tau}}_J$ and the fast time $\nu = (t - t_0)/\epsilon$ with (arbitrary) initial time t_0 reads as, see [85],

$$\frac{d^2\mathbf{z}}{d\nu^2} + \mathbf{K}_\epsilon \left(\mathbf{D}_E^{-1}(\mathbf{q}_J) + \mathbf{D}_M^{-1} \right) \mathbf{z} = \mathbf{K}_\epsilon \mathbf{D}_M^{-1} (\boldsymbol{\tau}_d - \bar{\boldsymbol{\tau}}_d) . \quad (3.69)$$

Because \mathbf{q}_J is considered to be constant in the fast time ν , the boundary layer system (3.69) is linear and time invariant.

3.6.3.2 Controller

A composite controller is used in the following. The control input $\boldsymbol{\tau}_d$ is split up into a slow component $\boldsymbol{\tau}_{d,s}$ and a fast component $\boldsymbol{\tau}_{d,f}$ such that the slow component only affects the quasi-steady state model (3.68) and the boundary layer model (3.69) is only affected by the fast component. The overall control input reads as $\boldsymbol{\tau}_d = \boldsymbol{\tau}_{d,s} + \boldsymbol{\tau}_{d,f}$.

The path following controller is designed for the quasi-steady state model (3.68), which corresponds to the rigid body model resulting in the slow control input component, compare (3.29),

$$\boldsymbol{\tau}_{d,s} = \mathbf{n}_E(\bar{\mathbf{q}}_J, \dot{\bar{\mathbf{q}}}_J) - \boldsymbol{\tau}_{ext} + (\mathbf{D}_E(\bar{\mathbf{q}}_J) + \mathbf{D}_M) \hat{\mathbf{J}}_p^{-1} \left(\mathbf{v}_p - \dot{\mathbf{J}}_p \dot{\bar{\mathbf{q}}}_J \right) . \quad (3.70)$$

Hence, the path parametrization $\boldsymbol{\sigma}(\theta)$ only has to be \mathcal{C}^3 and the position and compliance controllers as described in Section 3.5.3 can be used to compute \mathbf{v}_p . The fast control input component

$$\boldsymbol{\tau}_{d,f} = -\epsilon \mathbf{D}_b \dot{\boldsymbol{\tau}}_J , \quad (3.71)$$

with the positive definite damping matrix \mathbf{D}_b , gives rise to exponentially stable closed-loop dynamics of the boundary layer system (3.69), see [85]. Hence, applying the singular perturbation approach to pure elastic joint robots with relatively stiff transmissions tremendously simplifies the path following controller and allows to use the results from the PFC approach for fully actuated manipulators.

3.7 Implementation

The proposed static state feedback controllers are implemented on digital computers. Hence, a time discretization in the form $t_k = kT_s, k = 0, 1, 2, \dots$ is performed, with the sampling time T_s . For the discrete-time implementation of the PFC, the optimization problem (3.6), the integral (3.12), and the differential-algebraic equations (3.2) have to be solved numerically in real time.

3.7.1 Numerical solution of the optimization problem

The optimization problem (3.6) is numerically solved using the Newton method. For the initialization, the global optimum θ_0^* is needed. A sufficient number of evenly spread points on the path γ_t are chosen and the distances to $\mathbf{y}_t(0)$ are calculated. The point with shortest distance is used as starting point for the local minimum search to obtain θ_0^* . Then, the optimization problem (3.6) is iteratively solved in each time step $k = 1, 2, \dots$ for $i = 1, 2, \dots$ according to

$$\theta_{k,i} = \theta_{k,i-1} - \frac{J'(\theta_{k,i-1})}{J''(\theta_{k,i-1})}, \quad (3.72)$$

with initial condition $\theta_{k,0} = \theta_{k-1}^*$ and cost function $J(\theta_{k,i-1}) = \|\mathbf{y}_{t,k} - \boldsymbol{\sigma}_t(\theta_{k,i-1})\|_2^2$ until $|\theta_{k,i} - \theta_{k,i-1}| < \varepsilon$. The optimal solution $\theta_k^* = \theta_{k,i}$ is used to perform the numerical integration of (3.12), i.e.,

$$\eta_{1,k} = \eta_{1,k-1} + (\theta_k^* - \theta_{k-1}^*) \|\boldsymbol{\sigma}'_t(\theta_k^*)\|_2, \quad (3.73)$$

where $\eta_{1,k} = \eta_1(kT_s)$ and $\theta_k^* = \theta^*(kT_s)$.

3.7.2 Discretization of the parallel transport frame

A discrete method to calculate the normal vectors of the parallel transport frame is proposed in [25]. In every time step $k = 1, 2, \dots$, a rotation axis and an angle is determined from the tangential vectors $\mathbf{e}_{||,k}$ and $\mathbf{e}_{||,k-1}$ and the normal vectors are rotated with respect to them. The approach is ill-conditioned for small changes of the tangential vectors. Hence, another approach that directly solves the overdetermined problem (3.2) to find the first normal vector $\mathbf{e}_\perp(\theta)$ is presented. Application of the constant step-size backward Euler method, see, e.g., [82], to (3.2) for $i = \perp$ yields

$$\begin{aligned} \mathbf{e}_{\perp,k} &= \mathbf{e}_{\perp,k-1} + T_s \gamma_{\perp,k} \mathbf{e}_{||,k} \\ 0 &= 1 - \mathbf{e}_{\perp,k}^T \mathbf{e}_{\perp,k} \\ 0 &= \mathbf{e}_{||,k}^T \mathbf{e}_{\perp,k}, \end{aligned} \quad (3.74)$$

where $\mathbf{e}_{\perp,k} = \mathbf{e}_\perp(kT_s)$. The overdetermined equations (3.74) have no solution for $T_s > 0$. A straightforward idea is to search for a least-squares solution of (3.74), see [82]. Therefore, the constrained least-squares problem

$$\begin{aligned} \min_{\mathbf{p}_k \in \mathbb{R}^4} f(\mathbf{p}_k) &= \frac{1}{2} \|\mathbf{e}_{\perp,k-1} - \mathbf{e}_{\perp,k} + T_s \gamma_{\perp,k} \mathbf{e}_{||,k}\|_2^2 \\ \text{s.t. } g_1(\mathbf{p}_k) &= \frac{1}{2} (\mathbf{e}_{\perp,k}^T \mathbf{e}_{\perp,k} - 1) = 0 \\ g_2(\mathbf{p}_k) &= \mathbf{e}_{||,k}^T \mathbf{e}_{\perp,k} = 0, \end{aligned} \quad (3.75)$$

with $\mathbf{p}_k^T = [\mathbf{e}_{\perp,k}^T, \gamma_{\perp,k}]$ and fixed k , is considered. The optimal solution $(\mathbf{p}_k^*)^T = [(\mathbf{e}_{\perp,k}^*)^T, \gamma_{\perp,k}^*]$ of (3.75) is found using the first-order optimality condition of (3.75), i.e.,

$$(\nabla f)(\mathbf{p}_k^*) + \lambda_1^*(\nabla g_1)(\mathbf{p}_k^*) + \lambda_2^*(\nabla g_2)(\mathbf{p}_k^*) = \mathbf{0} \quad (3.76a)$$

$$g_1(\mathbf{p}_k^*) = 0 \quad (3.76b)$$

$$g_2(\mathbf{p}_k^*) = 0, \quad (3.76c)$$

with Lagrange multipliers λ_1^* and λ_2^* and gradients

$$\begin{aligned} (\nabla f)^T &= \frac{\partial f}{\partial \mathbf{p}_k} = \left[-\left(\mathbf{e}_{\perp,k-1}^* - \mathbf{e}_{\perp,k}^* + T_s \gamma_{\perp,k}^* \mathbf{e}_{\parallel,k}\right)^T, \right. \\ &\quad \left. T_s \left(\mathbf{e}_{\perp,k-1}^* - \mathbf{e}_{\perp,k}^* + T_s \gamma_{\perp,k}^* \mathbf{e}_{\parallel,k}\right)^T \mathbf{e}_{\parallel,k} \right] \\ (\nabla g_1)^T &= \frac{\partial g_1}{\partial \mathbf{p}_k} = \left[\left(\mathbf{e}_{\perp,k}^*\right)^T \quad 0 \right] \\ (\nabla g_2)^T &= \frac{\partial g_2}{\partial \mathbf{p}_k} = \left[\mathbf{e}_{\parallel,k}^T \quad 0 \right]. \end{aligned} \quad (3.77)$$

From the last row of (3.76a) and (3.76c), we obtain

$$\gamma_{\perp,k}^* = -\frac{1}{T_s} \mathbf{e}_{\parallel,k}^T \mathbf{e}_{\perp,k-1}^*. \quad (3.78)$$

Multiplying (3.76a) by $\mathbf{e}_{\parallel,k}^T$ and using (3.76b) and (3.76c) yields $\lambda_2^* = 0$ and

$$\mathbf{e}_{\perp,k}^* = \frac{1}{1 + \lambda_1^*} \left(\mathbf{e}_{\perp,k-1}^* - \mathbf{e}_{\parallel,k}^T \mathbf{e}_{\perp,k-1}^* \mathbf{e}_{\parallel,k} \right). \quad (3.79)$$

Inserting (3.79) into (3.76b) results in

$$1 + \lambda_1^* = \pm \sqrt{1 - \left(\mathbf{e}_{\parallel,k}^T \mathbf{e}_{\perp,k-1}^*\right)^2}. \quad (3.80)$$

Thus, (3.75) features the analytic solution

$$\mathbf{e}_{\perp,k}^* = \frac{\mathbf{e}_{\perp,k-1}^* - \mathbf{e}_{\parallel,k}^T \mathbf{e}_{\perp,k-1}^* \mathbf{e}_{\parallel,k}}{\sqrt{1 - \left(\mathbf{e}_{\parallel,k}^T \mathbf{e}_{\perp,k-1}^*\right)^2}}, \quad (3.81)$$

which represents an iteration for the first normal vector $\mathbf{e}_{\perp,k}$. Note that to ensure continuity of $\mathbf{e}_{\perp,k}^*$ the positive solution of (3.80) has to be used.

The iteration (3.81), without derivation, can also be found in [86]. The initial condition for the iteration (3.81), i.e., $\mathbf{e}_{\perp}(\theta_0) = \mathbf{e}_{\perp,1}$, must comply with the algebraic equations of (3.74). In the following, it is proved that the iteration

(3.81) converges to the solution of (3.2) for $T_s \rightarrow 0$. The difference equation (3.81) reads as

$$\mathbf{e}_{\perp,k} \sqrt{1 - (\mathbf{e}_{\perp,k-1}^T \mathbf{e}_{\parallel,k})^2} - \mathbf{e}_{\perp,k-1} = -\mathbf{e}_{\perp,k-1}^T \mathbf{e}_{\parallel,k} \mathbf{e}_{\parallel,k}. \quad (3.82)$$

Substitution of (3.78) on the left hand side in (3.82), dividing by T_s , and taking the limit results in

$$\begin{aligned} & \lim_{T_s \rightarrow 0} \frac{\sqrt{1 - T_s^2 \gamma_{\perp,k}^2} \mathbf{e}_{\perp,k} - \mathbf{e}_{\perp,k-1}}{T_s} \\ &= \dot{\mathbf{e}}_{\perp} + \lim_{T_s \rightarrow 0} \frac{\sqrt{1 - T_s^2 \gamma_{\perp,k}^2} - 1}{T_s} \mathbf{e}_{\perp,k} = \dot{\mathbf{e}}_{\perp} = \mathbf{e}'_{\perp} \dot{\theta}. \end{aligned} \quad (3.83)$$

Utilizing (3.76c) for $k-1$ on the right hand side in (3.82), dividing by T_s , and taking the limit yields

$$-\lim_{T_s \rightarrow 0} \mathbf{e}_{\perp,k-1}^T \frac{\mathbf{e}_{\parallel,k} - \mathbf{e}_{\parallel,k-1}}{T_s} \mathbf{e}_{\parallel,k} = -\mathbf{e}_{\perp}^T \dot{\mathbf{e}}_{\parallel} \mathbf{e}_{\parallel} = -\mathbf{e}_{\perp}^T \mathbf{e}'_{\parallel} \dot{\theta}. \quad (3.84)$$

Thus, in the limit case $T_s \rightarrow 0$, the iteration (3.81) resembles (3.4) and the iteration (3.81) numerically solves the differential-algebraic system (3.2). The second normal vector $\mathbf{e}_{\text{in}}(\theta)$ is calculated using (3.5).

3.7.3 Path parametrization

A path γ may be represented by any regular parametrization $\boldsymbol{\sigma}(\theta) : \mathcal{T} \mapsto \mathbb{R}^n$ of class \mathcal{C}^k . Arbitrary paths can be approximated using splines, where the \mathcal{C}^k curve is defined piecewise by polynomials. In the experiments with the DELTA robot presented in Section 3.8.1 and 3.8.2, quartic splines are used, which are of class \mathcal{C}^3 . In particular, a parametrization $\boldsymbol{\sigma}(\theta) : \mathcal{T} \mapsto \mathbb{R}^n$ of the form

$$\boldsymbol{\sigma}(\theta) = \begin{cases} \boldsymbol{\sigma}_0(\theta), & \theta_0 \leq \theta < \theta_1 \\ \boldsymbol{\sigma}_1(\theta), & \theta_1 \leq \theta < \theta_2 \\ \vdots \\ \boldsymbol{\sigma}_{l-1}(\theta), & \theta_{l-1} \leq \theta \leq \theta_l \end{cases} \quad (3.85a)$$

consisting of l path segments with

$$\boldsymbol{\sigma}_s(\theta) = \boldsymbol{\sigma}|_{[\theta_s, \theta_{s+1})}(\theta) = \sum_{j=0}^4 \begin{bmatrix} a_{j,s}(\theta - \theta_s)^j \\ b_{j,s}(\theta - \theta_s)^j \\ c_{j,s}(\theta - \theta_s)^j \end{bmatrix} \quad (3.85b)$$

and suitable coefficients $a_{j,s}, b_{j,s}, c_{j,s}$, $j = 0, \dots, 4$ and $s = 0, \dots, l-1$ is considered. Note that also non-uniform rational basis splines (NURBS), see [87], can be used to parametrize a path γ .

3.8 Applications

This section illustrates the characteristics and demonstrates the applicability of the control concepts presented in this chapter. PFC for fully actuated manipulators is implemented on the real-time system *DS1006* from dSPACE with a sampling time of $T_s = 1$ ms to control a DELTA robot of the type FESTO EXPT-45, see Fig. 3.4. Experimental results with position control are shown in Section 3.8.1 and with compliance control in Section 3.8.2.

The performance of the PFC for elastic joint robots is evaluated by simulation studies using MATLAB[®]/Simulink[®] R2016b on a 64 bit Windows 7 computer. In Section 3.8.3, PFC of a serial robot with three pure elastic rotational joints is shown. A simulation example of the simplified PFC for elastic joint robots using singular perturbation theory is given in Section 3.8.4, where a six axis industrial robot with pure elastic joints is stabilized on a spline path. Additionally,

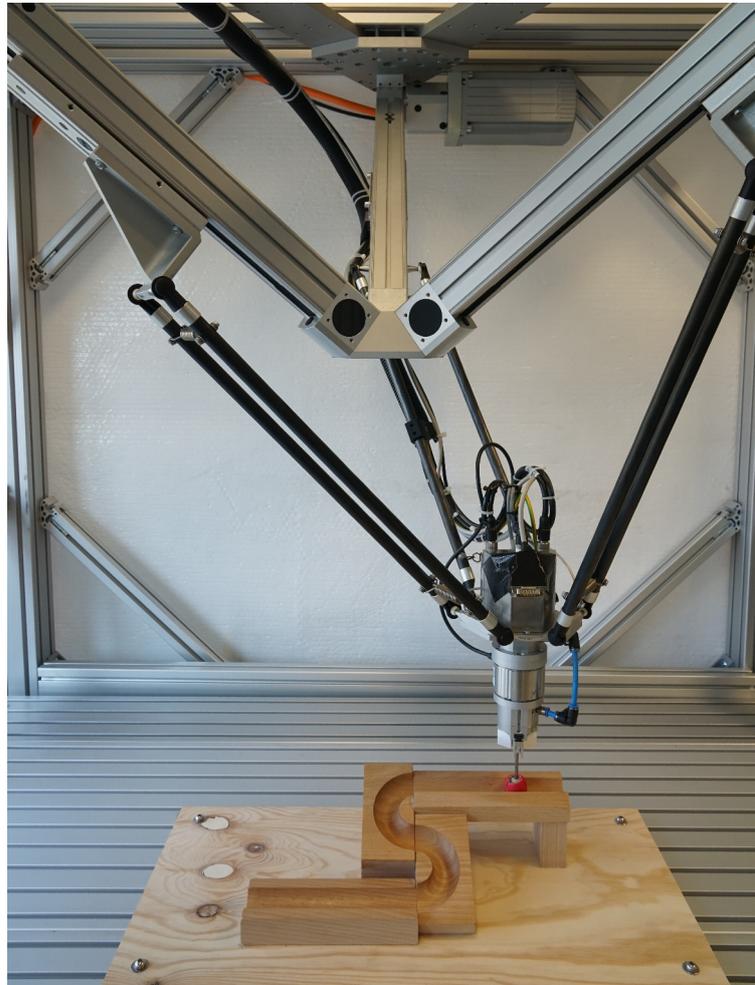


Figure 3.4: DELTA Robot FESTO EXPT-45. [70] © 2017 IEEE

a classical trajectory tracking controller (TTC) is simulated to demonstrate the advantages of PFC compared to TTC.

The application of PFC approach for fully actuated manipulators and the kinematic PFC approach to generate virtual fixtures for physical human-robot interaction is presented in Chapter 5.

3.8.1 Path following control of a Delta robot

In this experiment, the PFC law (3.29) with the Jacobian (2.33) and the position controller (3.33), where integral parts are added, is implemented on the real-time system *DS1006* and applied to the DELTA robot *FESTO EXPT-45*. Since a DELTA robot features three translational and no rotational degrees of freedom, the rotational subsystem ζ is omitted. The mathematical model of the DELTA robot is presented in Chapter 2. A teach-in procedure followed by a quartic spline interpolation was performed, using (3.85) to obtain the path γ_t , which is depicted in Fig. 3.5.

The position controller with integral parts is given by, cf. (3.33)

$$\mathbf{v}_p = \begin{bmatrix} \ddot{\eta}_1^p - a_{\eta,2}\dot{e}_\eta^p - a_{\eta,1}e_\eta^p - a_{\eta,0}\int_0^t e_\eta^p d\tau \\ \ddot{\xi}_1^p - a_{\xi,2}\dot{e}_{\xi_1}^p - a_{\xi,1}e_{\xi_1}^p - a_{\xi,0}\int_0^t e_{\xi_1}^p d\tau \\ \ddot{\xi}_3^p - a_{\xi,2}\dot{e}_{\xi_3}^p - a_{\xi,1}e_{\xi_3}^p - a_{\xi,0}\int_0^t e_{\xi_3}^p d\tau \end{bmatrix} \quad (3.86)$$

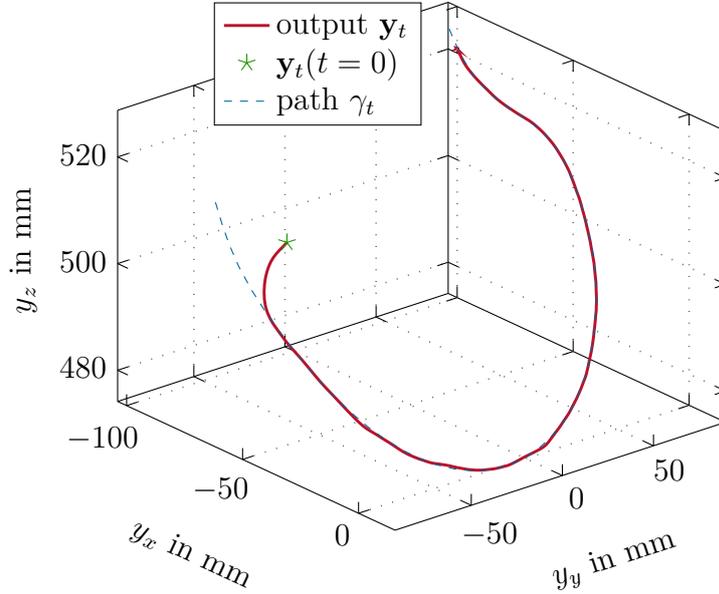


Figure 3.5: Path γ_t and measured output \mathbf{y}_t of the PFC experiment on a DELTA robot. [70] © 2017 IEEE

and the control parameters are shown in Table 3.1. A reference η_1^p for the tangential coordinate η_1 is generated, which smoothly connects the starting point $\eta_{1,0}^p = 0.022$ m and the end point $\eta_{1,T}^p = 0.31$ m with a maximum velocity of $\eta_{2,\max} = 0.1$ m/s. The references in transversal direction, ξ_1^p and ξ_3^p , and their derivatives are set to zero. The end-effector was initially placed next to the path. Fig. 3.5 depicts the measured output \mathbf{y}_t which obviously converges to the path γ_t . The tangential coordinate η_1 and the reference η_1^p is shown in Fig. 3.6(a). The measurements clearly show that control objective (O3) is fulfilled. The transversal states ξ_1 and ξ_3 , depicted in Fig. 3.6(b), quickly converge to zero and stay approximately at zero, hence, the control objectives (O1) and (O2) are also satisfied. The deviation of the transversal states from zero is mainly caused by the sticking friction in the linear drives. This gets clear by comparing the transversal states ξ_1 and ξ_3 with the generalized velocities $\dot{\mathbf{q}}$ of Fig. 3.6(e) and the linear drive forces $\boldsymbol{\tau}_d$ of Fig. 3.6(f). Fig. 3.6(c) shows in addition the virtual control input $\mathbf{v}_p^T = [v_{||}, v_{\perp}, v_{\theta}]$.

3.8.2 Compliant path following control of a Delta Robot

To illustrate the combined path following and compliance control strategy, an experiment is performed, in which the end-effector of the DELTA robot is operated in a notch. This is a typical task that may occur in automatic glue dispersion, where the environment is not exactly known. Fig. 3.4 shows the FESTO EXPT-45, a ball with a soft shell, which is fixed to the robot via a gripper, and a ball notch. A teach-in procedure is performed and the path γ is recorded and interpolated using quartic splines, as described in Section 3.7.3. The task of this experiment is to move the end-effector with the ball along a ball notch from $\eta_{1,0}^d = 0.02$ m to $\eta_{1,T}^d = 0.49$ m with a maximum velocity of $\eta_{2,\max}^d = 0.5$ m/s. If the recorded path exactly conforms with the real path this task is not a problem at all. However, if the workpiece is displaced with respect to the recorded path, the ball on the end-effector gets in contact with the environment. Pure position control would either damage the ball notch of the workpiece, break the end-effector or harm the ball joints of the DELTA robot.

In the considered experiment, the workpiece is displaced -2.5 mm in x -direction and 5.5 mm in y -direction. The feedback transformation (3.29), the position control law (3.86), and the impedance control law (3.42), with the control parameters from Tables 3.1 and 3.2, are executed in real-time on the dSPACE system with

Table 3.1: Control parameters.

Symbol	Value	Unit	Symbol	Value	Unit
$a_{\eta,0}$	42875	1/s ³	$a_{\xi,0}$	125000	1/s ³
$a_{\eta,1}$	3675	1/s ²	$a_{\xi,1}$	7500	1/s ²
$a_{\eta,2}$	105	1/s	$a_{\xi,2}$	150	1/s

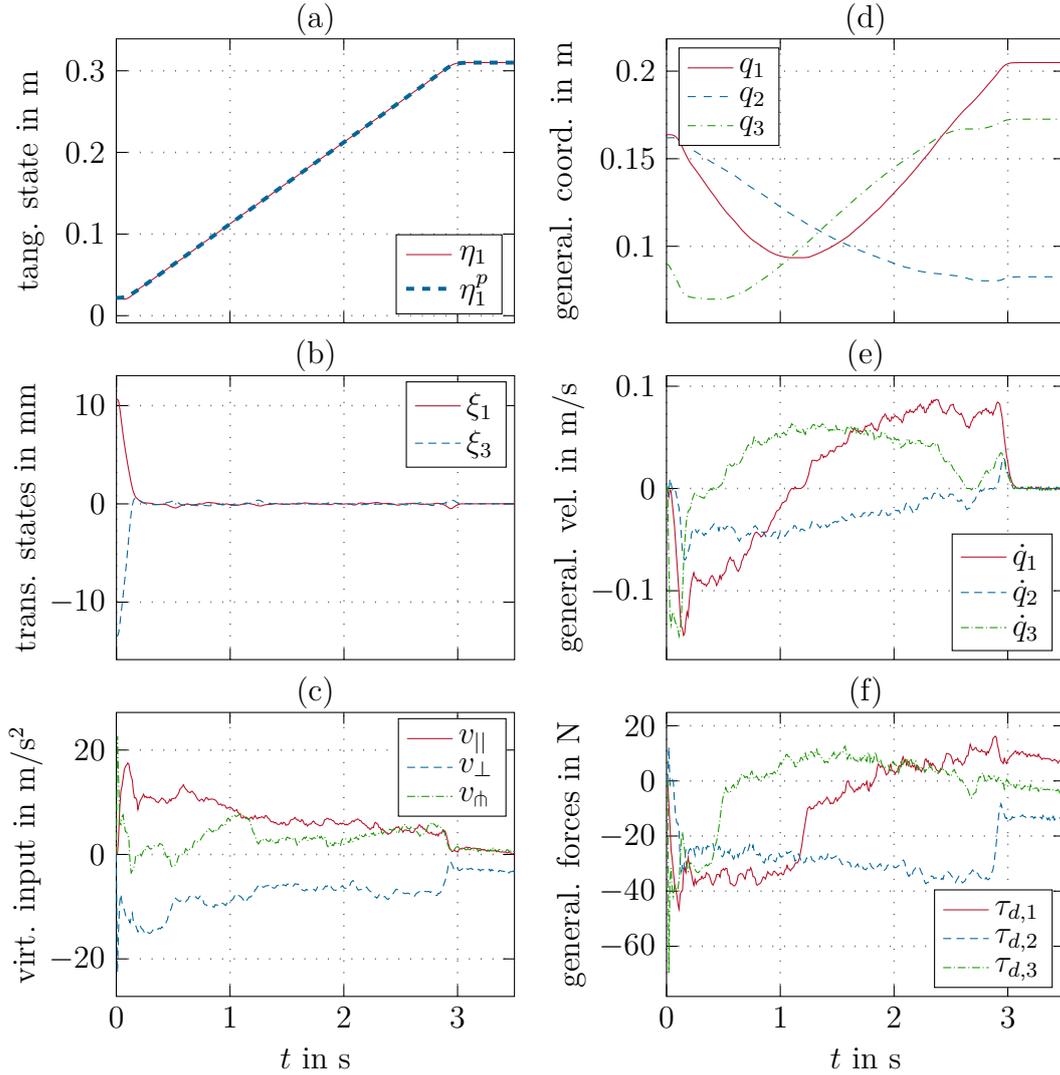


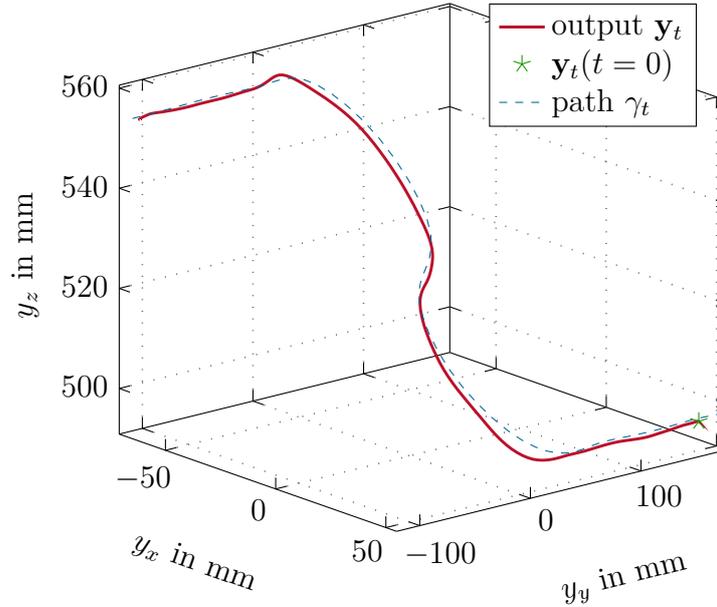
Figure 3.6: Experimental results of the PFC strategy applied to the DELTA robot. [70] © 2017 IEEE

a sampling time of $T_s = 1$ ms. The damping ratio of the virtual mass-spring-damper systems is defined as $\zeta^d = 5$, which results in the damping constants $d_i^d = 2\zeta^d \sqrt{m_i^d k_i^d}$, for $i \in \{||, \perp\}$. This relatively high damping is required to ensure contact stability, see, e.g., [31, 88] for more information. To measure the external force τ_{ext} , the six-axis force sensor $K6 - D40$ from ME-MESSSYSTEME is used.

Fig. 3.7 shows that the output \mathbf{y}_t deviates from the original path γ_t due to the displacement of the workpiece with the ball notch. However, the proposed control strategy is still able to move the ball with maximum velocity inside the notch. As shown in Fig. 3.8(a), the reference motion η_1^d along the path can be tracked very well. The position errors in the transversal states are smaller than

Table 3.2: Compliance control parameters.

Symbol	Value	Unit	Symbol	Value	Unit
m_{\parallel}^d	0.3	kg	m_{\perp}^d	0.3	kg
d_{\parallel}^d	300	Ns/m	d_{\perp}^d	122.47	Ns/m
k_{\parallel}^d	3	kN/m	k_{\perp}^d	0.5	kN/m

Figure 3.7: Path γ_t and measured output \mathbf{y}_t with the combined PFC and compliance control strategy. [70] © 2017 IEEE

1 mm, cf. Fig. 3.8(b). Note that the characteristics of ξ_1^p and ξ_3^p correspond to the displacement of the notch. The forces acting on the ball during the movement are depicted in Fig. 3.8(c). The force in tangential direction τ_{\parallel} is mainly caused by friction between the ball and the notch. In addition, the control inputs $\boldsymbol{\tau}_d$, i.e. the forces of the linear drives, are shown in Fig. 3.8(d).

3.8.3 Path following control of a 3R elastic joint robot

The PFC is applied to the industrial robot COMAU Racer 1.4, where it is assumed that only the first three joints are actuated and the last three joints are fixed. Only the position of the end-effector is considered in the simulations and the orientation is ignored. Hence, the 3R robot features three translational and no rotational degrees of freedom. In contrast to the real robot, the transmissions of the joints are assumed to be purely elastic with a relatively low stiffness of $\mathbf{K}_s = 10^3 \cdot \text{diag}([35, 35, 19])\text{Nm/rad}$ and the friction $\boldsymbol{\tau}_f$ is set to zero. Hence, the simplified equations of motion (2.65) with the joint torques (2.67) serves as basis

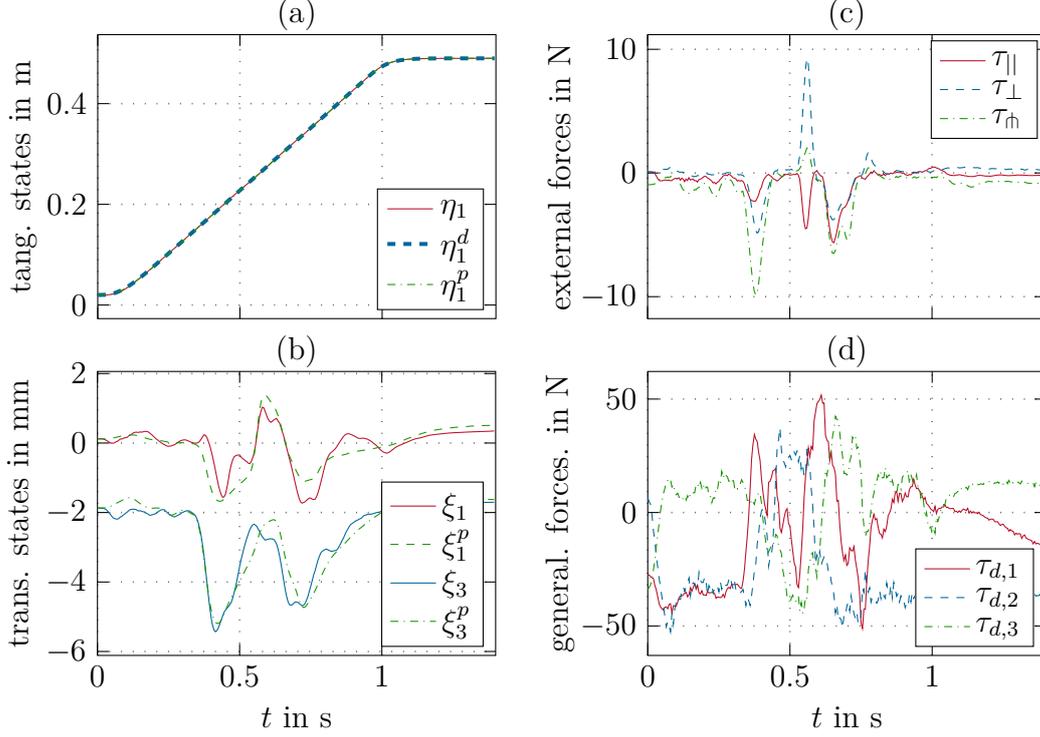


Figure 3.8: Experimental results of the combined PFC and compliance control strategy. [70] © 2017 IEEE

for the MATLAB[®]/Simulink[®] simulation model. Henceforth, the PFC for elastic joint robots presented in Section 3.6.2, the singular perturbation PFC approach of Section 3.6.3, and the PFC for fully actuated manipulators of Section 3.5 are compared for this elastic joint robot. For the latter PFC approach, the motor coordinates \mathbf{q}_M have to be used instead of the link coordinates \mathbf{q}_J because otherwise the closed-loop system would become unstable, see [7].

The smooth path γ_t is defined as a vertically bent lemniscate

$$\boldsymbol{\sigma}_t(\theta) = \begin{bmatrix} 0.25 \cos(\theta) + 0.7 \\ 0.25 \sin(\theta) \cos(\theta) + 0.2 \\ 0.1 \sin^2(\theta) + 1.103 \end{bmatrix} \quad (3.87)$$

and is depicted in Fig 3.9. All numbers are given in meters. The reference η_1^p for the tangential coordinate η_1 smoothly connects the starting point $\eta_1^p(t_0) = 0.96\text{m}$, the point $\eta_1^p(t_1) = 0.31\text{m}$ and the end point $\eta_1^p(t_2) = 1.99\text{m}$ with a maximum velocity of $|\dot{\eta}_{1,max}| = 1.5\text{m/s}$.

First, the results with the PFC law (3.65) and (3.67) for the elastic joint robot of Section 3.6.2, with the control parameters of Tab. 3.3, are presented. The simulated output \mathbf{y}_t is depicted in Fig 3.9 and the corresponding trajectories in Fig. 3.10. Fig. 3.10(a) shows that the desired motion on the path η_1^p is tracked

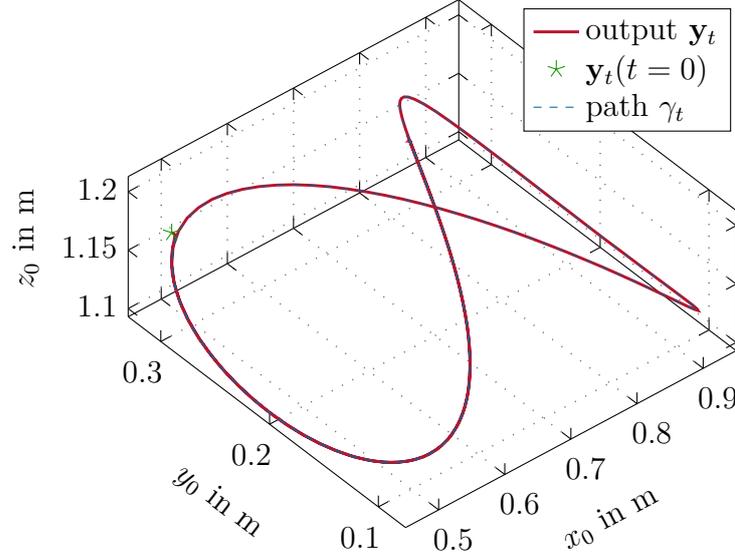


Figure 3.9: Path γ_t and simulated output \mathbf{y}_t with PFC approach for the 3R elastic joint robot.

Table 3.3: Control parameters of the PFC approach for the 3R elastic joint robot.

Symbol	Value	Unit	Symbol	Value	Unit
$a_{\eta,1,e}$	6250000	1/s ⁴	$a_{\xi,1,e}$	6250000	1/s ⁴
$a_{\eta,2,e}$	500000	1/s ³	$a_{\xi,2,e}$	500000	1/s ³
$a_{\eta,3,e}$	15000	1/s ²	$a_{\xi,3,e}$	15000	1/s ²
$a_{\eta,4,e}$	200	1/s	$a_{\xi,4,e}$	200	1/s

very well and Fig. 3.10(b) illustrates that the output \mathbf{y}_t converges to and then remains on the path γ_t . In Fig. 3.10(c), it can be seen that the motor velocities $\dot{q}_{M,i}$ slightly differ from the joint velocities $\dot{q}_{J,i}$, in particular for high drive torques $\tau_{d,i}$ due to the elastic transmissions, cf. Fig. 3.10(d).

Second, the results of the singular perturbation PFC approach according to (3.70) with (3.33) and (3.71) are presented. Thereby, the control parameters of Tab. 3.4 are used, where the damping matrix for the fast control input (3.71) reads as $\mathbf{D}_b = \text{diag}([d_{b,1}, d_{b,2}, d_{b,3}])$. Fig. 3.11 shows that the desired motion on the path η_1^d is tracked very well, but large deviations from the path of up to 40mm occur. A comparison of Fig. 3.11 with Fig. 3.10 reveals that the oscillations of the link coordinates $q_{J,i}$ and the motor torques $\tau_{d,i}$ are larger with the singular perturbation PFC approach. Hence, the stiffness \mathbf{K}_s in this simulation example is too small to properly apply the singular perturbation theory. Note that the singular perturbation PFC approach shows good results in the simulation example of Section 3.8.4, where the stiffness \mathbf{K}_s is ten times larger than in this simulation example.

Third, the results with the PFC approach for fully actuated manipulators

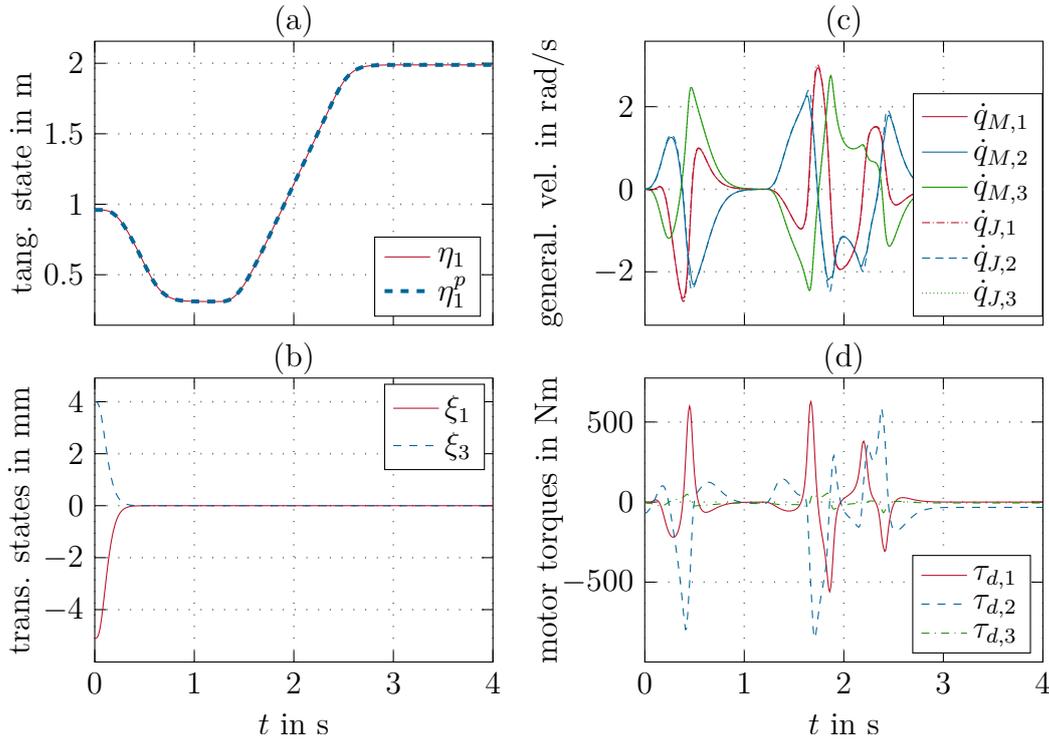


Figure 3.10: Simulated performance of the PFC approach for a 3R elastic joint robot.

of Section 3.5 with the control parameters of Tab. 3.4 are presented. Again, the desired motion on the path η_1^p is tracked very well, which can be seen in Fig. 3.12(a). The deviations from the path are below 20mm and, therefore, much smaller than with the singular perturbation PFC approach, cf. Fig. 3.12(b). A static error from the path remains because only the motor coordinates \mathbf{q}_M are used for the control law. The joint coordinates \mathbf{q}_J , which are relevant for the actual end-effector pose, differ due to gravity. The drive torques $\tau_{d,i}$ are also larger compared to the PFC approach for the elastic joint robot.

Table 3.4: Control parameters of the singular perturbation PFC and the PFC for fully actuated manipulators for the 3R elastic joint robot.

Symbol	Value	Unit	Symbol	Value	Unit
$a_{\eta,1}$	900	1/s ²	$a_{\xi,1}$	900	1/s ²
$a_{\eta,2}$	60	1/s	$a_{\xi,2}$	60	1/s
ϵ	$4 \cdot 10^{-5}$	1	$d_{b,1}$	1100	1/s
$d_{b,2}$	1100	1/s	$d_{b,3}$	640	1/s

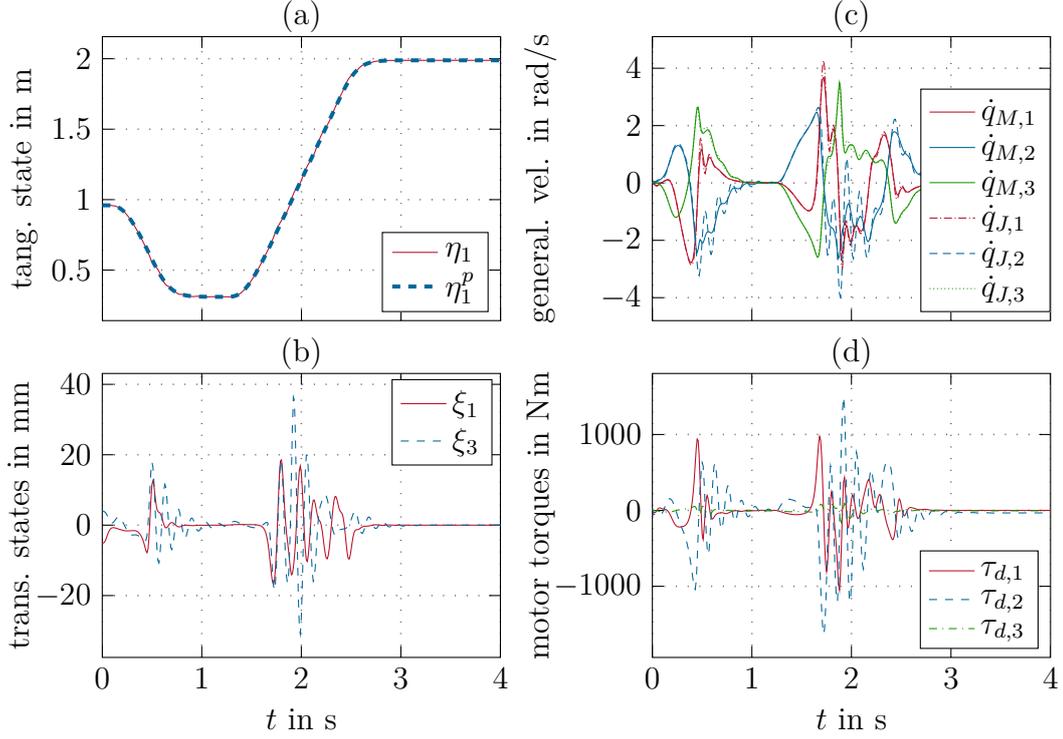


Figure 3.11: Simulated performance of the singular perturbation PFC for a 3R elastic joint robot.

3.8.4 Comparison between path following and trajectory tracking control

In this section, the PFC is compared with a classical trajectory tracking controller (TTC) to point out the differences between these two concepts. The industrial robot COMAU Racer 1.4 is chosen for the simulation model, but in contrast to the real robot the joints are assumed to be pure elastic with the stiffness $\mathbf{K}_s = 10^4 \cdot \text{diag}([35, 35, 19, 3, 1, 0.5])\text{Nms/rad}$ and the friction $\boldsymbol{\tau}_f$ is set to zero.

The singular perturbation PFC law (3.70) and (3.71) together with the position and orientation control laws (3.33) and (3.38) are used. The control parameters are listed in Table 3.5 and $\epsilon \mathbf{D}_b = 10^{-3} \cdot \text{diag}([22, 22, 8.8, 2.4, 4.4, 2.2])\text{Nms/rad}$. The path γ and, therefore, the parametrization $\boldsymbol{\sigma}^T(\theta) = [\boldsymbol{\sigma}_t^T(\theta), \boldsymbol{\sigma}_r^T(\theta)]$, is an

Table 3.5: Control parameters for the 6R elastic joint robot.

Symbol	Value	Unit	Symbol	Value	Unit
$a_{\eta,1}$	361	$1/\text{s}^2$	$a_{\eta,2}$	38	$1/\text{s}$
$a_{\xi,1}$	729	$1/\text{s}^2$	$a_{\xi,2}$	54	$1/\text{s}$
$a_{r,1}$	625	$1/\text{s}^2$	$a_{r,2}$	50	$1/\text{s}$

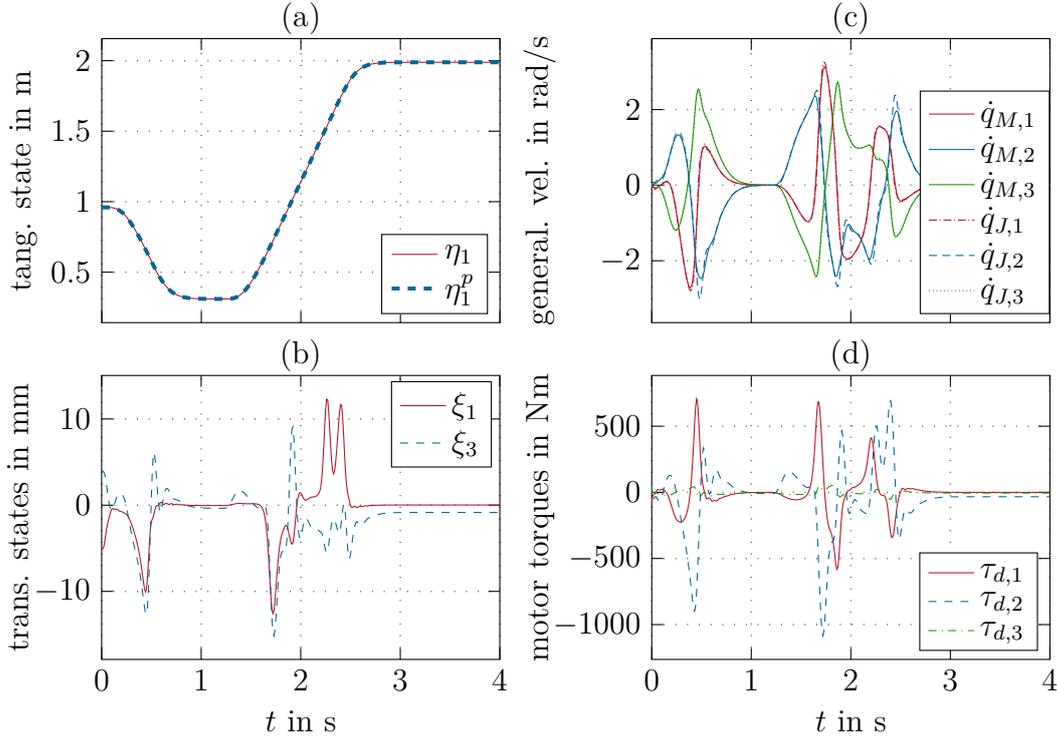


Figure 3.12: Simulated performance of the PFC for fully actuated manipulators for a 3R elastic joint robot.

approximation using quartic splines of the recorded end-effector position and orientation during an experiment on the real robot and its position part γ_t is depicted in Fig. 3.13. The reference η_1^p for the tangential coordinate η_1 smoothly connects the starting point $\eta_{1,0}^p = 0.026\text{m}$ and the end point $\eta_{1,T}^p = 0.954\text{m}$ with a maximum velocity of $\dot{\eta}_1^p = 0.3\text{m/s}$. An external force acts between $t = 1.5\text{s}$ and $t = 2.0\text{s}$ on the end-effector in a way that the system is slowed down in tangential direction with respect to the path to show the benefits of the PFC in comparison to a classical TTC.

Fig. 3.14(a) shows that the desired motion on the path η_1^p can be tracked very well except for the time period where the external force, as described above, is applied. The two transversal states ξ_1 and ξ_3 , which are depicted in Fig. 3.14(b), converge to zero and stay approximately at zero even when the external force is exerted on the robot. In Fig. 3.14(c) and Fig. 3.14(d), one can see that the Euler angles φ and ϑ of the end-effector's orientation ϕ_e quickly converge to the reference Euler angles $\varphi_d = \sigma_{r,1}(\theta^*)$ and $\vartheta_d = \sigma_{r,2}(\theta^*)$, respectively, corresponding to the actual optimal path parameter θ^* . The Euler angle ψ has a similar behavior and is omitted for brevity. In contrast to TTC, cf. Fig. 3.15, the progress of the reference orientation automatically slows down corresponding to the actual speed on the path. The motor torques τ_d are depicted in Fig. 3.14(e) and Fig. 3.14(f).

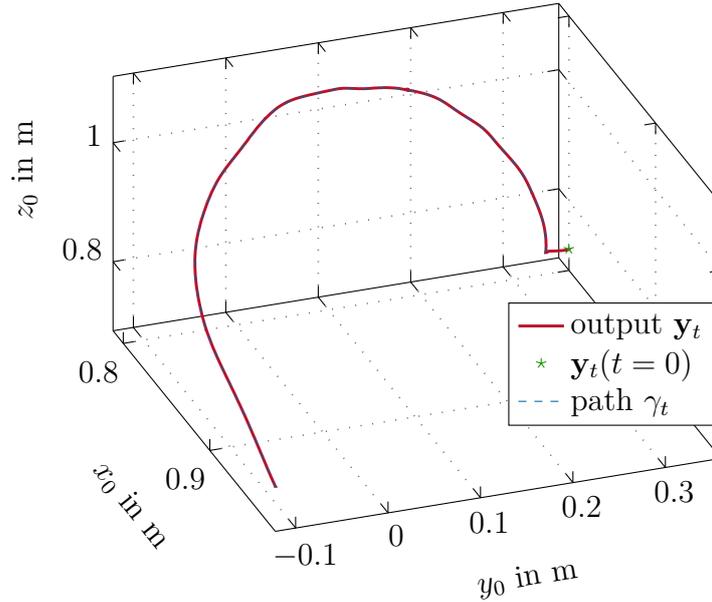


Figure 3.13: Path γ_t and simulated output \mathbf{y}_t for the 6R elastic joint robot.

For a comparison of the PFC with TTC, a computed torque controller, e.g. [7], is implemented for the same elastic joint robot. The reference trajectory is constructed using the path γ of Fig. 3.13 and the same reference motion on the path as for the PFC. The control parameters are chosen in such a way that the TTC has similar dynamics as the PFC. The same force acting between $t = 1.5$ s and $t = 2.0$ s is applied to the end-effector. The output \mathbf{y}_t is transformed into tangential and transversal coordinates η_1 , ξ_1 , and ξ_3 for better comparability. Fig. 3.15(a) shows that the motion along the path with TTC is similar to the PFC. In Fig. 3.15(b), the deviations to the path are depicted. The error is much bigger than for the PFC when the robot is slowed down in tangential direction to the path with an external force, cf. Fig. 3.14(b). Fig. 3.15(c) and Fig. 3.15(d) show the Euler angles φ and ϑ as well as their references φ_d and ϑ_d , respectively. Since the reference trajectory is parametrized in time, the progress of the orientation does not slow down. This is in contrast to the PFC, where the progress of the reference orientation is defined by the progress on the path.

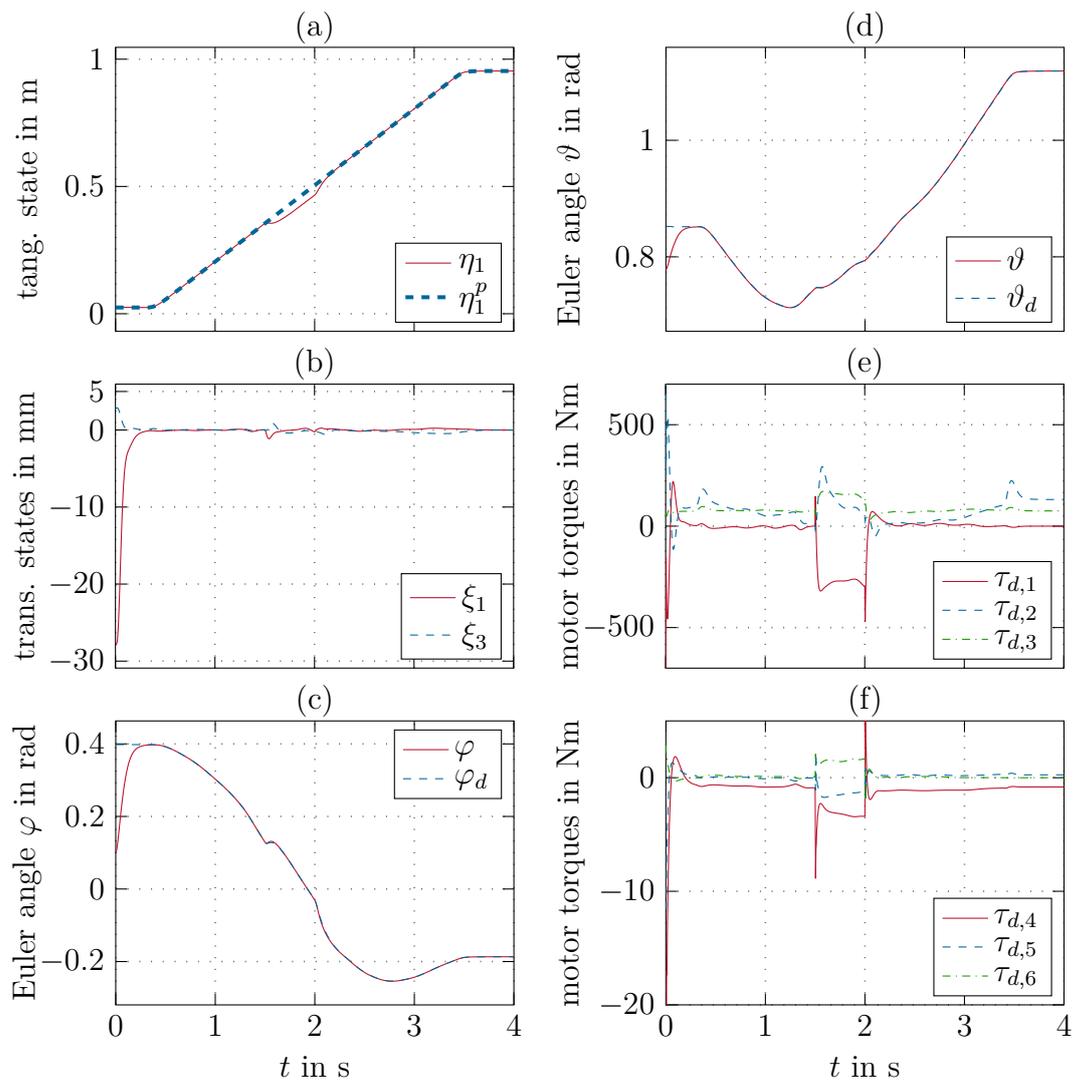


Figure 3.14: Simulated performance of the PFC for a 6R elastic joint robot.

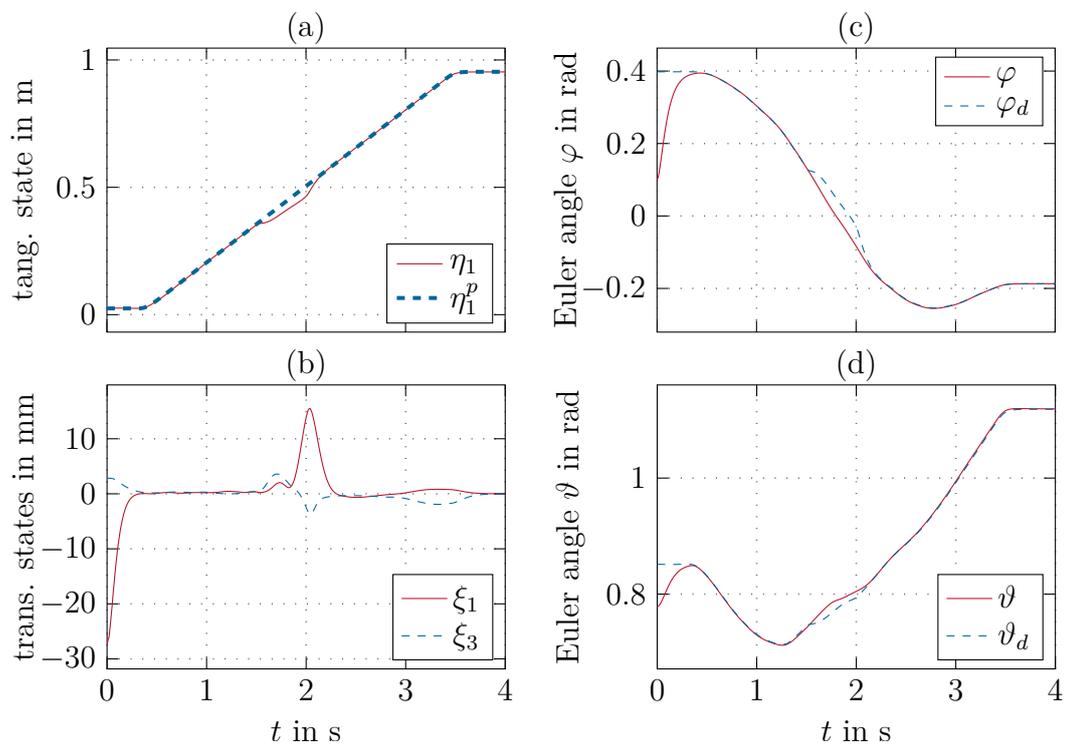


Figure 3.15: Simulated performance of the TTC for a 6R elastic joint robot.

Surface Following Control

This chapter presents surface following control (SFC) for fully actuated manipulators and elastic joint robots in three-dimensional space, which is based on input-output linearization, cf. [78]. The presented SFC approach is closely related to the PFC approach presented in the previous chapter, but here, the system is stabilized on a two-dimensional surface rather than a one-dimensional path. It can handle open, closed, and intersecting surfaces represented by a regular parametrization of class \mathcal{C}^k , which is in contrast to the approaches of [12–14] that require the implicit representation of the surface.

The proposed SFC approach for fully actuated manipulators is applied to a DELTA robot and the COMAU Racer 1.4 for a proof of concept. In the first experiment, the DELTA robot's position is stabilized on a cylinder and a reference motion on the cylinder is tracked. In the second experiment, the combination of SFC and compliance control is applied to the COMAU Racer 1.4 and a human operator moves the robot along a paraboloid of revolution. The SFC approach for visco-elastic joint robots is verified by a simulation study in MATLAB[®]/Simulink[®], where end-effector's position of a 3R elastic joint robot is stabilized on a paraboloid of revolution.

Parts of this chapter are published in similar form in [81, 89].

4.1 Surface assumptions

The surface \mathcal{S} is given by a regular \mathcal{C}^k parametrization $\boldsymbol{\sigma}^T(\boldsymbol{\theta}) = [\boldsymbol{\sigma}_t^T(\boldsymbol{\theta}), \boldsymbol{\sigma}_r^T(\boldsymbol{\theta})] : \mathcal{T}_s \mapsto \mathbb{R}^n$ with the parameter vector $\boldsymbol{\theta}^T = [\theta_1, \theta_2]$, which is an element of the nonempty set $\mathcal{T}_s \subseteq \mathbb{R}^2$. The surface \mathcal{S} can be separated into a position part \mathcal{S}_t defined by $\boldsymbol{\sigma}_t(\boldsymbol{\theta}) \in \mathbb{R}^{n_t}$ and an orientation part \mathcal{S}_r defined by $\boldsymbol{\sigma}_r(\boldsymbol{\theta}) \in \mathbb{R}^{n_r}$. The

parametrization $\boldsymbol{\sigma}(\boldsymbol{\theta})$ of the surface \mathcal{S} is regular, if $\boldsymbol{\sigma}_{t,\theta_1} \times \boldsymbol{\sigma}_{t,\theta_2} \neq \mathbf{0} \forall \boldsymbol{\theta} \in \mathcal{T}_s$, where $\boldsymbol{\sigma}_{t,\theta_i} = \partial \boldsymbol{\sigma}_t / \partial \theta_i$ for $i = 1, 2$. Hence, at each point of the regular surface \mathcal{S}_t there exist two linear independent tangent vectors $\boldsymbol{\sigma}_{t,\theta_i}$, $i = 1, 2$, with $\|\boldsymbol{\sigma}_{t,\theta_i}\| > 0$, and a normal unit vector $\mathbf{e}_\perp(\boldsymbol{\theta})$, which is the normalized cross product of the two tangent vectors.

4.2 Control objectives

The mapping $\|\mathbf{y}_t\|_{\mathcal{S}_t} = \inf_{\bar{\mathbf{y}}_t \in \mathcal{S}_t} \|\mathbf{y}_t - \bar{\mathbf{y}}_t\|_2$ is introduced to formulate the objectives of surface following control. It assigns each position \mathbf{y}_t in the output space a nonnegative real number that is the shortest distance to the surface \mathcal{S}_t . The control objectives of surface following control are defined as follows, see Fig. 4.1:

- (O1) *Asymptotic convergence to \mathcal{S}_t* : The position output \mathbf{y}_t of (2.1) converges asymptotically to the surface \mathcal{S}_t , i.e., $\|\mathbf{y}_t(t)\|_{\mathcal{S}_t} \rightarrow 0$ for $t \rightarrow \infty$.
- (O2) *Invariance property*: If the configuration coordinates and velocities of the dynamic system at time t_0 , i.e., $[\bar{\mathbf{q}}_J^T(t_0), \dot{\bar{\mathbf{q}}}_J^T(t_0)]^T$, are elements of the controlled invariant subset Γ^* of

$$\Gamma = \left\{ [\bar{\mathbf{q}}_J^T, \dot{\bar{\mathbf{q}}}_J^T]^T \in \mathbb{R}^{2n} : \mathbf{h}(\bar{\mathbf{q}}_J) \in \mathcal{S} \right\}, \quad (4.1)$$

then $\|\mathbf{y}_t(t)\|_{\mathcal{S}_t} = 0$, $\forall t \geq t_0$.

- (O3) *Tangential motion*: The motion on the surface \mathcal{S}_t meets application-specific requirements.

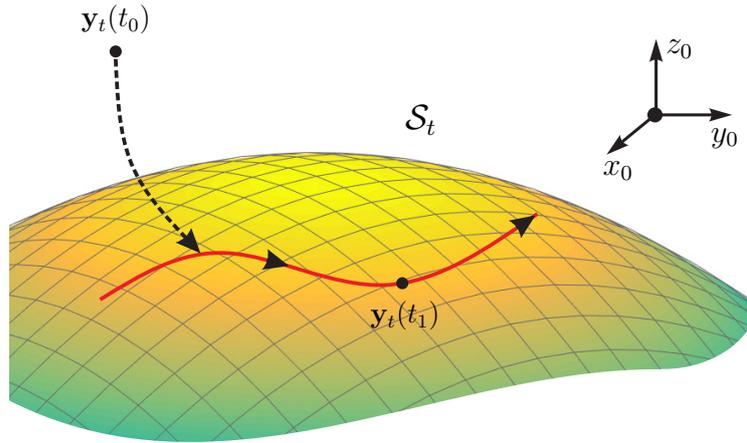


Figure 4.1: Surface following control objectives.

4.3 Moving frame

The tangent unit vectors of the surface parametrization $\boldsymbol{\sigma}_t(\boldsymbol{\theta})$ are given by

$$\mathbf{e}_{\parallel,i}(\boldsymbol{\theta}) = \frac{\boldsymbol{\sigma}_{t,\theta_i}(\boldsymbol{\theta})}{\|\boldsymbol{\sigma}_{t,\theta_i}(\boldsymbol{\theta})\|_2}, \quad i = 1, 2 \quad (4.2)$$

and span the tangent plane to \mathcal{S}_t at $\boldsymbol{\theta}$ as depicted in Fig. 4.2. The normal unit vector $\mathbf{e}_\perp(\boldsymbol{\theta})$ is obtained using the cross product

$$\mathbf{e}_\perp(\boldsymbol{\theta}) = \frac{\boldsymbol{\sigma}_{t,\theta_1}(\boldsymbol{\theta}) \times \boldsymbol{\sigma}_{t,\theta_2}(\boldsymbol{\theta})}{EG - F^2}, \quad (4.3)$$

where

$$EG - F^2 = \|\boldsymbol{\sigma}_{t,\theta_1}(\boldsymbol{\theta}) \times \boldsymbol{\sigma}_{t,\theta_2}(\boldsymbol{\theta})\|_2^2 > 0 \quad (4.4)$$

with the coefficients of the first fundamental form

$$E = \boldsymbol{\sigma}_{t,\theta_1}^\top \boldsymbol{\sigma}_{t,\theta_1} > 0, \quad F = \boldsymbol{\sigma}_{t,\theta_1}^\top \boldsymbol{\sigma}_{t,\theta_2}, \quad G = \boldsymbol{\sigma}_{t,\theta_2}^\top \boldsymbol{\sigma}_{t,\theta_2} > 0. \quad (4.5)$$

The angle between the tangent unit vectors $\mathbf{e}_{\parallel,1}(\boldsymbol{\theta})$ and $\mathbf{e}_{\parallel,2}(\boldsymbol{\theta})$ is given by $\alpha_\parallel(\boldsymbol{\theta}) = \arccos(F/\sqrt{EG})$ and due to the regularity of the surface \mathcal{S} , $\sin(\alpha_\parallel(\boldsymbol{\theta})) \neq 0$ holds. At each point on a regular surface \mathcal{S}_t , the three linear independent vectors $\mathbf{e}_{\parallel,1}$, $\mathbf{e}_{\parallel,2}$, and \mathbf{e}_\perp are defined and form a frame. Thus, any other vector can be represented as a linear combination of (4.2) and (4.3). It can be shown, see, e.g., [21, p.128], that the partial derivatives of (4.3) with respect to θ_1 and θ_2 , i.e., $\mathbf{e}_{\perp,\theta_i} = \partial \mathbf{e}_\perp / \partial \theta_i$, are given by the Weingarten equations

$$\begin{aligned} \mathbf{e}_{\perp,\theta_1}(\boldsymbol{\theta}) &= \alpha_{11}(\boldsymbol{\theta}) \boldsymbol{\sigma}_{t,\theta_1}(\boldsymbol{\theta}) + \alpha_{12}(\boldsymbol{\theta}) \boldsymbol{\sigma}_{t,\theta_2}(\boldsymbol{\theta}), \\ \mathbf{e}_{\perp,\theta_2}(\boldsymbol{\theta}) &= \alpha_{21}(\boldsymbol{\theta}) \boldsymbol{\sigma}_{t,\theta_1}(\boldsymbol{\theta}) + \alpha_{22}(\boldsymbol{\theta}) \boldsymbol{\sigma}_{t,\theta_2}(\boldsymbol{\theta}). \end{aligned} \quad (4.6)$$

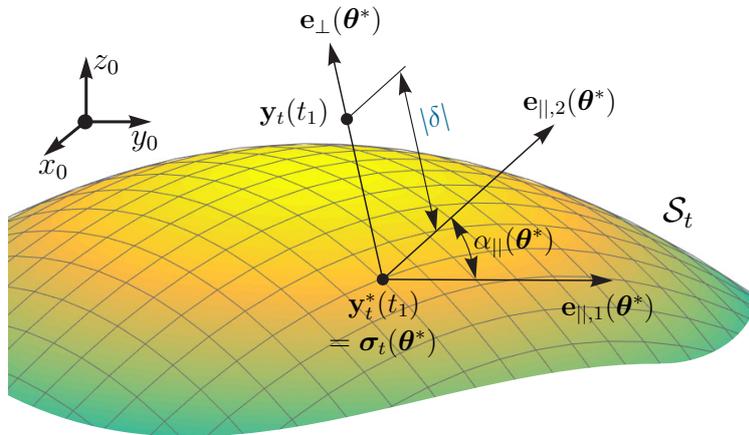


Figure 4.2: Moving frame of the surface following control.

The functions $\alpha_{ij}(\boldsymbol{\theta})$ in (4.6) read as

$$\begin{aligned}\alpha_{11}(\boldsymbol{\theta}) &= \frac{MF - LG}{EG - F^2}, & \alpha_{12}(\boldsymbol{\theta}) &= \frac{LF - ME}{EG - F^2}, \\ \alpha_{21}(\boldsymbol{\theta}) &= \frac{NF - MG}{EG - F^2}, & \alpha_{22}(\boldsymbol{\theta}) &= \frac{MF - NE}{EG - F^2},\end{aligned}\quad (4.7)$$

with the coefficients of the second fundamental form

$$L = \boldsymbol{\sigma}_{t,\theta_1\theta_1}^T \mathbf{e}_\perp, \quad M = \boldsymbol{\sigma}_{t,\theta_1\theta_2}^T \mathbf{e}_\perp, \quad N = \boldsymbol{\sigma}_{t,\theta_2\theta_2}^T \mathbf{e}_\perp, \quad (4.8)$$

where $\boldsymbol{\sigma}_{t,\theta_i\theta_j} = \partial^2 \boldsymbol{\sigma}_t / \partial \theta_i \partial \theta_j$.

4.4 Projection operator and feasible neighborhood

The orthogonal projection $P_{\mathcal{T}_s}(\mathbf{y}_t)$ is used to determine the closest point $\mathbf{y}_t^* = \boldsymbol{\sigma}_t(\boldsymbol{\theta}^*)$ on the surface \mathcal{S}_t to \mathbf{y}_t . Let $\boldsymbol{\sigma}(\boldsymbol{\theta})$ with $\boldsymbol{\theta} \in \mathcal{T}_s \subseteq \mathbb{R}^2$ be a parametrized surface, the orthogonal projection of \mathbf{y}_t onto $\boldsymbol{\sigma}_t(\boldsymbol{\theta})$ reads as

$$\boldsymbol{\theta}^* = P_{\mathcal{T}_s}(\mathbf{y}_t) = \arg \min_{\boldsymbol{\theta} \in \mathcal{T}_s} f(\mathbf{y}_t, \boldsymbol{\theta}) \in \mathcal{T}_s, \quad (4.9)$$

with

$$f(\mathbf{y}_t, \boldsymbol{\theta}) = \frac{1}{2} \|\mathbf{d}(\mathbf{y}_t, \boldsymbol{\theta})\|_2^2, \quad \mathbf{d}(\mathbf{y}_t, \boldsymbol{\theta}) = \mathbf{y}_t - \boldsymbol{\sigma}_t(\boldsymbol{\theta}).$$

If $\boldsymbol{\theta}^*$ is inside the set \mathcal{T}_s , (4.9) features a strict minimum in a feasible neighborhood. The optimum $\boldsymbol{\theta}^*$ fulfills the first-order necessary condition for optimality

$$\left(\frac{\partial f}{\partial \boldsymbol{\theta}} \right) (\mathbf{y}_t, \boldsymbol{\theta}^*) = \begin{bmatrix} -(\boldsymbol{\sigma}_{t,\theta_1}^*)^T \mathbf{d}^* \\ -(\boldsymbol{\sigma}_{t,\theta_2}^*)^T \mathbf{d}^* \end{bmatrix} = \mathbf{0}, \quad (4.10)$$

with $\boldsymbol{\sigma}_{t,\theta_i}^* = (\partial \boldsymbol{\sigma}_t / \partial \theta_i)(\boldsymbol{\theta}^*)$ for $i = 1, 2$ and $\mathbf{d}^* = \mathbf{d}(\mathbf{y}_t, \boldsymbol{\theta}^*)$, in addition to the second-order sufficient condition for optimality

$$\mathbf{H}^* = \mathbf{H}(\mathbf{y}_t, \boldsymbol{\theta}^*) = \left(\frac{\partial^2 f}{\partial \boldsymbol{\theta}^2} \right) (\mathbf{y}_t, \boldsymbol{\theta}^*) > 0, \quad (4.11)$$

with the Hessian

$$\mathbf{H}^* = \begin{bmatrix} E^* & F^* \\ F^* & G^* \end{bmatrix} - \begin{bmatrix} (\boldsymbol{\sigma}_{t,\theta_1\theta_1}^*)^T \mathbf{d}^* & (\boldsymbol{\sigma}_{t,\theta_1\theta_2}^*)^T \mathbf{d}^* \\ (\boldsymbol{\sigma}_{t,\theta_2\theta_1}^*)^T \mathbf{d}^* & (\boldsymbol{\sigma}_{t,\theta_2\theta_2}^*)^T \mathbf{d}^* \end{bmatrix}, \quad (4.12)$$

where $K^* = K(\boldsymbol{\theta}^*)$, $K \in \{E, F, G\}$, see, e.g., [83] for necessary and sufficient optimality conditions for static optimization problems. The second-order sufficient

condition for optimality (4.11) is given in terms of the leading principle minors of \mathbf{H}^* , i.e.,

$$\beta_1(\mathbf{y}_t) = E^* - \left(\boldsymbol{\sigma}_{t,\theta_1}^* \right)^T \mathbf{d}^* > 0 , \quad (4.13a)$$

$$\beta_2(\mathbf{y}_t) = \det(\mathbf{H}^*) > 0 . \quad (4.13b)$$

Hence, the feasible neighborhood of a surface is defined as $\mathcal{Y}_t = \{ \bar{\mathbf{y}}_t \in \mathbb{R}^{n_t} : \beta_i(\bar{\mathbf{y}}_t) > 0, i = 1, 2 \}$. Clearly, if $\mathbf{y}_t \in \mathcal{S}_t$, $\mathbf{d}^* = \mathbf{0}$ and $\mathbf{H}^* > 0$ holds because of (4.4) and (4.5).

Differentiating the first-order condition for optimality (4.10) with respect to the time results in

$$\dot{\boldsymbol{\theta}}^* = (\mathbf{H}^*)^{-1} \begin{bmatrix} \left(\boldsymbol{\sigma}_{t,\theta_1}^* \right)^T \\ \left(\boldsymbol{\sigma}_{t,\theta_2}^* \right)^T \end{bmatrix} \dot{\mathbf{y}}_t . \quad (4.14)$$

4.5 Surface following control for fully actuated manipulators

In this section, a surface following controller for fully actuated manipulators as described in Section 2.4.3, with the dynamics (2.42) and the output function (2.1), is presented. Additionally, a drastically simplified surface following concept using joint velocity controllers is introduced.

4.5.1 Coordinate transformation

A coordinate transformation is defined, which maps the generalized coordinates and velocities, \mathbf{q} and $\dot{\mathbf{q}}$, of the system (2.42) and (2.1) into tangential coordinates $\boldsymbol{\psi}^T = [\boldsymbol{\eta}_1^T, \boldsymbol{\eta}_2^T] \in \mathbb{R}^4$, transversal coordinates $\boldsymbol{\xi}^T = [\xi_1, \xi_2] \in \mathbb{R}^2$, and rotational coordinates $\boldsymbol{\zeta}^T = [\mathbf{y}_r^T, \dot{\mathbf{y}}_r^T] \in \mathbb{R}^{2n_r}$ with respect to a surface \mathcal{S} with regular parametrization $\boldsymbol{\sigma}(\boldsymbol{\theta})$ of class \mathcal{C}^3 . It will be shown that the coordinate transformation is a diffeomorphism in a feasible neighborhood \mathcal{Y}_t of the surface \mathcal{S} .

The rotational subsystem is the same as for the PFC for fully actuated manipulators defined in Section 3.5.1.3 and thus omitted in this section.

4.5.1.1 Tangential subsystem

The tangential coordinates are defined by

$$\boldsymbol{\eta}_1 = \mathbf{g}(\mathbf{y}_t) = \int_{t_0}^t \boldsymbol{\Sigma}^* \dot{\boldsymbol{\theta}}^* d\tau , \quad (4.15)$$

with initial time t_0 and the nonsingular matrix

$$\boldsymbol{\Sigma}^* = \begin{bmatrix} \|\boldsymbol{\sigma}_{t,\theta_1}(\boldsymbol{\theta}^*)\|_2 & \|\boldsymbol{\sigma}_{t,\theta_2}(\boldsymbol{\theta}^*)\|_2 \cos(\alpha_{\parallel}(\boldsymbol{\theta}^*)) \\ 0 & \|\boldsymbol{\sigma}_{t,\theta_2}(\boldsymbol{\theta}^*)\|_2 \sin(\alpha_{\parallel}(\boldsymbol{\theta}^*)) \end{bmatrix} ,$$

which ensures that the two components of $\boldsymbol{\eta}_1$ locally represent a physically interpretable length in orthogonal directions. Differentiation of (4.15) with respect to the time t and using (4.14) yields

$$\begin{aligned}\dot{\boldsymbol{\eta}}_1 &= \boldsymbol{\eta}_2 = \boldsymbol{\Sigma}^* \dot{\boldsymbol{\theta}}^* \\ &= \underbrace{\boldsymbol{\Sigma}^* (\mathbf{H}^*)^{-1} \begin{bmatrix} \boldsymbol{\sigma}_{t,\theta_1}^T(\boldsymbol{\theta}^*) \\ \boldsymbol{\sigma}_{t,\theta_2}^T(\boldsymbol{\theta}^*) \end{bmatrix}}_{\nabla \mathbf{g}} \dot{\mathbf{y}}_t, \end{aligned} \quad (4.16)$$

with the Jacobian $\nabla \mathbf{g} = \partial \mathbf{g} / \partial \mathbf{y}_t$.

4.5.1.2 Transversal subsystem

The transversal coordinate ξ_1 is defined as the projection of $\mathbf{d}(\boldsymbol{\theta}^*) = \mathbf{y}_t - \boldsymbol{\sigma}_t(\boldsymbol{\theta}^*)$ onto the normal vector \mathbf{e}_\perp , i.e.,

$$\xi_1 = \delta(\mathbf{y}_t) = \mathbf{e}_\perp^T(\boldsymbol{\theta}^*) (\mathbf{y}_t - \boldsymbol{\sigma}_t(\boldsymbol{\theta}^*)). \quad (4.17)$$

Differentiating (4.17) with respect to the time yields with (4.6)

$$\dot{\xi}_2 = \dot{\xi}_1 = \mathbf{e}_\perp^T(\boldsymbol{\theta}^*) \dot{\mathbf{y}}_t = (\nabla \delta)^T \mathbf{J}_t \dot{\mathbf{q}}, \quad (4.18)$$

with the gradient $(\nabla \delta)^T = \partial \delta / \partial \mathbf{y}_t$ and $\mathbf{J}_t = \partial \mathbf{h}_t(\mathbf{q}) / \partial \mathbf{q}$.

4.5.1.3 Diffeomorphism

A \mathcal{C}^1 -diffeomorphism is constructed, where the tangent, normal, and rotation maps (4.15)-(4.18) and (3.18), (3.19) are used. Introducing the virtual SFC output $\hat{\mathbf{y}}_s^T = \hat{\mathbf{h}}_s^T(\mathbf{q}) = [\boldsymbol{\eta}_1^T, \xi_1, \boldsymbol{\zeta}_1^T]$ allows us to define the mapping

$$\begin{bmatrix} \hat{\mathbf{y}}_s \\ \hat{\dot{\mathbf{y}}}_s \end{bmatrix} = \begin{bmatrix} \boldsymbol{\eta}_1 \\ \xi_1 \\ \boldsymbol{\zeta}_1 \\ \boldsymbol{\eta}_2 \\ \xi_2 \\ \boldsymbol{\zeta}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{g} \circ \mathbf{h}_t(\mathbf{q}) \\ \delta \circ \mathbf{h}_t(\mathbf{q}) \\ \mathbf{h}_r(\mathbf{q}) \\ \nabla \mathbf{g} \mathbf{J}_t \dot{\mathbf{q}} \\ (\nabla \delta)^T \mathbf{J}_t \dot{\mathbf{q}} \\ \mathbf{J}_r \dot{\mathbf{q}} \end{bmatrix} = \boldsymbol{\Phi}(\mathbf{q}, \dot{\mathbf{q}}). \quad (4.19)$$

Lemma 5. *The mapping $\boldsymbol{\Phi} : \mathcal{X} \mapsto \mathcal{Z}$ with $\mathcal{X} = \mathcal{Q} \times \mathcal{T}_q \mathcal{Q}$, $\mathcal{Q} = \{\bar{\mathbf{q}} \in \mathbb{R}^N : \beta_i \circ \mathbf{h}_t(\bar{\mathbf{q}}) > 0, i = 1, 2\}$, and tangent space $\mathcal{T}_q \mathcal{Q}$ is a \mathcal{C}^1 -diffeomorphism, if $\mathbf{J}(\mathbf{q})$ is nonsingular.*

Proof. According to the inverse function theorem, the following conditions have to be satisfied:

- (i.) \mathcal{X} and \mathcal{Z} are open in \mathbb{R}^{2N} ,

(ii.) $\Phi \in \mathcal{C}^1(\mathcal{X}, \mathcal{Z})$, and

(iii.) $\nabla\Phi = [\partial\Phi/\partial\mathbf{q}, \partial\Phi/\partial\dot{\mathbf{q}}]$ is nonsingular $\forall [\mathbf{q}^T, \dot{\mathbf{q}}^T]^T \in \mathcal{X}$.

Since \mathcal{Q} is an open subset of \mathbb{R}^N , \mathcal{X} and \mathcal{Z} are open in \mathbb{R}^{2N} . Moreover, the output $\mathbf{y} = \mathbf{h}(\mathbf{q})$ is assumed to be sufficiently smooth and $\sigma(\boldsymbol{\theta}) \in \mathcal{C}^3(\mathcal{T}_s, \mathbb{R}^N)$, $\Phi \in \mathcal{C}^1(\mathcal{X}, \mathcal{Z})$ holds. The Jacobian of Φ reads as

$$\nabla\Phi = \begin{bmatrix} \hat{\mathbf{J}}_s(\mathbf{q}) & \mathbf{0} \\ * & \hat{\mathbf{J}}_s(\mathbf{q}) \end{bmatrix}, \quad (4.20)$$

with the SFC Jacobian

$$\hat{\mathbf{J}}_s(\mathbf{q}) = \mathbf{L}_s(\mathbf{q})\mathbf{J}(\mathbf{q}) \quad (4.21)$$

and the matrices

$$\mathbf{L}_s(\mathbf{q}) = \begin{bmatrix} \mathbf{E}_s(\mathbf{q}) & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \quad \text{and} \quad \mathbf{E}_s(\mathbf{q}) = \begin{bmatrix} \nabla\mathbf{g} \\ \mathbf{e}_\perp^T \end{bmatrix}.$$

If $\mathbf{J}(\mathbf{q})$ is nonsingular and $\beta_i(\mathbf{y}_t) > 0$, $i = 1, 2$, then, $\mathbf{E}_s(\mathbf{q})$ and $\mathbf{L}_s(\mathbf{q})$ are nonsingular, and thus, $\hat{\mathbf{J}}_s$ and $\nabla\Phi$ are nonsingular for all $[\mathbf{q}^T, \dot{\mathbf{q}}^T]^T \in \mathcal{X}$. \square

From (4.19) and (4.21) the relation

$$\dot{\hat{\mathbf{y}}}_s = \mathbf{L}_s(\mathbf{q})\dot{\mathbf{y}} = \hat{\mathbf{J}}_s(\mathbf{q})\dot{\mathbf{q}} \quad (4.22)$$

can be derived, which is equal to (3.22) when replacing the subscript s with p .

4.5.2 Feedback linearization

Differentiating the tangential state $\boldsymbol{\eta}_2 = \dot{\boldsymbol{\eta}}_1$ from (4.16) with respect to the time yields

$$\ddot{\boldsymbol{\eta}}_1 = \frac{d}{dt}(\nabla\mathbf{g})\dot{\mathbf{y}}_t + \nabla\mathbf{g}\ddot{\mathbf{y}}_t. \quad (4.23)$$

The time derivative of the transversal state $\xi_2 = \dot{\xi}_1$ from (4.18) takes the form

$$\ddot{\xi}_1 = \left([\mathbf{e}_{\perp, \theta_1}(\boldsymbol{\theta}^*), \mathbf{e}_{\perp, \theta_2}(\boldsymbol{\theta}^*)] \dot{\boldsymbol{\theta}}^* \right)^T \dot{\mathbf{y}}_t + \mathbf{e}_\perp^T(\boldsymbol{\theta}^*)\ddot{\mathbf{y}}_t. \quad (4.24)$$

The equations (4.23), (4.24), and (3.27) can be written in matrix form as, see also (4.22)

$$\ddot{\hat{\mathbf{y}}}_s = \dot{\hat{\mathbf{J}}}_s(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \hat{\mathbf{J}}_s(\mathbf{q})\mathbf{D}^{-1}(\mathbf{q})\left(\boldsymbol{\tau}_d + \boldsymbol{\tau}_{ext} - \mathbf{n}(\mathbf{q}, \dot{\mathbf{q}})\right). \quad (4.25)$$

Note that (3.28) for $\ddot{\hat{\mathbf{y}}}_p$ is equal to (4.25) when $\hat{\mathbf{J}}_p$ is replaced with $\hat{\mathbf{J}}_s$. Hence, the SFC feedback transformation is obtained just by replacing $\hat{\mathbf{J}}_p$ with $\hat{\mathbf{J}}_s$ in the PFC feedback transformation (3.29) and yields

$$\boldsymbol{\tau}_d = \mathbf{n}(\mathbf{q}, \dot{\mathbf{q}}) - \boldsymbol{\tau}_{ext} + \mathbf{D}(\mathbf{q})\hat{\mathbf{J}}_s^{-1}(\mathbf{q})\left(\mathbf{v}_s - \dot{\hat{\mathbf{J}}}_s(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}}\right). \quad (4.26)$$

Application of (4.26) to the system (2.42) and (2.1), with the new control input $\mathbf{v}_s^T = [\mathbf{v}_{s,t}^T, \mathbf{v}_{s,r}^T]$, where $\mathbf{v}_{s,t}^T = [v_{||,1}, v_{||,2}, v_{\perp}]$, results in the linear input-output relation

$$\ddot{\hat{\mathbf{y}}}_s = \mathbf{v}_s. \quad (4.27)$$

Note that the dynamics of the transformed system are linear with respect to a nonlinear plant and a nonlinear surface \mathcal{S} . The virtual input in the direction of the normal vector v_{\perp} can effectively be used to fulfill objective (O1), i.e., to stabilize the transversal (ξ_1, ξ_2) -subsystem and to guarantee asymptotic convergence to the surface \mathcal{S}_t . The controlled invariant subset is given by

$$\Gamma^* = \left\{ \left[\bar{\mathbf{q}}^T, \dot{\bar{\mathbf{q}}}^T \right]^T \in \mathcal{X} : \Phi_{\xi}(\bar{\mathbf{q}}, \dot{\bar{\mathbf{q}}}) = \mathbf{0} \right\}, \quad (4.28)$$

with

$$\Phi_{\xi}(\mathbf{q}, \dot{\mathbf{q}}) = \begin{bmatrix} \delta \circ \mathbf{h}_t(\mathbf{q}) \\ (\nabla \delta)^T \mathbf{J}_t \dot{\mathbf{q}} \end{bmatrix}. \quad (4.29)$$

Thus, objective (O2) is met, because if $[\bar{\mathbf{q}}^T(t_0), \dot{\bar{\mathbf{q}}}^T(t_0)]^T \in \Gamma^*$, then by choosing $v_{\perp} = 0$, $\xi_1 = 0$ and $\|\mathbf{y}_t(t)\|_{\mathcal{S}_t} = 0 \forall t > t_0$. Moreover, the new inputs in tangential direction, $v_{||,1}$ and $v_{||,2}$, allow to control the motion on the surface, thus, objective (O3) can be fulfilled.

4.5.3 Stabilization of the linearized system

The new input \mathbf{v}_s can be computed by any controller that stabilizes the linearized system (4.27). In the following, a position controller and a compliance controller for the SFC are presented.

4.5.3.1 Position Control

The simple position control law, cf. Section 3.5.3,

$$\mathbf{v}_s = \begin{bmatrix} \mathbf{v}_{s,t} \\ \mathbf{v}_{s,r} \end{bmatrix} = \begin{bmatrix} \ddot{\boldsymbol{\eta}}_1^p - a_{\eta,2} \dot{\mathbf{e}}_{\eta}^p - a_{\eta,1} \mathbf{e}_{\eta}^p \\ \ddot{\xi}_1^p - a_{\xi,2} \dot{e}_{\xi}^p - a_{\xi,1} e_{\xi}^p \\ \ddot{\mathbf{y}}_r^p - a_{r,2} \dot{\mathbf{e}}_r^p - a_{r,1} \mathbf{e}_r^p \end{bmatrix}, \quad (4.30)$$

where $\mathbf{e}_{\eta}^p = \boldsymbol{\eta}_1 - \boldsymbol{\eta}_1^p$, $e_{\xi}^p = \xi_1 - \xi_1^p$, $\mathbf{e}_r^p = \mathbf{y}_r - \mathbf{y}_r^p$ yields an exponentially stable error dynamics, if $a_{i,j} > 0$ with $i \in \{\boldsymbol{\eta}, \xi, r\}$ and $j = 1, 2$. The reference position on the surface is denoted by $\boldsymbol{\eta}_1^p$ and the reference for the orthogonal state by ξ_1^p . The reference for the orientation is given by $\mathbf{y}_r^p = \boldsymbol{\sigma}_r(\boldsymbol{\theta}^*)$. The orthogonal state ξ_1 exponentially converges to the reference ξ_1^p and, thus, objective (O1) is fulfilled for $\xi_1^p = 0$. Additionally, if $[\mathbf{q}^T(t_0), \dot{\mathbf{q}}^T(t_0)]^T \in \Gamma^*$, then $v_{\perp} = 0$ holds, the output \mathbf{y}_t stays on the surface \mathcal{S}_t for all $t \geq t_0$, and objective (O2) is fulfilled. The tangential coordinate $\boldsymbol{\eta}_1$ exponentially converges to the reference $\boldsymbol{\eta}_1^p$. Hence, objective (O3) is also satisfied.

In the case of $\dim(\mathbf{y}_r) = n_r = 3$, the same method as presented in Section 3.5.3.1 can be used to deal with the representation singularities of the Euler angles.

4.5.3.2 Compliance control

A similar approach as in Section 3.5.3.2 for the PFC is used here to generate a desired compliance behavior for the SFC. With the SFC, the reference impedance of the motion along and orthogonal to the surface as well as the reference impedance of the orientation can be separately defined as

$$\begin{bmatrix} \boldsymbol{\tau}_{\parallel} \\ \tau_{\perp} \\ \boldsymbol{\tau}_r \end{bmatrix} = \begin{bmatrix} m_{\parallel}^d \ddot{\mathbf{e}}_{\eta}^d + d_{\parallel}^d \dot{\mathbf{e}}_{\eta}^d + k_{\parallel}^d \mathbf{e}_{\eta}^d \\ m_{\perp}^d \ddot{e}_{\xi}^d + d_{\perp}^d \dot{e}_{\xi}^d + k_{\perp}^d e_{\xi}^d \\ m_r^d \ddot{\mathbf{e}}_r^d + d_r^d \dot{\mathbf{e}}_r^d + k_r^d \mathbf{e}_r^d \end{bmatrix}, \quad (4.31)$$

where $\mathbf{e}_{\eta}^d = \boldsymbol{\eta}_1 - \boldsymbol{\eta}_1^d$ and $e_{\xi}^d = \xi_1 - \xi_1^d$ denote the errors between the coordinates $\boldsymbol{\eta}_1$, ξ_1 and the references $\boldsymbol{\eta}_1^d$ and ξ_1^d . Moreover, m_i^d , d_i^d , and k_i^d for $i \in \{\parallel, \perp, r\}$ are positive design parameters for the reference impedance model. The external (projected) generalized forces are given by

$$\begin{bmatrix} \boldsymbol{\tau}_{\parallel} \\ \tau_{\perp} \\ \boldsymbol{\tau}_r \end{bmatrix} = \hat{\mathbf{J}}_s^{-T} \boldsymbol{\tau}_{\text{ext}}. \quad (4.32)$$

The controller (4.30) is employed for the inner position control loop and assuming perfect tracking, the actual tangential and transversal coordinates $\boldsymbol{\eta}_1$ and ξ_1 as well as the orientation \mathbf{y}_r in (4.31) can be replaced by the references $\boldsymbol{\eta}_1^p$, ξ_1^p , and \mathbf{y}_r^p . The impedance control law then follows as

$$\ddot{\boldsymbol{\eta}}_1^p = \ddot{\boldsymbol{\eta}}_1^d + \frac{\boldsymbol{\tau}_{\parallel}}{m_{\parallel}^d} - \frac{d_{\parallel}^d}{m_{\parallel}^d} \dot{\mathbf{e}}_{\eta}^{pd} - \frac{k_{\parallel}^d}{m_{\parallel}^d} \mathbf{e}_{\eta}^{pd}, \quad (4.33a)$$

$$\dot{\boldsymbol{\eta}}_1^p = \int_0^t \ddot{\boldsymbol{\eta}}_1^p d\tau, \quad \boldsymbol{\eta}_1^p = \int_0^t \dot{\boldsymbol{\eta}}_1^p d\tau,$$

$$\ddot{\xi}_1^p = \ddot{\xi}_1^d + \frac{\tau_{\perp}}{m_{\perp}^d} - \frac{d_{\perp}^d}{m_{\perp}^d} \dot{e}_{\xi}^{pd} - \frac{k_{\perp}^d}{m_{\perp}^d} e_{\xi}^{pd}, \quad (4.33b)$$

$$\dot{\xi}_1^p = \int_0^t \ddot{\xi}_1^p d\tau, \quad \xi_1^p = \int_0^t \dot{\xi}_1^p d\tau,$$

with \mathbf{y}_r^p from (3.42d) and the errors $\mathbf{e}_{\eta}^{pd} = \boldsymbol{\eta}_1^p - \boldsymbol{\eta}_1^d$ and $e_{\xi}^{pd} = \xi_1^p - \xi_1^d$.

In the case of $\dim(\mathbf{y}_r) = n_r = 3$, the same method as presented in Section 3.5.3.2 can be used to deal with the representation singularities of the Euler angles.

4.5.4 Simplification using joint velocity controllers

Similar to the kinematic PFC, the feedback law for the SFC can be drastically simplified when subordinate velocity controllers are used that sufficiently satisfy the relation (3.48). Kinematic SFC is presented in the following, where $\boldsymbol{\sigma}(\boldsymbol{\theta})$ only has to be \mathcal{C}^2 .

4.5.4.1 Coordinate transformation

The coordinate transformation $\Phi : \mathbb{R}^N \rightarrow \mathbb{R}^N$ reads as

$$\hat{\mathbf{y}}_s = \begin{bmatrix} \mathbf{g} \circ \mathbf{h}_t(\mathbf{q}) \\ \delta \circ \mathbf{h}_t(\mathbf{q}) \\ \mathbf{h}_r(\mathbf{q}) \end{bmatrix} = \Phi(\mathbf{q}) , \quad (4.34)$$

with \mathbf{g} and δ from (4.15) and (4.17), respectively.

Lemma 6. *The mapping $\Phi : \mathcal{Q} \mapsto \mathcal{Z} \subset \mathbb{R}^N$, with $\mathcal{Q} = \{\bar{\mathbf{q}} \in \mathbb{R}^N : \beta_i \circ \mathbf{h}_t(\bar{\mathbf{q}}) > 0, i = 1, 2\}$ and β_i from (4.13), is a \mathcal{C}^1 -diffeomorphism, if $\mathbf{J}(\mathbf{q})$ is nonsingular.*

Proof. Based on the inverse function theorem it has to be shown that

- (i.) \mathcal{Q} and \mathcal{Z} are open in \mathbb{R}^N ,
- (ii.) $\Phi \in \mathcal{C}^1(\mathcal{Q}, \mathcal{Z})$, and
- (iii.) $\nabla \Phi = \partial \Phi / \partial \mathbf{q} = \hat{\mathbf{J}}_s(\mathbf{q})$ is nonsingular for all $\mathbf{q} \in \mathcal{Q}$.

Since \mathcal{Q} is an open subset of \mathbb{R}^N , \mathcal{Z} is open in \mathbb{R}^N . Since the output $\mathbf{y} = \mathbf{h}(\mathbf{q})$ is assumed to be sufficiently smooth and $\boldsymbol{\sigma}(\boldsymbol{\theta}) \in \mathcal{C}^2(\mathcal{T}_s, \mathbb{R}^N)$, $\Phi \in \mathcal{C}^1(\mathcal{Q}, \mathcal{Z})$ holds. If $\mathbf{J}(\mathbf{q})$ is nonsingular and $\beta_i(\mathbf{y}_t) > 0, i = 1, 2$, then $\hat{\mathbf{J}}_s = \nabla \Phi$ is nonsingular for all $\mathbf{q} \in \mathcal{Q}$. \square

4.5.4.2 Feedback linearization

Because of (4.22), application of the feedback transformation

$$\dot{\mathbf{q}}_{ref} = \hat{\mathbf{J}}_s^{-1} \mathbf{v}_{s,k} \quad (4.35)$$

to the system (3.48) and (2.1) results in the linear input-output relation

$$\dot{\hat{\mathbf{y}}}_s = \mathbf{v}_{s,k} . \quad (4.36)$$

In (4.35) and (4.36) the new control input is denoted by $\mathbf{v}_{s,k}^T = [\mathbf{v}_{s,t,k}^T, \mathbf{v}_{s,r,k}^T]$, where $\mathbf{v}_{s,t,k}^T = [v_{||,1,k}, v_{||,2,k}, v_{\perp,k}]$. Unlike the SFC feedback law for the dynamic model, see (4.26), the computational demanding time-derivative of the Jacobian $\dot{\hat{\mathbf{J}}}_s$ does not appear in the kinematic SFC law (4.35).

4.5.4.3 Stabilization of the linearized system

Similar to Section 3.5.4.3, the position control law is chosen as

$$\mathbf{v}_{s,k} = \begin{bmatrix} \mathbf{v}_{s,t,k} \\ \mathbf{v}_{s,r,k} \end{bmatrix} = \begin{bmatrix} \dot{\boldsymbol{\eta}}_1^p - a_\eta \mathbf{e}_\eta^p \\ \dot{\xi}_1^p - a_\xi \mathbf{e}_\xi^p \\ \dot{\mathbf{y}}_r^p - a_r \mathbf{e}_r^p \end{bmatrix}, \quad (4.37)$$

which yields an exponentially stable error dynamics if $a_i > 0$ with $i \in \{\eta, \xi, r\}$.

A compliant behavior is achieved by using (4.33) and (3.42d) to compute the references $\boldsymbol{\eta}_1^p$, ξ_1^p , and \mathbf{y}_r^p together with the position controller (4.37). The same method as for the kinematic PFC presented in Section 3.5.4.3 can then be used to deal with the representation singularities of the Euler angles in the case of $\dim(\mathbf{y}_r) = n_r = 3$.

4.6 Surface following control for elastic joint robots

This section presents two surface following controllers for the simplified elastic joint manipulator dynamics (2.65) with joint forces $\boldsymbol{\tau}_J$ including viscosity (2.66) and without viscosity (2.67), which are both input-output linearizable via static state feedback. The two SFC approaches are based on the same ideas as the PFC approaches for elastic joint robots presented in Section 3.6.

4.6.1 Robots with visco-elastic joints

The (vector) relative degree of the simplified elastic joint manipulator dynamics (2.65) with joint forces $\boldsymbol{\tau}_J$ from (2.66) and output function (2.1) reads as $\{3, 3, \dots, 3\}$.

4.6.1.1 Coordinate transformation

The mapping Φ transforms the joint coordinates \mathbf{q}_J and the motor coordinates \mathbf{q}_M as well as their first time derivatives into the virtual output $\hat{\mathbf{y}}_s^T = [\boldsymbol{\eta}_1^T, \xi_1, \boldsymbol{\zeta}_1^T]$ and their first and second time derivative and reads as

$$\begin{bmatrix} \hat{\mathbf{y}}_s \\ \dot{\hat{\mathbf{y}}}_s \\ \ddot{\hat{\mathbf{y}}}_s \\ \dot{\mathbf{q}}_M \end{bmatrix} = \begin{bmatrix} \mathbf{g} \circ \mathbf{h}_t(\mathbf{q}_J) \\ \delta \circ \mathbf{h}_t(\mathbf{q}_J) \\ \mathbf{h}_r(\mathbf{q}_J) \\ \hat{\mathbf{J}}_s \dot{\mathbf{q}}_J \\ \hat{\mathbf{J}}_s \dot{\mathbf{q}}_J + \hat{\mathbf{J}}_s \mathbf{D}_E^{-1}(\boldsymbol{\tau}_J + \boldsymbol{\tau}_{ext} - \mathbf{n}_E) \\ \dot{\mathbf{q}}_M \end{bmatrix} = \Phi(\mathbf{q}_J, \dot{\mathbf{q}}_J, \mathbf{q}_M, \dot{\mathbf{q}}_M), \quad (4.38)$$

with the SFC Jacobian $\hat{\mathbf{J}}_s$ of (4.21).

Lemma 7. *The mapping $\Phi : \mathcal{X} \mapsto \mathcal{Z}$ with $\mathcal{X} = \mathcal{Q} \times \mathcal{T}_{\mathbf{q}}\mathcal{Q}$, $\mathcal{Q} = \{[\bar{\mathbf{q}}_J^T, \bar{\mathbf{q}}_M^T]^T \in \mathbb{R}^{2N} : \beta_i \circ \mathbf{h}_t(\bar{\mathbf{q}}_J) > 0, i = 1, 2\}$ and tangential space $\mathcal{T}_{\mathbf{q}}\mathcal{Q}$, is a \mathcal{C}^1 -diffeomorphism, if $\mathbf{J}(\mathbf{q}_J)$ is nonsingular.*

The proof is identical to the proof of Lemma 3 by just replacing $\hat{\mathbf{J}}_p$ by $\hat{\mathbf{J}}_s$.

4.6.1.2 Feedback linearization

The third time derivative of the virtual output $\hat{\mathbf{y}}_s$ is given by (3.57), where $\hat{\mathbf{J}}_p$ is replaced by $\hat{\mathbf{J}}_s$. Hence, the feedback transformation that yields the linear input-output relation

$$\hat{\mathbf{y}}_s^{(3)} = \mathbf{v}_{s,v} \quad (4.39)$$

and the non-observable internal dynamics (3.59b) is similar to (3.58) and given by

$$\boldsymbol{\tau}_d = \mathbf{D}_{s,v}^{-1} (\mathbf{v}_{s,v} - \mathbf{b}_{s,v}(\mathbf{q}_J, \dot{\mathbf{q}}_J, \mathbf{q}_M, \dot{\mathbf{q}}_M)) \quad , \quad (4.40a)$$

with the new control input $\mathbf{v}_{s,v}^T = [\mathbf{v}_{s,t,v}^T, \mathbf{v}_{s,r,v}^T] \in \mathbb{R}^N$, the decoupling matrix

$$\mathbf{D}_{s,v}(\mathbf{q}_J) = \hat{\mathbf{J}}_s \mathbf{D}_E^{-1} \mathbf{K}_d \mathbf{D}_M^{-1} \quad , \quad (4.40b)$$

and the vector

$$\begin{aligned} \mathbf{b}_{s,v}(\mathbf{q}_J, \dot{\mathbf{q}}_J, \mathbf{q}_M, \dot{\mathbf{q}}_M) = & \ddot{\hat{\mathbf{J}}}_s \dot{\mathbf{q}}_J + 2\dot{\hat{\mathbf{J}}}_s \ddot{\mathbf{q}}_J - \hat{\mathbf{J}}_s \mathbf{D}_E^{-1} \left((\dot{\mathbf{D}}_E + \mathbf{K}_d) \ddot{\mathbf{q}}_J + \dot{\mathbf{n}}_E \right. \\ & \left. - \dot{\boldsymbol{\tau}}_{ext} - \mathbf{K}_s(\dot{\mathbf{q}}_M - \dot{\mathbf{q}}_J) + \mathbf{K}_d \mathbf{D}_M^{-1}(\boldsymbol{\tau}_J + \boldsymbol{\tau}_{f,M}) \right) . \end{aligned} \quad (4.40c)$$

The stability proof of the zero dynamics is given in Section 3.6.1.2. The new input $v_{\perp,v}$ of $\mathbf{v}_{s,t,v}^T = [v_{\parallel,1,v}, v_{\parallel,2,v}, v_{\perp,v}]$ can be used to fulfill the control objectives (O1) and (O2) and with $v_{\parallel,1,v}$ and $v_{\parallel,2,v}$ an application specific motion along the surface can be achieved. Note that the decoupling matrix $\mathbf{D}_{s,v}(\mathbf{q}_J)$ is nonsingular for $[\mathbf{q}_J^T, \mathbf{q}_M^T] \in \mathcal{Q}$.

4.6.1.3 Stabilization of the linearized system

Similar to the PFC for visco-elastic joint robots, the position control law

$$\mathbf{v}_{s,v} = \begin{bmatrix} (\boldsymbol{\eta}_1^p)^{(3)} - a_{\boldsymbol{\eta},3,v} \ddot{\mathbf{e}}_{\boldsymbol{\eta}} - a_{\boldsymbol{\eta},2,v} \dot{\mathbf{e}}_{\boldsymbol{\eta}} - a_{\boldsymbol{\eta},1,v} \mathbf{e}_{\boldsymbol{\eta}} \\ (\boldsymbol{\xi}_1^p)^{(3)} - a_{\boldsymbol{\xi},3,v} \ddot{\mathbf{e}}_{\boldsymbol{\xi}_1} - a_{\boldsymbol{\xi},2,v} \dot{\mathbf{e}}_{\boldsymbol{\xi}_1} - a_{\boldsymbol{\xi},1,v} \mathbf{e}_{\boldsymbol{\xi}_1} \\ (\mathbf{y}_r^p)^{(3)} - a_{r,3,v} \ddot{\mathbf{e}}_r - a_{r,2,v} \dot{\mathbf{e}}_r - a_{r,1,v} \mathbf{e}_r \end{bmatrix} \quad (4.41)$$

asymptotically stabilizes the linear system (4.39) if $p_i(s) = s^3 + a_{i,3,v} s^2 + a_{i,2,v} s + a_{i,1,v}$ constitutes a Hurwitz polynomial for $i \in \{\boldsymbol{\eta}, \boldsymbol{\xi}, r\}$. In (4.41), $\boldsymbol{\eta}_1^p$ and $\boldsymbol{\xi}_1^p$ denote \mathcal{C}^3 -references for the position on the surface and the deviation from the surface \mathcal{S}_t . A combination with compliance control is possible as described in Section 3.6.1.3.

4.6.2 Robots with pure elastic joints

The joint forces $\boldsymbol{\tau}_J$ of the dynamic system (2.65) are given by (2.67) and the system has a (vector) relative degree of $\{4, 4, \dots, 4\}$, when the viscosity inside the joints is negligibly small.

4.6.2.1 Coordinate transformation

A coordinate transformation is derived that maps the joint coordinates \mathbf{q}_J and the motor coordinates \mathbf{q}_M as well as their first time derivative onto the virtual output $\hat{\mathbf{y}}_s^T = [\boldsymbol{\eta}_1^T, \xi_1, \boldsymbol{\zeta}_1^T]$ and their first, second, and third time derivative with respect to a \mathcal{C}^5 surface \mathcal{S} . The coordinate transformation Φ reads as

$$\begin{bmatrix} \hat{\mathbf{y}}_s \\ \dot{\hat{\mathbf{y}}}_s \\ \ddot{\hat{\mathbf{y}}}_s \\ \hat{\mathbf{y}}_s^{(3)} \end{bmatrix} = \begin{bmatrix} \mathbf{g} \circ \mathbf{h}_t(\mathbf{q}_J) \\ \delta \circ \mathbf{h}_t(\mathbf{q}_J) \\ \mathbf{h}_r(\mathbf{q}_J) \\ \hat{\mathbf{J}}_s \dot{\mathbf{q}}_J \\ \dot{\hat{\mathbf{J}}}_s \dot{\mathbf{q}}_J + \hat{\mathbf{J}}_s \mathbf{D}_E^{-1}(\boldsymbol{\tau}_J + \boldsymbol{\tau}_{ext} - \mathbf{n}_E) \\ \mathbf{r}_s + \hat{\mathbf{J}}_s \mathbf{D}_E^{-1}(\dot{\boldsymbol{\tau}}_J + \dot{\boldsymbol{\tau}}_{ext} - \dot{\mathbf{n}}_E - \dot{\mathbf{D}}_E \ddot{\mathbf{q}}_J) \end{bmatrix} = \Phi(\mathbf{q}_J, \dot{\mathbf{q}}_J, \mathbf{q}_M, \dot{\mathbf{q}}_M), \quad (4.42)$$

with the SFC Jacobian $\hat{\mathbf{J}}_s$ of (4.21) and the vector $\mathbf{r}_s = \ddot{\hat{\mathbf{J}}}_s(\mathbf{q}_J, \dot{\mathbf{q}}_J, \ddot{\mathbf{q}}_J) \dot{\mathbf{q}}_J + 2\dot{\hat{\mathbf{J}}}_s(\mathbf{q}_J, \dot{\mathbf{q}}_J) \ddot{\mathbf{q}}_J$.

Lemma 8. *The mapping $\Phi : \mathcal{X} \mapsto \mathcal{Z}$ with $\mathcal{X} = \mathcal{Q} \times \mathcal{T}_q \mathcal{Q}$, $\mathcal{Q} = \{[\bar{\mathbf{q}}_J^T, \bar{\mathbf{q}}_M^T]^T \in \mathbb{R}^{2N} : \beta_i \circ \mathbf{h}_t(\bar{\mathbf{q}}_J) > 0, i = 1, 2\}$ and tangential space $\mathcal{T}_q \mathcal{Q}$, is a \mathcal{C}^1 -diffeomorphism, if $\mathbf{J}(\mathbf{q}_J)$ is nonsingular.*

The proof is identical to the proof of Lemma 4 by just replacing $\hat{\mathbf{J}}_p$ by $\hat{\mathbf{J}}_s$.

4.6.2.2 Feedback linearization

Differentiating $\hat{\mathbf{y}}_s$ four times with respect to time yields (3.64), where $\hat{\mathbf{J}}_p$ is replaced by $\hat{\mathbf{J}}_s$. Hence, the linear input-output relation

$$\hat{\mathbf{y}}_s^{(4)} = \mathbf{v}_{s,e} \quad (4.43)$$

is obtained by applying the feedback transformation (3.65), where $\hat{\mathbf{J}}_p$ is replaced by $\hat{\mathbf{J}}_s$, to the system (2.65), joint forces $\boldsymbol{\tau}_J$ of (2.67) and output function (2.1). For the sake of completeness, the feedback transformation is recapitulated here

$$\boldsymbol{\tau}_d = \mathbf{D}_{s,e}^{-1}(\mathbf{v}_{s,e} - \mathbf{b}_{s,e}(\mathbf{q}_J, \dot{\mathbf{q}}_J, \mathbf{q}_M, \dot{\mathbf{q}}_M)), \quad (4.44)$$

with the new control input $\mathbf{v}_{s,e}^T = [\mathbf{v}_{s,t,e}^T, \mathbf{v}_{s,r,e}^T] \in \mathbb{R}^N$, the decoupling matrix

$$\mathbf{D}_{s,e}(\mathbf{q}_J) = \hat{\mathbf{J}}_s \mathbf{D}_E^{-1} \mathbf{K}_s \mathbf{D}_M^{-1}, \quad (4.45)$$

and the vector

$$\begin{aligned} \mathbf{b}_{s,e}(\mathbf{q}_J, \dot{\mathbf{q}}_J, \mathbf{q}_M, \dot{\mathbf{q}}_M) &= \hat{\mathbf{J}}_s^{(3)} \dot{\mathbf{q}}_J + 3\hat{\mathbf{J}}_s \ddot{\mathbf{q}}_J + 3\dot{\hat{\mathbf{J}}}_s \mathbf{q}_J^{(3)} - \mathbf{D}_{s,e}(\boldsymbol{\tau}_J - \boldsymbol{\tau}_{f,M}) \\ &\quad - \hat{\mathbf{J}}_s \mathbf{D}_E^{-1} \left(2\dot{\mathbf{D}}_E \mathbf{q}_J^{(3)} + \ddot{\mathbf{D}}_E \ddot{\mathbf{q}}_J + \ddot{\mathbf{n}}_E - \ddot{\boldsymbol{\tau}}_{ext} + \mathbf{K}_s \ddot{\mathbf{q}}_J \right). \end{aligned} \quad (4.46)$$

The new input $v_{\perp,e}$ of $\mathbf{v}_{s,t,e}^T = [v_{\parallel,1,e}, v_{\parallel,2,e}, v_{\perp,e}]$ can be used to fulfill the control objectives (O1) and (O2) and with $v_{\parallel,1,e}$ and $v_{\parallel,2,e}$ an application specific motion on the surface \mathcal{S}_t can be achieved.

4.6.2.3 Stabilization of the linearized system

Similar to the PFC for pure elastic joint robots, the position control law

$$\mathbf{v}_{s,e} = \begin{bmatrix} (\boldsymbol{\eta}_1^p)^{(4)} - a_{\eta,4,e} \mathbf{e}_{\eta}^{(3)} - a_{\eta,3,e} \ddot{\mathbf{e}}_{\eta} - a_{\eta,2,e} \dot{\mathbf{e}}_{\eta} - a_{\eta,1,e} \mathbf{e}_{\eta} \\ (\boldsymbol{\xi}_1^p)^{(4)} - a_{\xi,4,e} \mathbf{e}_{\xi}^{(3)} - a_{\xi,3,e} \ddot{\mathbf{e}}_{\xi} - a_{\xi,2,e} \dot{\mathbf{e}}_{\xi} - a_{\xi,1,e} \mathbf{e}_{\xi} \\ (\mathbf{y}_r^p)^{(4)} - a_{r,4,e} \mathbf{e}_r^{(3)} - a_{r,3,e} \ddot{\mathbf{e}}_r - a_{r,2,e} \dot{\mathbf{e}}_r - a_{r,1,e} \mathbf{e}_r \end{bmatrix} \quad (4.47)$$

asymptotically stabilizes the linear system (4.43) if $p_i(s) = s^4 + a_{i,4,e} s^3 + a_{i,3,e} s^2 + a_{i,2,e} s + a_{i,1,e}$ constitutes a Hurwitz polynomial for $i \in \{\eta, \xi, r\}$. The references in (4.47) have to be of class \mathcal{C}^4 . The same arguments are valid regarding the combination with compliance control as stated in Section 3.6.2.3.

4.7 Implementation

As for the PFC, the static state feedback controllers are implemented on digital computers with the sampling time T_s . The optimization problem (4.9) and the integral (4.15) have to be numerically solved in real time.

The optimization problem (4.9) is solved using the Newton method. For the initialization, the global optimum $\boldsymbol{\theta}_0^*$ is needed. A sufficient number of evenly spread points on the surface \mathcal{S}_t are chosen and the distances to $\mathbf{y}_t(0)$ are calculated. The point with shortest distance is used as starting point for the local minimum search to obtain $\boldsymbol{\theta}_0^*$. Then, the optimization problem (4.9) is iteratively solved in each time step $k = 1, 2, \dots$ for $i = 1, 2, \dots$ according to

$$\boldsymbol{\theta}_{k,i} = \boldsymbol{\theta}_{k,i-1} - \mathbf{H}^{-1}(\mathbf{y}_{t,k}, \boldsymbol{\theta}_{k,i-1}) \left(\frac{\partial f}{\partial \boldsymbol{\theta}} \right) (\mathbf{y}_{t,k}, \boldsymbol{\theta}_{k,i-1}), \quad (4.48)$$

with $\mathbf{y}_{t,k} = \mathbf{y}_t(kT_s)$, f from (4.9), and the initial condition $\boldsymbol{\theta}_{k,0} = \boldsymbol{\theta}_{k-1}^*$ until $\|\boldsymbol{\theta}_{k,i} - \boldsymbol{\theta}_{k,i-1}\|_2 < \varepsilon$. The optimal solution $\boldsymbol{\theta}_k^* = \boldsymbol{\theta}_{k,i}$ is used to perform the numerical integration of (4.15), i.e.,

$$\boldsymbol{\eta}_{1,k} = \boldsymbol{\eta}_{1,k-1} + \boldsymbol{\Sigma}^*(\boldsymbol{\theta}_k^*) \dot{\boldsymbol{\theta}}_k^* T_s, \quad (4.49)$$

where

$$\dot{\boldsymbol{\theta}}_k^* = (\mathbf{H}^*(\mathbf{y}_{t,k}, \boldsymbol{\theta}_k^*))^{-1} \begin{bmatrix} \left(\begin{smallmatrix} \boldsymbol{\sigma}_{t,\theta_1}^* \\ \boldsymbol{\sigma}_{t,\theta_2}^* \end{smallmatrix} \right) (\boldsymbol{\theta}_k^*)^T \\ \left(\begin{smallmatrix} \boldsymbol{\sigma}_{t,\theta_1}^* \\ \boldsymbol{\sigma}_{t,\theta_2}^* \end{smallmatrix} \right) (\boldsymbol{\theta}_k^*)^T \end{bmatrix} \dot{\mathbf{y}}_{t,k} ,$$

$$\dot{\mathbf{y}}_{t,k} = \mathbf{J}_t(\mathbf{q}_{J,k}) \dot{\mathbf{q}}_{J,k}, \quad \mathbf{q}_{J,k} = \mathbf{q}_J(kT_s), \quad \dot{\mathbf{q}}_{J,k} = (\mathbf{q}_{J,k} - \mathbf{q}_{J,k-1})/T_s, \quad \text{and} \quad \boldsymbol{\eta}_{1,k} = \boldsymbol{\eta}_1(kT_s).$$

4.8 Applications

SFC for fully actuated manipulators is implemented on the real-time system *DS1006* from DSPACE with a sampling time of $T_s = 1$ ms to control a DELTA robot of the type FESTO EXPT-45, see Fig. 3.4, and the six-axis industrial robot COMAU Racer 1.4. Experimental results with position control are shown in Section 4.8.1 and with the combination of SFC and compliance control in Section 4.8.2.

The performance of the SFC approach for visco-elastic joint robots is evaluated by a simulation study using MATLAB[®]/Simulink[®] R2016b on a 64 bit Windows 7 computer and the results are presented in Section 4.8.3. Due to the similarity of the SFC approach to the PFC approach of Chapter 3, no simulation example is given for the SFC approach for elastic joint robots.

An application of the kinematic SFC to generate virtual fixtures for physical human-robot interaction is presented in Chapter 5.

4.8.1 Surface following control of a Delta robot

In this experiment, the SFC law (4.26) with the Jacobian (2.33) and the position controller (4.30) is implemented on the real-time system *DS1006* and applied to the DELTA robot FESTO EXPT-45.

A cylinder with parametrization $\boldsymbol{\sigma}_t(\boldsymbol{\theta}) : \mathcal{T}_s \mapsto \mathbb{R}^3$ is used to demonstrate the surface following control approach, see Fig. 4.3. The parametrization is given by

$$\boldsymbol{\sigma}_t(\boldsymbol{\theta}) = \begin{bmatrix} r \cos(\theta_1) \\ \theta_2 \\ r \sin(\theta_1) + c \end{bmatrix}, \quad (4.50)$$

with radius $r = 90$ mm and offset $c = 505$ mm, $\mathcal{T}_s = [0, 2\pi] \times [\underline{\theta}_2, \bar{\theta}_2]$, and $\underline{\theta}_2 < \bar{\theta}_2 \in \mathbb{R}$. Points on the surface $\boldsymbol{\eta}_{1,i}^p$, $i = 0, 1, 2, 3$ are chosen for the tangential coordinates. They are smoothly connected with a maximal velocity of $\boldsymbol{\eta}_{2,\max}^p = [0.2, 0.2]^T$ m/s to generate the reference in tangential direction $\boldsymbol{\eta}_1^p$. The sequence is defined as

$$\begin{array}{ccc} \boldsymbol{\eta}_{1,0}^p = [0, 0]^T \text{m} & \rightarrow & \boldsymbol{\eta}_{1,1}^p = [0.25, 0]^T \text{m} \\ \uparrow & & \downarrow \\ \boldsymbol{\eta}_{1,3}^p = [0, 0.15]^T \text{m} & \leftarrow & \boldsymbol{\eta}_{1,2}^p = [0.25, 0.15]^T \text{m} . \end{array} \quad (4.51)$$

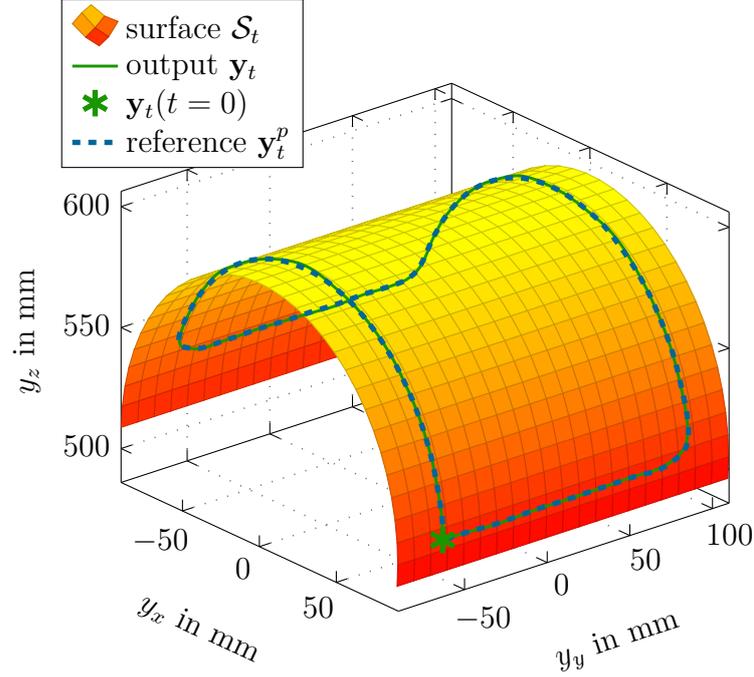


Figure 4.3: Surface \mathcal{S}_t and measured output \mathbf{y}_t of the SFC experiment on a DELTA robot.

Table 4.1: Control parameters.

Symbol	Value	Unit	Symbol	Value	Unit
$a_{\eta,0}$	64000	1/s ³	$a_{\xi,0}$	64000	1/s ³
$a_{\eta,1}$	4800	1/s ²	$a_{\xi,1}$	4800	1/s ²
$a_{\eta,2}$	120	1/s	$a_{\xi,2}$	120	1/s

The reference in transversal direction ξ_1^p and their derivatives are set to zero. Integral parts are added to the position controller (4.30), which reads as

$$\mathbf{v}_s = \begin{bmatrix} \ddot{\eta}_1^p - a_{\eta,2}\dot{e}_\eta^p - a_{\eta,1}e_\eta^p - a_{\eta,0} \int_0^t e_\eta^p d\tau \\ \ddot{\xi}_1^p - a_{\xi,2}\dot{e}_\xi^p - a_{\xi,1}e_\xi^p - a_{\xi,0} \int_0^t e_\xi^p d\tau \end{bmatrix} \quad (4.52)$$

for this experiment. The parameters of the position controller are shown in Table 4.1. For the experiment, the end-effector is initially placed next to the surface. Fig. 4.3 depicts the cylinder \mathcal{S}_t , the starting point $\mathbf{y}_t^T(t_0) = [89.9, -62.8, 509.2]$ mm, with respective initial parameters $\theta_{1,0} = 0.0467$ rad and $\theta_{2,0} = -0.0628$ m, the output \mathbf{y}_t , and the reference $\mathbf{y}_t^p = \mathbf{h}_D \circ \Phi^{-1}(\hat{\mathbf{y}}_s^p, \dot{\hat{\mathbf{y}}}_s^p)$ with $(\hat{\mathbf{y}}_s^p)^T = [(\boldsymbol{\eta}_1^p)^T, \xi_1^p = 0]$. It can be inferred from Fig. 4.3 that the output \mathbf{y}_t stays on the surface \mathcal{S}_t for all times and that the reference \mathbf{y}_t^p is tracked quite well.

Fig. 4.4(a) shows the tracking behavior of the references in tangential directions

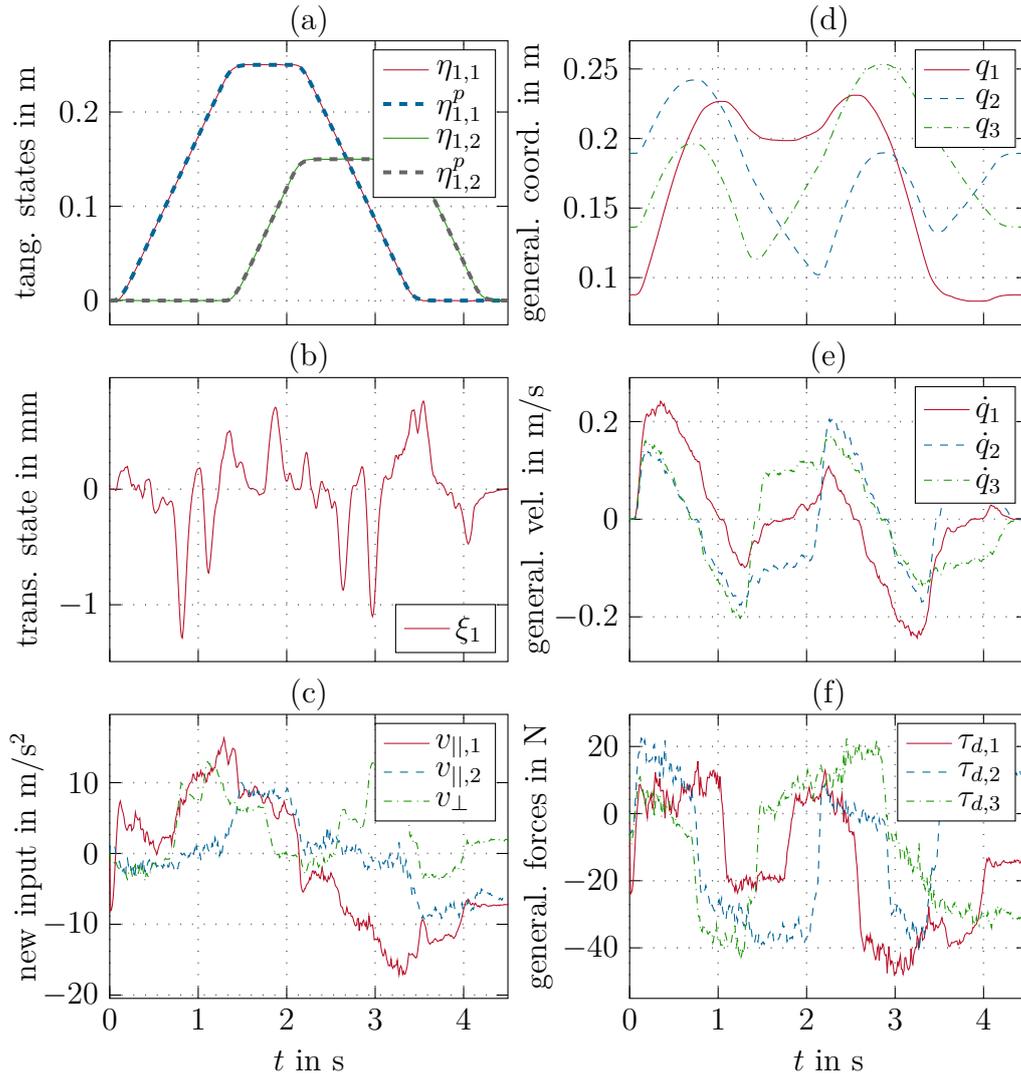


Figure 4.4: Experimental results of the SFC approach applied to the DELTA robot.

$\eta_{1,1}^p$ and $\eta_{1,2}^p$. The position error in the transversal state ξ_1 is in the range of 1 mm, cf. Fig. 4.4(b). The new inputs $v_{||,1}$, $v_{||,2}$, and v_{\perp} vary quite smoothly, see Fig. 4.4(c). The generalized coordinates q_1 , q_2 , and q_3 are depicted in Fig. 4.4(d) and the generalized velocities \dot{q}_1 , \dot{q}_2 , and \dot{q}_3 in Fig. 4.4(e). The deviations in the transversal state ξ_1 are mainly due to stick-slip friction in the linear drives. This becomes clear by comparing the zero crossings of the generalized velocities with the deviations in the transversal direction. Moreover, the input forces $\tau_{d,1}$, $\tau_{d,2}$, and $\tau_{d,3}$ of Fig. 4.4(f) also clearly show the influence of the friction.

4.8.2 Compliant surface following control of an industrial robot

The combination of SFC with compliance control for fully actuated manipulators introduced in Section 4.5 is experimentally validated on the six-axis industrial robot COMAU Racer 1.4. Thereby, the control laws (4.26), (4.52), and (4.33) are used to stabilize the robot on a paraboloid of revolution. A human operator is able to move the robot along the surface without effort by using low stiffness k_{\parallel}^d and damping d_{\parallel}^d in tangential direction. The stiffness k_{\perp}^d and damping d_{\perp}^d in orthogonal direction are chosen rather high to limit the deviations from the surface. To avoid representation singularities of the orientation, the compliance control laws (3.44) and (3.46) are utilized to compute the new orientation input $\mathbf{v}_{s,o}$. The control parameters for (4.52) and (3.46) are listed in Tab. 4.2 and the impedance parameters for (4.33) and (3.44) in Tab. 4.3.

The COMAU Racer 1.4, which is shown in Fig. 4.5 and described in Chapter 2, is used for this experiment. The 6D-force/torque sensor *K6-D40* from ME-MESSYSTEME is attached to the robot's end-effector and used as haptic input device. The SFC is implemented on the real-time system *DS1006* from dSPACE with a sampling time of $T_s = 1$ ms and the torque commands $\boldsymbol{\tau}_d$ are sent to joint servo controllers. A communication delay of approximately 2ms between the *DS1006* and the servo controllers limit the control gains in the position controller (4.30). The experimental setup is depicted in Fig. 4.6.

Table 4.2: Control parameters for the SFC.

Symbol	Value	Unit	Symbol	Value	Unit
$a_{\eta,0}$	4913	1/s ³	$a_{\xi,0}$	17576	1/s ³
$a_{\eta,1}$	867	1/s ²	$a_{\xi,1}$	2028	1/s ²
$a_{\eta,2}$	51	1/s	$a_{\xi,2}$	78	1/s
$a_{r,1}$	2700	1/s ²	$a_{r,2}$	90	1/s

Table 4.3: Compliance control parameters.

Symbol	Value	Unit	Symbol	Value	Unit
m_{\parallel}^d	3	kg	m_{\perp}^d	3	kg
d_{\parallel}^d	90	Ns/m	d_{\perp}^d	150	Ns/m
k_{\parallel}^d	1.5	N/m	k_{\perp}^d	5	kN/m
m_r^d	0.2	kgm ²	k_r^d	34.4	Nm/rad
d_r^d	24.6	Nms/rad			

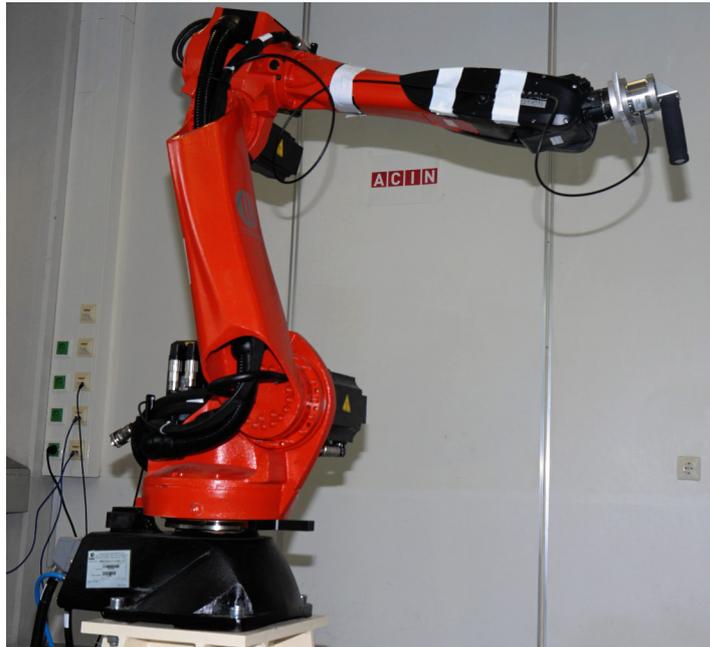


Figure 4.5: COMAU Racer 1.4 equipped with the 6D-force/torque sensor *K6-D40*. [81] © 2018 IEEE

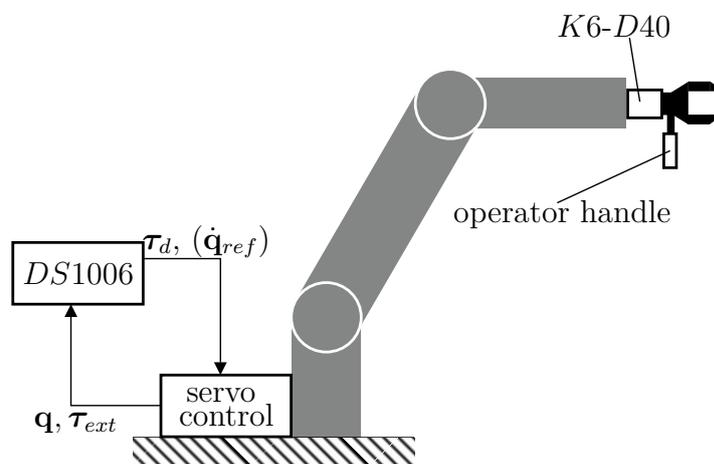


Figure 4.6: Experimental setup. [81] © 2018 IEEE

4.8.2.1 Measurement results of the compliant SFC

The surface \mathcal{S} is defined as a paraboloid of revolution with constant orientation using the smooth parametrization

$$\boldsymbol{\sigma}(\boldsymbol{\theta}) = \begin{bmatrix} \boldsymbol{\sigma}_t(\boldsymbol{\theta}) \\ \boldsymbol{\sigma}_r \end{bmatrix} = \begin{bmatrix} \theta_1 + 0.83 \\ \theta_2 \\ 12.5(\theta_1^2 + \theta_2^2) + 0.6 \\ 0 \\ \pi/4 \\ \pi - 0.1 \end{bmatrix} \quad (4.53)$$

and is depicted in Fig 4.7. All numbers for $\boldsymbol{\sigma}_t(\boldsymbol{\theta})$ are given in meters.

The external references $\boldsymbol{\eta}_1^d$ and $\boldsymbol{\xi}_1^d$ are set to zero. Hence, the robot's motion is only caused by the input forces of the operator via the impedance control law (4.33) and is depicted in Fig. 4.7. Fig. 4.8(a) shows that the desired motion on the surface $\boldsymbol{\eta}_1^p = [\eta_{1,1}^p, \eta_{1,2}^p]^T$ is tracked very well. The desired deviation from the surface \mathcal{S}_t corresponds with the operator's input force in orthogonal direction τ_\perp , see Fig. 4.8(b) and Fig. 4.8(c). An operator force of less than 20N is necessary for the motion along the surface. Fig. 4.8(d) and Fig. 4.8(e) show that the impedance of the orientation also behaves as specified. The first three joint torques are depicted in Fig. 4.8(f).

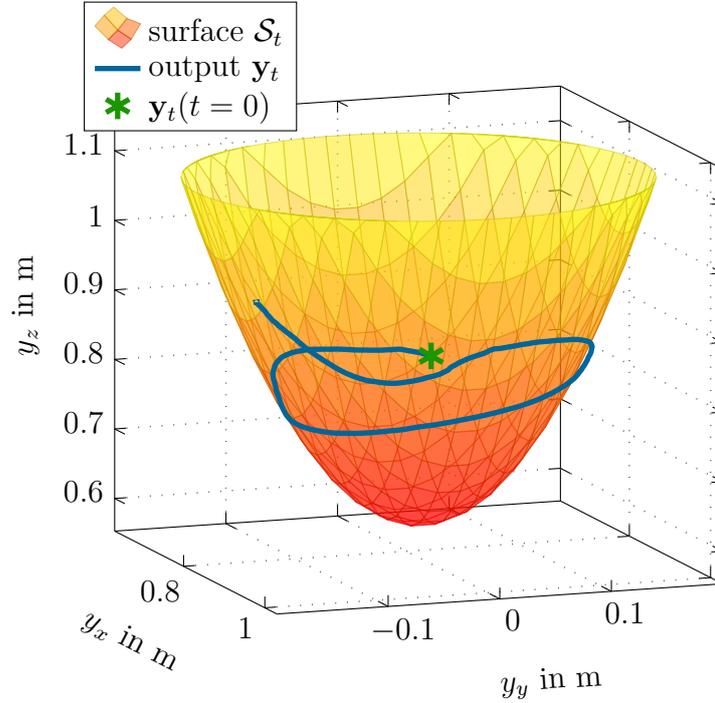


Figure 4.7: Surface \mathcal{S}_t and measured output \mathbf{y}_t of the combined SFC and compliance control strategy. [81] © 2018 IEEE

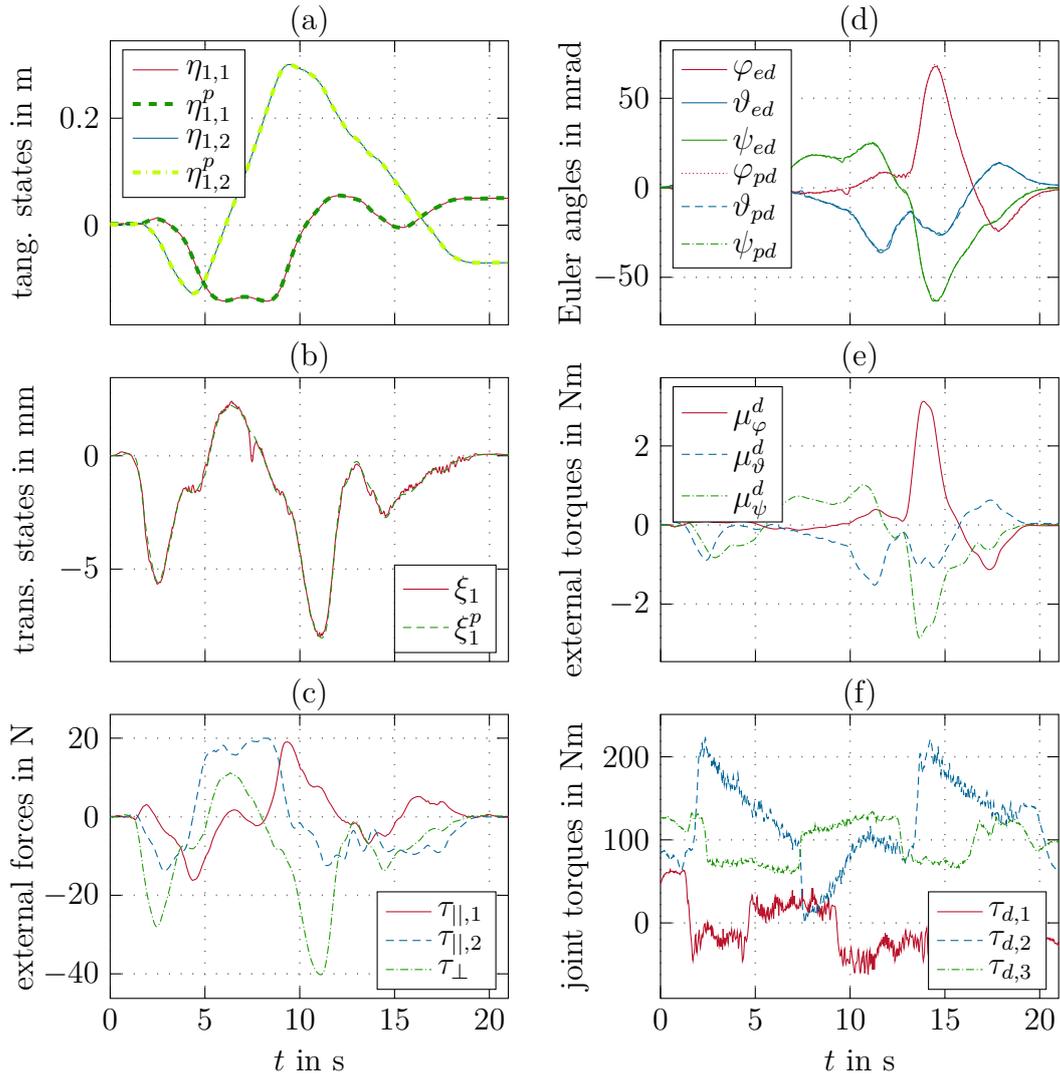


Figure 4.8: Experimental results of the combined SFC and compliance control strategy. [81] © 2018 IEEE

Hence, the results of this experiment confirm that the proposed SFC approach is well suited to be combined with compliance control.

4.8.3 Surface following control of a 3R visco-elastic joint robot

The SFC approach for robots with visco-elastic joints presented in Section 4.6.1 is verified by a simulation study in MATLAB[®]/Simulink[®] and is applied to the COMAU Racer 1.4, where only the first three joints are actuated and the last three joints are fixed. In contrast to the real robot, the transmissions of the

joints are assumed to be visco-elastic with a relatively low stiffness of $\mathbf{K}_s = 10^3 \cdot \text{diag}([35, 35, 19])\text{Nm/rad}$ and a viscosity of $\mathbf{K}_d = \text{diag}([10, 10, 10])\text{Nms/rad}$. Hence, the simplified equations of motion are given by (2.65) with the joint torques (2.66). Moreover, the friction $\boldsymbol{\tau}_f$ is set to zero.

The smooth surface \mathcal{S}_t is defined as a paraboloid of revolution using the parametrization

$$\boldsymbol{\sigma}_t(\boldsymbol{\theta}) = \begin{bmatrix} \theta_1 + 0.7 \\ \theta_2 \\ 1.667(\theta_1^2 + \theta_2^2) + 1.11 \end{bmatrix} \quad (4.54)$$

and is depicted in Fig 4.9. All numbers are given in meters.

The tangential reference $\boldsymbol{\eta}_1^p = [\eta_{1,1}^p, \eta_{1,2}^p]^T$ is a smooth function connecting the points $\eta_{1,1}^p(t_0) = 0$, $\eta_{1,1}^p(t_{1,1}) = 0.3\text{m}$, and $\eta_{1,1}^p(t_{1,2}) = -0.25\text{m}$ with a maximum velocity of $|\dot{\eta}_{1,1}^p| = 1.5\text{m/s}$, as well as $\eta_{1,2}^p(t_0) = 0$, $\eta_{1,2}^p(t_{2,1}) = 0.2\text{m}$, and $\eta_{1,2}^p(t_{2,2}) = -0.2\text{m}$ with a maximum velocity of $|\dot{\eta}_{1,2}^p| = 1\text{m/s}$. The trajectory of $\boldsymbol{\eta}_1^p$ is depicted in the left upper graph of Fig. 4.10. The starting point of the end-effector is defined as $\mathbf{y}_t(t_0) = [0.7, 0, 1.1]^T\text{m}$, which is 10mm off the surface \mathcal{S}_t . The reference in transversal direction ξ_1^p and their derivatives are set to zero. The SFC law (4.40) and the position controller (4.41), with the parameters from Tab. 4.4, are implemented in MATLAB[®]/Simulink[®].

The simulated output \mathbf{y}_t is depicted in Fig. 4.9 and the trajectories in Fig. 4.10.

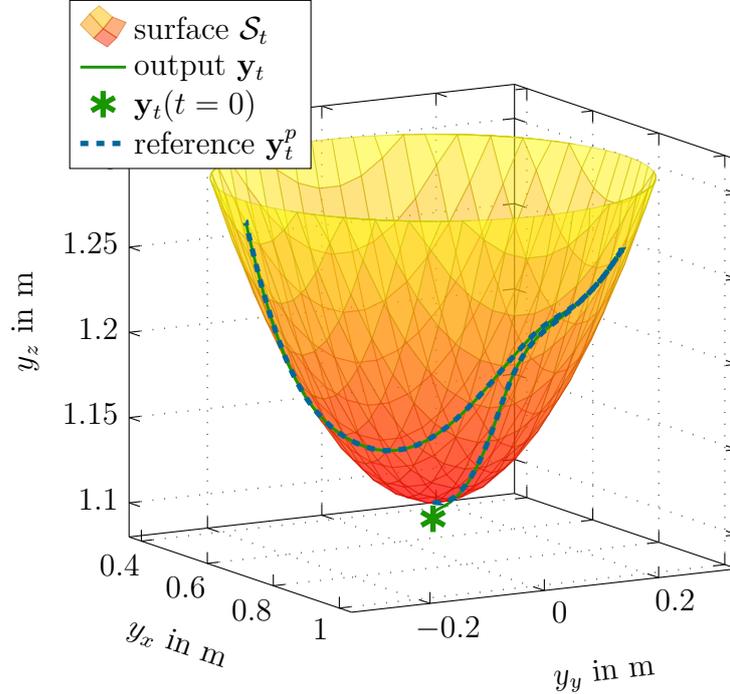


Figure 4.9: Surface \mathcal{S}_t and simulated output \mathbf{y}_t for the SFC applied to a 3R visco-elastic joint robot.

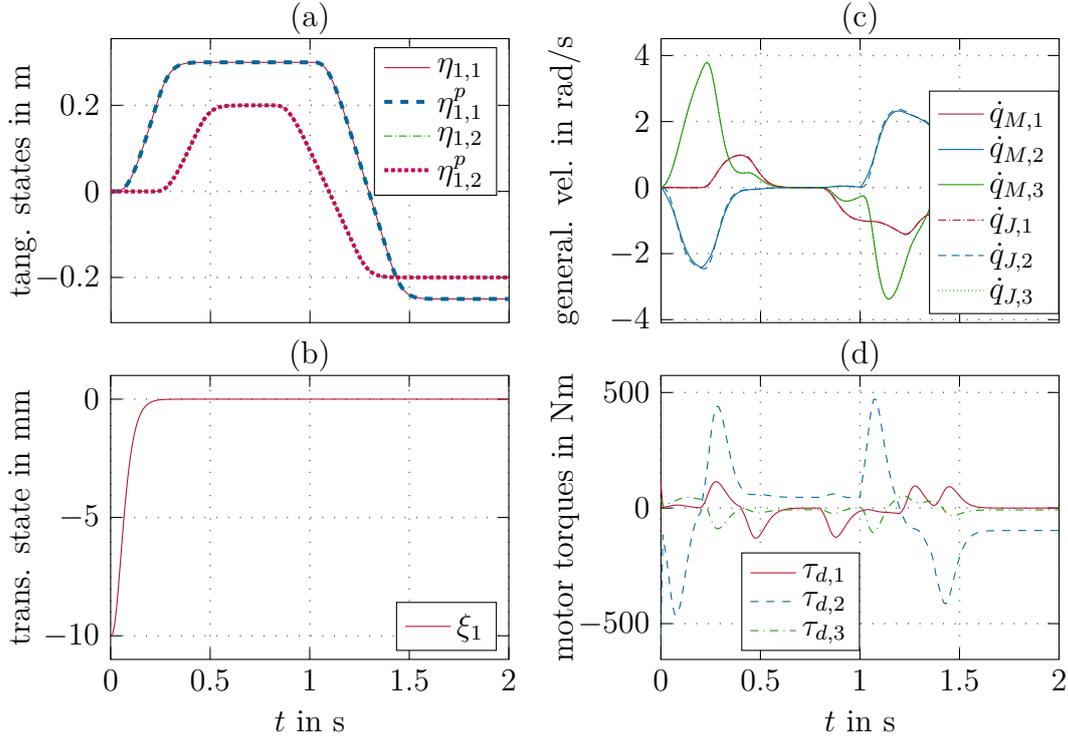


Figure 4.10: Simulated performance of the the SFC approach for a 3R visco-elastic joint robot.

Table 4.4: Control parameters.

Symbol	Value	Unit	Symbol	Value	Unit
$a_{\eta,1,v}$	64000	$1/s^3$	$a_{\xi,1,v}$	64000	$1/s^3$
$a_{\eta,2,v}$	4800	$1/s^2$	$a_{\xi,2,v}$	4800	$1/s^2$
$a_{\eta,3,v}$	120	$1/s$	$a_{\xi,3,v}$	120	$1/s$

Fig. 4.10(a) shows that the desired motion on the surface $\boldsymbol{\eta}_1^p$ is tracked very well and Fig. 4.10(b) illustrates that the output \mathbf{y}_t converges to and then remains on the surface \mathcal{S}_t . The motor velocities $\dot{q}_{M,i}$ slightly differ from the joint velocities $\dot{q}_{J,i}$, in particular for high drive torques $\tau_{d,i}$, due to the elastic transmissions, see Fig. 4.10(c) and Fig. 4.10(d).

CHAPTER 5

Path and Surface Following Control for Human-Robot Interaction

In this chapter, applications of path and surface following control for physical human-robot interaction are presented, in which an operator is hand-guiding a robot. PFC and SFC allow to restrict the motion of a robot to a path or surface. Hence, these control approaches can be used to systematically generate a large number of virtual fixtures.

The concepts to generate virtual fixtures introduced in this chapter are based on the PFC and SFC approaches for fully actuated manipulators, where the feedback linearizations (3.29) and (4.26) are denoted as dynamic PFC and SFC, respectively, and the feedback linearizations (3.50) and (4.35) are denoted as kinematic PFC and SFC. Note that the concepts to generate virtual fixtures with the dynamic PFC/SFC can also be applied to the PFC/SFC approaches for elastic joint robots.

Four experiments to generate various virtual fixtures in free space on the six-axis robot COMAU Racer 1.4 are presented in Section 5.2. These are soft guidance virtual fixtures with path constraint, hard guidance virtual fixtures with path constraint, dynamic guidance virtual fixtures with path constraint, and hard forbidden region virtual fixtures, where the robot's workspace is restricted to a cylinder and, additionally, to the outside of a paraboloid of revolution, which is placed inside the cylinder. Section 5.3 presents an application of the virtual fixture approach to a semi-automation production use case, in which a plate is taken from a magazine and assembled into a frame.

Parts of this chapter are published in similar form in [81].

5.1 Generating virtual fixtures with path and surface following control

The PFC and SFC approaches introduced in Chapter 3 and Chapter 4 decouple and exactly linearize the dynamics in tangential and orthogonal direction to a path γ or surface \mathcal{S} . Thereby, PFC and SFC are only applied to the position parametrization of the path or surface, γ_t and \mathcal{S}_t , respectively. The reference orientation is coupled to the optimal path parameter θ^* (or optimal surface parameter vector $\boldsymbol{\theta}^*$) and given by the trajectory $\mathbf{y}_r^p(t) = \boldsymbol{\sigma}_r(\theta^*(t))$. Hence, the behavior of the robot along a path or surface and away from it can be defined independently and in a physically interpretable manner. This allows to systematically realize a large number of possible virtual fixtures. In contrast to other virtual fixture methods, a systematic proof of the closed-loop stability including the dynamics of the manipulator can be given. A compliant behavior of the manipulator is achieved by measuring the external force/torque $\mathbf{f}_{op} = \mathbf{J}_g^{-T} \boldsymbol{\tau}_{ext}$, which is exerted on the end-effector by an operator, see also (2.11) for the manipulator Jacobian \mathbf{J}_g . The generalized external forces $\boldsymbol{\tau}_{ext}$ are then transformed into tangential, transversal, and orientation coordinates according to (3.41) and (4.32) and the impedance control laws (3.42) or (4.33) are used as presented in the Sections 3.5.3.2 and 4.5.3.2, respectively. Note that for systems with very low (or compensated) friction, a desired compliance behavior can also be achieved without a force sensor by PFC/SFC for the dynamic system and the position control (3.33) or (4.30) with small gains. For such systems, the operator's input force \mathbf{f}_{op} on the end-effector can be estimated from the drive forces/torques $\boldsymbol{\tau}_d$.

The survey papers [40] and [43] distinguish between several properties and classifications of virtual fixtures. An overview of the six most prominent methods to generate virtual fixtures is also given in Section 1.2. Numerous methods to define the geometry of the virtual fixtures can be found in literature including point clouds and mesh grids. The paths for PFC and the surfaces for SFC can be defined by splines allowing for high flexibility to represent different geometries and, additionally, leading to a continuous control output. Three classifications of virtual fixtures, which can be realized with PFC and SFC, are discussed in the following and are experimentally validated in Section 5.2.

5.1.1 Guidance and forbidden region virtual fixtures

Virtual fixtures can either guide an operator to and along a submanifold of the workspace (guidance virtual fixture) or prevent the operator from entering specific areas of the workspace (forbidden region virtual fixture). Most of the existing methods to generate virtual fixtures enable only one of the two possibilities.

PFC and SFC ensure that the manipulator converges to and then remains on a path or surface. Compliance control in tangential direction with low stiffness and

damping enables the operator to easily move the manipulator along the path or surface. Hence, guidance virtual fixtures can be implemented using kinematic or dynamic PFC/SFC by simply adjusting the parameters in the position controllers and the impedance control laws (3.42) and (4.33), respectively.

In the case of forbidden region virtual fixtures, the operator is able to move the manipulator's end-effector freely inside the admissible region of the restricted workspace without any manifold stabilization. Various control concepts like Cartesian impedance control exist to generate such an unconstrained motion, see, e.g., [3, 26, 29, 32, 90]. The limits of the restricted workspace are defined by M parametrized surfaces $\mathcal{S}_{t,i}$, $i = 1, \dots, M$, and inside the admissible region, the shortest distance to each surface has to be computed. This can be achieved by solving a global optimization problem to obtain each optimal surface parameter vector $\boldsymbol{\theta}_i^*$, which is in general computationally quite expensive. SFC with the corresponding surface $\mathcal{S}_{t,i}$ gets activated to prevent from entering the forbidden region, when the manipulator's end-effector contacts $\mathcal{S}_{t,i}$ and the input force of the operator \mathbf{f}_{op} points into the forbidden region, implying that $\|\mathbf{y}_t - \boldsymbol{\sigma}_{t,i}(\boldsymbol{\theta}_i^*)\| < d_{fr}$ as well as $s_{fr,i} \mathbf{e}_{i,\perp}^T(\boldsymbol{\theta}_i^*)[\mathbf{I} \ \mathbf{0}]\mathbf{f}_{op} < 0$ holds, where $\boldsymbol{\sigma}_{t,i}$ is the position parametrization of the surface $\mathcal{S}_{t,i}$, $d_{fr} > 0$ is the distance threshold, \mathbf{I} is the 3×3 identity matrix, and $s_{fr,i} = 1$ when the normal vector $\mathbf{e}_{i,\perp}$ onto the surface $\mathcal{S}_{t,i}$ points into the admissible region and $s_{fr,i} = -1$ when $\mathbf{e}_{i,\perp}$ points into the forbidden region. The motion is then restricted by SFC to a tangential direction of the surface $\mathcal{S}_{t,i}$ as long as the operator's input force \mathbf{f}_{op} points into the forbidden region. Unconstrained motion is activated again once the input force of the operator points into the admissible region, i.e., $s_{fr,i} \mathbf{e}_{i,\perp}^T(\boldsymbol{\theta}_i^*)[\mathbf{I} \ \mathbf{0}]\mathbf{f}_{op} > f_{fr}$ holds, where $f_{fr} > 0$ is a small force threshold that prevents from chattering between the control laws.

Assume that the two surfaces $\mathcal{S}_{t,i}$ and $\mathcal{S}_{t,j}$ intersect in the curve $\gamma_{t,ij}$ as depicted in Fig. 5.1. Then, during active SFC with surface $\mathcal{S}_{t,i}$, $\mathbf{y}_t \in \mathcal{S}_{t,i}$ holds, and the shortest distance $d_{c,ij}$ to the intersection curve $\gamma_{t,ij}$ has to be computed, which can be done by solving the optimization problem (3.6) to compute $\boldsymbol{\theta}_{ij}^*$. Once the manipulator's end-effector contacts the intersection curve $\gamma_{t,ij}$, thus also the surface $\mathcal{S}_{t,j}$, and the operator's input force \mathbf{f}_{op} points into the forbidden region of the surface $\mathcal{S}_{t,j}$, i.e., $d_{c,ij} = \|\mathbf{y}_t - \boldsymbol{\sigma}_{t,j}(\boldsymbol{\theta}_j^*)\| < d_{fr}$ as well as $s_{fr,j} \mathbf{e}_{j,\perp}^T(\boldsymbol{\theta}_j^*)[\mathbf{I} \ \mathbf{0}]\mathbf{f}_{op} < 0$ holds, PFC along the intersection curve $\gamma_{t,ij}$ gets activated. With active PFC, the optimal parameter vectors $\boldsymbol{\theta}_i^*$ and $\boldsymbol{\theta}_j^*$ of the two intersecting surfaces have to be computed to be able to switch back to SFC. The transition back to SFC with surface $\mathcal{S}_{t,i}$ takes place when $s_{fr,j} \mathbf{e}_{j,\perp}^T(\boldsymbol{\theta}_j^*)[\mathbf{I} \ \mathbf{0}]\mathbf{f}_{op} > f_{fr}$ holds and the transition to SFC with surface $\mathcal{S}_{t,j}$ takes place when $s_{fr,i} \mathbf{e}_{i,\perp}^T(\boldsymbol{\theta}_i^*)[\mathbf{I} \ \mathbf{0}]\mathbf{f}_{op} > f_{fr}$ holds. The resulting state machine to generate forbidden region virtual fixtures with SFC and PFC is depicted in Fig. 5.2. Note that in the special case of three intersecting surfaces in one point, set-point stabilization has to be added as control state with similar transitions to PFC as described above.

It is worth noting that no stability proof is provided for the overall switched system including the state machine of Fig. 5.2. While this is not a big issue for

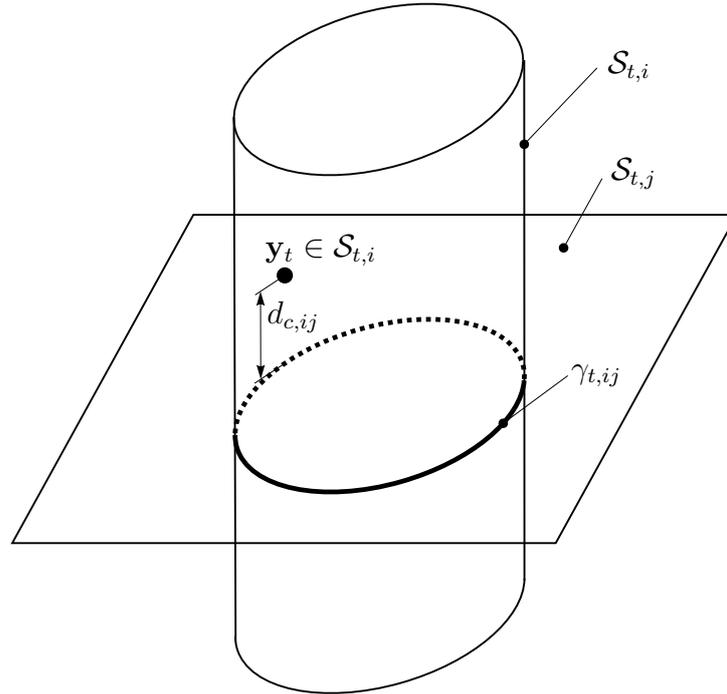


Figure 5.1: Intersection of two surfaces. [81] © 2018 IEEE

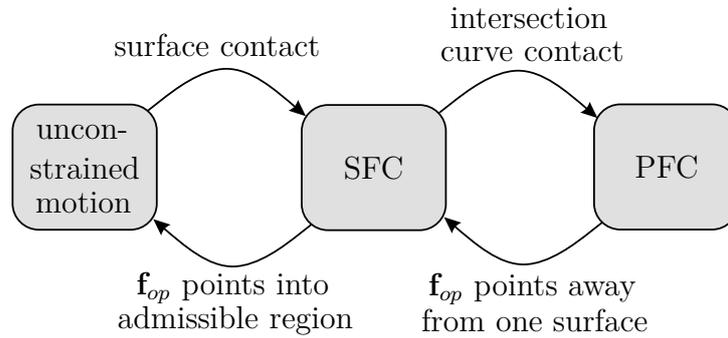


Figure 5.2: Forbidden region virtual fixtures state machine. [81] © 2018 IEEE

static forbidden region virtual fixtures, it has to be investigated in more detail for the dynamic case.

5.1.2 Hard and soft constraints

The behavior away from the virtual fixture defines the level of guidance. Hard constraints do not allow any motion off the virtual fixture (negligible deviations always occur in practice due to limited control gains), while soft constraints give some compliance to allow the operator little freedom to deviate from the fixture.

Hard constraints can be implemented with kinematic or dynamic PFC/SFC

by using the position controller (3.33), (3.52), (4.30), or (4.37) for the transversal directions with high gains.

Using the combination of compliance control with PFC or SFC enables to realize soft constraints. The stiffness away from the constraint can be adjusted with k_{\perp}^d . The impedance control laws (3.42) and (4.33) remain stable for a variable stiffness $k_{\perp}^d = k_{\perp}^d(\xi_i) > 0$, with $i = 1, 3$. Hence, also nonlinear virtual springs can be implemented. Note that for manipulators with very low (or compensated) friction, soft constraints can also be generated by using dynamic PFC/SFC together with the position controller (3.33) or (4.30) for the transversal directions with small gains.

5.1.3 Static and dynamic virtual fixtures

Normally, the virtual fixtures do not change over time and are static. However, in some applications the constraints have to be changed dynamically to adapt to a changing environment, e.g., in robot-assisted heart surgery [91].

Such dynamic virtual fixtures can also be implemented with PFC or SFC. Thereby, the path γ_t or surface \mathcal{S}_t remains constant, but the reference path or surface deviation is adapted corresponding to the dynamic virtual fixture. For PFC, the desired path deviation $\Delta\sigma_t(t) \in \mathcal{C}^2$ at $\sigma_t(\theta^*)$ is projected onto the normal vectors \mathbf{e}_{\perp} and \mathbf{e}_{th} leading to the transversal references

$$\xi_1^i = \mathbf{e}_{\perp}^T \Delta\sigma_t \quad \text{and} \quad \xi_3^i = \mathbf{e}_{\text{th}}^T \Delta\sigma_t, \quad (5.1)$$

where $i = p$ for hard constraints and $i = d$ for soft constraints. For SFC, there is only one transversal direction onto the surface \mathcal{S}_t and, hence, its reference is given by

$$\xi_1^i = \Delta\sigma_t, \quad (5.2)$$

with $i = p$ for hard constraints, $i = d$ for soft constraints, and the scalar desired surface deviation $\Delta\sigma_t(t) \in \mathcal{C}^2$.

A desired path deviation of class \mathcal{C}^2 implies a continuous output τ_d or $\dot{\mathbf{q}}_{\text{ref}}$ of the PFC/SFC feedback transformation. Note that the maximum deviation from a path or surface is limited by the feasible neighborhood, cf. Section 3.4 and Section 4.4.

5.1.4 Summary of the virtual fixture generation with PFC and SFC

A large number of virtual fixtures can be generated with PFC and SFC. Tab. 5.1 lists the types of guidance virtual fixtures that can be generated with either dynamic or kinematic PFC/SFC for a manipulator equipped with a sensor to measure the generalized forces exerted on the end-effector by an operator. If the

Table 5.1: Guidance virtual fixtures for manipulators with force sensor. [81] © 2018 IEEE

guidance virtual fixture type			control laws	
stat./dyn. manifold	guidance level	manifold type	dynamic PFC/SFC	kinematic PFC/SFC
static	hard	path	(3.29),(3.33),(3.42a)	(3.50),(3.52),(3.42a)
		surface	(4.26),(4.30),(4.33a)	(4.35),(4.37),(4.33a)
	soft	path	(3.29),(3.33),(3.42)	(3.50),(3.52),(3.42)
		surface	(4.26),(4.30),(4.33)	(4.35),(4.37),(4.33)
dynamic	hard	path	(3.29),(3.33) (3.42a),(5.1)	(3.50),(3.52) (3.42a),(5.1)
		surface	(4.26),(4.30) (4.33a),(5.2)	(4.35),(4.37) (4.33a),(5.2)
	soft	path	(3.29),(3.33) (3.42),(5.1)	(3.50),(3.52) (3.42),(5.1)
		surface	(4.26),(4.30) (4.33),(5.2)	(4.35),(4.37) (4.33),(5.2)

manipulator has very low (or compensated) friction, it is highly back-drive able and virtual fixtures can be generated using PFC/SFC without a force sensor, as listed in Tab. 5.2. In this case, kinematic PFC/SFC cannot be used due to the high gains of the velocity controllers, which disable the back-drive ability.

Compared to guidance virtual fixtures, the generation of forbidden region virtual fixtures with PFC/SFC requires more implementation and computational effort due to the switching between the control laws. With the method described in this work, only static forbidden region virtual fixtures with hard constraints can be generated, where the same control laws as listed in Tab. 5.1 or Tab. 5.2

Table 5.2: Guidance virtual fixtures for back-drive able manipulators. [81] © 2018 IEEE

guidance virtual fixture type			control laws
static/dynamic manifold	guidance level	manifold type	dynamic PFC/SFC
static	hard	path	(3.29),(3.33)
		surface	(4.26),(4.30)
	soft	path	(3.29),(3.33)
		surface	(4.26),(4.30)
dynamic	hard	path	(3.29),(3.33),(5.1)
		surface	(4.26),(4.30),(5.2)
	soft	path	(3.29),(3.33),(5.1)
		surface	(4.26),(4.30),(5.2)

for a static manifold and hard guidance level are used. Dynamic forbidden region virtual fixtures or soft constraints require different switching strategies, which are subject to further research.

5.2 Virtual fixture implementation examples

This section presents four implementation examples of virtual fixtures generated with PFC/SFC, which demonstrate several combinations of the virtual fixture classes and properties described in Section 5.1. For this, the experimental setup of Section 4.8.2 with the industrial robot COMAU Racer 1.4 is used. These examples comprise the following virtual fixtures:

Example 1: Static guidance virtual fixtures on a path with soft constraints

Example 2: Static guidance virtual fixtures on a path with hard constraints

Example 3: Dynamic guidance virtual fixtures on a path with hard constraints

Example 4: Static forbidden region virtual fixtures with hard constraints

PFC and SFC are implemented on the real-time system *DS1006* from dSPACE with a sampling time of $T_s = 1$ ms and the torque commands $\boldsymbol{\tau}_d$ are sent to joint servo controllers. When the kinematic PFC of Section 3.5.4 or the kinematic SFC of Section 4.5.4 are used, the reference joint velocities $\dot{\mathbf{q}}_{\text{ref}}$ are transferred to the servo controllers instead of the torques. A communication delay of approximately 2ms between the *DS1006* and the servo controllers limit the control gains in the dynamic PFC/SFC.

The operator does not have any visual feedback about the path/surface or the deviation from the path/surface during the experiments.

5.2.1 Example 1: Static guidance virtual fixtures on a path with soft constraints

Guidance virtual fixtures on a path γ with soft constraints are generated by a combination of the dynamic PFC (3.29) with compliance control (3.42) and inner position controller (3.86). The robot's end-effector is supposed to move on a horizontal circle with radius $r_{pc} = 0.15$ m and center $\mathbf{y}_{pc}^T = [0.83, 0, 0.9]$ m and the constant orientation $\boldsymbol{\sigma}_r^T = [0, \pi/4, \pi - 0.1]$ rad, cf. Fig. 5.3. The parametrization of the path is given by

$$\boldsymbol{\sigma}(\theta) = \begin{bmatrix} \boldsymbol{\sigma}_t \\ \boldsymbol{\sigma}_r \end{bmatrix} = \begin{bmatrix} y_{pc,x} + r_{pc} \cos \theta \\ y_{pc,y} + r_{pc} \sin \theta \\ y_{pc,z} \\ 0 \\ \pi/4 \\ \pi - 0.1 \end{bmatrix}. \quad (5.3)$$

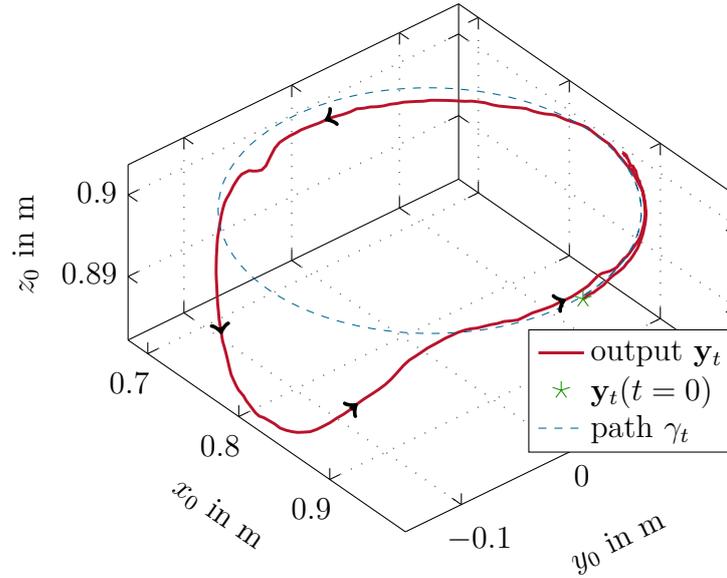


Figure 5.3: Path γ_t and output \mathbf{y}_t with soft guidance virtual fixtures (Example 1). [81] © 2018 IEEE

A low stiffness k_{\parallel}^d and damping d_{\parallel}^d in tangential direction enable the operator to move the robot along the circle without effort. The stiffness k_{\perp}^d and damping d_{\perp}^d in orthogonal direction are chosen rather high to limit the deviations from the path. The orientation is also made compliant with respect to σ_r by applying the control law (3.46) for the input $\mathbf{v}_{p,o}$. The control parameters for (3.86) and (3.46) are listed in Tab. 4.2 and the impedance parameters in Tab. 4.3.

In the experiment, the operator moves the robot along the circle for approximately $1\frac{1}{4}$ revolutions. In the middle of the experiment, the robot is pushed down in negative z_0 -direction to deviate considerably from the path, cf. Fig. 5.3.

Fig. 5.4(a) shows that the arc length on the circle η_1 follows the reference η_1^p , which is the output of the impedance control law (3.42a). The robot deviates from the circle corresponding to the external forces. This gets clear by comparing the transversal states ξ_1^p and ξ_3^p in Fig. 5.4(b) with the external forces τ_{\perp} and τ_{th} in Fig. 5.4(c). The deviations of ξ_i from ξ_i^p , $i = 1, 3$, mainly occur at joint velocity zero crossings and are caused by uncompensated friction effects.

The rotational coordinates, the external torques as well as the first three joint torques are depicted in Fig. 5.4(d), Fig. 5.4(e), and Fig. 5.4(f), respectively. The Euler angles φ_{pd} , ϑ_{pd} , and ψ_{pd} represent the desired deviation from the reference orientation σ_r , which is caused by the external torques μ_{φ}^d , μ_{ϑ}^d , and μ_{ψ}^d . Again, errors between the actual deviation φ_{ed} , ϑ_{ed} , and ψ_{ed} and the reference deviation φ_{pd} , ϑ_{pd} , and ψ_{pd} mainly occur at joint velocity zero crossings.

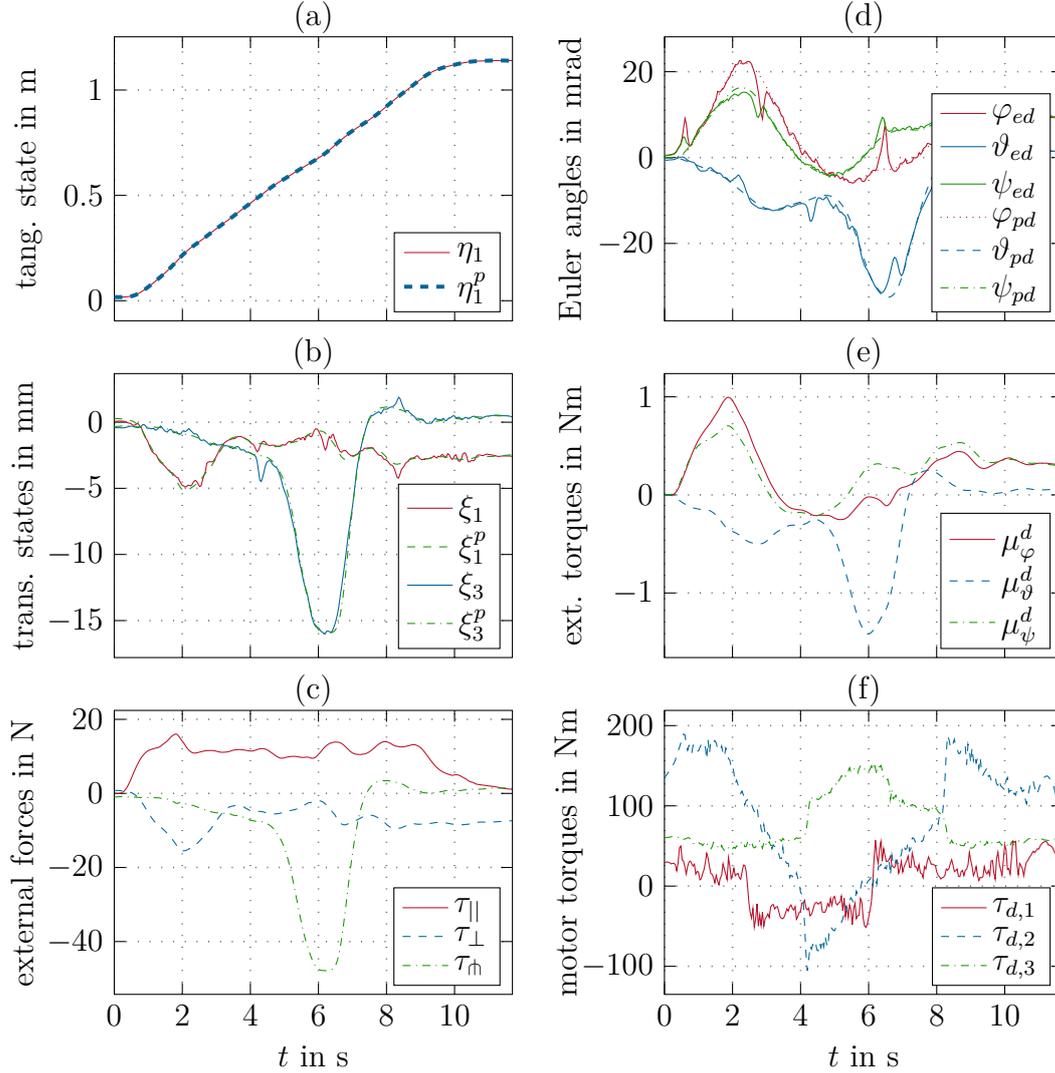


Figure 5.4: Experimental results for soft guidance virtual fixtures (Example 1). [81] © 2018 IEEE

5.2.2 Example 2: Static guidance virtual fixtures on a path with hard constraints

In this experiment, the kinematic PFC (3.50) with the position controller (3.52) and the analytic Jacobian \mathbf{J}_a (2.16) in (3.22) is used to restrict the robot's motion to a path. Additionally, the impedance control (3.42a) for the tangential direction, with parameters m_{\parallel}^d , d_{\parallel}^d , and k_{\parallel}^d from Tab. 4.3, allows the operator to move the robot along the path without effort. Hard constraints are ensured by setting the references for the transversal states ξ_1^p and ξ_3^p as well as their derivatives to zero. The orientation is also made stiff by setting the reference to $(\mathbf{y}_r^p)^T = \boldsymbol{\sigma}_r^T(\theta^*) = [\varphi_p, \vartheta_p, \psi_p]$. Tab. 5.3 lists the control parameters for the

Table 5.3: Control parameters for the kinematic PFC in Example 2.

Symbol	Value	Unit	Symbol	Value	Unit
a_η	10	1/s	a_ξ	15	1/s
a_r	12	1/s			

kinematic PFC. The path γ is defined by cubic splines and its position part γ_t is depicted in Fig. 5.5. The operator moves the robot from the beginning of the spline path until the end and then backwards again, cf. Fig. 5.6(a). During the backwards motion, a considerable force is applied in normal direction to the path, which can be seen in Fig. 5.6(c). Fig. 5.6(b) shows that the robot remains on the path with deviations of less than 0.4mm. The orientation of the end-effector $\mathbf{y}_r^T = [\varphi_e, \vartheta_e, \psi_e]$ follows the reference orientation $\boldsymbol{\sigma}_r(\theta^*)$ without noticeable errors even with applied external torques, cf. Fig. 5.6(d).

5.2.3 Example 3: Dynamic guidance virtual fixtures on a path with hard constraints

The same control law of Section 5.2.2 and path of Section 5.2.1 with constant orientation $\boldsymbol{\sigma}_r^T = [0, \pi/4, \pi - 0.1]$ rad are employed in this experiment, but now, the radius of the circle is a function of time and given by $r_p(t) = r_{pc} + a_{rp} \sin(2\pi f_{rp}t)$, with $a_{rp} = 20$ mm and $f_{rp} = 0.5$ Hz. Hence, the radius of the circle constraint oscillates between $r_{p,\min} = 0.13$ m and $r_{p,\max} = 0.17$ m with a period of 2s and

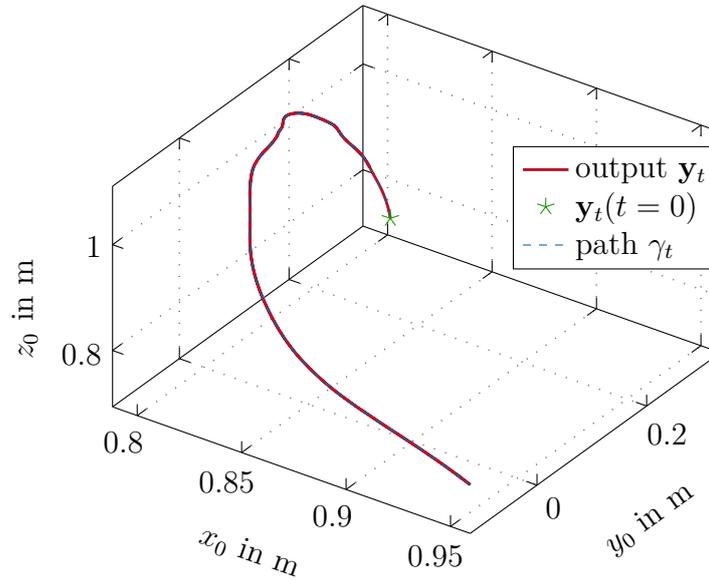


Figure 5.5: Path γ_t and output \mathbf{y}_t with hard guidance virtual fixtures (Example 2). [81] © 2018 IEEE

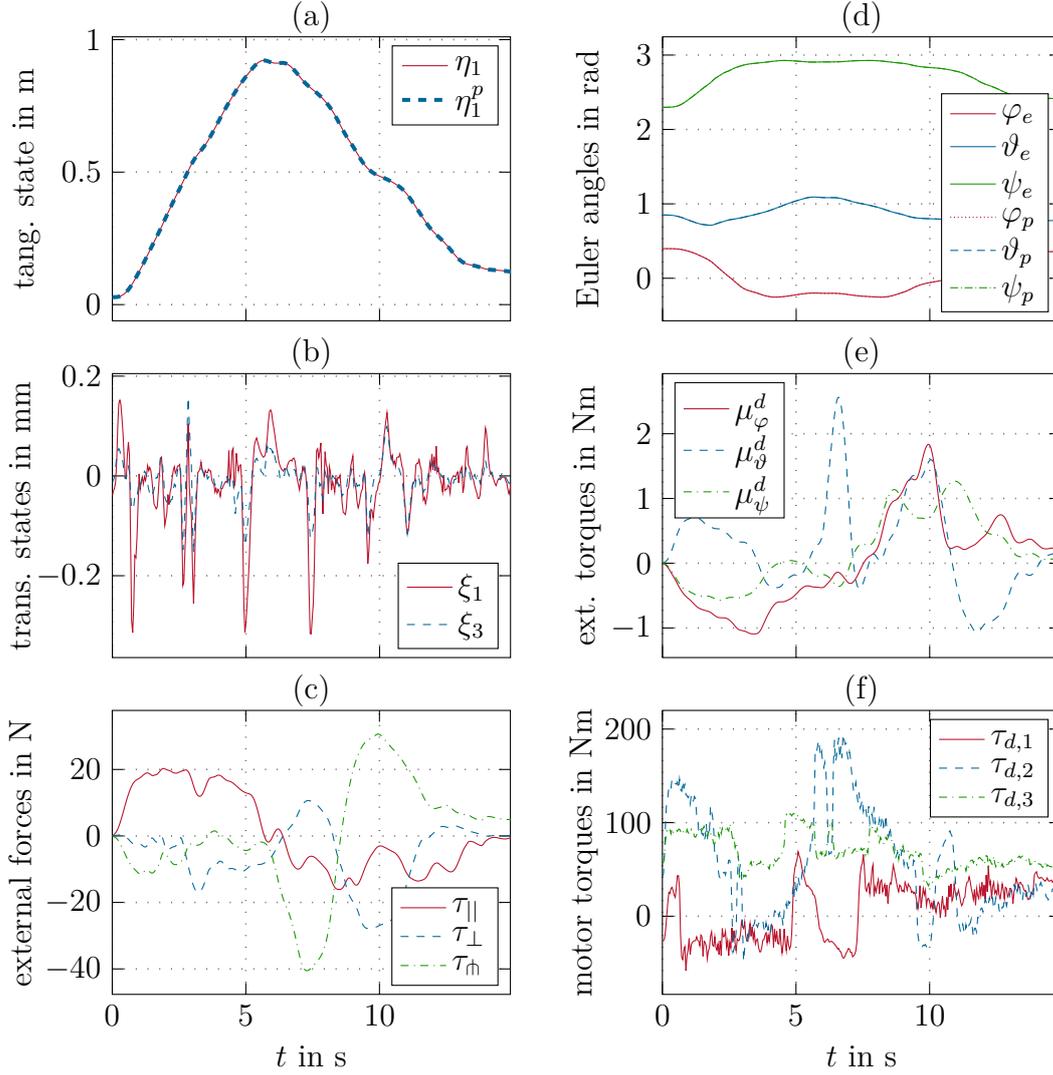


Figure 5.6: Experimental results for hard guidance virtual fixtures (Example 2). [81] © 2018 IEEE

$\Delta\sigma_t^T(t) = a_{rp} \sin(2\pi f_{rp}t) [\cos(\theta), \sin(\theta), 0]$, see (5.3). The impedance control (3.42a) for the tangential direction, with parameters from Tab. 4.3, allows the operator to move the robot along the path without effort.

Hard constraints and dynamic virtual fixtures are implemented by setting the references for the transversal states to $\xi_1^p = \mathbf{e}_\perp^T \Delta\sigma_t = a_{rp} \sin(2\pi f_{rp}t)$ and $\xi_3^p = \mathbf{e}_\uparrow^T \Delta\sigma_t = 0$, because \mathbf{e}_\perp is pointing in radial direction of the circle and \mathbf{e}_\uparrow is pointing in direction of the z_0 -axis. The PFC control parameters are listed in Tab. 5.3.

In this experiment, the operator moves the robot along the oscillating circle for almost one revolution. The velocity of the robot along the circle is increased

during the experiment to show the time dependency of the virtual fixture. The resulting output trajectory \mathbf{y}_t is depicted in Fig. 5.7, together with the circles with the minimum and maximum radius, $r_{p,\min}$ and $r_{p,\max}$.

Fig. 5.8 shows the tangential, transversal states, the external forces, and the torques of the first three motors. The transversal reference ξ_1^p oscillates at 0.5Hz with an amplitude of 20mm and the state ξ_1 follows without noticeable errors, cf. Fig. 5.8(b).

5.2.4 Example 4: Static forbidden region virtual fixtures with hard constraints

Forbidden region virtual fixtures with hard constraints are implemented in this experiment. The robot's motion is restricted to a cylinder with radius $r_{cy} = 0.2\text{m}$, height $h_{cy} = 0.4\text{m}$, and vertical rotation axis with the coordinates $x_{ra} = 0.9\text{m}$ and $y_{ra} = 0$. Additionally, a paraboloid of revolution with height $h_{pr} = 0.2\text{m}$, radius $r_{pr} = 0.076\text{m}$, and rotation axis coordinates $x_{pr} = 0.95\text{m}$ and $y_{pr} = 0$ further restricts the workspace inside the cylinder, cf. Fig. 5.9. A Cartesian position-based impedance control [29, 32] (often denoted as Cartesian admittance control) is used to enable an unconstrained movement of the manipulator inside the admissible region of the restricted workspace. Thereby, the parameters m_{\parallel}^d , k_{\parallel}^d , and d_{\parallel}^d from Tab. 4.3 are used.

Once a limiting surface is reached, kinematic SFC (4.35) with position control

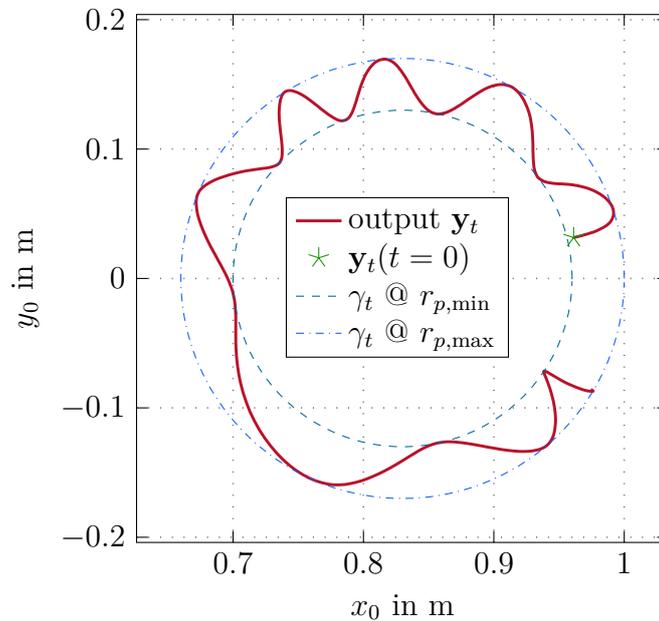


Figure 5.7: Path $\gamma_t @ r_{p,\min} = 0.13\text{m}$ and $r_{p,\max} = 0.17\text{m}$ and output \mathbf{y}_t with dynamic guidance virtual fixtures (Example 3). [81] © 2018 IEEE

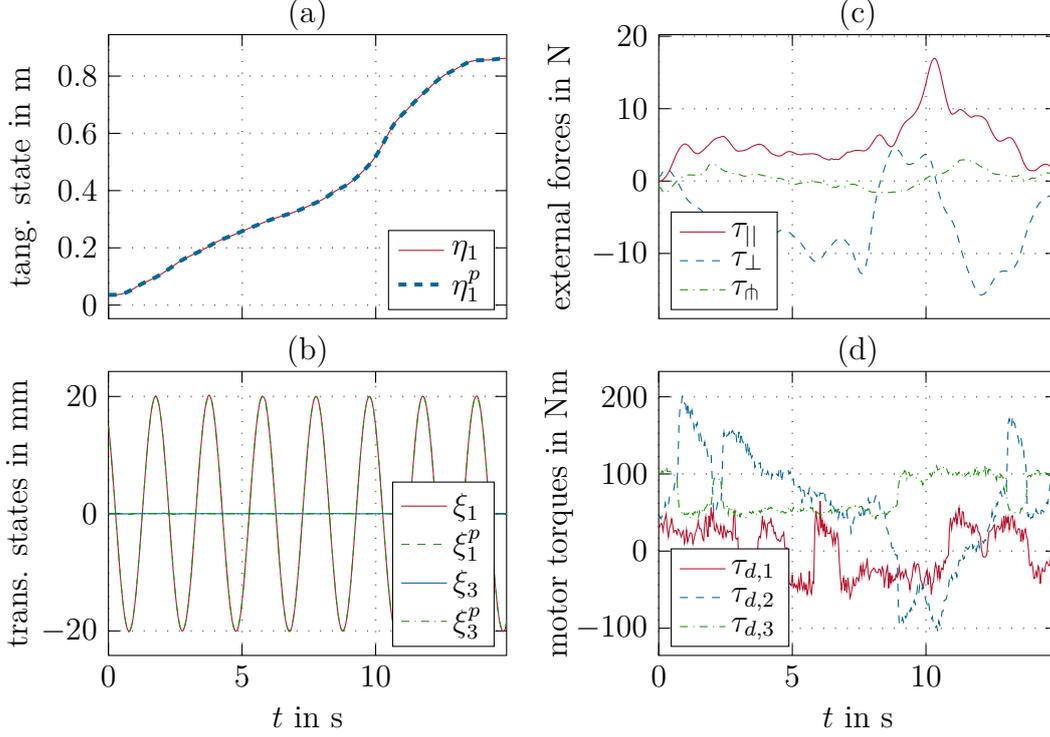


Figure 5.8: Experimental results for hard guidance dynamic virtual fixtures (Example 3). [81] © 2018 IEEE

(4.37) and $\xi_1^p \equiv 0$ is activated to prevent the robot from entering the forbidden region. The impedance control law (4.33a) is used to compute the two elements of the reference $\boldsymbol{\eta}_1^p$ with parameters from Tab. 4.3.

Kinematic PFC (3.50) with the controller (3.52) becomes active when the robot reaches the intersection of the two surfaces, i.e., a plane intersects the cylinder or the paraboloid of revolution, hence, the path is a circle with radius r_{cy} or r_{pr} . Additionally, the impedance control (3.42a) for the tangential direction, with parameters from Tab. 4.3, allows the operator to move the robot along the path without effort. Admittance control (AdmC) is activated again once the intended motion from the operator points away from the forbidden region.

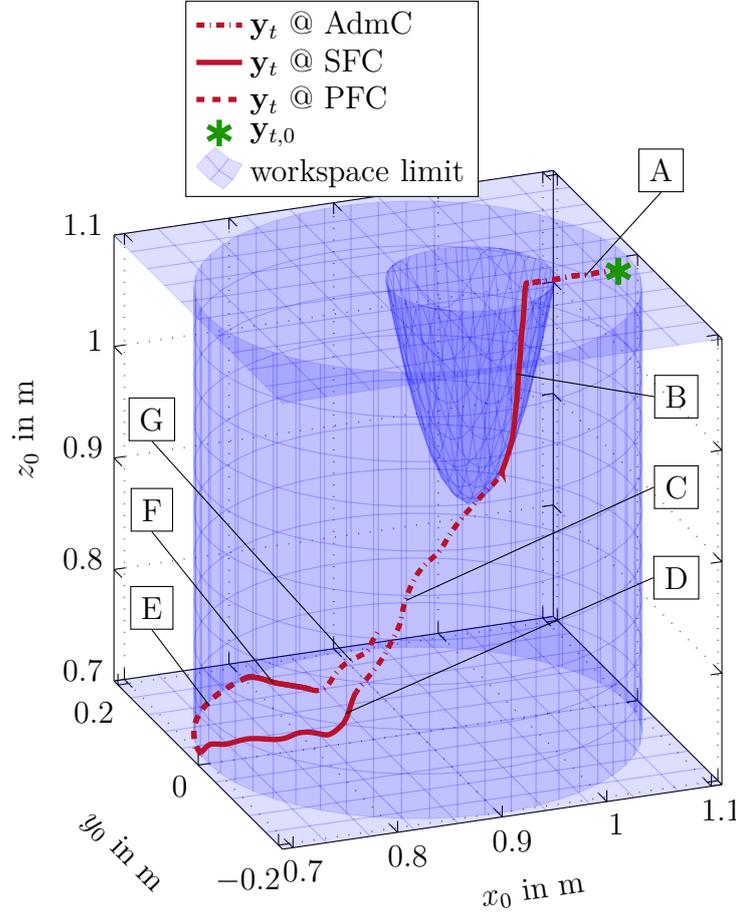
Continuous tangential velocities at the transitions are ensured by setting the initial tangential references to $\dot{\boldsymbol{\eta}}_1^p(t_{\text{switch}}) = [\mathbf{I} \ \mathbf{0}] \hat{\mathbf{J}}_s \dot{\mathbf{q}}(t_{\text{switch}})$ and $\dot{\eta}_1^p(t_{\text{switch}}) = [1, \mathbf{0}] \hat{\mathbf{J}}_p \dot{\mathbf{q}}(t_{\text{switch}})$, respectively. The analytic Jacobian \mathbf{J}_a is used for the PFC (3.50) and SFC (4.35). Tab. 5.4 lists the control parameters.

In the experiment, the robot starts inside the admissible region at point $\mathbf{y}_{t,0}^T = [1.094, 0.02, 1.077]\text{m}$, which is marked with an asterisk in Fig. 5.9.

AdmC is activated and the operator moves the robot towards the paraboloid of revolution (path A). Once the paraboloid of revolution is reached, SFC gets enabled to prevent the robot from entering the forbidden region (path B). AdmC

Table 5.4: Control parameters for the kinematic PFC/SFC in Example 4.

Symbol	Value	Unit	Symbol	Value	Unit
a_η	5	1/s	a_ξ	5	1/s
a_r	12	1/s			

Figure 5.9: Forbidden region and measured output \mathbf{y}_t (Example 4). [81] © 2018 IEEE

gets active again once the input force of the operator points away from the surface of the paraboloid of revolution. Now, the operator moves the robot towards the cylinder (path C). SFC gets enabled again when the surface of the cylinder is reached and the operator moves the robot along the cylinder towards the lower vertical limit $z_{0,\min} = 0.7\text{m}$ (path D). When the lower vertical limit is reached, PFC on a circle with radius r_{cy} gets active because the operator's intended motion points in negative z_0 direction and to the outside of the cylinder (path E). The operator's input force then points into the inside of the cylinder and SFC on the xy -plane with $z_{0,\min}$ is enabled (path F). On the last path segment, the robot is

moved in positive z_0 -direction with AdmC (path G).

Fig. 5.10 shows the tangential states, the transversal state, the external forces, and the first three motor torques of the motion along the cylinder with SFC (path D). The first tangential state $\eta_{1,1}$ represents the arc length of the trajectory projected to the xy -plane and the second tangential state $\eta_{1,2}$ represents the z -component of the trajectory. The transversal state ξ_1 describes the deviation from the cylinder, where negative values are a penetration of the forbidden region. Hence, the penetration is less than 0.2mm, cf. Fig. 5.10(b).

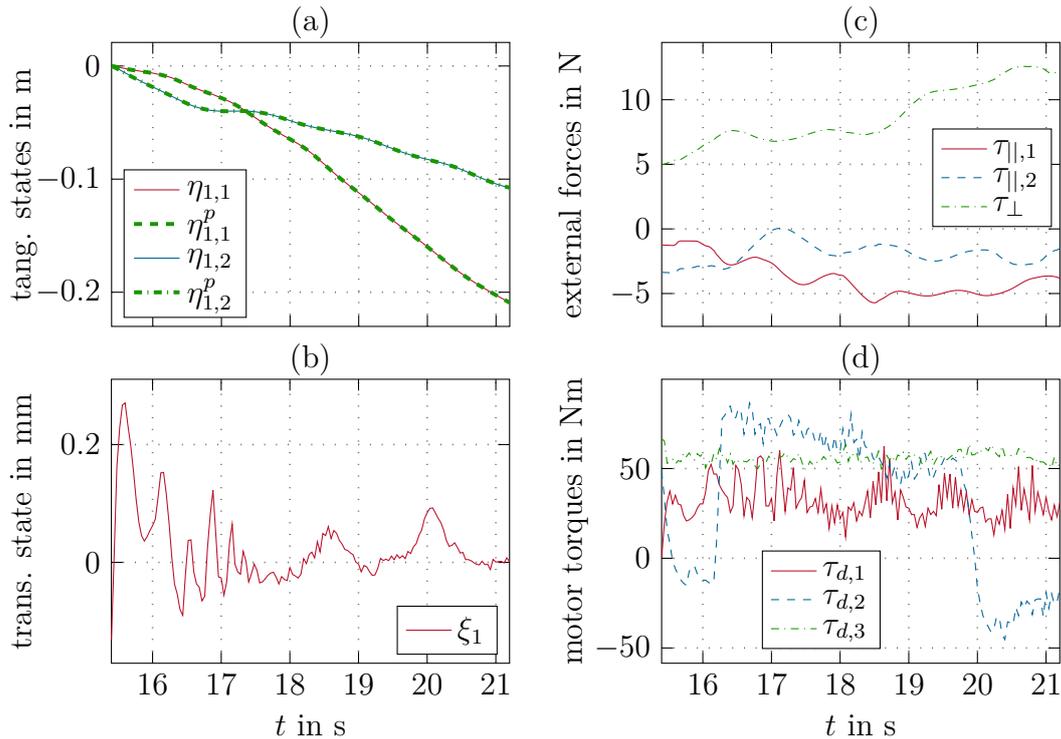


Figure 5.10: Experimental results for forbidden region virtual fixtures for the path segment D in Fig. 5.9 (Example 4). [81] © 2018 IEEE

5.3 Semi-automation production use case

Handling of a plate using human-robot collaboration is chosen as semi-automation production use case. Guidance virtual fixtures lead a human operator along the path γ_1 to handle a plate from a magazine to a frame, cf. Fig. 5.11, and along the path γ_2 to move the robot's end-effector back to the magazine. The mounting pins of the plate have to be inserted into the boreholes of the frame, of which the diameter is only 0.3mm larger than the diameter of the pins. In the test setup, the positions and orientations of the magazine and the frame do not exactly match

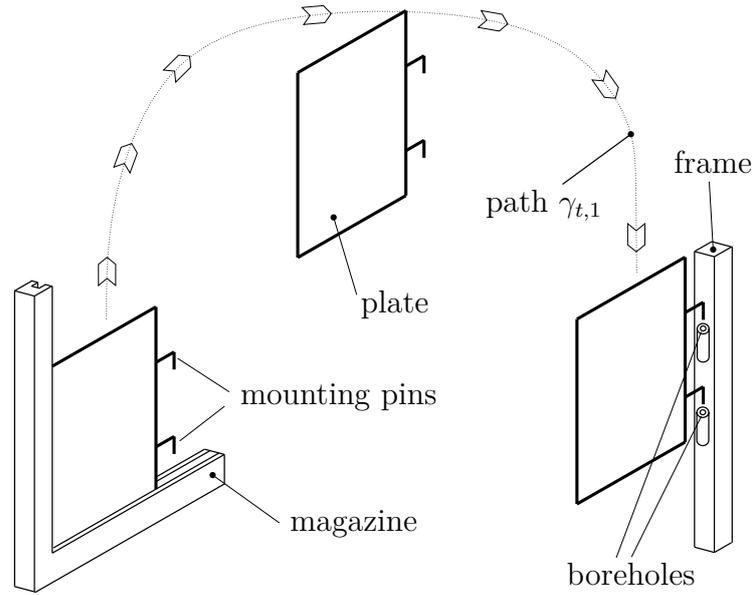


Figure 5.11: Plate mounting using guidance virtual fixtures on a path.

with the path and, hence, the operator has to deviate from the path to fulfill the task. This is achieved by using virtual fixtures with soft constraints.

The virtual fixtures are generated by a combination of the kinematic PFC (3.50), where the geometric Jacobian \mathbf{J}_g (2.11) in (3.22) is used, with compliance control (3.42a), (3.42b), (3.42c), and (3.52). The control law (3.46) stabilizes the orientation with the impedance of (3.44). The demonstration system is the same as described in Section 4.8.2, which includes the industrial robot COMAU Racer 1.4, the 6D-force/torque sensor *K6-D40* as input device, and the real-time system *DS1006* from dSPACE. A gripper equipped with four vacuum suction cups is attached to the robot's end-effector to grasp the plate, see Fig. 5.12. The force/torque sensor is attached in series to the end-effector. This implies that also the contact forces with the environment are measured. Contact occurs when the plate in the magazine is gripped and when the plate pins are inserted into the boreholes. In these cases, the compliance control limits the contact forces to prevent damages and the damping parameters d_i^d of (3.42) and (3.46), for $i = \{\parallel, \perp, r\}$, have to be high enough to ensure contact stability, see, [88].

Two paths are necessary for the handling task. The first path γ_1 starts with taking the plate from the magazine and ends with assembling the pins of the plate into the boreholes of the frame. The second path γ_2 starts with releasing the plate at the frame and ends at the magazine to grip the next plate, cf. Fig. 5.12. Both paths are recorded during a teach-in procedure, where the robot is freely movable in space via admittance control. The paths are stored as cubic splines for the kinematic PFC.

The damping in tangential direction to the path d_{\parallel}^d limits the maximum

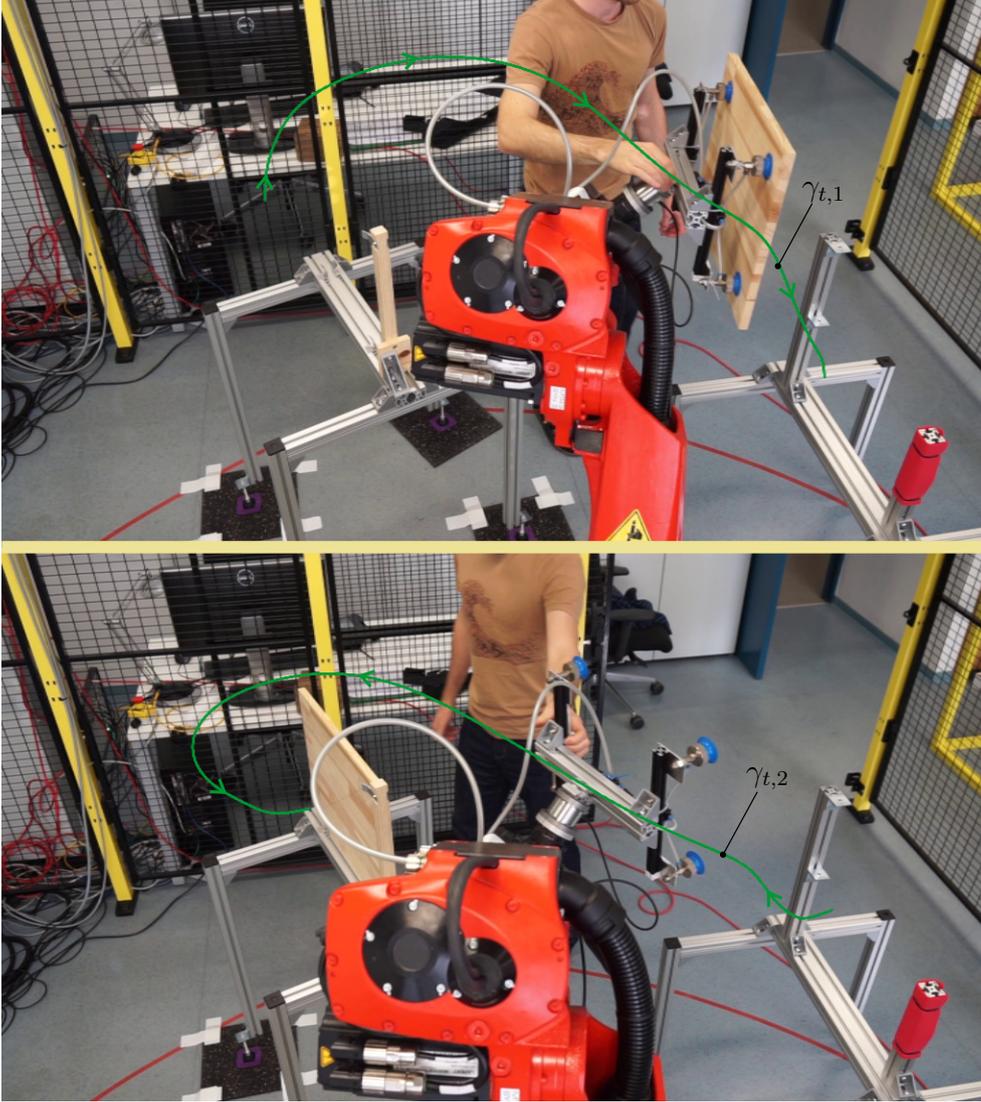


Figure 5.12: Semi-automation production use case: plate mounting.

velocity of η_1 . In proximity of a stiff environment, the robot has to move relatively slow to limit the forces in case of a contact and the damping should therefore be quite high, which is also essential to ensure contact stability. On the other hand, a higher speed is preferable in the free space between the magazine and the frame to reduce the cycle time of the production step. Hence, the variable damping in tangential direction is defined as the smooth function

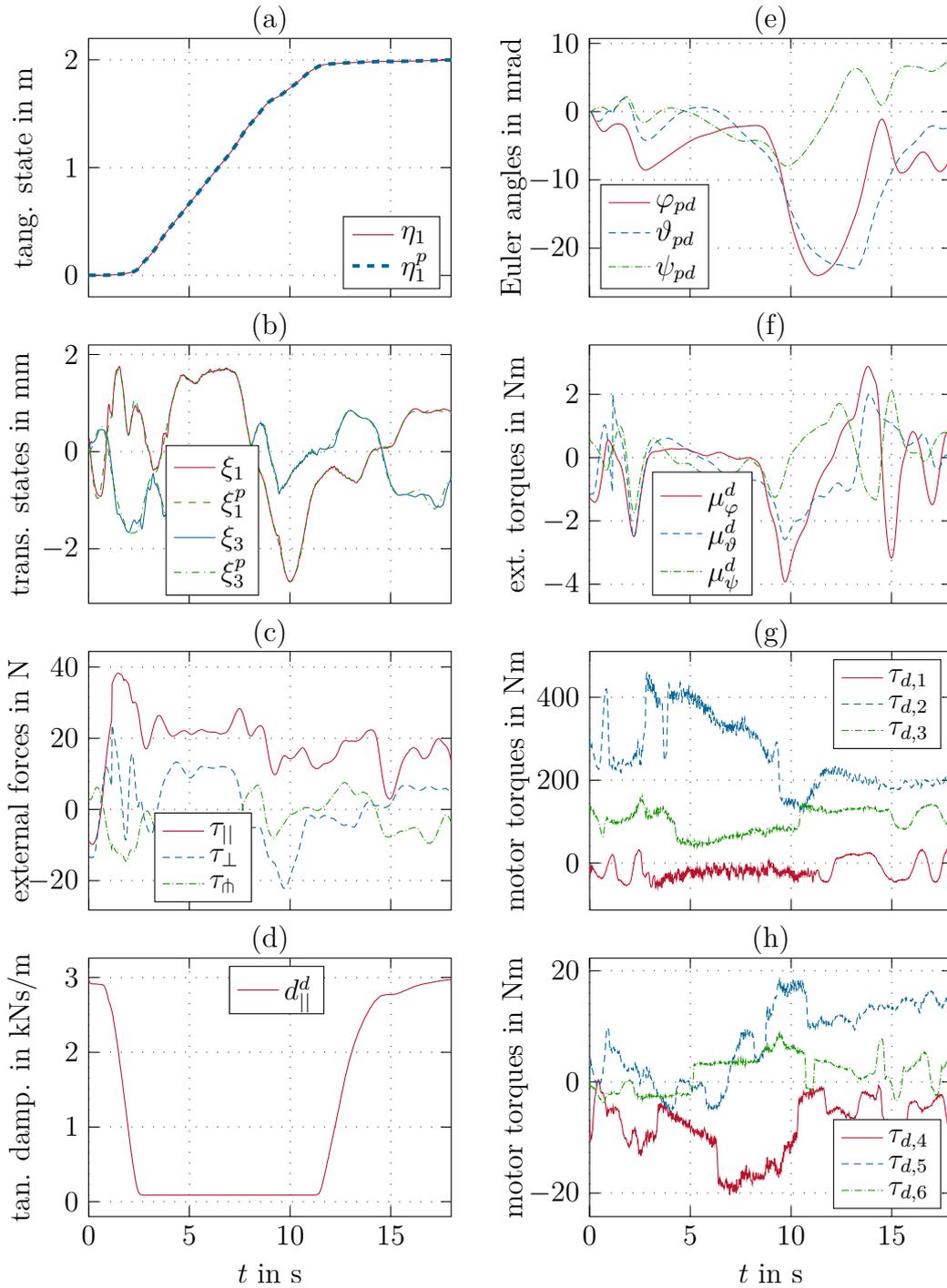
$$d_{||}^d(\eta_1) = d_{||,\min} + (d_{||,\max} - d_{||,\min}) \left(3 - 2 \tanh(40\eta_1) + \tanh(80(\eta_1 - \eta_{1,\max})) \right), \quad (5.4)$$

where $\eta_{1,\max}$ denotes the total length of the path $\gamma_{t,1}$ or $\gamma_{t,2}$, respectively. The control parameters for the kinematic PFC are listed in Tab. 5.3 and the compliance control parameters in Tab. 5.5.

Table 5.5: Compliance control parameters for plate assembly.

Symbol	Value	Unit	Symbol	Value	Unit
$m_{ }^d$	3	kg	m_{\perp}^d	3	kg
$d_{ ,\min}$	90	Ns/m	d_{\perp}^d	3	kNs/m
$d_{ ,\max}$	3	kNs/m	k_{\perp}^d	7	kN/m
$k_{ }^d$	1	N/m	m_r^d	0.2	kgm ²
d_r^d	229	Nms/rad	k_r^d	34.4	Nm/rad

Fig. 5.13 shows trajectories of the motion along the path γ_1 from the grasping of the plate until the release of the plate after assembling the mounting pins into the boreholes of the frame. The tangential states η_1 and η_1^p are depicted in Fig. 5.13(a). The frame is reached after approximately 12s and six more seconds are required for the assembling of the mounting pins. The deviation from the path $\gamma_{t,1}$ at the pin insertion adds up to about 1.1mm, which can be seen in Fig. 5.13(b). Fig. 5.13(c) shows that the input forces of the operator stay below 45N during the handling task. The trajectory of the damping in tangential direction $d_{||}^d$ is depicted in Fig. 5.13(d), which is computed by (5.4). The orientation of the plate when inserting the mounting pins into the boreholes has also to be adapted for approximately 13mrad, cf. Fig. 5.13(e). The motor torques $\tau_{d,i}$ are depicted in Fig. 5.13(g) and Fig. 5.13(h).

Figure 5.13: Experimental results for plate assembly on path γ_1 .

CHAPTER 6

Conclusions and Outlook

The goals of this work were to develop a simple and efficient manifold stabilization approach for industrial robots, to combine it with compliance control, and to show possible applications with a focus on physical human-robot interaction. The manifolds can constitute either a one-dimensional path or a two-dimensional surface defined by a regular parametrization in the output space of the robot. This allows to approximate arbitrary paths and surfaces by splines giving high flexibility for the geometric shapes. Moreover, feasible neighborhoods of the manifolds are defined for which a diffeomorphism can be found that maps the generalized coordinates (joint coordinates) to tangential, transversal, and rotational coordinates, and which allows to systematically prove the closed-loop stability. The presented path following control (PFC) and surface following control (SFC) approaches decouple and exactly linearize the dynamics of the orientation and in tangential and transversal direction to a path or a surface. The decoupled dynamics can be controlled independently to meet the requirements of the specific application, e.g., to introduce a well defined impedance in tangential direction to a path or surface. The manifold stabilization approaches are formulated for fully actuated manipulators as well as for robots with elastic joints. The combination of PFC and SFC with compliance control is used to systematically generate numerous different types of virtual fixtures for physical human-robot interaction, which include guidance and forbidden region virtual fixtures, soft and hard constraints as well as static and dynamic virtual fixtures and their combinations. The proposed approaches are validated by simulation studies in MATLAB[®]/Simulink[®] and by experiments on the DELTA robot FESTO EXPT-45 and on the six-axis industrial robot COMAU Racer 1.4.

6.1 Conclusions

The novel PFC approach was introduced in Chapter 3, which can handle open, closed, and intersecting paths parametrized as regular continuously differentiable curves. In contrast to, e.g., [22], the parametrization was split into the position and orientation parametrization and the transversal feedback linearization was only applied to the position parametrization. Application of the parallel transport frame in the PFC approach not only allows to handle paths with zero curvature, but also drastically simplifies the path following control law compared to, e.g., the Frenet-Serret frame, which is typically used in literature. Additionally, due to the structure of the proposed PFC concept, this strategy is able to be combined with compliance control with respect to the path.

The PFC approach was formulated for fully actuated manipulators as well as for elastic joint robots. The elastic joint robots can contain either visco-elastic or pure elastic transmissions between the rotors of the actuators and the links. For fully actuated manipulators, also a simplified control concept was proposed, where the subordinate joint velocity controllers are assumed to ideally track the velocity reference. The feedback transformation for pure elastic joint robots contains the second order time derivative of the mass matrix and the Coriolis matrix and becomes very extensive for robots with complex dynamics. To simplify the control law, the PFC was also developed for a reduced order model based on the singular perturbation theory, which is applicable to robots with rather stiff transmissions and exhibits the same structure and complexity as the PFC for fully actuated manipulators.

For experimental validation, the proposed PFC approach for fully actuated manipulators was applied to the DELTA robot FESTO EXPT-45. These experiments include standard path following control in free space as well as the combination with compliance control. It was shown that the combination of PFC with compliance control enables applications in which the classical trajectory tracking control fails. The PFC approaches for elastic joint robots were validated by simulation studies in MATLAB[®]/Simulink[®]. One simulation example also demonstrated the limits of the singular perturbation based approach regarding the stiffness of the joints. Another simulation example pointed out the advantages of the PFC compared to classical trajectory tracking control, when external disturbances prevent the robot from following the reference trajectory with the predefined speed.

Chapter 4 is devoted to the novel SFC approach, which can handle open, closed, and intersecting surfaces represented by a regular and continuously differentiable parametrization. This is in contrast to the existing approaches known from literature, see, e.g., [12–14], which require the implicit representation of the surface. Again, the parametrization was split into the position and orientation parametrization and the transversal feedback linearization was only applied to the position parametrization. The feedback linearization ensures that the nonlinear

system dynamics transform into a decoupled linear system comprising two orthogonal tangential states and one transversal state with a physically interpretable length with respect to the surface. This enables the combination with compliance control that ensures a well defined compliance in tangential as well as in orthogonal direction to the surface. The SFC approach was formulated for fully actuated manipulators as well as for elastic joint robots. A simplified control concept for fully actuated manipulators was again proposed, where the subordinate joint velocity controllers were assumed to be ideal. For experimental validation, the proposed SFC approach for fully actuated manipulators was applied to the DELTA robot FESTO EXPT-45, where the robot was stabilized on a cylinder and to the six-axis industrial robot COMAU Racer 1.4. Simulation studies in MATLAB[®]/Simulink[®] were used to validate the SFC approach for elastic joint robots.

The application of the PFC and SFC approaches to physical human-robot interaction, where an operator is hand-guiding a robot, was presented in Chapter 5. The PFC and SFC approaches introduced in this work decouple and exactly linearize the dynamics for the orientation and in tangential and transversal direction to a path or surface. Therefore, these methods can be used to systematically generate numerous different constraint types like guidance and forbidden region virtual fixtures, hard and soft constraints as well as static and dynamic virtual fixtures and their combinations. This is in contrast to the existing approaches known from literature, which usually cover only a few different virtual fixture types. Guidance virtual fixtures with hard or soft constraints can simply be generated in a straightforward manner with PFC or SFC. Forbidden region virtual fixtures require more implementation and computational effort because the distance to each surface in the unconstrained motion state and to each intersection path in the SFC state has to be calculated in every sampling instance. Four experiments on the six-axis industrial robot COMAU Racer 1.4 were performed to show the ability of the PFC/SFC approach to generate a variety of virtual fixtures. Finally, a semi-automation production use case was presented. This use case shows that the virtual fixture approach can effectively be used to solve handling tasks, where the strengths of the human operator and the robot are synergetically utilized.

6.2 Outlook

The presented manifold stabilization approach is applicable to most of the existing industrial robots. However, the class of redundant manipulators is not covered yet. In recent years, lightweight robots especially designed for physical human-robot interaction are getting more popular. Some of these robots are realized with redundancies and feature seven actuated joints like the KUKA LBR iiwa. The redundancy resolution methods described in, e.g., [22, 85, 92] can be used to extend the proposed manifold stabilization approach to redundant manipulators.

A combination of PFC with compliance control is used in [88, 93] to manipulate

highly deformable materials like fiberglass strips. These applications are restricted to the two-dimensional space. The manifold stabilization approaches presented in this work are formulated in three-dimensional space, which enables to solve more complex applications in which highly deformable materials are handled.

The generation of forbidden region virtual fixtures with PFC/SFC requires the computation of the distance to each surface in the unconstrained motion state. In this work, a global optimization problem is solved to obtain each optimal surface parameter vector, which is in general computationally quite expensive. However, there is no need to compute the distance to each surface in each sampling instance due to the slow motion of the robot compared to the sampling time. The time interval between the computation of the distance to a specific surface could additionally depend on the value of the distance. Hence, there is a lot of room to further decrease the computational costs. In addition, further research could deal with dynamic forbidden region virtual fixtures and forbidden region virtual fixtures with soft constraints, where different switching strategies than the ones proposed in Section 5.1.1 have to be found.

This work considers hand-guiding physical human-robot interaction, while virtual fixtures are also common in teleoperation, where the tool-carrying slave manipulator copies the motion of a master device that is controlled by the human operator. The application of the PFC/SFC approach to generate virtual fixtures for teleoperation is also an open topic of research.

APPENDIX A

Friction Observer

In the robot dynamics (2.42), the second time derivative of the generalized coordinates \mathbf{q} appears. For external force estimation, De Luca showed that the measurement or estimation of $\ddot{\mathbf{q}}$ can be avoided by using the generalized momenta $\mathbf{p} = \mathbf{D}(\mathbf{q})\dot{\mathbf{q}}$ [94]. This technique can also be applied to design a friction observer.

The dynamics (2.42) with $\boldsymbol{\tau}_{ext} = \mathbf{0}$, where friction $\boldsymbol{\tau}_f(\dot{\mathbf{q}}) \in \mathbb{R}^m$ is modeled by (2.41), can be written as

$$\begin{aligned} \mathbf{D}(\mathbf{q})\ddot{\mathbf{q}} + \underbrace{\left(\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{C}^T(\mathbf{q}, \dot{\mathbf{q}})\right)}_{\dot{\mathbf{D}}(\mathbf{q}, \dot{\mathbf{q}}), \text{ see (2.40)}} \dot{\mathbf{q}} &= \dot{\mathbf{p}} \\ &= \boldsymbol{\tau} - \boldsymbol{\tau}_f - \mathbf{g}(\mathbf{q}) + \mathbf{C}^T(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}}. \end{aligned} \quad (\text{A.1})$$

The friction parameters \mathbf{f}_c and \mathbf{f}_v according to (2.41) are assumed to be unknown but constant, which is why the disturbance model $\dot{\mathbf{f}}_c = \mathbf{0}$ and $\dot{\mathbf{f}}_v = \mathbf{0}$ is added to the system (A.1). Since $\mathbf{q}(t)$ and $\dot{\mathbf{q}}(t)$ are considered as known functions of time t , the following linear time-varying system

$$\begin{aligned} \dot{\mathbf{x}}_o &= \underbrace{\begin{bmatrix} \mathbf{0} & -\text{diag}(\text{sign}(\dot{\mathbf{q}})) & -\text{diag}(\dot{\mathbf{q}}) \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}}_{\mathbf{A}(t)} \mathbf{x}_o + \\ &\underbrace{\begin{bmatrix} \mathbf{I} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}}_{\mathbf{B}} \underbrace{\left(\boldsymbol{\tau} - \mathbf{g}(\mathbf{q}) + \mathbf{C}^T(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}}\right)}_{\mathbf{u}}, \end{aligned} \quad (\text{A.2a})$$

with state $\mathbf{x}_o^T = [\mathbf{p}^T \mathbf{f}_c^T \mathbf{f}_v^T]$ and the output function

$$\mathbf{p} = \underbrace{[\mathbf{I} \ \mathbf{0} \ \mathbf{0}]}_{\mathbf{C}_o} \mathbf{x}_o \quad (\text{A.2b})$$

serves as the basis for the observer design. System (A.2) contains six decoupled subsystems representing each axis $i = 1, \dots, 6$

$$\dot{\mathbf{x}}_{o,i} = \underbrace{\begin{bmatrix} 0 & -\text{sign}(\dot{q}_i) & -\dot{q}_i \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}}_{\mathbf{A}_i(t)} \mathbf{x}_{o,i} + \underbrace{\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}}_{\mathbf{b}_i} u_i, \quad (\text{A.3a})$$

with the output function

$$p_i = \underbrace{[1 \ 0 \ 0]}_{\mathbf{c}_i} \mathbf{x}_{o,i} \quad (\text{A.3b})$$

and the state $\mathbf{x}_{o,i}^T = [p_i \ f_{c,i} \ f_{v,i}]$. The transition matrix of the system (A.3) follows as

$$\Phi_i(t, t_0) = \begin{bmatrix} 1 & -\int_{t_0}^t \text{sign}(\dot{q}_i(\tau)) d\tau & -\int_{t_0}^t \dot{q}_i(\tau) d\tau \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (\text{A.4})$$

The observer is implemented on a digital computer and, therefore, a discrete-time representation of the system (A.2) is required. The discrete-time dynamics of the subsystem (A.3) is given by $\mathbf{x}_{o,i,k+1} = \Phi_i(t_{k+1}, t_k) \mathbf{x}_{o,i,k} + \int_{t_k}^{t_{k+1}} \Phi_i(t_{k+1}, \tau) d\tau \mathbf{b}_i u_{i,k}$, with $\mathbf{x}_{o,i,k} = \mathbf{x}_{o,i}(kT_s)$ and the sampling time $T_s = t_{k+1} - t_k$. Let us assume that $\dot{q}_i(\tau) = \dot{q}_i(kT_s) = \dot{q}_{i,k}$ for $kT_s \leq \tau < (k+1)T_s$. Then the overall discrete-time system reads as

$$\mathbf{x}_{o,k+1} = \underbrace{\begin{bmatrix} \mathbf{I} & -T_s \text{diag}(\text{sign}(\dot{\mathbf{q}}_k)) & -T_s \text{diag}(\dot{\mathbf{q}}_k) \\ \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} \end{bmatrix}}_{\Phi_k} \mathbf{x}_{o,k} + \underbrace{\begin{bmatrix} T_s \mathbf{I} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}}_{\Gamma} \mathbf{u}_k + \mathbf{w}_k \quad (\text{A.5})$$

$$\mathbf{p}_k = \mathbf{C}_o \mathbf{x}_{o,k} + \mathbf{v}_k,$$

where the zero-mean Gaussian process and measurement noise \mathbf{w}_k and \mathbf{v}_k , respectively, are added. A linear time-varying Kalman-filter, see, e.g. [95], is used as state observer for the system (A.5). The covariance matrix $\mathbf{R}_k > 0$ of the measurement noise \mathbf{v}_k and the covariance matrix $\mathbf{Q}_k > 0$ of the process noise \mathbf{w}_k are the design parameters for the Kalman-filter and determine the dynamics of the estimation error. The estimated friction parameters $\hat{\mathbf{f}}_c$ and $\hat{\mathbf{f}}_v$ can be used to compute an estimation of the friction torque $\hat{\boldsymbol{\tau}}_f(\dot{\mathbf{q}}) = \hat{\mathbf{F}}_c \text{sign}(\dot{\mathbf{q}}) + \hat{\mathbf{F}}_v \dot{\mathbf{q}}$.

A proof of concept for the friction observer is done by simulation in MATLAB[®]/Simulink[®]. The mathematical model of the fully actuated COMAU Racer 1.4 is used, with the static friction of (2.41) and friction parameters

$$\begin{aligned}\mathbf{f}_v^T &= [11.69, 108.56, 39.47, 1.342, 0.886, 0.7790] \text{ Nms}, \\ \mathbf{f}_c^T &= [40.8, 89.9, 19.47, 4.8, 3.64, 4.17] \text{ Nm}.\end{aligned}$$

The discrete linear time-variant Kalman-filter is implemented with a sampling time of $T_s = 1\text{ms}$ and the covariance matrices \mathbf{Q}_k and \mathbf{R}_k are chosen as constant diagonal matrices

$$\begin{aligned}\mathbf{R} &= \text{diag}([1, 1, 0.7, 0.05, 0.05, 0.05]) , \\ \mathbf{Q} &= \text{diag}([1, 1, 1, 0.1, 0.1, 0.1, 1, 1, 1, 0.1, 0.1, 0.1, 1, 1, 1, 0.1, 0.1, 0.1]) .\end{aligned}$$

The state estimation error covariance matrix for $k = 0$ is chosen as $\mathbf{P}_0 = 10^3 \cdot \mathbf{I}$, with the identity matrix \mathbf{I} , and the initial estimates of the friction parameters are set to 70% of the real values, i.e. $\hat{\mathbf{f}}_c(k = 0) = 0.7 \cdot \mathbf{f}_c$ and $\hat{\mathbf{f}}_v(k = 0) = 0.7 \cdot \mathbf{f}_v$. The path following controller of Section 3 stabilizes the robot on the bent lemniscate of Fig. 3.9 with a tangential reference velocity of $\dot{\eta}_1^p = 1\text{m/s}$. In Fig. A.1, the results of the friction observer simulation are depicted, where the upper two graphs show the generalized momenta $\mathbf{p} = \mathbf{D}(\mathbf{q})\dot{\mathbf{q}}$. The estimate of the generalized momenta $\hat{\mathbf{p}}$ corresponds very well with \mathbf{p} and is not depicted. The four lower graphs of Fig. A.1 show that the estimates of the friction parameters converge to the real friction parameters within 10s. Hence, a linear time-varying Kalman-filter is suitable to estimate the friction parameters of the nonlinear dynamic robot model without knowledge of the joint accelerations $\ddot{\mathbf{q}}$.

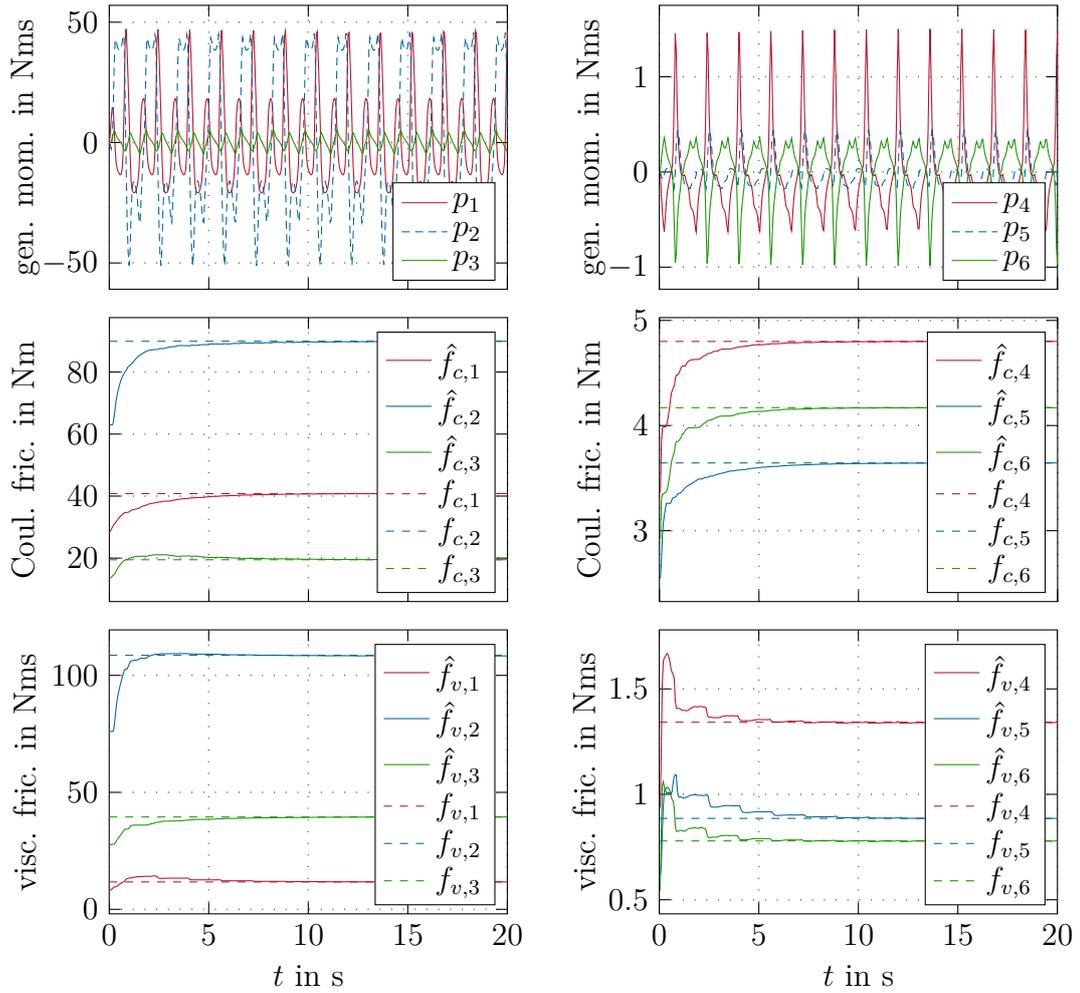


Figure A.1: Simulated results of the friction observer.

APPENDIX B

Base Parameters of the Comau Racer

This section revisits the method of [59] to determine the base parameters of serial manipulators with application to the COMAU Racer 1.4. The number of inertial base parameters is given by

$$n_{b,i} = 7N - 4\beta_1 \quad (\text{B.1})$$

or if the first joint axis is parallel to the gravity vector by

$$n_{b,i} = 7N - 4\beta_1 - 2, \quad (\text{B.2})$$

where β_1 is the number of links connected by joints whose axes are always parallel to the first joint axis. Using (B.2), the number of linear independent inertial base parameters is $n_{b,i} = 36$ for the COMAU Racer 1.4, where $\beta_1 = 1$.

The base parameters are not unique. Every nonsingular quadratic transformation of a base parameter set is also a base parameter set. Mayeda et al. give one possibility of an inertial base parameter set. For this, the N links are divided into link clusters with parallel joint axes. The first link i of a cluster j is denoted by $\alpha_j = i$ beginning with $\alpha_1 = 1$ and the last link k whose axis is parallel to the link i axis is denoted by $\beta_j = k$. The number of cluster j that contains link i is given by $c(i) = j$. The total number of clusters of a manipulator is denoted by K . The clusters of the COMAU Racer are listed in Table B.1, where $K = 5$.

To apply the method of [59] to the COMAU Racer, the coordinate systems have to be defined as depicted in Fig. B.1, where the z_i -axis corresponds to the axis of joint i and the joint angle θ_i is measured between x_{i-1} and x_i . The constant vector from the origin 0_i to the origin 0_{i+1} expressed in the coordinate system

Table B.1: Cluster parameter of the COMAU Racer.

$\alpha_1 = 1$	$\beta_1 = 1$
$\alpha_2 = 2$	$\beta_2 = 3$
$\alpha_3 = 4$	$\beta_3 = 4$
$\alpha_4 = 5$	$\beta_4 = 5$
$\alpha_5 = 6$	$\beta_5 = 6$

$(0_i, x_i, y_i, z_i)$ is denoted by \mathbf{l}_i^i . For the COMAU Racer the vectors \mathbf{l}_i^i are given by

$$\mathbf{l}_1^1 = \begin{bmatrix} a_1 \\ 0 \\ d_1 \end{bmatrix}, \quad \mathbf{l}_2^2 = \begin{bmatrix} a_2 \\ 0 \\ 0 \end{bmatrix}, \quad \mathbf{l}_3^3 = \begin{bmatrix} a_3 \\ 0 \\ 0 \end{bmatrix}, \quad \mathbf{l}_4^4 = \begin{bmatrix} 0 \\ 0 \\ d_4 \end{bmatrix}, \quad \mathbf{l}_5^5 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad (\text{B.3})$$

with the Denavit-Hartenberg parameters a_i and d_i for $1 \leq i \leq 4$ of Table 2.1.

The inertial base parameters for a robot with the first joint axis parallel to

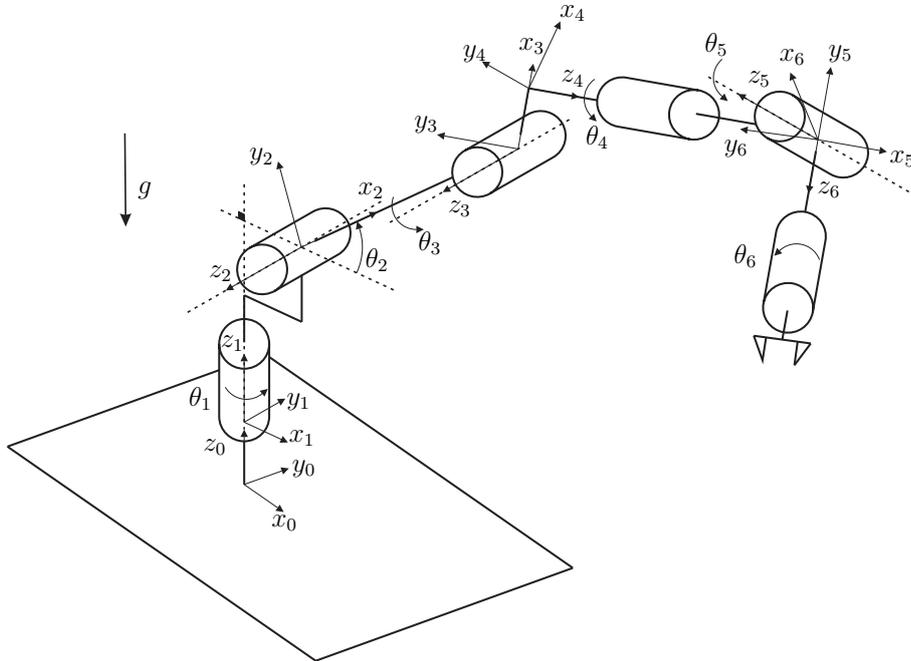


Figure B.1: Coordinate system definition of the COMAU Racer 1.4 for base parameter identification.

the gravity vector and $\alpha_2 = 2$ are given by

$$\begin{aligned}
\varrho_{b,1} &= \tilde{I}_{1,zz} + v_1 \\
\varrho_{b,2+7(i-2)} &= \tilde{I}_{i,zz} + v_i \\
\varrho_{b,3+7(i-2)} &= \rho_{i,x} \\
\varrho_{b,4+7(i-2)} &= \rho_{i,y} - \zeta_{b,i} \\
\varrho_{b,5+7(i-2)} &= \tilde{I}_{i,xx} - \tilde{I}_{i,yy} + v_i \\
\varrho_{b,6+7(i-2)} &= \tilde{I}_{i,xy} + l_{i,x}^i \zeta_{b,i} \\
\varrho_{b,7+7(i-2)} &= \tilde{I}_{i,xz} - l_{i,x}^i \zeta_{a,i} \\
\varrho_{b,8+7(i-2)} &= \tilde{I}_{i,yz} + l_{i,z}^i \zeta_{b,i} ,
\end{aligned} \tag{B.4}$$

for $2 \leq i \leq N$, with

$$\boldsymbol{\rho}_i = m_i \mathbf{p}_{c,i}^i + \mathbf{l}_i^i \sum_{j=i+1}^N m_{L,j} = [\rho_{i,x} \quad \rho_{i,y} \quad \rho_{i,z}]^T , \tag{B.5a}$$

$$\tilde{\mathbf{I}}_i = \bar{\mathbf{I}}_i + \mathbf{S}^T(\mathbf{l}_i^i) \mathbf{S}(\mathbf{l}_i^i) \sum_{j=i+1}^N m_{L,j} = \begin{bmatrix} \tilde{I}_{i,xx} & \tilde{I}_{i,xy} & \tilde{I}_{i,xz} \\ \tilde{I}_{i,xy} & \tilde{I}_{i,yy} & \tilde{I}_{i,yz} \\ \tilde{I}_{i,xz} & \tilde{I}_{i,yz} & \tilde{I}_{i,zz} \end{bmatrix} , \tag{B.5b}$$

$$\zeta_{a,i} = \begin{cases} 0, & i = \beta_{c(i)} \\ \sum_{j=i+1}^{\beta_{c(i)}} \rho_{j,z}, & \text{otherwise} \end{cases} , \tag{B.5c}$$

$$\zeta_{b,i} = \begin{cases} \sum_{j=\alpha_{c(i)+1}}^{\beta_{c(i)+1}} z \rho_{j,z}, & i = \beta_{c(i)} \text{ and } c(i) \neq K \\ 0, & \text{otherwise} \end{cases} , \tag{B.5d}$$

and

$$v_i = \begin{cases} \sum_{j=\alpha_{c(i)+1}}^{\beta_{c(i)+1}} \tilde{I}_{j,yy} - 2 l_{j,z} \zeta_{a,j}, & i = \beta_{c(i)} \text{ and } c(i) \neq K \\ 0, & \text{otherwise} \end{cases} . \tag{B.5e}$$

In (B.5b), $\bar{\mathbf{I}}_i$ denotes the inertia tensor of link i with respect to the coordinate system $(0_i, x_i, y_i, z_i)$, where

$$\bar{\mathbf{I}}_i = \begin{bmatrix} I_{i,xx} & I_{i,xy} & I_{i,xz} \\ I_{i,xy} & I_{i,yy} & I_{i,yz} \\ I_{i,xz} & I_{i,yz} & I_{i,zz} \end{bmatrix}$$

and the skew-symmetric matrix operator $\mathbf{S}(\cdot)$ performs the cross product $\mathbf{S}(\mathbf{a})\mathbf{b} = \mathbf{a} \times \mathbf{b}$. The resulting $n_b = 52$ base parameters $\boldsymbol{\varrho}_{b,C}^T = [\varrho_{b,C,1}, \dots, \varrho_{b,C,n_b}]$ for the COMAU Racer 1.4 are listed in Table B.2, where $\mathbf{p}_{c,i}^i = [p_{c,i,x} \quad p_{c,i,y} \quad p_{c,i,z}]^T$.

Table B.2: Base parameters $\rho_{b,C,j}$ of the COMAU Racer.

j	corresponding physical parameters (PP)	j	PP
1	$I_{1,zz} + \sum_{i=2}^3 I_{i,yy} + a_1^2 \sum_{i=2}^6 m_i$ $+ a_2^2 \sum_{i=3}^6 m_i + a_3^2 \sum_{i=4}^6 m_i$	27	$I_{5,xy}$
2	$I_{2,zz} + a_2^2 \sum_{i=3}^6 m_i$	28	$I_{5,xz}$
3	$m_2 p_{c,2,x} + a_2 \sum_{i=3}^6 m_i$	29	$I_{5,yz}$
4	$m_2 p_{c,2,y}$	30	$I_{6,zz}$
5	$I_{2,xx} - I_{2,yy} - a_2^2 \sum_{i=3}^6 m_i$	31	$m_6 p_{c,6,x}$
6	$I_{2,xy}$	32	$m_6 p_{c,6,y}$
7	$I_{2,xz} - a_2 m_3 p_{c,3,z}$	33	$I_{6,xx} - I_{6,yy}$
8	$I_{2,yz}$	34	$I_{6,xy}$
9	$I_{3,zz} + I_{4,yy} + a_3^2 \sum_{i=4}^6 m_i + d_4^2 \sum_{i=5}^6 m_i$	35	$I_{6,xz}$
10	$m_3 p_{c,3,x} + a_3 \sum_{i=4}^6 m_i$	36	$I_{6,yz}$
11	$m_3 p_{c,3,y} - m_4 p_{c,4,z} - d_4 \sum_{i=5}^6 m_i$	37	$I_{m,z,3}$
12	$I_{3,xx} - I_{3,yy} + I_{4,yy} + a_3^2 \sum_{i=4}^6 m_i + d_4^2 \sum_{i=5}^6 m_i$	38	$I_{m,z,4}$
13	$I_{3,xy} + a_3 (m_4 p_{c,4,z} + d_4 \sum_{i=5}^6 m_i)$	39	$I_{m,z,5}$
14	$I_{3,xz}$	40	$I_{m,z,6}$
15	$I_{3,yz}$	41	$f_{c,1}$
16	$I_{4,zz} + I_{5,yy}$	42	$f_{v,1}$
17	$m_4 p_{c,4,x}$	43	$f_{c,2}$
18	$m_4 p_{c,4,y} - m_5 p_{c,5,z}$	44	$f_{v,2}$
19	$I_{4,xx} - I_{4,yy} + I_{5,yy}$	45	$f_{c,3}$
20	$I_{4,xy}$	46	$f_{v,3}$
21	$I_{4,xz}$	47	$f_{c,4}$
22	$I_{4,yz} + d_4 m_5 p_{c,5,z}$	48	$f_{v,4}$
23	$I_{5,zz} + I_{6,yy}$	49	$f_{c,5}$
24	$m_5 p_{c,5,x}$	50	$f_{v,5}$
25	$m_5 p_{c,5,y} - m_6 p_{c,6,z}$	51	$f_{c,6}$
26	$I_{5,xx} - I_{5,yy} + I_{6,yy}$	52	$f_{v,6}$

Parameter Estimation of the Comau Racer

The linear regression approach of Section 2.5 is used to estimate the base parameters $\boldsymbol{q}_{b,C}$ of the COMAU Racer 1.4.

C.1 Excitation trajectory

The period T_f of the periodic function (2.78) was chosen as 10s, which is an integer multiple of the sampling time $T_s = 1\text{ms}$. The number of harmonics M in (2.78) was set to 5. Instead of using all harmonics, the amplitudes of the frequencies $\omega = 2\omega_f$ and $\omega = 4\omega_f$ are set to zero to reduce the degrees of freedom for the optimization problem (2.79) from 66 to 42. The joint position limits, the joint velocity limits, and the joint acceleration limits of Tab. C.1 are used as constraints \mathbf{g}_1 . The workspace constraints \mathbf{g}_2 are defined in a way that the robot cannot collide with the ground or the socket. The resulting $k_\nu = 42$ parameters $\boldsymbol{\nu}$ of the optimal excitation trajectory are listed in Tab. C.2 and the corresponding time functions are depicted in Fig. C.1.

Table C.1: Joint limitations of the COMAU Racer.

Joint	q_{min}	q_{max}	$ \dot{q}_{max} $	$ \ddot{q}_{max} $
1	-165°	165°	$220^\circ/s$	$400^\circ/s^2$
2	-65°	175°	$250^\circ/s$	$550^\circ/s^2$
3	-70°	80°	$300^\circ/s$	$666^\circ/s^2$
4	-180°	180°	$550^\circ/s$	$1270^\circ/s^2$
5	-120°	135°	$550^\circ/s$	$1270^\circ/s^2$
6	-180°	180°	$600^\circ/s$	$1390^\circ/s^2$

Table C.2: Parameters of the optimal excitation trajectory for the i -th joint.

	$i = 1$	$i = 2$	$i = 3$	$i = 4$	$i = 5$	$i = 6$
$q_{i,0}$	0.0332	1.3758	0.2091	0.0075	0.1167	0.0017
$a_{i,1}$	-0.1176	-0.0713	-0.0176	0.9083	-0.5208	-0.0867
$b_{i,1}$	0.3297	-0.2599	-0.0181	0.0713	0.5076	-0.7961
$a_{i,3}$	1.9736	0.2130	-0.0886	0.4240	1.0258	0.3671
$b_{i,3}$	-0.3291	0.2930	-0.1763	0.7724	0.5667	0.9517
$a_{i,5}$	0.2785	1.6351	0.9626	2.7082	0.6510	1.2767
$b_{i,5}$	0.4256	1.2841	2.3068	1.4445	2.0331	2.1298

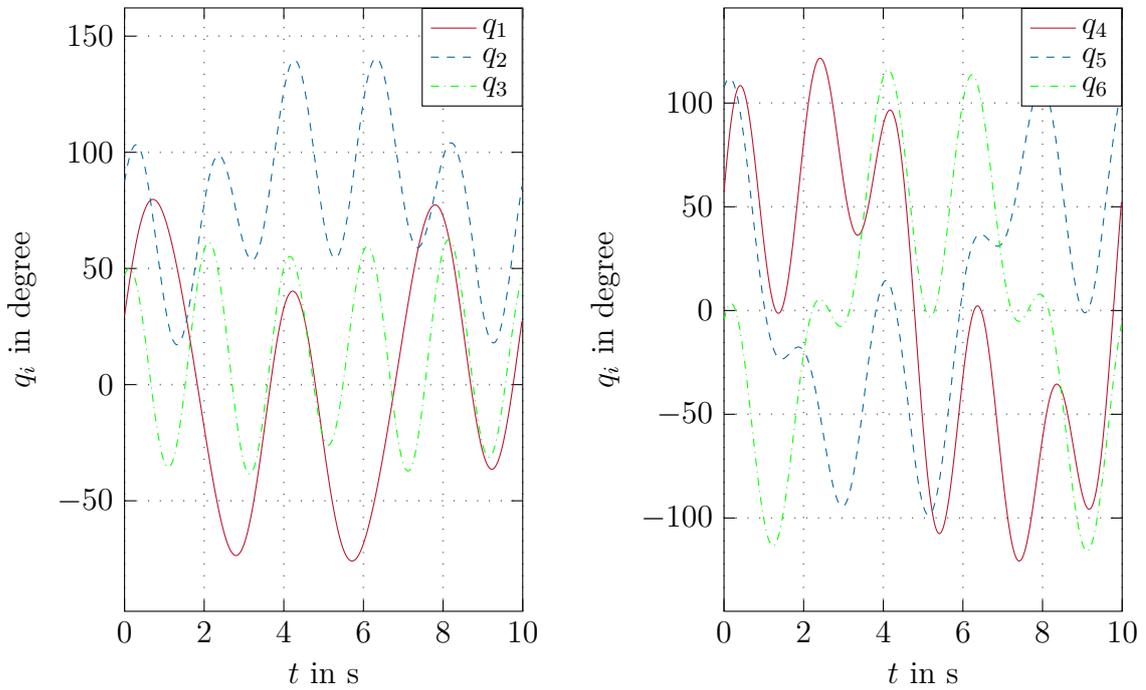


Figure C.1: One period of the optimal excitation trajectory.

C.2 Estimated parameters

The joint positions and torques are recorded for five periods of the excitation trajectory. Only the last four periods are used for the signal processing and parameter estimation to avoid transient operation. The filter-bandwidth for the positions is set to $f_g = 20/T_f$. The parameters are estimated using the WLSE

algorithm (2.80) with the torque covariance matrix

$$\Upsilon = \begin{bmatrix} 159.53 & 0 & 0 & 0 & 0 & 0 \\ 0 & 135 & 0 & 0 & 0 & 0 \\ 0 & 0 & 38.688 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.5024 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.247 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.08 \end{bmatrix} \text{N}^2\text{m}^2,$$

The identification procedure is carried out seven times and the estimated parameters of each run are compared. The mean values of the identified base parameters $\hat{\varrho}_{b,C,j}$ of the seven identification procedures as well as the relative standard deviation $\hat{\sigma}_{rel,j}$ are listed in Tab. C.3. Most of the estimated parameters have a relative

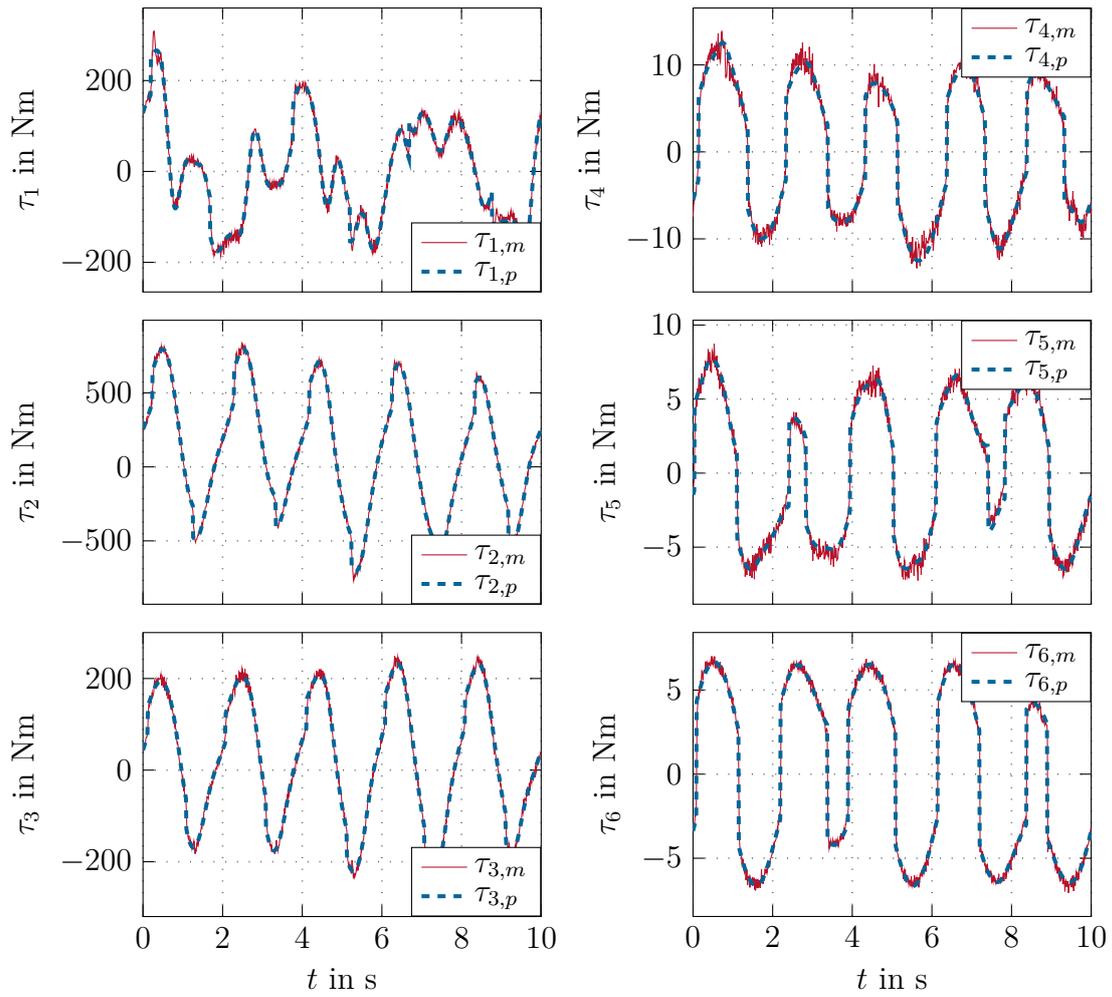
Table C.3: Identified base parameters $\hat{\varrho}_{b,C,j}$ of the COMAU Racer.

j	$\hat{\varrho}_{b,C,j}$	unit	$\hat{\sigma}_{rel,j}$	j	$\hat{\varrho}_{b,C,j}$	unit	$\hat{\sigma}_{rel,j}$
1	32.79	kgm ²	0.479%	27	0.0199	kgm ²	8.07%
2	26.14	kgm ²	0.196%	28	0.00305	kgm ²	25.5%
3	27.81	kgm	0.169%	29	0.0113	kgm ²	4.78%
4	-0.396	kgm	17.3%	30	-0.0223	kgm ²	6.29%
5	-17.7	kgm ²	1.65%	31	-0.00607	kgm	12.8%
6	-1.4	kgm ²	5.85%	32	-0.0106	kgm	5.47%
7	-1.28	kgm ²	5.81%	33	-0.0116	kgm ²	8.63%
8	1.929	kgm ²	3.31%	34	-0.00136	kgm ²	20.3%
9	4.299	kgm ²	1.56%	35	0.00904	kgm ²	7.93%
10	3.71	kgm	1.01%	36	-0.00282	kgm ²	15.6%
11	-6.4	kgm	0.536%	37	0.000611	kgm ²	2.64%
12	3.545	kgm ²	1.17%	38	7.19e - 05	kgm ²	2.37%
13	0.433	kgm ²	5.67%	39	6.75e - 05	kgm ²	2.12%
14	0.516	kgm ²	8.81%	40	5.08e - 05	kgm ²	3.87%
15	-0.234	kgm ²	7.35%	41	41.63	Nm	1.82%
16	0.0745	kgm ²	3.61%	42	14.68	Nms/rad	1.71%
17	-0.0396	kgm	7.12%	43	101.7	Nm	3.03%
18	0.129	kgm	2.91%	44	97.29	Nms/rad	8.11%
19	0.0595	kgm ²	9.99%	45	21	Nm	2.29%
20	0.0404	kgm ²	3.54%	46	39.69	Nms/rad	7.14%
21	-0.014	kgm ²	16.2%	47	4.194	Nm	3.29%
22	-0.0853	kgm ²	3.19%	48	1.812	Nms/rad	5.84%
23	0.00828	kgm ²	8.44%	49	2.688	Nm	2.1%
24	0.0329	kgm	2.99%	50	1.695	Nms/rad	4.98%
25	-0.0426	kgm	1.09%	51	3.746	Nm	2.03%
26	-0.0362	kgm ²	4.83%	52	1.102	Nms/rad	3.77%

standard deviation $\hat{\sigma}_{rel,j}$ of less than five percent. Other parameters seem to be badly identifiable with a relative standard deviation of over 15%. Especially the parameter $\hat{q}_{b,C,28} = I_{5,xz}$ varies a lot over the individual identification runs with a relative standard deviation of $\hat{\sigma}_{rel,28} = 25.5\%$. However, the predicted torque of the model (2.74) with the estimated parameters of all runs match the measured torque very well and the root mean square of the torque prediction error has the same order of magnitude as the standard deviation of the torque measurements, which can be seen in Fig. C.2 and Tab. C.4.

Table C.4: Prediction error and noise level of the torques.

	Joint 1	Joint 2	Joint 3	Joint 4	Joint 5	Joint 6
error RMS in Nm	11.2	21.8	9.6	0.98	0.57	0.47
noise level in Nm	12.6	11.6	6.22	0.71	0.5	0.28

Figure C.2: Measured τ_m and predicted τ_p torque at the excitation trajectory.

References

- [1] International Federation of Robotics. (2017). Executive summary world robotics 2017 industrial robots, [Online]. Available: https://ifr.org/downloads/press/Executive_Summary_WR_2017_Industrial_Robots.pdf (visited on 01/05/2018).
- [2] M. Hägele, K. Nilsson, and J. N. Pires, “Industrial robotics”, in *Springer handbook of robotics*, B. Siciliano and O. Khatib, Eds., Berlin Heidelberg: Springer, 2008, pp. 963–986.
- [3] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, *Robotics: Modelling, planning and control*. London: Springer, 2009.
- [4] M. W. Spong, S. Hutchinson, and M. Vidyasagar, *Robot modeling and control*, 3rd ed. New York: John Wiley & Sons, 2006.
- [5] W. Khalil and E. Dombre, *Modeling, identification and control of robots*. London: Butterworth-Heinemann, 2004.
- [6] F. Caccavale, C. Natale, B. Siciliano, and L. Villani, “Resolved-acceleration control of robot manipulators: A critical review with experiments”, *Robotica*, vol. 16, no. 5, pp. 565–573, 1998.
- [7] M. W. Spong, “Modeling and control of elastic joint robots”, *Journal of dynamic systems, measurement, and control*, vol. 109, no. 4, pp. 310–319, 1987.
- [8] A. De Luca and L. Lanari, “Robots with elastic joints are linearizable via dynamic feedback”, in *Proceedings of the 34th IEEE Conference on Decision and Control*, IEEE, vol. 4, New Orleans, LA, USA, Dec. 1995, pp. 3895–3897.
- [9] A. De Luca, “Trajectory control of flexible manipulators”, *Control Problems in Robotics and Automation*, pp. 83–104, 1998.

- [10] C. Samson, “Path following and time-varying feedback stabilization of a wheeled mobile robot”, in *Proceedings of the International Conference on Automation, Robotics and Computer Vision (ICARCV) 1992*, Singapore, Sep. 1992, RO-13.1.1-RO-13.1.5.
- [11] A. Banaszuk and J. Hauser, “Feedback linearization of transverse dynamics for periodic orbits”, in *Proceedings of the 33rd IEEE Conference on Decision and Control 1994*, vol. 2, Lake Buena Vista, FL, USA, Dec. 1994, pp. 1639–1644.
- [12] C. Nielsen and M. Maggiore, “Maneuver regulation via transverse feedback linearization: Theory and examples”, *IFAC Proceedings Volumes*, vol. 37, no. 13, pp. 57–64, 2004.
- [13] ———, “On local transverse feedback linearization”, *SIAM Journal on Control and Optimization*, vol. 47, no. 5, pp. 2227–2250, 2008.
- [14] M. Böck and A. Kugi, “Manifold stabilization and path following control for flat systems with application to a laboratory tower crane”, in *Proceedings of the IEEE Conference on Decision and Control*, Los Angeles, CA, USA, Dec. 2014, pp. 4529–4535.
- [15] C. Nielsen and D. Wang, “Path following using transverse feedback linearization: Application to a maglev positioning system”, in *Proceedings of the American Control Conference*, St. Louis, MO, USA, Jun. 2009, pp. 3045–3050.
- [16] A. Hladio, C. Nielsen, and D. Wang, “Path following controller design for a class of mechanical systems”, *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 10 331–10 336, 2011.
- [17] ———, “Path following for mechanical systems: Experiments and examples”, in *Proceedings of the American Control Conference (ACC), 2011*, IEEE, San Francisco, CA, USA, Jun. 2011, pp. 551–556.
- [18] ———, “Path following for a class of mechanical systems”, *IEEE Transactions on Control Systems Technology*, vol. 21, no. 6, pp. 2380–2390, Nov. 2013.
- [19] R. Gill, D. Kulić, and C. Nielsen, “Robust path following for robot manipulators”, in *Proceedings of the IEEE Conference on Intelligent Robots and Systems*, Tokyo, Japan, Nov. 2013, pp. 3412–3418.
- [20] L. Consolini, M. Maggiore, C. Nielsen, and M. Tosques, “Path following for the pvtol aircraft”, *Automatica*, vol. 46, pp. 1284–1296, 2010.
- [21] V. A. Toponogov, *Differential geometry of curves and surfaces*. Boston: Birkhäuser, 2006.
- [22] R. Gill, D. Kulić, and C. Nielsen, “Spline path following for redundant mechanical systems”, *IEEE Transaction on Robotics*, vol. 31, no. 6, pp. 1378–1392, 2015.

- [23] D. Carrol, E. Köse, and I. Sterling, “Improving frenet’s frame using bishop’s frame”, *Journal of Mathematics Research*, vol. 5, no. 4, pp. 97–106, 2013.
- [24] R. L. Bishop, “There is more than one way to frame a curve”, *American Mathematical Monthly*, vol. 82, no. 3, pp. 246–251, 1975.
- [25] A. J. Hanson, *Visualizing quaternions*. San Francisco: Morgan Kaufmann, 2006.
- [26] N. Hogan, “Impedance control: An approach to manipulation”, *Journal of Dynamic Systems, Measurement, and Control*, vol. 107, pp. 1–24, 1985.
- [27] D. Wedel and G. Saridis, “An experiment in hybrid position/force control of a six dof revolute manipulator”, in *Proceedings of the IEEE International Conference on Robotics and Automation*, Philadelphia, PA, USA, Apr. 1988, 1638–1642.
- [28] J. E. Colgate and N. Hogan, “Robust control of dynamically interacting systems”, *International Journal of Control*, vol. 48, no. 1, pp. 65–88, 1988.
- [29] W.-S. Lu and Q.-H. Meng, “Impedance control with adaptation for robotic manipulators”, *IEEE Transactions on Robotics and Automation*, vol. 7, no. 3, pp. 408–415, 1991.
- [30] S. Chiaverini, B. Siciliano, and L. Villani, “A stable force/position controller for robot manipulators”, in *Proceedings of the 31st IEEE Conference on Decision and Control*, IEEE, Tucson, AZ, USA, Dec. 1992, pp. 1869–1874.
- [31] D. Surdilović, “Contact stability issues in position based impedance control: Theory and experiments”, in *Proceedings of the 1996 IEEE International Conference on Robotics and Automation*, vol. 2, San Diego, CA, USA, Oct. 1996, pp. 1675–1680.
- [32] S. Chiaverini, B. Siciliano, and L. Villani, “A survey of robot interaction control schemes with experimental comparison”, *IEEE Transactions on Mechatronics*, vol. 4, no. 3, pp. 273–285, 1999.
- [33] F. Caccavale, C. Natale, B. Siciliano, and L. Villani, “Six-dof impedance control based on angle/axis representations”, *IEEE Transactions on Robotics and Automation*, vol. 15, no. 2, pp. 289–300, 1999.
- [34] D. Surdilović, “Robust control design of impedance control for industrial robots”, in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, San Diego, CA, USA, Oct. 2007, pp. 3572–3579.
- [35] P. Akella, M. Peshkin, E. Colgate, W. Wannasuphprasit, N. Nagesh, J. Wells, S. Holland, T. Pearson, and B. Peacock, “Cobots for the automobile assembly line”, in *Proceedings of the IEEE International Conference on Robotics and Automation*, Detroit, Michigan, USA, May 1999, pp. 728–733.

- [36] A. Bicchi, M. Peshkin, and E. Colgate, “Safety for physical human-robot interaction”, in *Springer handbook of robotics*, B. Siciliano and O. Khatib, Eds., Berlin Heidelberg: Springer, 2008, pp. 1335–1348.
- [37] S. Haddadin, *Towards safe robots: Approaching asimov’s 1st law*. Berlin Heidelberg: Springer, 2014.
- [38] A. Bicchi and G. Tonietti, “Fast and "soft-arm"tactics”, *IEEE Robotics & Automation Magazine*, vol. 11, no. 2, pp. 22–33, 2004.
- [39] B. Gillespie, E. Colgate, and M. Peshkin, “A general framework for cobot control”, *IEEE Transactions on Robotics and Automation*, vol. 17, no. 4, pp. 391–401, 2001.
- [40] J. Abbott, P. Marayong, and A. Okamura, “Haptic virtual fixtures for robot-assisted manipulation”, in *Robotics research*, S. Thrun, R. Brooks, and H. Durrant-Whyte, Eds., Berlin Heidelberg: Springer, 2007, pp. 49–64.
- [41] M. Peshkin, E. Colgate, W. Wannasuphprasit, C. Moore, B. Gillespie, and P. Akella, “Cobot architecture”, *IEEE Transactions on Robotics and Automation*, vol. 17, no. 4, pp. 377–390, 2001.
- [42] J. Funda, R. Taylor, B. Eldridge, S. Gomory, and K. Gruben, “Constrained cartesian motion control for teleoperated surgical robots”, *IEEE Transactions on Robotics and Automation*, vol. 12, no. 3, pp. 453–465, 1996.
- [43] S. Bowyer, B. Davies, and F. y Baena, “Active constraints/virtual fixtures: A survey”, *IEEE Transactions on Robotics*, vol. 30, no. 1, pp. 138–157, 2014.
- [44] R. Prada and S. Payandeh, “On study of design and implementation of virtual fixtures”, *Virtual reality*, vol. 13, no. 2, pp. 117–129, 2009.
- [45] J. Abbott and A. Okamura, “Stable forbidden-region virtual fixtures for bilateral telemanipulation”, *Journal of dynamic systems, measurement, and control*, vol. 128, no. 1, pp. 53–64, 2006.
- [46] N. Turro and O. Khatib, “Haptically augmented teleoperation”, in *Experimental Robotics VII*, D. Rus and S. Singh, Eds., Berlin Heidelberg: Springer, 2001, pp. 1–10.
- [47] R. Kikuuwe, N. Takesue, and H. Fujimoto, “A control framework to generate nonenergy-storing virtual fixtures: Use of simulated plasticity”, *IEEE Transactions on Robotics*, vol. 24, no. 4, pp. 781–793, 2008.
- [48] S. Bowyer and F. y Baena, “Dynamic frictional constraints for robot assisted surgery”, in *Proceedings of the IEEE World Haptics Conference*, Daejeon, Korea, Apr. 2013, pp. 319–324.
- [49] ———, “Dynamic frictional constraints in translation and rotation”, in *Proceedings of the IEEE Conference on Robotics and Automation*, Hong Kong, China, May 2014, pp. 2685–2692.

- [50] ———, “Dissipative control for physical human–robot interaction”, *IEEE Transactions on Robotics*, vol. 31, no. 6, pp. 1281–1293, 2015.
- [51] M. Li, A. Kapoor, and R. Taylor, “A constrained optimization approach to virtual fixtures”, in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Edmonton, Canada, Aug. 2005, pp. 1408–1413.
- [52] A. Bettini, P. Marayong, S. Lang, A. Okamura, and G. Hager, “Vision-assisted control for manipulation using virtual fixtures”, *IEEE Transactions on Robotics*, vol. 20, no. 6, pp. 953–966, 2004.
- [53] R. Castillo-Cruces and J. Wahrburg, “Virtual fixtures with autonomous error compensation for human-robot cooperative tasks”, *Robotica*, vol. 28, no. 2, pp. 267–277, 2010.
- [54] W. Wannasuphoprasit, B. Gillespie, E. Colgate, and M. Peshkin, “Cobot control”, in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 4, Albuquerque, New Mexico, USA, Apr. 1997, pp. 3571–3576.
- [55] L. W. Tsai, *Robot analysis*. New York: John Wiley & Sons, 1999.
- [56] J.-P. Merlet, *Parallel robots*, 2nd ed. Dordrecht, The Netherlands: Springer, 2006.
- [57] R. M. Murray, Z. Li, and S. S. Sastry, *Robotic manipulation*. Boca Raton, FL: CRC Press, 1994.
- [58] F. L. Lewis, C. T. Abdallah, and D. M. Dawson, *Control of robot manipulators*. New York: Macmillan Publishing Company, 1993.
- [59] H. Mayeda, K. Yoshida, and K. Osuka, “Base parameters of manipulator dynamic models”, *IEEE Transactions on Robotics and Automation*, vol. 6, no. 3, pp. 312–321, 1990.
- [60] J. Hollerbach, W. Khalil, and M. Gautier, “Model identification”, in *Springer handbook of robotics*, B. Siciliano and O. Khatib, Eds., Berlin Heidelberg: Springer, 2008, pp. 321–344.
- [61] J. Wu, J. Wang, and Z. You, “An overview of dynamic parameter identification of robots”, *Robotics and computer-integrated manufacturing*, vol. 26, no. 5, pp. 414–419, 2010.
- [62] J. Swevers, C. Ganseman, B. Tukel, J. De Schutter, and H. V. Brussel, “Optimal robot excitation and identification”, *IEEE Transactions on Robotics and Automation*, vol. 13, no. 5, pp. 730–740, 1997.
- [63] J. Swevers, W. Verdonck, and J. De Schutter, “Dynamic model identification for industrial robots”, *IEEE Control Systems Magazine*, vol. 27, no. 5, pp. 58–71, 2007.

- [64] B. Armstrong, “On finding exciting trajectories for identification experiments involving systems with nonlinear dynamics”, *The International journal of robotics research*, vol. 8, no. 6, pp. 28–48, 1989.
- [65] M. Gautier and W. Khalil, “Exciting trajectories for the identification of base inertial parameters of robots”, *The International journal of robotics research*, vol. 11, no. 4, pp. 362–375, 1992.
- [66] R. Clavel, “Delta, a fast robot with parallel geometry”, in *Proceedings of the 18th International Symposium on Industrial Robots (ISIR)*, Lausanne, Switzerland, Apr. 1988, pp. 91–100.
- [67] —, “Conception d’un robot parallèle rapide à 4 degrés de liberté”, PhD thesis, Swiss Federal Institute of Technology Lausanne (EPFL), Lausanne, 1991.
- [68] A. Codourey, “Contribution à la commande des robots rapides et précis. application au robot delta à entraînement direct”, PhD thesis, Swiss Federal Institute of Technology Lausanne (EPFL), Lausanne, 1991.
- [69] —, “Dynamic modelling and mass matrix evaluation of the delta parallel robot for axes decoupling control”, in *Proceedings of the 1996 IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 3, Osaka, Japan, Nov. 1996, pp. 1211–1218.
- [70] B. Bischof, T. Glück, and A. Kugi, “Combined path following and compliance control for fully actuated rigid body systems in 3-d space”, *IEEE Transactions on Control Systems Technology*, vol. 25, no. 5, pp. 1750–1760, 2017.
- [71] B. Armstrong-Hélouvry, P. Dupont, and C. Canudas De Wit, “A survey of models, analysis tools and compensation methods for the control of machines with friction”, *Automatica*, vol. 30, no. 7, pp. 1083–1138, 1994.
- [72] H. Olsson, K. Åström, C. Canudas De Wit, M. Gäfvert, and P. Lischinsky, “Friction models and friction compensation”, *European journal of control*, vol. 4, no. 3, pp. 176–195, 1998.
- [73] C. Canudas De Wit and P. Lischinsky, “Adaptive friction compensation with partially known dynamic friction model”, *International journal of adaptive control and signal processing*, vol. 11, no. 1, pp. 65–80, 1997.
- [74] A. De Luca, R. Farina, and P. Lucibello, “On the control of robots with visco-elastic joints”, in *Proceedings of the IEEE International Conference on Robotics and Automation*, Barcelona, Spain, Apr. 2005, pp. 4297–4302.
- [75] P. Tomei, “A simple pd controller for robots with elastic joints”, *IEEE Transactions on Automatic Control*, vol. 36, no. 10, pp. 1208–1213, 1991.
- [76] H. K. Khalil, *Nonlinear systems*, 3rd ed. Englewood Cliffs: Prentice Hall, 2002.

- [77] M. Gautier, “Numerical calculation of the base inertial parameters of robots”, *Journal of Field Robotics*, vol. 8, no. 4, pp. 485–506, 1991.
- [78] A. Isidori, *Nonlinear control systems*, 3rd ed. London: Springer, 1995.
- [79] S. Flixeder, T. Glück, M. Böck, and A. Kugi, “Combined path following and compliance control with application to a biaxial gantry robot”, in *Proceedings of the 2014 IEEE Conference on Control Applications (CCA)*, Antibes, France, Oct. 2014, pp. 796–801.
- [80] B. Bischof, T. Glück, M. Böck, and A. Kugi, “Path following control for elastic joint robots”, *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 4806–4811, 2017.
- [81] —, “A path/surface following control approach to generate virtual fixtures”, *IEEE Transactions on Robotics*, pp. 1–16, 2018. DOI: 10.1109/TR0.2018.2861913.
- [82] E. Hairer and G. Wanner, *Solving ordinary differential equations ii: Stiff and differential-algebraic problems*, 2nd ed. Berlin Heidelberg: Springer, 2010.
- [83] D. Luenberger and Y. Ye, *Linear and nonlinear programming*, 3rd ed. New-York: Springer, 2008.
- [84] J. R. Munkres, *Analysis on manifolds*. Redwood City: Addison-Wesley, 1993.
- [85] C. Ott, *Cartesian impedance control of redundant and flexible-joint robots*. Berlin Heidelberg: Springer, 2008.
- [86] H. Ma, “Curve and surface framing for scientific visualization and domain dependent navigation”, PhD thesis, Indiana University, Department of Computer Science, Bloomington, Indiana, USA, 1996.
- [87] L. Piegl and W. Tiller, *The nurbs book*. Berlin: Springer Science, 2012.
- [88] S. Flixeder, T. Glück, and A. Kugi, “Modeling and force control for the collaborative manipulation of deformable strip-like materials”, *IFAC-PapersOnLine*, vol. 49, no. 21, pp. 95–102, 2016.
- [89] T. Glück, B. Bischof, and A. Kugi, “Surface following control for fully actuated rigid body systems in three-dimensional euclidean space”, *IFAC-PapersOnLine*, vol. 49, no. 18, pp. 594–599, 2016.
- [90] C. Ott, A. Albu-Schäffer, A. Kugi, and G. Hirzinger, “On the passivity-based impedance control of flexible joint robots”, *IEEE Transactions on Robotics*, vol. 24, no. 2, pp. 416–429, 2008.
- [91] S. Bowyer and F. y Baena, “Deformation invariant bounding spheres for dynamic active constraints in surgery”, *Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine*, vol. 228, no. 4, pp. 350–361, 2014.

- [92] F. Ficuciello, L. Villani, and B. Siciliano, “Variable impedance control of redundant manipulators for intuitive human–robot physical interaction”, *IEEE Transactions on Robotics*, vol. 31, no. 4, pp. 850–863, 2015.
- [93] S. Flixeder, T. Glück, and A. Kugi, “Force-based cooperative handling and lay-up of deformable materials: Mechatronic design, modeling, and control of a demonstrator”, *Mechatronics*, vol. 47, pp. 246–261, 2017.
- [94] A. De Luca and R. Mattone, “Actuator failure detection and isolation using generalized momenta”, in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 1, Taipei, Taiwan, Sep. 2003, pp. 634–639.
- [95] L. Ljung, *System identification*, 2nd ed. New Jersey: Prentice Hall, 1999.