Lehrstuhl für Steuerungs- und Regelungstechnik
Technische Universität München
Univ.-Prof. Dr.-Ing./Univ. Tokio Martin Buss

Hybrid Dynamical System Methods for Legged Robot Locomotion with Variable Ground Contact

Marion Sobotka

Vollständiger Abdruck der von der Fakultät für Elektrotechnik und Informationstechnik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktor-Ingenieurs (Dr.-Ing.)

genehmigten Dissertation.

Vorsitzender: Univ.-Prof. Dr.-Ing. Wolfgang Utschick

Prüfer der Dissertation:

1. Univ.-Prof. Dr.-Ing./Univ. Tokio Martin Buss

2. Univ.-Prof. Dr.-Ing. habil. Oliver Sawodny,
Universität Stuttgart

Die Dissertation wurde am 25.10.2006 bei der Technischen Universität München eingereicht und durch die Fakultät für Elektrotechnik und Informationstechnik am 12.02.2007 angenommen.

Foreword

This thesis summarizes research results of the last four years. I began my PhD studies at the Control Systems Group at Technische Universität Berlin in 2003. After one year, because of my advisors change of affiliation, I also changed to the Institute of Automatic Control Engineering at Technische Universität München.

I thank my "Doktorvater" Prof. Martin Buss for providing the topic and guiding me through the thesis from the working-in to the presented finalization. At his institute in Berlin as well as in Munich, I enjoyed the inspiring working environment.

Furthermore, I thank Dr. Dirk Wollherr for guidance, for fruitful discussions, and for proof-reading. I thank Jan Wolff who contributed many ideas to this research project, finally resulting in a cooperation on invariance control for balance maintenance. Also, I thank both, Dirk and Jan, for being great office mates and for their open ear for all my computer problems. I thank Mathias Bachmayer for the alongside hardware design. Thanks to all LSR-colleagues for entertaining lunch times, joyful board game evenings, challenging hiking tours, etc.

Many students contributed to this work. I especially thank Tobias Gern, Maik Blankenburg, Yilu Bao, Björn Langhof, and Christian Raubitschek.

I thank my parents for supporting me without reservation. Thank you Christian for ongoing efforts in convincing me that there is a life besides thesis writing.

Munich, 2006. Marion Sobotka

Contents

Notations

Abbreviations

BVP	Boundary Value Problem
CoM	Center of Mass
FRI	Foot Rotation Indicator
HSM	Hybrid State Model
ODE	Ordinary Differential Equation
ZMP	Zero Moment Point

Scalars, Vectors, and Matrices

Scalars are denoted by upper and lower case letters in italic type. Vectors are denoted by lower case letters in boldface type, and a vector $\boldsymbol{x}$ is composed of elements x_i. Only vectorial forces are denoted by upper case letters, and a force vector $\boldsymbol{F}$ is composed of elements F_x, F_y, and F_z. Matrices are denoted by upper case letters in boldface type, and a matrix $\boldsymbol{M}$ is composed of elements m_{ij} (i-th row, j-th column).

x	scalar
$\boldsymbol{x}$	vector
$\boldsymbol{X}$	matrix or force
$f(\cdot)$	scalar function
$\boldsymbol{f}(\cdot)$	vector function
$\dot{\boldsymbol{x}}$, $\ddot{\boldsymbol{x}}$	equivalent to $\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{x}$ and $\frac{\mathrm{d}^2}{\mathrm{d}t^2}\boldsymbol{x}$
$\boldsymbol{M}^T$	transposed of matrix $\boldsymbol{M}$
$\boldsymbol{M}^{-1}$	inverse of matrix $\boldsymbol{M}$
$\boldsymbol{M}^+$	pseudoinverse of matrix $\boldsymbol{M}$

Subscripts and Superscripts

$v_x,\ v_y,\ v_z$	component of vector $\boldsymbol{v}$ in x-, y-, z-direction
t^-, t^+	limit from the left, limit from the right of time t
$\boldsymbol{x}^-$, $\boldsymbol{x}^+$	state $\boldsymbol{x}$ at time t^- or time t^+
t_0, t_f	initial time, final time
$\boldsymbol{x}_0$, $\boldsymbol{x}_f$	initial value, final value of state $\boldsymbol{x}$
x^d	desired trajectory for x
x_l, x_b	upper boundary, lower boundary for x

General

$\mathbb{R}$	real numbers
$\mathbb{Z}$	integers
$\boldsymbol{e}_x$, $\boldsymbol{e}_y$, $\boldsymbol{e}_z$	cartesian directions

Hybrid Modeling

t	time
$\boldsymbol{\zeta}$	hybrid state vector
$\boldsymbol{x}$	continuous state vector
x_d	discrete state
n	dimension of $\boldsymbol{x}$
N_d	number of discrete states x_d
$\boldsymbol{u}$	continuous control input
u_d	discrete control input
$\boldsymbol{y}$	continuous output vector
y_d	discrete output
$\boldsymbol{f}(\cdot)$	right hand side of differential equation
S	transition surface
$s(\cdot) = 0$	algebraic description of transition surface
$\boldsymbol{\varphi}(\cdot)$	jump map for hybrid state $\boldsymbol{\zeta}$
$\boldsymbol{g}(\cdot)$	jump map for continuous state $\boldsymbol{x}$
$g_d(\cdot)$	jump map for discrete state x_d
$\boldsymbol{h}(\cdot)$	output function
$\delta_{i,j}$	Kronecker delta

Modeling of Legged Locomotion

$\boldsymbol{q}$	generalized coordinate vector
n_q	dimension of $\boldsymbol{q}$
ξ	coordinate in cartesian x-direction
η	coordinate in cartesian y-direction
α_i	coordinates for passive joint angles
β_i	coordinates for actuated joint angles
U	kinetic energy
V	potential energy
$L(\cdot)$, $L^*(\cdot)$	Lagrange function
$I(\cdot)$, $I^*(\cdot)$	cost function
m_i	masses of links
$\boldsymbol{M}$	inertia matrix
$\boldsymbol{n}$	vector of coriolis, centrifugal, and inertia terms
g	earth acceleration $g = 9.81\,\mathrm{m/s^2}$
$\boldsymbol{c}(\cdot)$	vector of holonomic constraints
N_{x_d}	number of holonomic constraints in contact situation x_d

$\boldsymbol{\lambda}$	Lagrange multipliers
$\boldsymbol{J}$	Jacobian matrix
$\boldsymbol{\Lambda}$	impulsive reaction force
t_c	collision time
R_y, $R_{y,L}$, $R_{y,R}$	vertical component of contact force
T_z	horizontal component of contact moment
l_L, l_R	foot geometry constants
$\boldsymbol{R}_i$	force acting on link i
$\boldsymbol{T}_i$	moment acting on link i
$\boldsymbol{r}_i$	position vector of center of mass of link i
$\boldsymbol{r}_{\text{zmp}}$	position of Zero Moment Point (ZMP)
$\boldsymbol{T}_{\text{feet}}$, $\boldsymbol{T}_{\text{edges}}$	symmetry transformations
$\boldsymbol{B}$, $\boldsymbol{b}$	elements of complementarity problem
F_x, F_y	spring-damper forces
k_i, $i = 1, ..., 4$	spring-damper parameterization

Compass Gait Robot

m_l	mass of leg
m_h	mass of hip
l	length of leg
a, b	geometry constants of leg

Monoped Robot

m_f	mass of foot
m_l	mass of link
l_f	length of half foot
h_f	height of foot
l_l	length of link
$h_{cm,f}$	geometry constant of foot
$\boldsymbol{I}_f$, $\boldsymbol{I}_l$	principal moments of inertia of foot and link

Gymnast Robot

m_f	mass of foot
m_l	mass of link
l_f	length of foot
l_l	length of link

Trajectory Planning

T	period length
$\boldsymbol{K}_P$, $\boldsymbol{K}_D$	proportional gain matrix, derivative gain matrix
$\boldsymbol{v}$	control input after feedback linearization
$\boldsymbol{f}_{\text{int}}(\cdot)$	right hand side of internal dynamics ODE
$\boldsymbol{\varrho}(\cdot)$	boundary conditions
$\boldsymbol{p}$, $\boldsymbol{p}_{\text{in}}$, $\boldsymbol{p}_{\text{out}}$	parameter vectors
n_p	dimension of parameter vector $\boldsymbol{p}$
t_0, t_1, $t_2, \ldots, t_f$	initial time, switching times, and final time
A, B, ω	shape parameters of trajectory planning

Stability

$\boldsymbol{\phi}_t(\cdot)$, $\boldsymbol{\phi}_t^H(\cdot)$	flux of (hybrid) dynamical system
$\boldsymbol{\Phi}_t(\cdot)$, $\boldsymbol{\Phi}_t^H(\cdot)$	trajectory sensitivity of (hybrid) initial value problem
U_ε	environment
ε, δ	small scalar values
$\boldsymbol{P}(\cdot)$, $\boldsymbol{DP}(\cdot)$	Poincaré map, derivative of Poincaré map
$\tau(\cdot)$, $\boldsymbol{D}\tau(\cdot)$	first return time, derivative of first return time
γ	invariant set, closed orbit
Σ	transversal cross section
$T\Sigma$	tangent space of transversal cross section
$\boldsymbol{x}^*$	fixed point
ψ	coordinate chart for local coordinates
SS	switching sequence of hybrid system
θ	normalized time
$\tilde{\boldsymbol{x}}$	enhanced state vector
λ	eigenvalue of $\boldsymbol{DP}$
σ	singular value of $\boldsymbol{DP}$
V	discrete-time Lyapunov function
v	locomotion progression velocity

Balance Control

$\boldsymbol{f}$, $\boldsymbol{g}$, $\boldsymbol{G}$, $\boldsymbol{h}$	components of control system
$\boldsymbol{u}_{\text{nom}}$, $\boldsymbol{u}_{\text{corr}}$	nominal/corrective control signal
$\boldsymbol{v}_{\text{nom}}$, $\boldsymbol{v}_{\text{corr}}$	nominal/corrective control signal for feedback controlled system
$\boldsymbol{A}$	matrix for linear equation of constraint compliance
$\boldsymbol{b}$	right hand side for linear equation of constraint compliance
$\boldsymbol{W}$	weighting matrix
$\boldsymbol{I}$	identity matrix
$\boldsymbol{\tau}$	torque vector of refined model
$\boldsymbol{K}$	parameter matrix of motor model
$\boldsymbol{F}_{\text{push}}$	pushing force

Abstract

This thesis investigates the variable contact situations of rigid robot feet in legged robot locomotion. One major goal is to include the rotation around foot edges in locomotion cycles. For walking robots they are referred to as the toe roll phase and the heel roll phase. The alternation between underactuated motion phases and completely actuated motion phases is believed to contribute in a decisive manner in enabling dynamic locomotion. Dynamic locomotion comprises e.g. walking, running, hopping, standing up, and many more motion patterns characterized by variable contact with the environment. A control-theoretic approach to legged robot locomotion is presented that uses an event-based hybrid (discrete-continuous) model with underlying rigid-body assumption. Events occur whenever the ground contact situation changes; this is either when a contact resolves or when a contact is established. The continuous time dynamics that is generally different for all contact situations is allowed to switch at these contact changes. Then discontinuous collision behavior is taken into account. To obtain periodic locomotion cycles, a trajectory planning algorithm is proposed where the boundary value problem is solved that relates the initial and final configuration of the robot. The resulting periodic robot locomotion is investigated for orbital stability using Poincaré map analysis of the hybrid trajectories. Finally, a hybrid control strategy is presented for balance control which makes use of the invariance control method. Throughout this thesis, the methods are demonstrated for three example robots: a compass gait robot, a monoped robot, and a gymnast robot.

Kurzfassung

In dieser Arbeit werden die variablen, wechselnden Kontaktsituationen zwischen Boden und Füßen bei zweibeiniger Roboterfortbewegung untersucht. Schwerpunkt dabei ist die Integration von nichtvollaktuierten Kontaktarten in die Bewegung, wie das Abrollen über Ferse oder Zehen bei einer Laufbewegung. Speziell der Wechsel zwischen unteraktuierten Kontaktsituationen und vollaktuierten Kontaktsitationen prägt den Charakter dynamischer Fortbewegung. Bewegungen wie gehen, rennen, springen oder aufstehen sind nur Beispiele für Bewegungen, bei welchen das Eingehen und das Lösen von Kontakten wichtiger Bestandteil ist. In der vorliegenden Arbeit wird ein regelungstechnischer Ansatz zur Realisierung von Fortbewegung zweibeiniger Roboter vorgestellt. Basis ist ein ereignisorientiertes hybrides (diskret-kontinuierliches) Modell der Starrkörperdynamik des Roboters. Ereignisse treten auf, wenn sich die Kontaktsituation verändert, also wenn sich ein Kontakt löst oder wenn ein neuer Kontakt eingegangen wird. Die kontinuierliche dynamische Beschreibung unterscheidet sich je nach Kontaktsituation. Bei einem Kontaktartwechsel müssen außerdem Kollisionen mitberücksichtigt werden. Eine Methode zur Trajektorienplanung für periodische Fortbewegung wird vorgestellt. Dazu wird das Randwertproblem gelöst, dass den Anfangszustand der Bewegung mit dem Endzustand verknüpft. Die orbitale Stabilität der resultierenden Trajektorien wird mit Hilfe von Poincaré Abbildungen untersucht. Abschließend wird ein Verfahren zur Gleichgewichtsregelung vorgestellt, basierend auf einer Modifikation der Invarianzregelung. Begleitend zur allgemeinen Darstellung werden die vorgestellten Methoden an drei Beispielrobotern illustriert: dem *compass gait* Roboter, einem *monoped* Roboter und einem Gymnastikroboter.

1 Introduction

The human environment is constructed by humans and in first place for humans. For robots that are meant to assist in a human environment, it is thus potentially easier to cope with this environment if they have humanoid properties; see Fig. 1.1 for an assistance robot. There are yet some areas where robot skills already exceed human skills, even in a human-made environment. One example are tasks that have to be repeated many times in exactly the same way, e.g. at an assembly line. In other areas robot skills are nevertheless still inferior to human skills. So it is assumed that it will still take until 2050 when a robot soccer team has a chance to win a match against a human team [107]. Figure 1.1 shows a snapshot from a robot soccer match in the humanoid league at the championship RoboCup 2006. The reasons for todays inferiority are manifold: The artificial vision systems of humanoid robots can not compete with the human eye and visual pathway in the brain, and also intelligent decision making in cooperation with team members is not developed far enough. However, still also machine locomotion on legs is one of the major problems. It is not yet as stable, dexterous, versatile, and fast as needed. Biped robots that perform a bicycle kick and stand up afterwards without any damage are hard to imagine if one has in mind even todays most powerful walking machines like Honda Asimo [63] or the entertainment robot Sony Qrio [94].

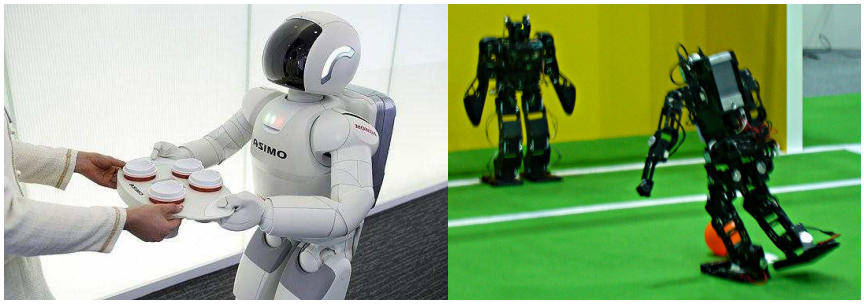

Figure 1.1: Robots in human environments. Left: Honda Asimo as assistance robot (© Honda Motor Corp.) [63]. Right: Robots playing football at RoboCup 2006 (© RoboCup Humanoid League) [107].

Research on legged locomotion, monoped, biped, or even multiped, is an interdisciplinary area. Approaches from kinesiology are biology-oriented, and it is believed that a thorough understanding of biological locomotion principles is the key for successful legged robot construction and control. Areas of research include for example the investigation of the energetics of human and animal locomotion. Here, Cavagna *et al.* [29] were one of the first to find out that human locomotion on level ground has a large portion of passive leg swinging. Another research area is the construction of artificial muscles providing actuators for robot design with energy storage and stiffness similar to human muscles [33]. Closely related are approaches from rehabilitation science, where e.g. actuated orthoses are used

to substitute limbs after an amputation [17]. From a mechanical engineering point of view the best construction of robots may but most not be humanlike: Still all humanoid robots have only a fraction of the joints that human beings have. There, the Sony QRIO [94] with yet 38 joints is one of the robots with the highest number of degrees of freedom. In electrical engineering, control theory is essential for realization of joint control for powerful and variable drives. Actual topics of research are joints that are capable to switch between actuated and passive modes, and complex, nonlinear control laws are needed to enable stable switching between the modes [139, 140]. Nonlinear control theory also contributes with results on stability and stabilization of the overall locomotion cycle to compensate for unexpected disturbances. In addition, advanced numerical algorithms for optimal control are used to calculate optimal motion [39].

Important for the realization of a walking motion and interesting from a control theoretic point of view is the stability of locomotion. A first very rough and general definition of stability implies that a stable robot is able to continue locomotion even in the presence of unexpected disturbances. Instability leads to falling without a possibility to return to the original desired pattern of motion. Local analysis of walking trajectories describes the reaction on small disturbances. The return to the desired motion is generally fast and the compensation motion is close to the desired motion. Mombaur *et al.* [90] present an optimal control problem that finds the trajectory in which convergence to the desired solution is fastest after a disturbance occurred. Much more difficult are global statements or approaches where an appropriate reaction on any kind of disturbance has to be available. In this case it might be necessary to leave the planned motion pattern, insert a correction motion, and continue the desired motion after some delay.

This thesis presents a control-theoretic approach to legged locomotion. Of special interest is the variable ground contact situation of a robot foot resulting in a consecution of ground contact situations with different dynamical properties in one locomotion cycle. A hybrid (discrete-continuous) model is used as basis for trajectory planning and control because hybrid models can account for the variable dynamical properties.

1.1 Legged Locomotion and Hybrid Systems

Essential for theoretical analysis and simulation as well as for application of model-based control methods is a mathematical model of the robot dynamics. This model can be used to preview robot motion when a torque trajectory or a control law is applied thus providing a basis for trajectory planning, control design, and stability analysis. The formulation and degree of abstraction of the model determines the possible areas of application. A structurally simplified model that considers only a single support phase and neglects collisions might still be useful for trajectory planning if the foot is desired to touch the ground with zero velocity. For simulation, in contrast, the model has to be refined because collisions between foot and ground will nevertheless occur under disturbances. In general, for a comprehensive control-theoretic analysis, models are used that consider the exchange of the stance legs with the associated collision. It is believed that the collisions between feet and ground in a locomotion cycle are a major characteristic of legged locomotion [72].

Two distinct modeling approaches for numerical simulation are found in literature: Under the assumption that ground or foot are slightly compliant, the robot feet are allowed to just a little intrude into the ground. The ground contact is thereby modeled by spring-damper elements [38]. In contrast, one can also assume a rigid ground, where after making contact the robot foot either sticks to the ground or the contact between foot and ground immediately dissolves [72].

In both approaches the dynamical description of the robot depends on the actual robot configuration. The more contact points between ground and robot, the more degrees of freedom are either constrained by spring-damper dynamics or constrained by constraint equations. The model is thus required to account for changes in the contact situation when a foot touches ground or detaches from ground. A hybrid modeling framework can be used to formalize the dynamical description of these contact changes.

Most hybrid modeling frameworks allow an event-based description of the underlying dynamics. That means, the model is continuous almost everywhere with ordinary differential equations to describe the behavior. If the trajectory arrives at a specific submanifold of the state space, an event is triggered and specific event actions are allowed to take place. For example, it is possible to exchange the dynamical properties to account for a modified number of contact points [20]. It is thus possible to combine different ordinary differential equations in one model. Some hybrid system formulations also allow discontinuities in the state vector as event action [26]. State discontinuities are necessary to account for velocity jumps that result when collisions are assumed to be instantaneous.

A theory of hybrid dynamical systems has been first proposed by Witsenhausen in 1966 [136], and since then various publications proposed hybrid modeling frameworks, see for example [25] for an overview. A unified theory with general results is still in progress. A description of an application in the hybrid framework is advantageous since it allows to use results that are formulated for general problems. And also the reverse: Results that are derived for specific applications can be generalized for arbitrary hybrid dynamical systems with equivalent structure.

The variety of topics in research on hybrid control systems is large. Many results and methods that are available for continuous control systems have yet been adapted to be applicable also for hybrid systems. So there are existence and uniqueness results for solutions and stability analysis tools for fixed points of hybrid dynamics using Lyapunov methods [75]. Also spadework from nonlinear control theory is in focus of adaptation for hybrid system control, e.g. identification methods [92] or observer design methods [12]. Actual areas of research are reachability, verification, and safety [7], with strong interest in industrial application [44]. In recent years hybrid systems with stochastic properties have been investigated [65].

The range of application of hybrid dynamical system modeling is manifold. Many systems subject to control have intrinsic hybrid properties. That means, neither a continuous nor a discrete dynamical description is satisfactory and covers all important properties. Examples for this class, besides legged robots, are mechatronic systems like robot hands where grasping implies repeatedly making and dissolving contact [111]. Also manipulators that come in contact with the environment have similar modeling properties. See for example Botturi *et al.* [18] for hybrid optimal control of a puncturing task. In automotive engineer-

ing, hybrid modeling finds application in engine modeling and control [13]. Also in process engineering, hybrid models are sometimes the only means appropriate to describe the dynamics of an underlying reaction process [43]. Further application of hybrid dynamical system models in nontechnical applications are found for example in models of neuronal activity [37]. Also systems where the hybrid character is not intrinsic are sometimes subject to hybrid control theory if the control strategy is chosen hybrid. In controller design hybrid systems result if a switching control law is applied as for example in invariance control [137], where a nominal controller and a corrective controller alternate.

Certainly, legged robotic systems also have very special properties that make them unique in the hybrid system context, and where special approaches have to be found that cannot be shared with the whole range of hybrid dynamical systems. In first place many insights can be drawn from observing human locomotion behavior. Balancing on one foot is a unique problem that is not carried forward to many other applications. In control theory much work is done on inverted pendulum control [8], and certainly results can be used to improve legged locomotion since a balancing system is very similar to an inverted pendulum. Another particular characteristic of legged locomotion is that underactuated contact phases occur in alternation with completely actuated contact situations where the number of actuators is equal to the degrees of freedom of the robot and that collision separate the contact situations.

1.2 Passive Joints versus Active Joints

From 1990 on, starting with McGeers findings on the similarities of a rolling rimless wheel and an unactuated walking motion [88], passive walking is a field of intensive and successful research [35]. Passive walking machines do not need actuation in the joints, therefore locomotion is only possible downhill, where the energy loss due to the collisions with the ground is compensated by potential energy. The appeal of passive walking is that the robotic system realizes an inherent, natural dynamical trajectory. The main drawback is the lacking versatility of passive walking machines. With their natural dynamics most of them are only robust against small changes in inclination of the walking plane.

A first step to compensate for this drawback is to allow small supporting controllers that mainly enhance the robustness on changes in inclination and widen the region of attraction resulting in acceptance of small disturbances, see results on nearly-passive walking [80]. Still, comparison of the specific costs of transport reveals that nearly-passive walking machines can compete with humans in walking at constant speed, thereby consuming a tenth of the energy of other walking machines [36]. Nevertheless, versatility cannot be enhanced as much as to allow motion patterns different from walking at constant speed. Examples for motion patterns that are not possible with standard passive approaches are climbing stairs, stopping, falling and standing up, etc. All of these are task which, also for a human, require powerful joints.

Versatility demands justify the need of full actuation that means actuators in every joint. Robots with full actuation are for example the HONDA Asimo robot [63], the entertainment robot Sony QRIO [94], the Toyota partner robots [129], the japanese research

platform HRP [78], or the Johnnie robot and its successor Lola from TU München [76]. The present challenge is to link the energy-efficiency and natural motion generation of passive walking with the versatility of actuated robots. Passive and nearly-passive locomotion is well suited for periodic tasks that do not need much force, whereas actuation is necessary for any other tasks. Yet, besides the nearly-passive walkers, robots and robot control approaches are developed that combine passive and actuated properties: One group of robots are those that have mainly actuated joints, but some joints like the knee joint are passive or at least allowed to switch between an actuated mode and a passive mode, like the UT-Theta from University of Tokyo [98]. Another robot type is actuated in all joints, but the contact between robot foot and ground is assumed to be passive. The foot has only point contact with the ground, and the lower leg rotates freely around the contact point. Acceleration of this passive joint is a consequence of dynamic coupling between active and passive degrees of freedom. An example robot is the RABBIT platform [31] that is designed specifically for running. Finally, there are robots with actuated joints and ground contact that alternates between free rotation around foot edges, ballistic flight phases, and stable support. Allowing for alternation of the contact type between foot and ground is essential for dynamic motion patterns, as for example hopping [14]. Robots of this type must not differ from fully actuated robots in construction, but more in the methods that are applied to realize locomotion with the variable ground contact situation [34]. In Fig. 1.2 robots with different passivity properties are shown for illustration.

Figure 1.2: Walking robots from left to right: McGeer's passive walker (© Simon Fraser University, Canada) [88], Collins and Ruina's nearly-passive walker (© Cornell University, US) [36], Honda Asimo (© Honda Motor Corp.) [63], UT-Theta (© University of Tokyo, Japan) [98], and RABBIT (© INRIA, France) [31].

Trajectory planning for robots with variable ground contact is a challenge, although there are planning methods that are applicable for robots with only actuated joints as well as for robots with passive joints or with a mixture of passive and actuated joints. For example, the notation framework and solution algorithm in optimal control trajectory planning does not explicitly make a difference between actuated joints and non-actuated joints. Only the complexity of the numerical problem increases for passive joints and thus the convergence rate to solutions decreases or regions of attraction shrink, and often a numerical solution is not found. Walking trajectories with different consecutive contact situations under consideration of constraints were determined by Buss *et al.* [27] and in

parallel by Denk *et al.* [39] with the direct collocation algorithm DIRCOL [131]. But in both approaches the transition times between different contact situations were preset and not subject to optimization. In addition the considered contact situations were completely actuated. Approaches including an underactuated contact situation to humanoid walking trajectory planning using DIRCOL failed until now [16] and still present a challenge. One of the few approaches where optimization-based trajectory planning was successful, even with a passive joint, was presented by Fujimoto [45] for a five-link biped.

Nevertheless many methods for trajectory planning that were developed for actuated robots fail for underactuated machines. These methods include in particular any static planning approaches [68] because of the dynamic coupling between actuated and nonactuated joints. The inverted pendulum method [77] is, in principal, applicable at least for robots with nonactuated ground contact. For this method the actuated ankle joint is often used to stabilize the walking motion. There is no reference yet how inverted pendulum methods apply to robots with alternating actuated and underactuated dynamical description.

1.3 Stability and Control in Legged Locomotion

For stability and stability control two definitions have to be discerned carefully. On the one hand stability is often used in the meaning of balance, and then stability control is the equivalent to balance control. On the other hand stability is defined in an orbital sense. Balance controllers prevent the robot from falling via unwanted tilting around the foot edges and are thus used for robots with full actuation to prevent underactuation. The control algorithm is often based on measurement of the Zero Moment Point that was introduced by Vukobratović in 1969 [133] and that provides a measure for balance in the distance of the Zero Moment Point to the closest foot edge. In most approaches, it is switched between the nominal controller and a corrective controller where the corrective controller acts whenever a violation of the Zero Moment Point invariance is predicted [99]. Other approaches modify the desired trajectory to avoid balance loss [66]. The resulting deviation from the original desired trajectory, however, makes it necessary to introduce an additional foot landing time control [66].

Orbital stability for a periodic locomotion cycle implies that small deviations from the desired trajectory can be compensated. Investigation of orbital stability began with Raiberts pneumatically actuated hopping robots [105] in the 1980s, and similar robots and concepts are used today to explore robot running. Simplifications of the governing equations of motion that assume legs with springs yield analytical results that give insight in the mechanisms of stability of running [47]. Stability is commonly investigated by first-return maps (Poincaré maps) [60]. Stability analysis for the periodic continuous system is then reduced to stability analysis of an underlying discrete map that considers the orbit only once in a period. In the approach from Westervelt *et al.* [134] the first-return map of the hybrid zero dynamics is considered while the walking motion is achieved by an input-output linearizing controller. The method was originally presented for a robot with underactuated ground contact. In recent times, modifications of the method have been published that account for variable ground contact. This occurs for running where a ballistic phase and a single support phase alternate [32] and also for walking where an actuated

contact situation alternates with an underactuated contact situation [34]. While stability in this approach is obtained and improved by offline optimization of the first-return map, in Tedrakes approach [128] stability is optimized online by reinforcement learning. This strategy is successfully implemented in the robot Toddler.

Few other approaches are not based on first-return maps, nevertheless providing provable stability for walking robots. Song *et al.* [123] presented an approach splitting the coordinates into a part transverse to the trajectory and a tangential part, finally designing a controller by solving linear matrix inequality equations. Duindam *et al.* [41] present an approach based on a port-Hamiltonian formulation of the equations of motion. The robot is fully actuated in this particular approach. Here, the question arises what role orbital stability is playing for fully actuated robots. Hurmuzlu gave reference in 1993 [70] that also for actuated robots, unstable periodic trajectories exist. In this context, also results of Djoudi *et al.* [40] are interesting: they consider a Zero Moment Point controlled actuated robot as underactuated system in the sense that there are more outputs than inputs if the Zero Moment Point is considered as additional output.

The above cited approaches consider stability either in the sense of balance or as local property of trajectory control. Wieber [135] points out that still this is not comprehensive enough. He proposes to consider a viability region that comprises all states of the robot that are consistent with force constraints of the ground and excludes conditions where the robot is fallen. The primary goal of control is then to keep the viability region invariant which results in global stable behavior of the legged robot.

1.4 Main Contribution and Outline of Dissertation

Passive Contact Situations. One of the goals in legged robots research is to achieve energy-efficient and nevertheless versatile locomotion. In most actuated robots, passive rotation around foot edges is not used. Including passive rotation around foot edges is the most intuitive approach to adapt fully actuated robots to the challenge of dynamic locomotion and furthermore does not need major reconstruction of the robots. Modeling, trajectory planning, stability analysis, and control have to be designed such that they can cope with passive ground contact on the one hand and alternating ground contact situations, that means, switching between actuated and underactuated behavior, on the other hand.

In this work, robots are considered that are allowed to take underactuated ground contact. For simplification only planar robots with simple geometrical construction are analyzed to concentrate on the problem of variable ground contact. A compass gait robot, a monoped robot, and a gymnast robot are subject of investigation, see Fig. 1.3 for illustration. The gymnast robot with five links, two feet, and three possible contact situations for every foot combines the features of the compass gait with one possible contact situation for the feet and the features of the monoped with three possible contact situations for the single foot. The robots are introduced in Chap. 2. Variable contact situations of robotic feet are still rarely considered. One reason is certainly that models and the following

trajectory synthesis and stability investigation become more complicated. See [34, 126] for comparable approaches.

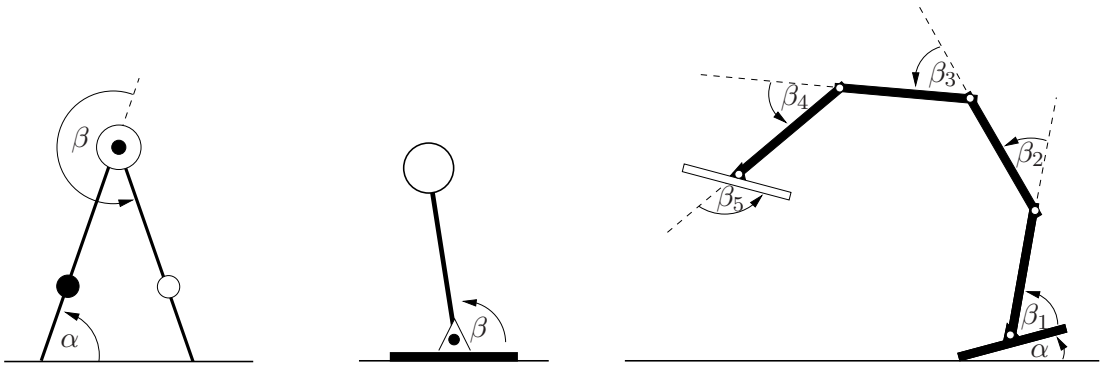

Figure 1.3: Investigated legged robots. Left: Compass gait robot. Middle: Monoped robot. Right: Gymnast robot.

Hybrid Modeling. Dynamical models are needed that are tractable for control theory. On the one hand, a model is required to approximate real behavior as good as possible. On the other hand, the model should be as accessible for control theory as possible. The model is basis for trajectory planning, stability analysis, and control.

In this work a hybrid model is chosen. The hybrid formulation of the legged locomotion problem is convenient if only a small number of contact situations is considered. If in addition, double support and sliding on the ground are taken into account, e.g., a formulation as a complementarity problem is useful. In the following, research on legged locomotion is embedded in hybrid systems theory. In Chap. 2 the three robots are described by their hybrid models. The basics for dynamical modeling, as e.g. derivation of continuous descriptions, collision equations, and contact forces, are presented in advance.

Trajectory Planning. The trajectory planning method has to account for the hybrid character of the legged robot model. Interesting trajectories include switching of the ground contact situation. Some of the contact phases might even be underactuated. Examples are walking, hopping, and similar patterns. General optimal control approaches are not easy to apply since the trajectory planning problem is complex. The main difficulties are the underactuatedness and the unknown transition times between different contact situations.

In this thesis, a simplified problem is solved for trajectory planning that reduces the optimal control problem to the underlying boundary value problem. The trajectory planning method is specified and illustrated using the example robotic systems in Chap. 3. Solving for trajectories as solutions of boundary value problems is common in the literature [69]. The approach presented in this thesis expands the boundary value trajectory planning to robots with several contact phases where multi-point boundary value problems arise.

Stability of Periodic Locomotion. Important for the practibility of trajectories is their orbital stability. Unstable trajectories are completely useless because only a tiny distur-

bance results in non-correctable deviations from the desired trajectory and, for a legged robot, finally to falling. Poincaré map analysis is applied to determine orbital stability. Poincaré maps are an analysis tool that is usable for stability analysis of cyclic orbits of arbitrary hybrid systems.

In Chap. 4 the conditions for stability are presented, and stability is discussed for trajectories of the example robots. Due to the nonlinearity of the equations of motion, stability results can only be obtained numerically. Since legged robots are supposed to perform in versatile tasks, the ability of periodic locomotion has to be enhanced making for example stopping and starting possible, as well as accelerated and decelerated motion. It is shown for the gymnast robot in simulation how switching between trajectories determined in Chap. 3 enables decelerated and accelerated walking. It is used that the trajectories exist as parameterized family where parameters can be changed at a set of allowed times. There, the mathematical foundation is the intersection of basins of attractions. Still, control for walking at constant speed presents a challenge and approaches beyond that are rare. Although the trajectory generation and stability control presented in this work is still basic, it is believed that further development towards dexterous locomotion is one of the important goals.

Balance Control. Improved balance control is decisive for dynamic locomotion. Balance control avoids non-actuated tilting around foot edges and should be able to compensate a range of disturbances as large as possible. Balance control in general results in a non-minimum phase control system. There, only switching between different control laws yields stability of the overall control system. A key question is also, how the nominal control task interacts with necessary balance control.

The concept of balance control is embedded in the control theoretic framework of invariance control in Chap. 5, using the Zero Moment Point as output that has to be kept in an invariance region. The presented description provides a general and clear concept for balance control and does not assume a specific robot structure as this is the case for many other published approaches. Also, the control theory based approach will allow an analytic analysis of balance control, concerning stability and robustness.

2 Modeling of Legged Locomotion

2.1 Introduction and State of the Art

Locomotion on legs is characterized by repetitive contacting and detaching of feet and ground which results in a sequence of distinguishable contact situations. A transition between two contact situations occurs if contact forces or contact moments become zero or if a robot foot collides with the ground with non-zero velocity.

A hybrid, event-based modeling approach is well-suited to describe the dynamics of legged locomotion. It allows to describe the variable dynamical properties of the different ground contact situations. Furthermore, a collision model is included that quantifies the instantaneous changes in joint velocities. An event for a legged system is either the incorporation of a constraint when a robot foot makes contact or the elimination of a constraint when a contact force becomes zero. Hence, in order to detect the occurrence of events, contact forces and moments as well as the position of the robot foot edges have to be supervised.

Different formulations of the hybrid model are possible depending on how to include ground contact. If the ground is chosen compliant, small penetrations of the foot into the ground are allowed and ground contact is modeled by spring-damper elements, see [38]. If ground and robot foot are assumed to be rigid bodies, penetration is not allowed and the transition between contact situations is instantaneous. Both modeling assumptions yield a hybrid model. For compliant ground models, the dimension of the generalized coordinates is constant, independent from the contact situation. The constrained degrees of freedom are coupled with the ground by spring-damper elements. Rigid ground models allow to use minimal coordinates for every contact situation. In the following, a description in minimal coordinates is chosen that is advantageous for control theory since the complexity of control is reduced if the model equations are simple in structure. Another variant for modeling of legged locomotion is the complementarity framework [72]. Therefore a solution of the dynamical equations requires the repetitive solution of linear complementarity systems. The complementarity formulation is based on the assumption of rigid ground. However, the description of the individual contact situation is not in minimal coordinates.

A formal introduction of legged robotic systems is given in Sec. 2.1.1, and the hybrid system modeling framework used throughout the thesis is introduced in Sec. 2.1.2. Section 2.2 provides the prerequisites needed for modeling of legged systems in any modeling framework. Then in Sec. 2.3, hybrid modeling of legged robots is formalized and illustrated in examples. In Sec. 2.4 two alternative modeling approaches are shortly introduced and discussed, complementarity modeling and compliant ground modeling. The chapter is summarized in Sec. 2.5.

2.1.1 Legged Robotic Systems

Legged robotic systems are made up from rigid links that are connected by rotational or translational joints. A foot is a special link that may but must not take ground contact. For simplicity only planar constructions are considered where motion is restricted to the xy-plane of a reference frame. Figure 2.1 displays example robots.

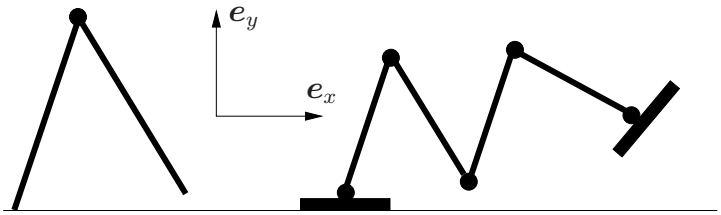

Figure 2.1: Examples of legged robots. Left: Robot with two links, two feet, and one actuated joint. The contact between foot and ground is passive. Right: Robot with four links, two feet, and five actuated joints. The robot is fully actuated as long as no rotation around foot edges occurs.

The robot is driven by motors in the joints that apply torques. A motor-driven joint is also termed an actuated joint. If a joint is not motor-controlled, it is called passive or unactuated. With this convention, the degrees of freedom that connect ground and robot are passive links, see Fig. 2.2 for an example. A contact situation is underactuated if the number of actuators is smaller than the number of degrees of freedom.

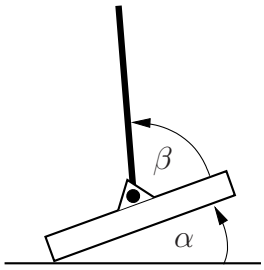

Figure 2.2: Passive and actuated links. The orientation α of the foot is passive, the deflection of the leg β is assumed to be actuated.

The generalized coordinates $\boldsymbol{q}$ define the configuration of the mechanical system and are hence the joint angles. Often the set of generalized coordinates is termed joint space. In what follows, greek letters ξ, η, α_i, and β_i are used for the components of the joint vector, where ξ and η are used for cartesian distances. Rotational deflections are generally described by angles α_i and β_i, where passive rotational degrees of freedom are labeled α_i and actuated rotational degrees of freedom are labeled β_i. The state of the control system comprises generalized coordinates $\boldsymbol{q}$ and associated generalized velocities $\dot{\boldsymbol{q}}$ summarized in the state vector

$$\boldsymbol{x} = \begin{pmatrix} \boldsymbol{q} \\ \dot{\boldsymbol{q}} \end{pmatrix}. \tag{2.1}$$

Furthermore, external torques that act in the joints are denoted by $\boldsymbol{u}$. A cartesian description of the posture with cartesian coordinates of the links is in general redundant.

2.1.2 Hybrid Control Systems

Many systems that appear in control applications can neither be described by a purely continuous nor by a purely discrete model. The continuous and the discrete aspects of the dynamics are coupled in a way such that neglecting one of the aspects yields useless results for modeling and controller design. In the literature those systems are termed hybrid (discrete-continuous) dynamical systems. See Witsenhausen's publication in 1966 [136] for one of the first definitions of a hybrid system, the book series "Hybrid Systems" [2–5, 56] for a collection of publications on hybrid systems from 1993 to 1999, and the collection edited by Engell *et al.* [42] in 2002.

The hybrid state model (HSM) describes discrete-continuous control systems and is outlined in the following. The HSM was introduced for control of robot finger grasping [109]. For example [24] gives a detailed description of this hybrid modeling framework in a general context.

Hybrid State Vector. In accordance with the state vector definition for a purely continuous system, the hybrid state vector $\boldsymbol{\zeta}(t)$ is composed from the continuous state vector $\boldsymbol{x}(t) \in \mathbb{R}^n$ and the discrete scalar state variable $x_d(t) \in \mathbb{Z}$. Let N_d denote the number of possible discrete states, then $x_d \in \{i_1, i_2, \ldots, i_{N_d}\} \subset \mathbb{Z}$ and

$$\boldsymbol{\zeta}(t) = \begin{pmatrix} \boldsymbol{x}(t) \\ x_d(t) \end{pmatrix} \in \mathbb{R}^n \times \mathbb{Z}. \tag{2.2}$$

For hybrid systems without external excitation, the system behavior is manifested for all times $t > t_0$ if the state vector $\boldsymbol{\zeta}$ is known for an initial time t_0.

Inputs and Outputs. Possible external inputs are divided into continuous control inputs $\boldsymbol{u}(t) \in \mathbb{R}^m$ and discrete control inputs $u_d(t) \in \mathbb{Z}$. In addition, hybrid systems are allowed to have continuous as well as discrete outputs, denoted as $\boldsymbol{y}(t)$ and $y_d(t)$. The outputs are determined by an output function $\boldsymbol{h}(\boldsymbol{x}, \boldsymbol{u}, x_d, u_d, t)$.

Continuous Dynamics. It is assumed that the system shows continuous behavior almost everywhere and that the dynamics for a constant discrete state x_d is modeled by ordinary differential equations.

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, x_d, t)$$

The vector field $\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, x_d, t)$ is a smooth function of the continuous state $\boldsymbol{x}$, the continuous control input $\boldsymbol{u}$, and of time t. If the discrete states are $x_d = 1, 2, \ldots, N_d$, vector field switches between the corresponding vector fields $\boldsymbol{f}_1(\boldsymbol{x}, \boldsymbol{u}, t), \ldots, \boldsymbol{f}_{N_d}(\boldsymbol{x}, \boldsymbol{u}, t)$ are allowed

and realized by choosing $\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, x_d, t)$ as follows:

$$\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, x_d, t) = \sum_k \delta_{k,x_d} \boldsymbol{f}_k(\boldsymbol{x}, \boldsymbol{u}, t) = \begin{cases} \boldsymbol{f}_1(\boldsymbol{x}, \boldsymbol{u}, t) & \text{if } x_d = 1 \\ \boldsymbol{f}_2(\boldsymbol{x}, \boldsymbol{u}, t) & \text{if } x_d = 2 \\ \vdots & \\ \boldsymbol{f}_{N_d}(\boldsymbol{x}, \boldsymbol{u}, t) & \text{if } x_d = N_d \end{cases} \tag{2.3}$$

Here the Kronecker delta δ_{k,x_d} is 1 if $k = x_d$ and zero elsewhere.

Discrete Dynamics. The occurrence of an event is defined through the extended state vector $(\boldsymbol{x}, \boldsymbol{u}, x_d, u_d)$ crossing one of the transition surfaces S_i that are denoted by

$$S_i : \ s_i(\boldsymbol{x}, \boldsymbol{u}, x_d, u_d) = 0, \ i \in I.$$

The set $I \subset \mathbb{Z}$ is a finite index set. Discontinuous behavior in the hybrid state is allowed at event times and is realized by jump (transition) maps $\boldsymbol{\varphi}_i(\boldsymbol{x}, \boldsymbol{u}, x_d, u_d, t^-)$ that determine the hybrid state

$$\boldsymbol{\zeta}^+ = \boldsymbol{\varphi}_i(\boldsymbol{x}, \boldsymbol{u}, x_d, u_d, t^-)$$

immediately after the event given the hybrid state $\boldsymbol{\zeta}^-$ immediately before the event. The notation $\boldsymbol{\zeta}^+ = \boldsymbol{\zeta}(t^+)$ denotes the successor state (limit from the right) of $\boldsymbol{\zeta}$ at time t. The hybrid state $\boldsymbol{\zeta}^- = \boldsymbol{\zeta}(t^-)$ is the predecessor state (limit from the left). The transition map $\boldsymbol{\zeta}^+ = \boldsymbol{\varphi}_i(\boldsymbol{x}, \boldsymbol{u}, x_d, u_d, t^-)$ allows a discontinuity in the continuous state $\boldsymbol{x}$ and a reset of the discrete state x_d. The latter may then result in a vector field switch.

Sometimes it is convenient to split the jump map into two parts to separate the continuous state behavior from the discrete state behavior. Then $\boldsymbol{\varphi}$ is split into mappings $\boldsymbol{g}$ and g_d:

$$\begin{pmatrix} \boldsymbol{x}^+ \\ x_d^+ \end{pmatrix} = \boldsymbol{\varphi}(\boldsymbol{x}, \boldsymbol{u}, x_d, u_d, t^-) = \begin{pmatrix} \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{u}, x_d, u_d, t^-) \\ g_d(\boldsymbol{x}, \boldsymbol{u}, x_d, u_d, t^-) \end{pmatrix}$$

In many applications transition surfaces are only valid for one particular transition from x_d^- to x_d^+ and will be denoted by $s_{x_d^-,x_d^+}(\boldsymbol{x}, \boldsymbol{u}, u_d)$. Then the corresponding jump map is $\boldsymbol{\varphi}_{x_d^-,x_d^+}(\boldsymbol{x}, \boldsymbol{u}, u_d, t^-)$.

To fit the hybrid state model, an isomorphism

$$s_i(\boldsymbol{x}, \boldsymbol{u}, x_d, u_d) = \begin{cases} s_{x_d^-,x_d^+}(\boldsymbol{x}, \boldsymbol{u}, u_d) & \text{if } i(x_d^-, x_d^+) = i \\ 1 & \text{else} \end{cases}$$

is introduced, where $i : \mathbb{Z} \times \mathbb{Z} \to I$.

Discrete-Continuous Dynamics. In summary, the above introduced notation allows a compact form for the hybrid state model:

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, x_d, t) \quad \text{if} \quad s_i(\boldsymbol{x}, \boldsymbol{u}, x_d, u_d, t) \neq 0 \text{ for all } i \tag{2.4a}$$

$$\boldsymbol{\zeta}^+ = \boldsymbol{\varphi}_j(\boldsymbol{x}, \boldsymbol{u}, x_d, u_d, t^-) \quad \text{if} \quad s_j(\boldsymbol{x}, \boldsymbol{u}, x_d, u_d, t^-) = 0 \text{ for } j \in I \tag{2.4b}$$

$$\begin{pmatrix} \boldsymbol{y} \\ y_d \end{pmatrix} = \boldsymbol{h}(\boldsymbol{x}, \boldsymbol{u}, x_d, u_d, t). \tag{2.4c}$$

Figure 2.3 illustrates this structure.

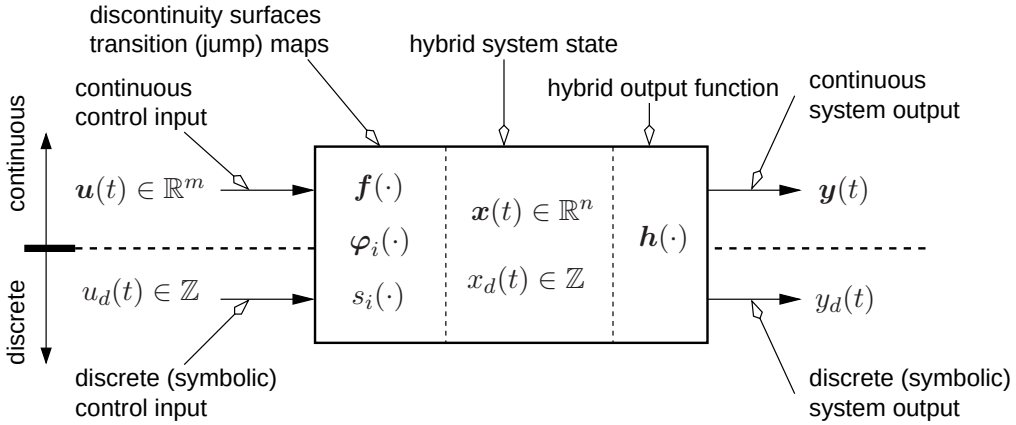

Figure 2.3: Interaction of continuous and discrete aspect in HSM. Figure adapted from [25].

Hybrid State Model and State Space Model. The hybrid state model (2.4) provides an extension to the state space model

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t)$$
$$\boldsymbol{y} = \boldsymbol{h}(\boldsymbol{x}, \boldsymbol{u}, t)$$

that is used in control theory [73]. Hence, the hybrid state model offers a framework for hybrid system control theory. Many other frameworks from different application fields can be found in literature. Buss [25] outlines and compares between approaches from Tavernini [127], Back *et al.* [11], Nerode *et al.* [97], Antsaklis *et al.* [6], Brockett [21], and Branicky *et al.* [19]. Other propositions come from Peleties *et al.*, [102], Michel *et al.* [89], or Simić *et al.* [115].

Hybrid Modeling of Mechatronic Systems. For mechatronic systems, the continuous state vector $\boldsymbol{x}$ has a position component $\boldsymbol{q}$ and a velocity component $\dot{\boldsymbol{q}}$, see (2.1). The vector $\boldsymbol{q} \in \mathbb{R}^{n_q}$ summarizes the generalized coordinates. Generalized coordinates are a minimal set of variables necessary to describe the posture. Since $\boldsymbol{x} \in \mathbb{R}^n$, $n = 2n_q$.

For mechatronic systems, the discrete state variable x_d can for example be used to code the contact situation with the environment, different charging, or different control modes. For the legged robotic systems considered henceforth, the discrete state codes the ground contact situation of the feet.

2.2 Underlying Mechanical Equations

The following derivation of equations of motion (Sec. 2.2.1), collision law (Sec. 2.2.2), and transition conditions (Sec. 2.2.3) provides the components for legged robot models in any modeling framework.

2.2.1 Equations of Motion

The equations of motion connect the input torques $\boldsymbol{u}$ and the resulting trajectories in joint space $(\boldsymbol{q}, \dot{\boldsymbol{q}})$ in a differential way. A robot is a mechanical system and robot motion results from an equilibrium of external torques and internal forces, respectively moments. A Lagrangian analysis can be used to derive equations of motion using generalized coordinates $\boldsymbol{q} = (q_1, \ldots, q_{n_q})^T$ that form a minimal set of coordinates specifying the posture of the robot. The generalized coordinate vector contains joint angles as well as the cartesian position of one reference point on the robot relative to the origin. We use $\dot{\boldsymbol{q}}$ for the derivatives of the generalized coordinates that correspond to angular or linear velocities in joint space.

The Hamiltonian principle of least action says that for a dynamical system without external excitation the action integral

$$I\left[\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t)\right] = \int_{t_0}^{t_f} U\left(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t)\right) - V\left(\boldsymbol{q}(t)\right) \, \mathrm{d}t \tag{2.5}$$

takes its extremal value. The total kinetic energy is denoted by $U(\boldsymbol{q}, \dot{\boldsymbol{q}})$, and $V(\boldsymbol{q})$ is the total potential energy. For a given initial configuration $(\boldsymbol{q}(t_0),\ \dot{\boldsymbol{q}}(t_0))$ and a given initial and final time t_0 and t_f, the extremal value of I defines the path $\boldsymbol{q}(t)$ of the generalized coordinates. The Lagrange function $L(\boldsymbol{q}, \dot{\boldsymbol{q}})$ abbreviates the difference between total kinetic energy $U(\boldsymbol{q}, \dot{\boldsymbol{q}})$ and total potential energy $V(\boldsymbol{q})$ of the robot.

$$L(\boldsymbol{q}, \dot{\boldsymbol{q}}) = U(\boldsymbol{q}, \dot{\boldsymbol{q}}) - V(\boldsymbol{q}) \tag{2.6}$$

The total kinetic energy is commonly written as $U(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \frac{1}{2}\dot{\boldsymbol{q}}^T \boldsymbol{M}(\boldsymbol{q})\dot{\boldsymbol{q}}$, where $\boldsymbol{M}(\boldsymbol{q})$ is the symmetric inertia matrix. Obviously $\boldsymbol{M}(\boldsymbol{q})$ is positive semidefinite since the kinetic energy takes only values greater or equal than zero.

Extremal value analysis leads to the Euler-Lagrange equations

$$\frac{\partial L}{\partial \boldsymbol{q}} - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial \dot{\boldsymbol{q}}} = \boldsymbol{0} \tag{2.7}$$

that determine solutions for the optimization problem in (2.5) in terms of differential equations. If the robot has n_q degrees of freedom, one obtains n_q second order ordinary differential equations.

Externally applied forces and torques denoted by $\boldsymbol{u} = (u_1, \ldots, u_{n_q})^T$ can be included in (2.7) through

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial \dot{\boldsymbol{q}}} - \frac{\partial L}{\partial \boldsymbol{q}} = \boldsymbol{u}. \tag{2.8}$$

If the i-th joint is passive, the respective force vanishes, $u_i = 0$. This is in particular the case for the joints that connect ground and foot in underactuated motion phases.

In robotics literature, the equations of motion are often denoted as

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{n}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \boldsymbol{u}, \tag{2.9}$$

which results from rearranging (2.8). In $\boldsymbol{n}(\boldsymbol{q}, \dot{\boldsymbol{q}})$ the influence of coriolis, centrifugal, and gravitational forces is summarized. If necessary, the equations of motion are transformed to $n = 2n_q$ first-order differential equations with state vector $\boldsymbol{x} = (\boldsymbol{q}^T, \dot{\boldsymbol{q}}^T)^T$ from (2.1):

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}) = \begin{pmatrix} \dot{\boldsymbol{q}} \\ \boldsymbol{M}(\boldsymbol{x})^{-1} \left[\boldsymbol{u} - \boldsymbol{n}(\boldsymbol{x})\right] \end{pmatrix} \tag{2.10}$$

For further details on the derivation of equations of motion see an introduction to theoretical mechanics, e.g. [49], for a robotics oriented introduction see [93].

2.2.2 Constraints and Collisions

The equations of motion of a robot are different for different contact situations, e.g. due to the actual number of degrees of freedom. One possibility to derive equations of motion for all contact situations is to use the Euler-Lagrange approach with different sets of generalized coordinates. A more convenient approach is to incorporate constraining conditions to the full dynamics of the robot. The constraints characterize the contact situation. The benefit of this approach is that one obtains additional equations for contact forces and moments.

Impact modeling is closely related to constraint modeling. Whenever the number of acting constraints increases, a collision has to be considered to conserve the angular momentum of the system. Using Newtons law results in discontinuities in the velocities.

Constraints. Every contact situation is specified by a set of scalar holonomic constraints represented by a set of constraint equations

$$c_i(\boldsymbol{q}) = 0 \qquad i = 1, \ldots, N_{x_d},$$

where N_{x_d} is the number of constraint conditions that act in the contact situation x_d. It is assumed that the constraints are independent, so N_{x_d} is the minimal number of equations needed to characterize the contact situation. Figure 2.4 illustrates the definition of constraint equations.

The constraints can be considered in the optimization problem (2.5) and as a consequence in the equations of motion (2.9) using Lagrange multipliers $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_{N_{x_d}})^T$. The cost function for a motion subject to constraints $\boldsymbol{c}(\boldsymbol{q}) = (c_1(\boldsymbol{q}), \ldots, c_{N_{x_d}}(\boldsymbol{q}))^T = \boldsymbol{0}$ is

$$I^* \left[\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t), \boldsymbol{\lambda}(t)\right] = \int_{t_0}^{t_f} U\left(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t)\right) - V\left(\boldsymbol{q}(t)\right) + \boldsymbol{\lambda}(t)^T \boldsymbol{c}\left(\boldsymbol{q}(t)\right) \mathrm{d}t. \tag{2.11}$$

Evaluation of the Euler-Lagrange equations (2.7) for the extended Lagrange function

$$L^*(\boldsymbol{q}, \dot{\boldsymbol{q}}, \boldsymbol{\lambda}) = \underbrace{U(\boldsymbol{q}, \dot{\boldsymbol{q}}) - V(\boldsymbol{q})}_{L(\boldsymbol{q}, \dot{\boldsymbol{q}})} + \boldsymbol{\lambda}^T \boldsymbol{c}(\boldsymbol{q})$$

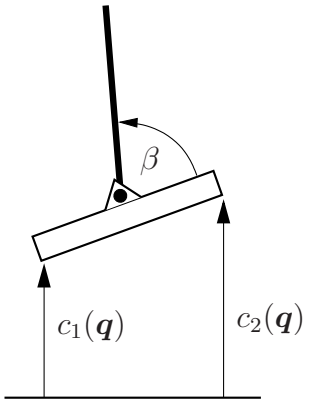

Figure 2.4: Holonomic constraints. If both constraints $c_1(\boldsymbol{q}) = 0$ and $c_2(\boldsymbol{q}) = 0$ are fulfilled, the robot has contact with the whole foot.

yields

$$\frac{\partial L}{\partial \boldsymbol{q}} - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial \dot{\boldsymbol{q}}} + \left(\frac{\partial \boldsymbol{c}(\boldsymbol{q})}{\partial \boldsymbol{q}}\right)^T \boldsymbol{\lambda} = \boldsymbol{0} \tag{2.12a}$$

and

$$\boldsymbol{c}(\boldsymbol{q}) = \boldsymbol{0}. \tag{2.12b}$$

The first equation (2.12a) results from application of the Lagrange equations (2.7) with respect to $\boldsymbol{q}$, the second equation (2.12b) from application with respect to $\boldsymbol{\lambda}$. The Jacobian $\boldsymbol{J}(\boldsymbol{q})$ of the constraints is used to abbreviate the derivative of the constraint function $\boldsymbol{c}(\boldsymbol{q})$:

$$\boldsymbol{J}(\boldsymbol{q}) = \frac{\partial \boldsymbol{c}(\boldsymbol{q})}{\partial \boldsymbol{q}} = \begin{pmatrix} \frac{\partial c_1}{\partial q_1} & \cdots & \frac{\partial c_1}{\partial q_{n_d}} \\ \vdots & \cdots & \vdots \\ \frac{\partial c_{N_{x_d}}}{\partial q_1} & \cdots & \frac{\partial c_{N_{x_d}}}{\partial q_{n_d}} \end{pmatrix}$$

The constraint equations $\boldsymbol{c}(\boldsymbol{q}) = \boldsymbol{0}$ are satisfied if $\boldsymbol{c}\left(\boldsymbol{q}(t)\right) = \boldsymbol{0}$ for all times t, and thus

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{c}(\boldsymbol{q}) = \boldsymbol{J}(\boldsymbol{q})\dot{\boldsymbol{q}} = \boldsymbol{0}.$$

Since $\boldsymbol{c}\left(\boldsymbol{q}(t)\right) = \boldsymbol{0}$ and $\dot{\boldsymbol{c}}\left(\boldsymbol{q}(t)\right) = \boldsymbol{0}$ there is also

$$\frac{\mathrm{d}^2}{\mathrm{d}t^2}\boldsymbol{c}(\boldsymbol{q}) = \boldsymbol{J}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \dot{\boldsymbol{J}}(\boldsymbol{q})\dot{\boldsymbol{q}} = \boldsymbol{0}. \tag{2.13}$$

Reformulation of (2.12a) in the usual notation with external forces $\boldsymbol{u}$ then provides

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{n}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \boldsymbol{u} + \boldsymbol{J}(\boldsymbol{q})^T\boldsymbol{\lambda}. \tag{2.14}$$

A physical interpretation is that $\boldsymbol{\lambda}$ are auxiliary forces and torques that have to act to satisfy the constraints. Therefore the components of $\boldsymbol{\lambda}$ are the contact forces and moments that act in the particular contact situation.

The system of equations (2.13) and (2.14) can now be solved for $\boldsymbol{\lambda}$ resulting in separate equations for the contact forces and new equations of motion. A combination of (2.13) and (2.14) yields contact forces:

$$\boldsymbol{\lambda} = (\boldsymbol{J}\boldsymbol{M}^{-1}\boldsymbol{J}^T)^{-1}(\boldsymbol{J}\boldsymbol{M}^{-1}\boldsymbol{n} - \dot{\boldsymbol{J}}\dot{\boldsymbol{q}}) - (\boldsymbol{J}\boldsymbol{M}^{-1}\boldsymbol{J}^T)^{-1}\boldsymbol{J}\boldsymbol{M}^{-1}\boldsymbol{u} \tag{2.15}$$

The new dynamical equations are (2.14) with (2.15) substituted. The number of generalized coordinates that are necessary to describe the system is reduced by the number N_{x_d} of constraints. With the reduced vector of generalized coordinates and the reduced input torque $\boldsymbol{u}$

$$\boldsymbol{M}_{x_d}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{n}_{x_d}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \boldsymbol{u}$$

is obtained, in which an appropriate mass matrix $\boldsymbol{M}_{x_d}(\boldsymbol{q})$ and appropriate forces $\boldsymbol{n}_{x_d}(\boldsymbol{q}, \dot{\boldsymbol{q}})$ for the contact situation are used.

An overview how to incorporate constraints is given in [93] for the example of robotic hands. Also, treatises on non-smooth mechanics, e.g. [22] and [48] cover the topic.

Collisions. For legged robots, impacts occur if either robot and ground get in contact or if the robot links collide with themselves. Internal collisions of robot links are neglected in the following, only robot to ground collisions are considered.

The following assumptions are made to derive collision equations following Newtons approach:

- the duration of the collision is arbitrarily short,
- no friction is considered,
- there is only a finite number of collisions in finite time,
- the collision is inelastic, and
- a collision acts at the same time for all robot links.

In (2.14), the dynamical equations of the robotic system with acting external forces $\boldsymbol{\lambda}$ are written as

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{n}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \boldsymbol{u} + \boldsymbol{J}(\boldsymbol{q})^T\boldsymbol{\lambda}.$$

For a collision investigation, $\boldsymbol{J}(\boldsymbol{q})$ is the Jacobian of the contact points that participate in the collision. The participating set of contact points is a subset of the points where $c(\boldsymbol{q}) = 0$ is true. It is assumed that the collision is instantaneous that means it takes place for a time t_c. The times $t_c^- = \lim_{t \leq \to t_c} t$ and $t_c^+ = \lim_{t \geq \to t_c} t$ are immediately before collision and immediately after collision. Conservation of momentum results in

$$\int_{t_c^-}^{t_c^+} \boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{n}(\boldsymbol{q}, \dot{\boldsymbol{q}})\,\mathrm{d}t = \int_{t_c^-}^{t_c^+} \boldsymbol{u} + \boldsymbol{J}(\boldsymbol{q})^T\boldsymbol{\lambda}\,\mathrm{d}t. \tag{2.16}$$

If discontinuities in the positions are not possible, such that $\boldsymbol{q}(t_c^+) = \boldsymbol{q}(t_c^-)$, evaluation of the integral (2.16) yields

$$\boldsymbol{M}(\boldsymbol{q})\left(\dot{\boldsymbol{q}}(t_c^+) - \dot{\boldsymbol{q}}(t_c^-)\right) = \boldsymbol{J}(\boldsymbol{q})^T \int_{t_c^-}^{t_c^+} \boldsymbol{\lambda}\,\mathrm{d}t. \tag{2.17}$$

The inelasticity assumption realizes that after collision no rebound is allowed

$$\boldsymbol{J}(\boldsymbol{q})\dot{\boldsymbol{q}}(t_c^+) = \dot{\boldsymbol{c}}\left(\boldsymbol{q}(t_c^+)\right) = \boldsymbol{0}. \tag{2.18}$$

Equation (2.17) together with (2.18) provides an equation system that can be solved for $\dot{\boldsymbol{q}}(t_c^+)$ and the acting impulsive reaction force

$$\boldsymbol{\Lambda} = \int_{t_c^-}^{t_c^+} \boldsymbol{\lambda}\,\mathrm{d}t.$$

Finally, combining (2.17) and (2.18), it is stated that:

$$\begin{aligned} \boldsymbol{\Lambda} &= -\left(\boldsymbol{J}\boldsymbol{M}^{-1}\boldsymbol{J}^T\right)^{-1}\boldsymbol{J}\dot{\boldsymbol{q}}^- \\ \dot{\boldsymbol{q}}^+ &= \dot{\boldsymbol{q}}^- - \boldsymbol{M}^{-1}\boldsymbol{J}^T\left(\boldsymbol{J}\boldsymbol{M}^{-1}\boldsymbol{J}^T\right)^{-1}\boldsymbol{J}\dot{\boldsymbol{q}}^- \end{aligned} \tag{2.19}$$

Not all points that have ground contact at collision time necessarily participate in the collision, and the set of participating contact points can only be determined by iterative evaluation of possible collision equations. After evaluation of the collision law for an initial assumption of participating points, it has to be verified that all components of the impulsive force $\boldsymbol{\Lambda}$ are greater than zero to agree with the demand of unilateral constraints. If it turns out that one or more components are less or equal than zero, the corresponding points do not participate in the collisions and the collision law (2.17) has to be reevaluated using a modified set of participating points. Analog checking has to be done for the contact points that were assumed not to participate. If the velocity of one of those points is negative after evaluation of the collision law, the collision has to be reevaluated including this contact point in the set of participating points.

The above derivation presented the simple collision model using Newtons approach that is widely used in legged robot modeling [14, 53, 103]. Details on the Newton method as well as on the Poisson collision model, which considers friction, are introduced in [22, 48]. Hurmuzlu *et al.* [71] address the problem of collision modeling specifically for mechanical actuators with chain structure.

2.2.3 Contact Forces and Moments

For simulation and control of legged robots, it is important to know the contact forces and moments between feet and ground. The ground acts as unilateral constraint to the robotic system. That means, the forces between ground and robot act only in one direction: The robot is supported by ground reaction forces but not attracted to ground.

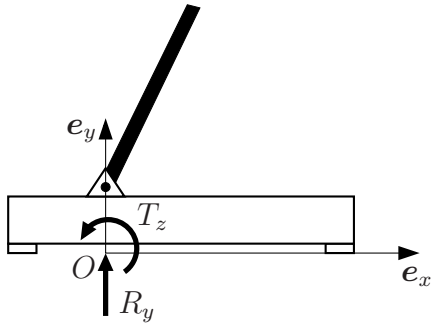

Figure 2.5: Contact force R_y and contact moment T_z.

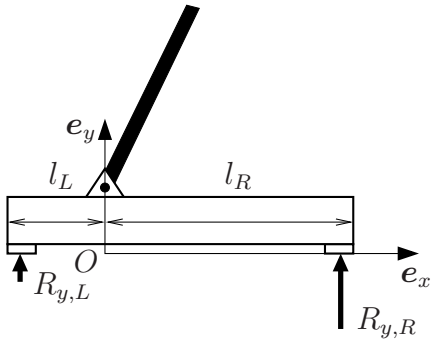

Figure 2.6: Contact force for left ($R_{y,L}$) and right foot edge ($R_{y,R}$).

In order to derive conditions for detachment, one considers the contact forces. It is shown that the condition for tipping over can be expressed in terms of contact forces and moments which leads to the common concept of Zero Moment Point (ZMP) or Foot Rotation Indicator (FRI).

Derivation of Detachment Condition for Planar Robots. To detect detachment, the vertical contact force component R_y has to be observed, see Fig. 2.5. As long as $R_y > 0$, the ground supports the robot weight. If $R_y = 0$, the support stops. Since attraction is impossible, the foot detaches, and a new contact phase starts.

Derivation of Tipping-over Condition for Planar Robots. Consider again the robot foot with its vertical contact force R_y and its horizontal contact moment T_z to find a condition for tipping over around one foot edge. Figure 2.5 depicts the total vertical contact force R_y and the total horizontal contact moment T_z with reference to the origin O of a given world coordinate system.

Alternatively, the total vertical contact force R_y is considered split up into two components

$$R_y = R_{y,L} + R_{y,R}, \tag{2.20}$$

where $R_{y,L}$ is the contact force for the left foot edge and $R_{y,R}$ for the right foot edge, see Fig. 2.6. As a consequence, the contact moment T_z can also be split up into two parts

$$T_z = l_R R_{y,R} - l_L R_{y,L}, \tag{2.21}$$

where $l = l_L + l_R$ is the length of the foot. If the contact force at the left foot edge vanishes, $R_{y,L} = 0$, the foot begins to tip over the right foot edge. Combination of (2.20) and (2.21) with $R_{y,L} = 0$ yields the tip-over condition:

$$\frac{T_z}{R_y} = l_R \tag{2.22a}$$

Analogously, the tip-over condition for the left foot edge is derived as

$$\frac{T_z}{R_y} = -l_L. \tag{2.22b}$$

In the robotics literature the fraction T_z/R_y is often referred to as Zero Moment Point (ZMP) initially introduced by Vukobratović *et al.* [132, 133] and used by many other research groups for balance control, e.g. [59, 100, 125]. In this thesis, the vertical axis is the y-axis, whereas in other publications, the vertical axis is denoted as z-axis. This results in different signs of the ZMP expression in terms of contact forces and moments.

Tipping-over and Zero Moment Point. Following one of several possible equivalent definitions, the ZMP is the point on the walking ground surface, at which the horizontal components of the resultant moment generated by active forces and moments acting on the robot links are equal to zero.

To formalize this definition, some notations have to be introduced. Still O is the origin of a reference frame. The considered robots are allowed to have a finite number of rigid links. The mass center of link i has the position $\boldsymbol{r}_i$. Then, the force acting at the mass center of link i is $\boldsymbol{R}_i = m_i\ddot{\boldsymbol{r}}_i - m_i(0,\, -g,\, 0)^T$. Here, $g = 9.81\mathrm{m/s}^2$ abbreviates earth acceleration. Due to inertia properties of the links, an additional moment $\boldsymbol{T}_i$ acts for every link mass center.

The ZMP $\boldsymbol{r}_{\mathrm{zmp}} = (r_{\mathrm{zmp},x},\, 0,\, r_{\mathrm{zmp},z})^T$ follows from the equilibrium of the total horizontal moments. The vertical moment is allowed to take arbitrary values, denoted by an asterisk on the right hand side of the following equilibrium equation.

$$\sum_i (\boldsymbol{r}_i - \boldsymbol{r}_{\mathrm{zmp}}) \times \boldsymbol{R}_i + \sum_i \boldsymbol{T}_i = \begin{pmatrix} 0 \\ * \\ 0 \end{pmatrix} \tag{2.23}$$

A component-wise evaluation yields again (2.22):

$$r_{\mathrm{zmp},x} = \frac{T_z}{R_y} \quad \text{and} \quad r_{\mathrm{zmp},z} = -\frac{T_x}{R_y}$$

Here T_z and T_x are the z and x-component of $\sum \boldsymbol{T}_i + \sum \boldsymbol{r}_i \times \boldsymbol{R}_i$, and R_y is the y-component of $\sum \boldsymbol{R}_i$.

An equivalent definition of ZMP is the point on ground level where a vertical force has to act to balance the system. If the ZMP is in the supporting area, which is the region covered by the foot, the ground reaction force balances the system. If the ZMP leaves the supporting area, the robot cannot be balanced by the ground reaction force any more and starts tipping over the foot edge. Thus the condition for tipping over derived from the ZMP is consistent with the results in (2.22).

The ZMP is equal to the Center of Pressure [51], the point on ground level where the contact force actually acts. The Center of Pressure is of practical relevance because it is obtained by measurements of the contact force distribution of the foot.

The Foot Rotation Indicator (FRI) was introduced by Goswami in 1999 [51] to replace ZMP and is a generalization. For a balanced system where the foot is at rest, FRI equals ZMP. But unlike the ZMP, the FRI is also defined for a robot that is already tipping over and gives a measure of postural instability.

Computation of Contact Forces and Contact Moments. Since in the following the discussion is restricted to planar robots where is motion confined to the xy-plane, the z-component of ZMP is zero. Hence the notation

$$r_{\text{zmp}} = r_{\text{zmp},x} = \frac{T_z}{R_y} \tag{2.24}$$

is used to avoid unnecessary indices. The contact force R_y is obtained by summation over all mass points

$$R_y = \sum_i m_i \left(\ddot{r}_{i,y} + g\right).$$

In the same way T_z is evaluated, considering the cross product between force components and position vector of mass points.

$$T_z = \sum_i m_i r_{i,x} \left(\ddot{r}_{i,y} + g\right) - m_i r_{i,y} \ddot{r}_{i,x} + T_{i,z}$$

2.3 Hybrid Models for Legged Locomotion Systems

The hybrid model combines the equations of motion from Sec. 2.2.1, conditions for a contact situation transition from Sec. 2.2.3, and reinitialization rules given by the impact law in Sec. 2.2.2. Transitions between contact situations are assumed to be instantaneous. This enables a differential description in minimal coordinates for every contact situation. That means variables q_i for constrained degrees of freedom i vanish in the differential equations for the contact situation. To formally keep the system order constant, auxiliary differential equations $\dot{q}_i = 0$ and $\ddot{q}_i = 0$ can be added. The auxiliary degrees of freedom are necessary for simulation purposes to achieve a constant size of the state vector in all contact situations. In trajectory planning and control, the auxiliary differential equations are omitted.

Hybrid State Vector. The hybrid state vector $\boldsymbol{\zeta}$ for the legged robot model according to (2.2) is introduced that is a combination of the continuous state $\boldsymbol{x} = \left(\boldsymbol{q}^T, \dot{\boldsymbol{q}}^T\right)^T \in \mathbb{R}^n$ and the discrete state $x_d \in \mathbb{Z}$. The discrete state x_d codes the contact situation.

Equations of Motion. It is assumed that an equation of motion $\boldsymbol{M}_{x_d}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{n}_{x_d}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \boldsymbol{u}$ is available for any possible contact situation x_d. The derivation was presented in Sec. 2.2.2. The vector $\boldsymbol{q}$ is the set of minimal coordinates needed to define the posture uniquely for the particular contact situation. After the equations are transformed to first-order differential equations with state vector $\boldsymbol{x} = (\boldsymbol{q}^T, \dot{\boldsymbol{q}}^T)^T$, compare (2.10),

$$\dot{\boldsymbol{x}} = \boldsymbol{f}_{x_d}(\boldsymbol{x}, \boldsymbol{u}) = \begin{pmatrix} \dot{\boldsymbol{q}} \\ \boldsymbol{M}_{x_d}^{-1} \left[\boldsymbol{u} - \boldsymbol{n}_{x_d}\right] \end{pmatrix}.$$

The notation corresponds with the hybrid state model notation introduced in (2.3). The dynamical behavior of the robot is determined by the differential equations as long as no events occur and the contact situation remains unaltered. Expressions for the equations of motion can be derived using the symbolic manipulation tool Autolev [9] that is based on Kane's algorithm. Autolev offers a convenient possibility to obtain equations of motion in symbolic form, ready for implementation in Matlab, Fortran, or C-code.

In the following, two categories of events, collision events and detachment events, will be distinguished:

Collision Event. A collision event occurs if a foot edge that had no ground contact before touches ground. The occurrence of collision events is supervised by transition equations of the distance between foot edges and ground. This category of transition surface thus depends only on the actual configuration, not on velocities or acting forces:

$$s(\boldsymbol{x}, \boldsymbol{u}) = s(\boldsymbol{q}) = 0$$

If the foot edge touches ground with nonzero velocity, an impact occurs. In the hybrid model, the impact law is accounted for by a reinitialization rule for the hybrid state. More precisely only the velocities and the discrete state are reset, compare (2.19).

$$\begin{pmatrix} \boldsymbol{q}^+ \\ \dot{\boldsymbol{q}}^+ \\ x_d^+ \end{pmatrix} = \begin{pmatrix} \boldsymbol{q}^- \\ \dot{\boldsymbol{q}}^- - \boldsymbol{M}^{-1}\boldsymbol{J}^T(\boldsymbol{J}\boldsymbol{M}^{-1}\boldsymbol{J}^T)^{-1}\boldsymbol{J}\dot{\boldsymbol{q}}^- \\ g_d(\boldsymbol{x}^-, x_d^-) \end{pmatrix}$$

Detachment Event. A foot edge detaches from ground if a contact force changes sign or the ZMP crosses a boundary of the foot support area. The conditions to supervise depend on state $\boldsymbol{x}$ and control input $\boldsymbol{u}$:

$$s(\boldsymbol{x}, \boldsymbol{u}) = 0$$

The reinitialization rule does not affect positions and velocities. Only the discrete state variable x_d is reset.

$$\begin{pmatrix} \boldsymbol{q}^+ \\ \dot{\boldsymbol{q}}^+ \\ x_d^+ \end{pmatrix} = \begin{pmatrix} \boldsymbol{q}^- \\ \dot{\boldsymbol{q}}^- \\ g_d(\boldsymbol{x}^-, x_d^-) \end{pmatrix}$$

Simulation. For the simulation of hybrid systems an integrator for ordinary differential equations has to be augmented by additional features. It is required that the integration process stops whenever an event occurs. Then, the integration may be restarted after the execution of the reset rules. Therefore all transition conditions have to be evaluated for every integration step. Then, if one transition condition becomes true, an iterative procedure is started to find the exact time of transition [110]. The technical computing environment Matlab [87] provides numerical algorithms for the integration of ordinary differential equations with parallel surveillance of event functions. If an event is detected, the event time is determined precisely and the integration is stopped to allow for state resets.

2.3.1 Compass Gait Robot

Introduction. The simplest robot construction that can show a walking motion is a robot with two legs that are linked by one actuator, as shown in Fig. 2.7. This robot is often used as passive walker and referred to as compass gait robot in the literature. The rotation around the foot contact point is always passive. The joint between the two legs is assumed to be passive or active.

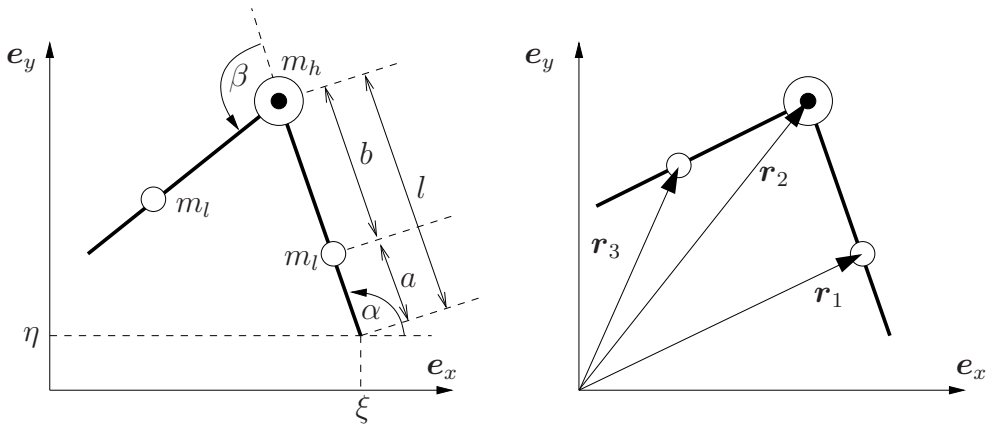

Figure 2.7: Compass gait robot. Illustration of geometry, masses, and coordinate system. The robot is made up from two links and one actuated joint. Leg masses are m_l, hip mass is m_h. Foot length is l; a and b are distances of leg mass centers from foot or hip. Generalized coordinates are $\boldsymbol{q} = (\xi,\ \eta,\ \alpha,\ \beta)^T$. The vectors $\boldsymbol{r}_1$, $\boldsymbol{r}_2$, and $\boldsymbol{r}_3$ are position vectors of mass points.

For simplicity, it is assumed that masses are point masses. The legs have mass m_l, and the hip has mass m_h. The legs have length l, where the mass center is at distance a from the foot. The position of the reference foot is $(\xi,\ \eta)$, α is the orientation of the reference foot, and β is the angle between the legs. The generalized coordinates are thus

$$\boldsymbol{q} = \begin{pmatrix} \xi \\ \eta \\ \alpha \\ \beta \end{pmatrix}.$$

In accordance with publications of Goswami *et al.* [52], Hiskens [61], or Spong *et al.* [80] the masses and lengths are chosen:

m_l	5.0	kg
m_h	10.0	kg
l	1.0	m
a, b	0.5	m

Hybrid State Vector. For the compass gait robot three contact situations are distinguished. The robot is either in flight, has contact with the reference foot, or contact with the non-reference foot. Other contact situations, as double support or sliding, are not

considered. A discrete valued state variable $x_d(t) \in \{-1, 0, 1\} \subset \mathbb{Z}$ encodes the contact situation as assigned in Fig. 2.8.

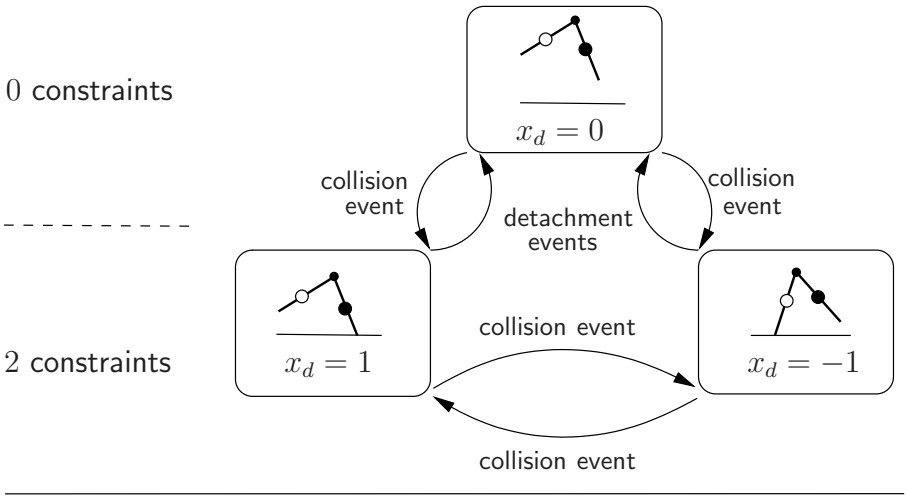

Figure 2.8: Transition graph for the compass gait robot.

The continuous state vector of the robot is composed from the vector of generalized coordinates $\boldsymbol{q} = (\xi,\ \eta,\ \alpha,\ \beta)^T$ and its derivative $\dot{\boldsymbol{q}}$ that are concatenated to the continuous state $\boldsymbol{x}$. The hybrid state vector $\boldsymbol{\zeta}$ combines the continuous and the discrete state. The actuator in the hip applies a torque u to actuate the joint angle β.

Equations of Motion for the Hybrid System. The equations of motion for the unconstrained robot are derived using the Euler-Lagrange approach outlined in Sec. 2.2.1. Through incorporation of constraint conditions, equations of motion and expressions for contact forces for the single support phases are generated. Equations of motion are needed for all three possible contact situations. If the three equations of motion are formulated as first order differential equations $\dot{\boldsymbol{x}} = \boldsymbol{f}_0(\boldsymbol{x}, u)$, $\dot{\boldsymbol{x}} = \boldsymbol{f}_1(\boldsymbol{x}, u)$, and $\dot{\boldsymbol{x}} = \boldsymbol{f}_{-1}(\boldsymbol{x}, u)$, the differential equations for the hybrid system are

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, x_d, u) = \delta_{x_d,0}\boldsymbol{f}_0(\boldsymbol{x}, u) + \delta_{x_d,1}\boldsymbol{f}_1(\boldsymbol{x}, u) + \delta_{x_d,-1}\boldsymbol{f}_{-1}(\boldsymbol{x}, u),$$

where $\delta_{i,j}$ is one only for $i = j$, else it is zero. Compare (2.3).

Equations of Motion for the Unconstrained System ($x_d = 0$). Before denoting the Lagrange function as basis for the derivation of the equations of motion, position vectors $\boldsymbol{r}_1$, $\boldsymbol{r}_2$, and $\boldsymbol{r}_3$ for the masses are introduced, compare Fig. 2.7 and Sec. A.1.1. If it is assumed that all masses are point masses, the Lagrange function reads

$$L(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \underbrace{\frac{1}{2}m_l\|\dot{\boldsymbol{r}}_1\|^2 + \frac{1}{2}m_h\|\dot{\boldsymbol{r}}_2\|^2 + \frac{1}{2}m_l\|\dot{\boldsymbol{r}}_3\|^2}_{U(\boldsymbol{q}, \dot{\boldsymbol{q}})} - \underbrace{(m_l g r_{1,y} + m_h g r_{2,y} + m_l g r_{3,y})}_{V(\boldsymbol{q})},$$

where $r_{i,y}$ is the y-component of $\boldsymbol{r}_i = (r_{i,x},\ r_{i,y})^T$.

Application of Euler-Lagrange equations (2.8) yields the equations of motion for $\boldsymbol{q} = (\xi,\, \eta,\, \alpha,\, \beta)^T$:

$$\underbrace{\begin{pmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{12} & m_{22} & m_{23} & m_{24} \\ m_{13} & m_{23} & m_{33} & m_{34} \\ m_{14} & m_{24} & m_{34} & m_{44} \end{pmatrix}}_{\boldsymbol{M}_0} \ddot{\boldsymbol{q}} + \underbrace{\begin{pmatrix} n_1 \\ n_2 \\ n_3 \\ n_4 \end{pmatrix}}_{\boldsymbol{n}_0} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ u \end{pmatrix} \tag{2.25}$$

See Sec. A.1.2 for explicit expressions of the matrix and vector elements m_{ij} and n_i. The equations of motion above describe the robot behavior for the robot in the ballistic motion phase coded by $x_d = 0$. The equations can be transformed to a first-order system of differential equations denoted by $\dot{\boldsymbol{x}} = \boldsymbol{f}_0(\boldsymbol{x}, u)$.

Equations of Motion for Single Support on Reference Foot ($x_d = 1$). For a single support phase, constraints have to be introduced that guarantee that the reference foot has persistent contact with the ground.

$$\boldsymbol{c}(\boldsymbol{q}) = \begin{pmatrix} \xi - \xi_0 \\ \eta \end{pmatrix} = \boldsymbol{0} \tag{2.26}$$

Here $(\xi_0, 0)$ is the desired cartesian position of the foot. The corresponding Jacobian $\boldsymbol{J}_1$ is

$$\boldsymbol{J}_1 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}. \tag{2.27}$$

Coupling of the Jacobian $\boldsymbol{J}_1$ to (2.25) results in the equations of motion for $\boldsymbol{q} = (\alpha,\, \beta)^T$

$$\underbrace{\begin{pmatrix} m_{11} & m_{12} \\ m_{12} & m_{22} \end{pmatrix}}_{\boldsymbol{M}_1} \ddot{\boldsymbol{q}} + \underbrace{\begin{pmatrix} n_1 \\ n_2 \end{pmatrix}}_{\boldsymbol{n}_1} = \begin{pmatrix} 0 \\ u \end{pmatrix}. \tag{2.28}$$

See again Sec. A.1.2 for expressions of the matrix and vector elements m_{ij} and n_i. The equations of motion do not consider the dynamics of ξ and η any more. They are restated in first-order differential equations $\dot{\boldsymbol{x}} = \boldsymbol{f}_1(\boldsymbol{x}, u)$. The contact forces are obtained by evaluation of $\boldsymbol{\lambda} = (R_x,\ R_y)^T$. Explicit expressions are given in Sec. A.1.3.

Equations of Motion for Single Support with Non-reference Foot ($x_d = -1$). Constraints for persistent contact with the non-reference foot are

$$\boldsymbol{c}(\boldsymbol{q}) = \begin{pmatrix} \xi + l\cos\alpha + l\cos(\alpha+\beta) - \xi_0 \\ \eta + l\sin\alpha + l\sin(\alpha+\beta) \end{pmatrix} = \boldsymbol{0} \tag{2.29}$$

where the corresponding Jacobian $\boldsymbol{J}_{-1}$ is

$$\boldsymbol{J}_{-1} = \begin{pmatrix} 1 & 0 & -l\sin\alpha - l\sin(\alpha+\beta) & -l\sin(\alpha+\beta) \\ 0 & 1 & l\cos\alpha + l\cos(\alpha+\beta) & l\cos(\alpha+\beta) \end{pmatrix}. \tag{2.30}$$

The constraints attach the x-coordinate and the y-coordinate of the non-reference foot to the ground. The equations of motion $\dot{\boldsymbol{x}} = \boldsymbol{f}_{-1}(\boldsymbol{x}, u)$ are derived in analogy to the derivation in the previous section for (2.28).

For practical implementations the equations of motion for contact with the non-reference foot ($x_d = -1$) are obtained by a coordinate transformation of the equations of motion for contact with the reference foot ($x_d = 1$) in (2.28). This is possible due to the symmetric construction. The transformation that maps coordinates of one foot to coordinates of the other foot is

$$\boldsymbol{T}_{\text{feet}}(\boldsymbol{x}) = \begin{pmatrix} \xi + l\sin\alpha + l\sin(\alpha+\beta) \\ \eta + l\cos\alpha + l\cos(\alpha+\beta) \\ \alpha + \beta - \pi \\ 2\pi - \beta \\ \dot{\xi} + l\cos\alpha\dot{\alpha} + l\cos(\alpha+\beta)(\dot{\alpha}+\dot{\beta}) \\ \dot{\eta} - l\sin\alpha\dot{\alpha} - l\sin(\alpha+\beta)(\dot{\alpha}+\dot{\beta}) \\ \dot{\alpha} + \dot{\beta} \\ -\dot{\beta} \end{pmatrix}. \tag{2.31}$$

Necessarily, applying the transformation $\boldsymbol{T}_{\text{feet}}$ twice results in the original state

$$\boldsymbol{x} = \boldsymbol{T}_{\text{feet}}\left(\boldsymbol{T}_{\text{feet}}(\boldsymbol{x})\right).$$

Transition Conditions. If the robot is in a ballistic motion phase ($x_d = 0$), two transitions are modeled. The robot lands on the reference foot ($x_d = 1$) or on the non-reference foot ($x_d = -1$). Hence the transition surfaces are $s_{0,1} = 0$ for the transition from ballistic ($x_d = 0$) to contact with the reference foot ($x_d = 1$) and $s_{0,-1} = 0$ for landing on the non-reference foot ($x_d = -1$). The algebraic expressions

$$\begin{aligned} s_{0,1}(\boldsymbol{x}) &= \eta \\ s_{0,-1}(\boldsymbol{x}) &= \eta + l\sin\alpha + l\sin(\alpha+\beta) \end{aligned}$$

that describe the y-coordinate of the feet have to be supervised on zero crossing.

Also if the robot is in contact with the reference foot ($x_d = 1$), two transitions may occur. Landing on the swing foot ($x_d = -1$) with a consecutive stance foot exchange is supervised by the transition surface

$$s_{1,-1}(\boldsymbol{x}) = \eta + l\sin\alpha + l\sin(\alpha+\beta).$$

A transition is though not allowed if $\beta = \pm\pi$. This is denoted foot scuffing problem in the literature: Due to the simple robot construction, it is not possible for the feet to pass each other without touching the ground of the swing foot when $\beta = \pm\pi$. For theoretical analysis this problem is neglected. In experimental studies flap mechanisms or similar auxiliary mechanisms are used.

Detaching to begin a ballistic phase ($x_d = 0$) occurs if the contact force R_y becomes zero:

$$s_{1,0}(\boldsymbol{x}, u) = R_y(\boldsymbol{x}, u) \tag{2.32}$$

An expression for R_y is given in (A.6) in terms of $(\boldsymbol{q}, \dot{\boldsymbol{q}}, \ddot{\boldsymbol{q}})$. To express R_y in terms of $(\boldsymbol{q}, \dot{\boldsymbol{q}}, u)$, $\ddot{\alpha}$ and $\ddot{\beta}$ are replaced by

$$\begin{pmatrix} \ddot{\alpha} \\ \ddot{\beta} \end{pmatrix} = \boldsymbol{M}_1^{-1} \left[-\boldsymbol{n}_1 + \begin{pmatrix} 0 \\ u \end{pmatrix} \right]$$

Analogously, the transition surfaces $s_{-1,1} = 0$ and $s_{-1,0} = 0$ for the robot in contact with the non-reference foot are derived.

Reinitialization. A collision event occurs either when landing from a ballistic motion phase or when the roles of stance and swing foot change. Then, the hybrid state and in particular the joint velocities are allowed to behave discontinuous, $\boldsymbol{\zeta}^+ = \boldsymbol{\varphi}(\boldsymbol{\zeta}^-)$. A general expression for the impact law is from (2.19)

$$\dot{\boldsymbol{q}}^+ = \dot{\boldsymbol{q}}^- - \boldsymbol{M}^{-1}\boldsymbol{J}^T(\boldsymbol{J}\boldsymbol{M}^{-1}\boldsymbol{J}^T)^{-1}\boldsymbol{J}\dot{\boldsymbol{q}}^- \tag{2.33}$$

with $\boldsymbol{M} = \boldsymbol{M}_0$ from (2.25) and $\boldsymbol{J}$ chosen according to the constraints that participate in the collision.

For the transition from a ballistic phase ($x_d = 0$) to contact with the reference foot ($x_d = 1$) or from non-reference foot contact ($x_d = -1$) to reference foot contact ($x_d = 1$), the active constraints are given in (2.26) with Jacobian $\boldsymbol{J}_1$. The constraints result in reset functions $\boldsymbol{\varphi}_{0,1}(\boldsymbol{\zeta})$ and $\boldsymbol{\varphi}_{-1,1}(\boldsymbol{\zeta})$ written as

$$\boldsymbol{\varphi}_{-1,1}(\boldsymbol{\zeta}) = \boldsymbol{\varphi}_{0,1}(\boldsymbol{\zeta}) = \begin{pmatrix} \boldsymbol{q}^- \\ \dot{\boldsymbol{q}}^- - \boldsymbol{M}_0^{-1}\boldsymbol{J}_1^T(\boldsymbol{J}_1\boldsymbol{M}_0^{-1}\boldsymbol{J}_1^T)^{-1}\boldsymbol{J}_1\dot{\boldsymbol{q}}^- \\ 1 \end{pmatrix}.$$

For the transition from a ballistic phase ($x_d = 0$) to a contact situation where the non-reference foot has contact ($x_d = -1$) or from contact with the reference foot ($x_d = 1$) to contact with the non-reference foot ($x_d = 1$), reset functions $\boldsymbol{\varphi}_{0,-1}(\boldsymbol{\zeta})$ and $\boldsymbol{\varphi}_{1,-1}(\boldsymbol{\zeta})$ are used. The corresponding constraints can be found in (2.29).

For the case when the swing foot and the stance foot change its roles, both feet have instantaneous contact:

$$\begin{aligned} c_1(\boldsymbol{q}) &= \xi - \xi_0 = 0 \\ c_2(\boldsymbol{q}) &= \eta = 0. \\ c_3(\boldsymbol{q}) &= \xi + l\cos\alpha + l\cos(\alpha+\beta) - \xi_1 = 0 \\ c_4(\boldsymbol{q}) &= \eta + l\sin\alpha + l\sin(\alpha+\beta) = 0. \end{aligned}$$

If, after the collision, $\dot{c}_1(\boldsymbol{q}) \not> 0$ or $\dot{c}_2(\boldsymbol{q}) \not> 0$, the stance foot participates in the collision and c_1 and c_2 have to be considered in the Jacobian $\boldsymbol{J}$. The resulting contact situation is a double support phase which, however, is not considered in the model. The simulation is aborted if double support occurs.

Any transitions where constraints are dissolved do not induce discontinuities in the velocities. These are in particular the transition from reference foot contact ($x_d = 1$) to a ballistic phase ($x_d = 0$) and from non-reference foot contact ($x_d = -1$) to a ballistic phase ($x_d = 0$).

$$\boldsymbol{\varphi}_{1,0}(\boldsymbol{\zeta}) = \boldsymbol{\varphi}_{-1,0}(\boldsymbol{\zeta}) = \begin{pmatrix} \boldsymbol{q}^- \\ \dot{\boldsymbol{q}}^- \\ 0 \end{pmatrix}.$$

The detailed transition graph in Fig. 2.9 gives a summary of the hybrid model for the compass gait robot.

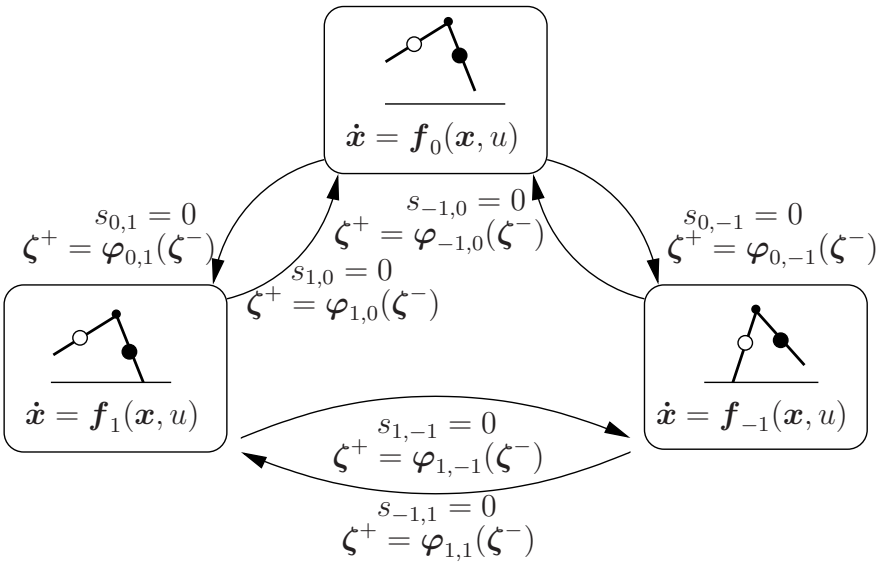

Figure 2.9: Detailed transition graph for the compass gait robot.

2.3.2 Monoped Robot

Introduction. The monoped robot has two links and one actuated joint that are arranged different than for the compass gait robot. One of the links is a foot that is allowed to have plane contact with the ground, thereby enabling a fully actuated contact situation. Figure 2.10 illustrates the robot construction, and Fig. 2.11 gives a magnified view of the foot.

An experimental platform is the basis for modeling, thus no simplification are presumed concerning the mass distribution and the positions of the mass centers. From the CAD model of the robot, the following parameter values are taken:

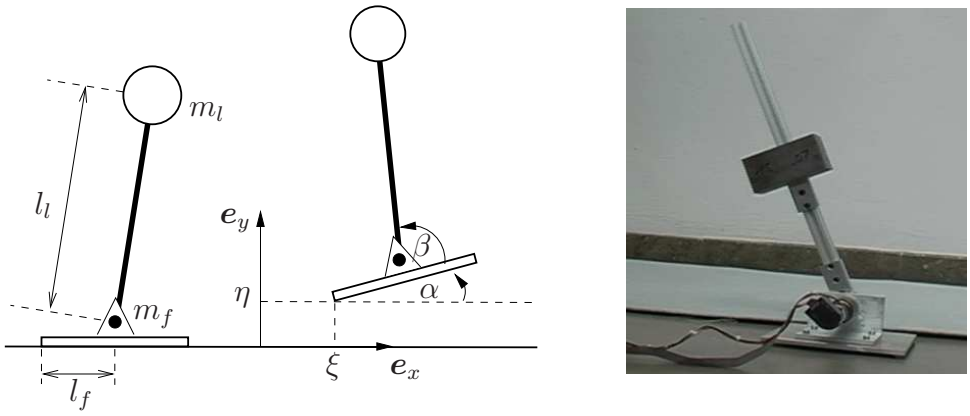

Figure 2.10: Monoped robot. Illustration of geometry, masses, and coordinate system. Masses are denoted m_f and m_l. Dimensions are denoted by l_l and l_f. The mass distribution is considered by principal moments of inertia $I_{f,z}$ and $I_{l,z}$ that are taken from a CAD model of the experimental platform depicted on the right hand side.

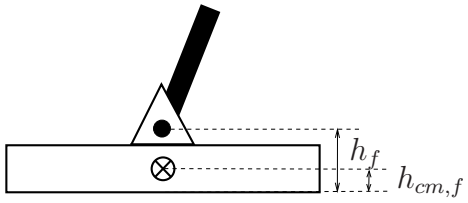

Figure 2.11: Details of monoped foot. The foot has height h_f, the center of mass of the foot has height $h_{cm,f}$.

m_f	0.299	kg
m_l	0.607	kg
l_l	0.405	m
l_f	0.1	m
h_f	0.025	m
$h_{cm,f}$	0.00353	m
$I_{f,z}$	0.0011573165	kg/m^2
$I_{l,z}$	0.00061034433	kg/m^2

Hybrid State Vector. For the monoped robot four contact situations are distinguished. The robot is either in flight, tilted around the left foot edge, tilted around the right foot edge, or has stable ground contact with the whole foot. Other ground contact situations are not considered in the model. A discrete state variable $x_d(t) \in \{0, 1, 2, 3\}$ encodes the contact situation as depicted in Fig. 2.12. The continuous state vector $\boldsymbol{x}$ of the robot is composed from the vector of generalized coordinates $\boldsymbol{q} = (\xi,\, \eta,\, \alpha,\, \beta)^T$ and its derivative $\dot{\boldsymbol{q}}$. The actuator applies a torque u to actuate the joint angle β.

Equations of Motion for the Hybrid System. First, differential descriptions are established for the unconstrained system according to the derivation in Sec. 2.2.1. The remaining

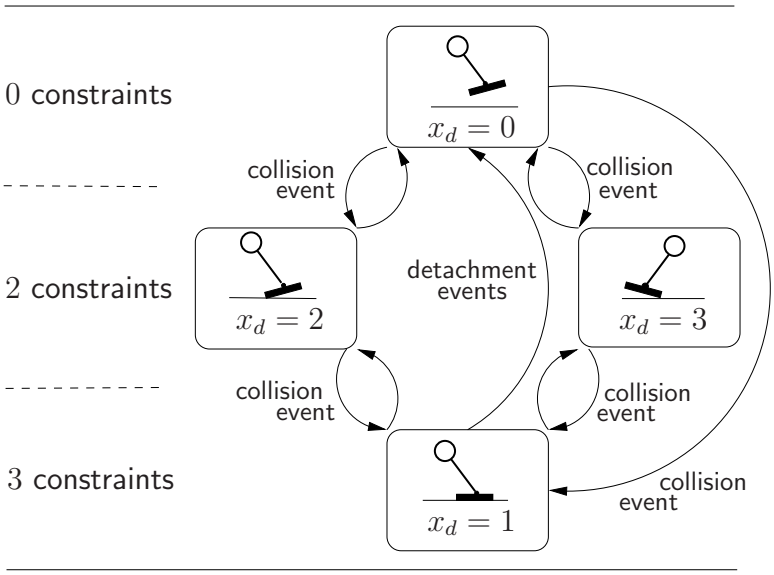

Figure 2.12: Transition graph for the monoped robot.

equations of motion are obtained by introduction of constraints. The differential equations $\boldsymbol{f}_{x_d}(\boldsymbol{x}, u)$ are combined to one vector field to fit into the hybrid state model notation (2.3):

$$\boldsymbol{f}(\boldsymbol{x}, x_d, u) = \sum_i \delta_{x_d,i} \boldsymbol{f}_i(\boldsymbol{x}, u)$$

Equations of Motion for the Unconstrained System ($x_d = 0$). Again, to prepare the Lagrange function, position vectors $\boldsymbol{r}_1$ and $\boldsymbol{r}_2$ for the mass centers of the links are introduced. See Sec. A.2.1 in the appendix for the exact expressions. Then the Lagrange function is

$$L(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \underbrace{\frac{1}{2} m_f \|\dot{\boldsymbol{r}}_1\|^2 + \frac{1}{2} I_{f,z} \dot{\alpha}^2 + \frac{1}{2} m_l \|\dot{\boldsymbol{r}}_2\|^2 + \frac{1}{2} I_{l,z} (\dot{\alpha} + \dot{\beta})^2}_{U(\boldsymbol{q}, \dot{\boldsymbol{q}})} - \underbrace{(m_f g r_{1,y} + m_l g r_{2,y})}_{V(\boldsymbol{q})}, \quad (2.34)$$

in which $I_{f,z}$ is the inertia of the foot and $I_{l,z}$ the inertia of the link. This results in equations of motion

$$\boldsymbol{M}_0 \ddot{\boldsymbol{q}} + \boldsymbol{n}_0 = \boldsymbol{u} \quad (2.35)$$

with details in (A.9). The equations of motion can be transformed to a first-order system $\dot{\boldsymbol{x}} = \boldsymbol{f}_0(\boldsymbol{x}, \boldsymbol{u})$.

Equations of Motion for Stable Support Contact ($x_d = 1$). Ground contact is introduced by algebraic constraints for the equations of motion for the unconstrained system (2.35). For stable contact with the ground, the foot has zero velocity relative to the ground. Therefore the constraints are

$$\boldsymbol{c}(\boldsymbol{q}) = \begin{pmatrix} \xi - \xi_0 \\ \eta \\ \alpha \end{pmatrix} = \boldsymbol{0}. \quad (2.36)$$

The resulting equations of motion are

$$\underbrace{(I_{l,z} + m_l l_l^2)}_{M_1} \ddot{\beta} + \underbrace{m_l g l_l \cos\beta}_{n_1} = u \tag{2.37}$$

and can be abbreviated $\dot{\boldsymbol{x}} = \boldsymbol{f}_1(\boldsymbol{x}, \boldsymbol{u})$. Additionally, equations for the contact forces and moments are produced, due to $\boldsymbol{\lambda} = (R_x,\ R_y,\ T_z)^T$. (See Sec. A.2.3.)

Equations of Motion for Contact on the Left Edge ($x_d = 2$). Accordingly, for contact on the left foot edge the constraints are

$$\boldsymbol{c}(\boldsymbol{q}) = \begin{pmatrix} \xi - \xi_0 \\ \eta \end{pmatrix} = \boldsymbol{0}.$$

The corresponding Jacobian matrix scaled by a multiplier $\boldsymbol{\lambda}$ is added to the equation of motion (2.35) resulting in

$$\boldsymbol{M}_2\, \ddot{\boldsymbol{q}} + \boldsymbol{n}_2 = \boldsymbol{u} \tag{2.38}$$

with matrix and vector elements m_{ij} and n_i given in Sec. A.2.2. An alternative notation is $\dot{\boldsymbol{x}} = \boldsymbol{f}_2(\boldsymbol{x}, \boldsymbol{u})$. For tilted ground contact $\boldsymbol{\lambda} = (R_x,\ R_y)^T$ consists only of the contact forces, see Sec. A.2.3.

Equations of Motion for Contact on the Right Edge ($x_d = 3$). The constraints for contact on the right foot edge are

$$\boldsymbol{c}(\boldsymbol{q}) = \begin{pmatrix} \xi + 2l_f \cos\alpha - \xi_0 \\ \eta + 2l_f \sin\alpha \end{pmatrix} = \boldsymbol{0}$$

and constrain the x and y-coordinate of the right foot edge.

The equations of motion for right tilting are obtained by a coordinate transformation of the equations for left tilting in (2.38) due to the symmetric construction. The symmetry transformation is:

$$\boldsymbol{T}_{\text{edges}}(\boldsymbol{x}) = \begin{pmatrix} \xi + 2l_f \sin\alpha \\ \eta + 2l_f \cos\alpha \\ -\alpha \\ \pi - \beta \\ \dot{\xi} + 2l_f \cos\alpha\ \dot{\alpha} \\ \dot{\eta} - 2l_f \sin\alpha\ \dot{\alpha} \\ -\dot{\alpha} \\ -\dot{\beta} \end{pmatrix} \tag{2.39}$$

Transition Conditions If the robot is in a ballistic phase ($x_d = 0$), three kinematic relations are supervised to detect a transition from this contact situation to another contact situation ($x_d = 1, 2, 3$):

$$\begin{aligned} s_{0,1}(\boldsymbol{x}) &= \eta = 0 \\ s_{0,2}(\boldsymbol{x}) &= \eta + 2l_f \sin\alpha = 0 \\ s_{0,3}(\boldsymbol{x}) &= \eta^2 + \alpha^2 = 0 \end{aligned}$$

If the robot is tilted on the left foot edge ($x_d = 2$), it is supervised if

$$s_{2,1}(\boldsymbol{x}) = \alpha = 0$$

to detect landing with the whole foot ($x_d = 1$). On the other hand the ground contact force is monitored to detect detaching:

$$s_{2,0}(\boldsymbol{x}, u) = R_y(\boldsymbol{x}, u)$$

Analog transition surfaces $s_{3,1}$ and $s_{3,0}$ are defined for contact with the right foot edge ($x_d = 3$). In the stable support contact phase ($x_d = 1$) the robot may detach either to reach a ballistic phase ($x_d = 0$) if

$$s_{1,0}(\boldsymbol{x}, u) = R_y(\boldsymbol{x}, u) = 0$$

or begins to tip over around one of the edges ($x_d = 2, 3$) where

$$\begin{aligned} s_{1,2}(\boldsymbol{x}, u) &= r_{\text{zmp}}(\boldsymbol{x}, u) = 0 \quad \text{or} \\ s_{1,3}(\boldsymbol{x}, u) &= r_{\text{zmp}}(\boldsymbol{x}, u) - 2l_f = 0. \end{aligned}$$

The ZMP r_{zmp} is calculated from the contact moment T_z and the contact force R_y. (See Sec. 2.2.3.)

Reinitialization. A collision event occurs either when landing from a ballistic motion phase or when landing from tilted. Again the hybrid state and in particular the joint velocities are allowed to behave discontinuous $\boldsymbol{\zeta}^+ = \boldsymbol{\varphi}(\boldsymbol{\zeta}^-)$.

If the robot is in a ballistic phase ($x_d = 0$) and makes contact, the contact is either with the left foot edge, the right foot edge, or with the whole foot. Then the mass matrix $\boldsymbol{M}_0$ from (2.35) is used with the appropriate Jacobian that differs for the three collision types. The resulting jump map is:

$$\boldsymbol{\varphi}_{0,\{1,2,3\}}(\boldsymbol{\zeta}) = \begin{pmatrix} \boldsymbol{q}^- \\ \dot{\boldsymbol{q}}^- - \boldsymbol{M}_0^{-1}\boldsymbol{J}_{\{1,2,3\}}^T(\boldsymbol{J}_{\{1,2,3\}}\boldsymbol{M}_0^{-1}\boldsymbol{J}_{\{1,2,3\}}^T)^{-1}\boldsymbol{J}_{\{1,2,3\}}\dot{\boldsymbol{q}}^- \\ \{1,2,3\} \end{pmatrix}$$

If the robot is in a tilting contact situation, and the tilt angle α becomes zero, the model is simplified in the following way: It is assumed that even if right tilting and left tilting consecute directly, the contact situation stable support is taken for a very short time. This assumption makes it possible to use the same collision law for all possible contact situations following a tilting contact phase. For a more accurate model, the collision law depends on the following contact situation and has thus to be chosen different if the next contact situation is stable contact or if the next contact situation is tilting contact again.

Again, $\boldsymbol{M}_0$ is taken from (2.35) and $\boldsymbol{J}_1$ is assumed to realize the constraints (2.36). Then, the collision law is:

$$\boldsymbol{\varphi}_{2,1}(\boldsymbol{\zeta}) = \begin{pmatrix} \boldsymbol{q}^- \\ \dot{\boldsymbol{q}}^- - \boldsymbol{M}_0^{-1}\boldsymbol{J}_1^T(\boldsymbol{J}_1\boldsymbol{M}_0^{-1}\boldsymbol{J}_1^T)^{-1}\boldsymbol{J}_1\dot{\boldsymbol{q}}^- \\ 1 \end{pmatrix} \tag{2.40}$$

In analogy an equation can be derived for $\boldsymbol{\varphi}_{3,1}(\boldsymbol{\zeta})$. Detachment events again do not cause discontinuities in $\boldsymbol{q}$ or $\dot{\boldsymbol{q}}$.

The transition graph in Fig. 2.13 gives a summary of the hybrid model for the monoped robot.

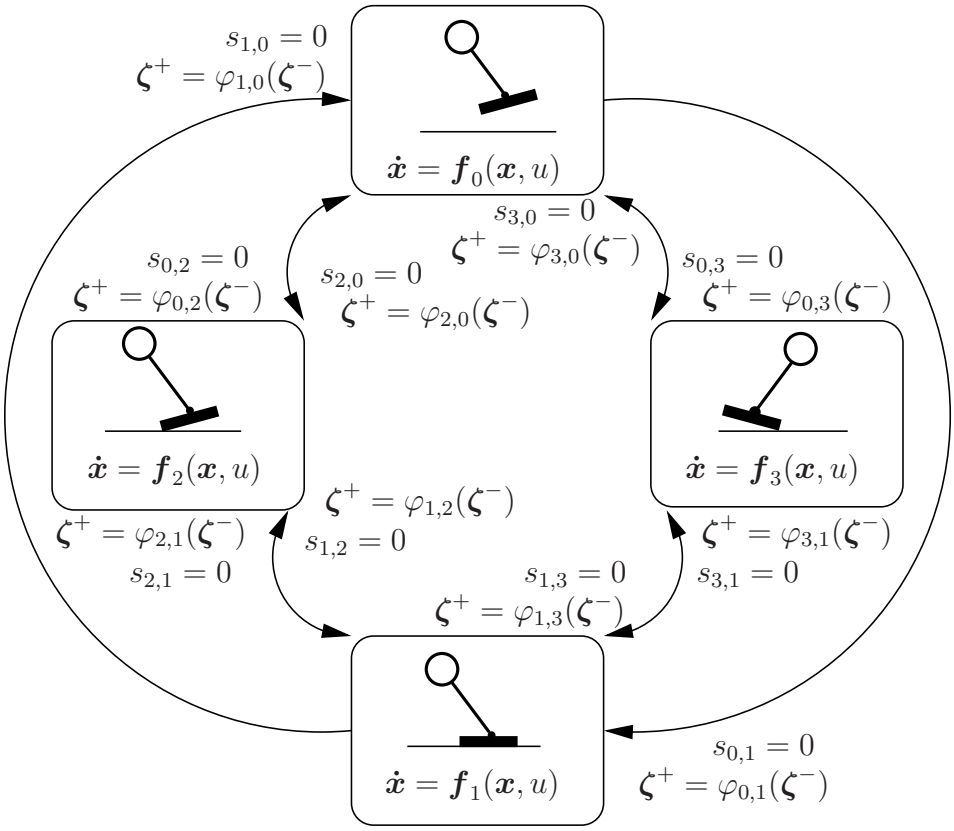

Figure 2.13: Detailed transition graph for the monoped robot.

2.3.3 Gymnast Robot

The gymnast robot that is considered in this section combines features of the compass gait robot from Sec. 2.3.1 and the monoped robot from Sec. 2.3.2. A sketch is given in Fig. 2.14. Like the compass gait robot, the gymnast has two feet. In contrast, the feet are not point feet but have an extended surface where contact is possible, similar to the monoped robot. Again, not all possible contact situation are considered in the hybrid model. Double support and sliding are omitted. Furthermore no direct transition between left tilted and right tilted is allowed. This results in seven possible contact situations and transitions as depicted in Fig. 2.15.

Figure 2.15 also gives an insight on how the gymnast robot is related to compass gait and monoped robot. The transition graph is equivalent to the compass gait transition graph in Fig. 2.8 if the contents of the shaded boxes are each summarized to one contact situation. The transition structure inside the shaded boxes is alike the transition structure of the monoped in Fig. 2.12.

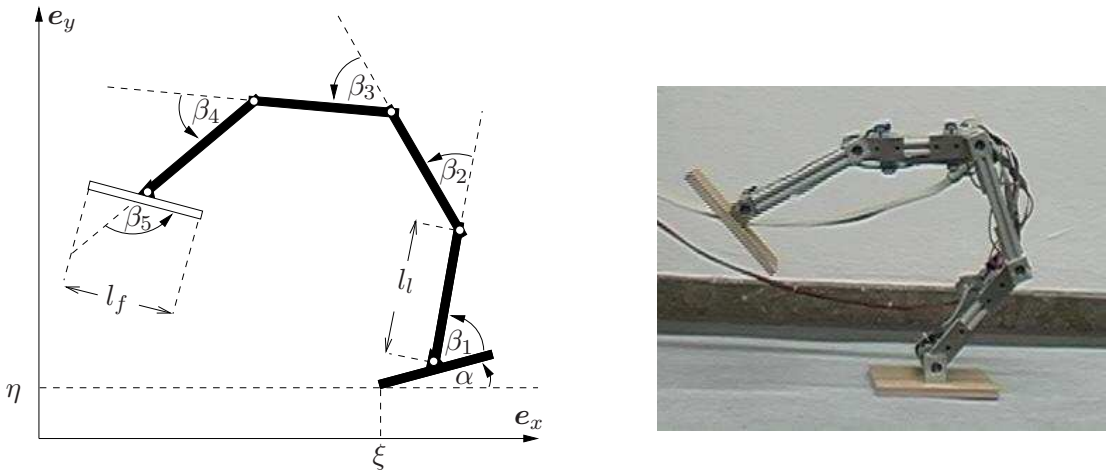

Figure 2.14: Gymnast robot with four links, five actuated joints, and two feet. The distance on a link between the joint axes is l_l, the total height is about 90 cm. Joint axes are parallel, allowing locomotion only in x-direction. Foot length is l_f. A foot has mass m_f, a link has mass m_l. The mass distribution is accounted for by inertia tensors that are obtained from a CAD model of the experimental platform.

For the unconstrained, ballistic contact situation, the robot has eight degrees of freedom, five of those are actuated by motors. The generalized coordinates vector for this robot is

$$\boldsymbol{q} = \begin{pmatrix} \xi \\ \eta \\ \alpha \\ \beta_1 \\ \vdots \\ \beta_5 \end{pmatrix}.$$

The masses and lengths used in the following chapters for trajectory planning and motion analysis are taken from a CAD model of an experimental platform.

m_f	0.300	kg
m_h	0.107	kg
l_l	20.5	cm
l_f	18.0	cm

The derivation of the hybrid model is analog to the derivation for compass gait or monoped robot, although analytic expression for equations of motion or transition conditions are lengthy and thus omitted here.

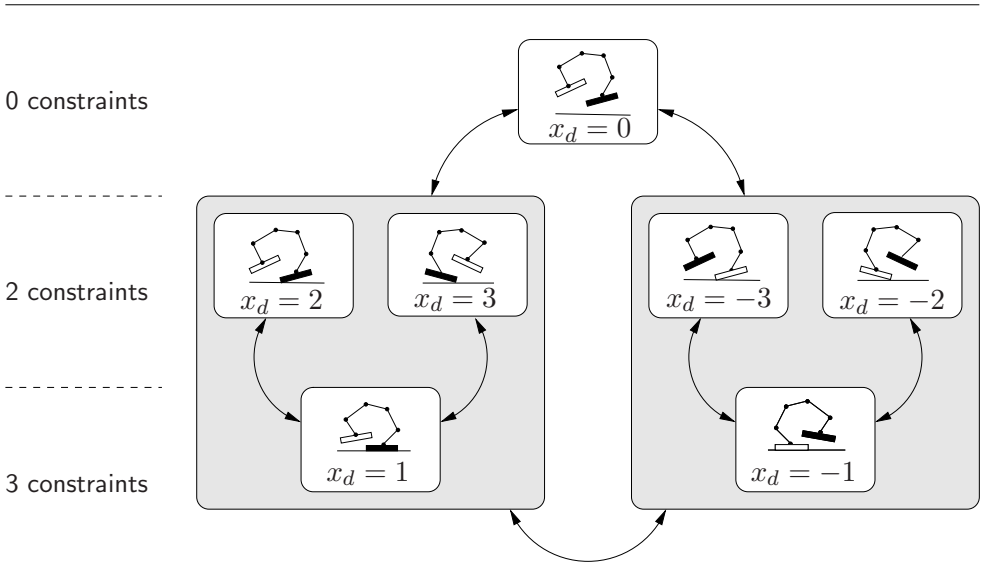

Figure 2.15: Transition graph for the gymnast robot.

2.4 Alternative Modeling Approaches

2.4.1 Complementarity Modeling

The complementarity modeling approach enables a compact description of the dynamics of a mechanical system with unilateral constraints [48, 72, 84]. The idea is to use linear complementarity problems to solve the dynamics with unilateral constraints and was introduced by Moreau in 1966 [91]. A simulation based on the complementarity model is easy to realize and allows to consider a higher number of contact situations without increasing effort of programming. Hence it is possible to consider also double support contact phases. An adaptation to sliding contact is also tractable and outlined in [48].

In the following, a mechanical complementarity model is presented in the simplest possible form. Therefore, consider again a robotic system with n_q degrees of freedom. In order to assure that no robot parts intersect ground the constraints $c_i(\boldsymbol{q}) \geq 0$ are introduced, where c_i $(1 \leq i \leq N_c)$ is a distance function of a potential contact point to the ground. The number of imposed constraints cannot exceed the number of degrees of freedom of the robot, $N_c \leq n_q$. See Fig. 2.16 for an illustration of the definition of constraint functions. In addition, it is required that the constraints are independent that means their Jacobian $\boldsymbol{J}(\boldsymbol{q})$ has full rank.

$$\boldsymbol{J}(\boldsymbol{q}) = \frac{\partial \boldsymbol{c}(\boldsymbol{q})}{\partial \boldsymbol{q}} = \begin{pmatrix} \frac{\partial c_1}{\partial q_1} & \cdots & \frac{\partial c_1}{\partial q_n} \\ \vdots & \cdots & \vdots \\ \frac{\partial c_{N_c}}{\partial q_1} & \cdots & \frac{\partial c_{N_c}}{\partial q_n} \end{pmatrix}$$

An expression for the equations of motion subject to inequality constraints $\boldsymbol{c}(\boldsymbol{q}) \geq \boldsymbol{0}$ can be derived by coupling of the inequality constraints to the dynamics (2.9) using Lagrange multipliers and auxiliary (slack) variables. The auxiliary variables are necessary to transform inequality constraints to equality constraints. See [84] for a derivation of the following

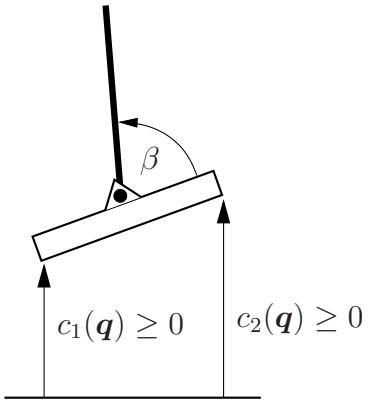

Figure 2.16: Constraint functions $c_i(\boldsymbol{q}) \geq 0$ for complementarity modeling.

equation system:

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{n}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \boldsymbol{u} + \boldsymbol{J}(\boldsymbol{q})^T \boldsymbol{\lambda} \tag{2.41}$$

$$\boldsymbol{c} \geq \boldsymbol{0},\ \boldsymbol{\lambda} \geq \boldsymbol{0} \tag{2.42}$$

$$\boldsymbol{\lambda}^T \boldsymbol{c} = 0 \tag{2.43}$$

The mechanical interpretation is intuitive: The inequalities in (2.42) demand compliance with the constraints $\boldsymbol{c}(\boldsymbol{q}) \geq \boldsymbol{0}$, and on the other hand ensure that ground contact forces act unilateral and only support is possible and not attraction. The equality constraint in (2.43) demands that either the contact force λ_i is zero or the distance c_i is zero. The resulting acceleration of the system subject to the acting constraint forces is obtained as solution of the equation of motion (2.41).

A solution of (2.41) respecting (2.42) and (2.43) is possible by solution of related linear complementarity problems. A linear complementarity problem solves for unknown variables $\boldsymbol{y}$ and $\boldsymbol{z}$ in:

$$\boldsymbol{y} - \boldsymbol{B}\boldsymbol{z} = \boldsymbol{b}$$

$$\boldsymbol{y} \geq \boldsymbol{0},\ \boldsymbol{z} \geq \boldsymbol{0}$$

$$\boldsymbol{y}^T \boldsymbol{z} = 0$$

See [84] for solution algorithms and geometrical interpretation of the linear complementarity problem. Equations (2.41), (2.42), and (2.43) can be reformulated to yield a linear complementarity problem if $\boldsymbol{c}(\boldsymbol{q}) \geq \boldsymbol{0}$ is differentiated twice. Then $\boldsymbol{B} = \boldsymbol{J}^T \boldsymbol{M}^{-1} \boldsymbol{J}$ and $\boldsymbol{b} = -\boldsymbol{J}^T \boldsymbol{M}^{-1} \boldsymbol{n} + \dot{\boldsymbol{J}}\dot{\boldsymbol{q}}$, and the linear complementarity problem solves for the unknowns $\boldsymbol{y} = \ddot{\boldsymbol{c}}$ and $\boldsymbol{z} = \boldsymbol{\lambda}$. Substituting $\boldsymbol{\lambda}$ into (2.41) enables a solution of the dynamics. The non-zero entries in $\boldsymbol{\lambda}$ indicate that the corresponding contact point participates in contact. Solving for the contact forces $\boldsymbol{\lambda}$ is in particular important when a collision occurs. It is then not clear which of the contacting points participate in collision and which of the contacting point dissolve while collision.

A related possibility for legged robot modeling is to apply a complementarity system in an event based formulation as proposed in [84]. It is then not necessary to calculate the contact situation in every integration step. Of interest are only times where a possible change of the contact situation occurs. This is either if a contact force becomes zero or if a new constraint is added to the set of active constraints. If the set of active constraints becomes larger, also collisions have to be considered. If neither of the events occur, the contact situation is not altered and a differential description in minimal coordinates can be used to integrate between events.

The above introduced formulation is still simplified. If only horizontal distances are constrained, vertical sliding with zero friction will occur in the simulation. Thus, also vertical constraints can be considered in the complementarity problem. Therefore friction between ground and contact points is introduced and additional auxiliary constraint forces are used. Since the constraint in the vertical direction is not unilateral, the auxiliary forces are not constrained to positive values. Also stiction and sliding friction have to be discerned. Glocker *et al.* [48] provide a formulation of the complementarity problem with frictional effects considered, where additional auxiliary states are introduced to transform the problem with mixed constraints to a complementarity problem.

2.4.2 Compliant Ground Modeling

The hybrid model presented in Sec. 2.3 and its complementarity formulation in Sec. 2.4.1 were based on a rigid body assumption. Another class of modeling approaches uses a compliant body assumption. Then contact situation changes and collisions are not modeled as instantaneous actions. Compliant ground models are used by Denk *et al.* [38] to model the controlled behavior for optimal trajectories. Also the simulation framework for the humanoid robots Johnnie and Lola at Technische Universität München [23] is based on the compliance assumption. Albro *et al.* [1] even use a compliant contact formulation as basis for optimization.

In compliant modeling of ground contact, virtual spring-damper elements are attached to contacting foot points to simulate the contact. The spring-damper elements apply forces that hold the robot foot on the walking plane after it makes contact. Therefore, methods that rely on compliance are also termed penalty methods.

The simulation framework of Denk *et al.* [38] applies the idea of Marhefka *et al.* [86] to use nonlinear spring-damper characteristics for the vertical contact. The horizontal contact is modeled by linear elements. Figure 2.17 illustrates the set-up for modeling. If the point of first contact is denoted $\boldsymbol{r}_0$, the force in vertical direction as F_y, the horizontal component as F_x, and the actual position of the contacting foot edge as $\boldsymbol{r}$, then the penalizing forces are computed as:

$$F_x = -k_1(r_x - r_{0,x}) - k_2\dot{r}_x$$

$$F_y = \begin{cases} -k_3 r_y - k_4 r_y \dot{r}_y & \text{if } -k_3 r_y - k_4 r_y \dot{r}_y \geq 0 \\ 0 & \text{if } -k_3 r_y - k_4 r_y \dot{r}_y < 0 \end{cases}$$

The case distinction for the vertical force components F_y inhibits attracting ground contact forces. The nonlinearity $r_y \dot{r}_y$ inhibits discontinuities in the force when the foot looses

contact for $r_y = 0$. The parameters k_1, k_2, k_3, and k_4 are constant. The contact forces are coupled into the dynamical model as input forces.

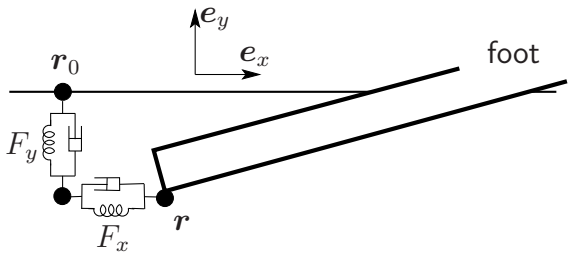

Figure 2.17: Compliant ground assumption. Action of the spring-damper elements connected to a contacting foot edge.

Compliant ground modeling is often used to simplify modeling of ground contact because the model has a constant number of degrees of freedom for all contact situations and a collision model is implicitly included. But nevertheless the modeling framework has hybrid characteristics. A surveillance function has to be included in the simulation to detect when the foot makes contact ($r_y = 0$, $\dot{r}_y < 0$) and when contact is dissolved ($r_y = 0$, $\dot{r}_y > 0$). The above stated formulation yields a smooth trajectory for the vertical contact force due to the nonlinearity that was included. But the horizontal force component behaves discontinuous at penetration time as well as at dissolving time. For reliable simulation results, the numerical integrator has to be stopped and restarted for these event times.

2.4.3 Discussion

The hybrid model has advantages for control theory. The dynamical description for every individual contact phase has a description in minimal coordinates. This enables efficient controller design based on fundamental methods of nonlinear control, as feedback linearization. Another advantage of hybrid models is that they enable efficient numerical simulation. In contrast, for compliant ground modeling the differential equations are often stiff and numerical integration is slow.

The present approach of hybrid modeling of legged locomotion also has weaknesses. Until now, for the sake of simplicity, for every event it was assumed that the consecutive contact situation is unique. A first model refinement should account for an iterative search for the consecutive contact situation after a collision occurred.

The present hybrid model considers only few of the possible contact situations and also the planned motion considers only few contact situations. As long as the simulated motion does not differ too much from the motion obtained from trajectory planning, unmodeled transitions will not occur in simulation. If in the simulation, an unmodeled contact situation is entered, the simulation is aborted. This applies in particular when double-support contact situations occur. Also, neglecting of sliding ground contact is seen as model simplification. Considering more contact situations in the hybrid modeling framework, however, enhances the programming complexity, in particular because the number of possible transitions increases.

Complementarity modeling is well suited for models with a high number of possible contact situations because the calculation of the consecutive ground contact situation is subject of an optimization problem. There, the dynamical description is not in minimal coordinates for every contact situation. A compromise is to solve the transitions as complementarity problem and nevertheless do the integration between the transitions in minimal coordinates. Although the complementarity formulation allows fast implementation of a high number of contact situations, complexity also increases with model refinements. Examples are the consideration of three-dimensional dynamics or the inclusion of friction during collisions. Until now, the complementarity formulation is not yet widely used in control theory for legged robots. Certainly, it will find its applications if the basic problems of stability and trajectory planning are solved for simple examples and new challenges are sought in refined models.

The compliant ground assumption in modeling results in numerically stiff differential equations, and hence, simulation is slow. Also for this modeling framework, the dynamical system is not in minimal coordinates. Since also in compliant ground modeling, collision detection is necessary resulting in a hybrid system description, it is believed that the simulation simplifies not very much. An additional question is how to parameterize the spring-damper elements to achieve realistic behavior.

Out of the scope of the presented approach, but still an important open question, is to judge how well the model approximates physical reality.

2.5 Summary

A modeling environment that allows variable ground contacts is essential for motion planning and controller design of legged locomotion systems. Also collisions between limbs and environment must be accounted for by the modeling framework. The variable ground contact situation when feet detach or make contact and collision events are seen as essential characteristics of legged robot locomotion.

The chapter illustrates how a hybrid (discrete-continuous) modeling framework fits the demands of legged robot modeling and how a hybrid model that is based on a rigid body assumption is related to compliant ground models. Also the similarities between complementarity models and hybrid models are discussed. Both approaches are rigid body formulations of the multi-body dynamics.

Hybrid models allow switches in the dynamical properties of the system description when the ground contact situation of a legged robot changes. In addition, the hybrid model allows for discontinuities in the states to represent instantaneously acting collisions. The discrete dynamics interacts when an event occurs. Events for legged robots occur when contact forces become zero or robot feet touch the ground. After introduction of the underlying mechanical equations, a hybrid model was presented for three example systems: a compass gait robot, a monoped robot, and a gymnast robot. In the following chapters, the models are used for trajectory planning, control, and stability analysis.

In the literature, the application of hybrid models is common for legged robot locomotion modeling. In the presented approach, robots are considered to especially have feet with various possibilities of contact. In particular, the alternation between tilting around the toe, tilting around the heel, and stable support is often neglected and presents a challenge for trajectory planning and control.

3 Trajectory Planning for Legged Robots

3.1 Introduction and State of the Art

Planning trajectories offline and replaying them during experiment is a common approach to realize locomotion of legged robotic systems, where a set of different preplanned trajectories is summarized in a database. For example in the approach of Denk *et al.* [38, 39], the database elements are trajectories that are determined by optimal control. The appropriate trajectory in the experiment is chosen depending on sensory information to fit actual requirements, e.g. in step length, velocity, or step height. The online planning task can thus be reduced to footstep planning [30]. Of special interest are periodic trajectories where repeated execution enables long locomotion cycles with small planning effort. This thesis considers preplanning of trajectories for dexterous locomotion in which different ground contact situations are part of the desired locomotion cycle. In particular underactuated contact situations will be part of the planned motion, such as free rotation around foot edges or ballistic phases. Online planning of trajectories is rarely possible with todays limited computing power because of the numerical complexity of the problem. Kondak *et al.* [82] propose an online planning method for biped robots where the trajectories do not include underactuated contact phases. Another type of online planning is the reinforcement learning approach by Tedrake [128]. There, the numerical optimization process is performed on the experimental platform Toddler.

A common and powerful tool for offline planning of trajectories is optimal control [124]. One of the most advanced numerical realizations of optimal control methods is the combination of direct collocation with sequential quadratic programming (SQP), e.g. used in the software package DIRCOL [130]. Optimal control is powerful but not easy to apply. In general, a good starting value has to be known to achieve convergence. The desired solution often results from an iterative process of problem refinement beginning with a rough approximation where a good initial guess is available. Trajectory planning for legged systems is therefore often not based on optimal control. Sometimes static approaches are used [68], or trajectories are adapted from human motion captured data [96]. Also common is the inverted pendulum approach [77], where the legged robot is approximated by an inverted pendulum pivoting around the ground contact point.

Optimal control has yet been applied successfully for hybrid system problems [28], and much research is done to generalize optimal control formulations and algorithms for the special requirements of hybrid systems, see [74] for an overview. One special field of mechatronic hybrid systems are legged robots. In [38, 39] optimization for a humanoid robot with full actuation is presented. For simplification of the optimization, transition times between different contact situations are preset and not subject to optimization. This is only possible if transition conditions are static in the sense that they are only functions of

the posture. As soon as transition conditions are dynamic and depend nonlinearly on the full state and input torques, this simplification is not applicable any more. Similar results for optimal trajectories of a walking robot are given in [58]. Optimal control formulations allow to include constraints, as angle or torque limitations for the joints or contact force and ZMP restrictions. In the approach of Mombaur *et al.* [90], the eigenvalues of the first-return map are used to find periodic solutions with fastest convergence to the desired trajectories after disturbances.

The problems considered in this thesis address some difficulties of optimization: Underactuated motion phases are an essential part of planned motion resulting in an alternation of fully actuated and underactuated motion phases. Additionally, the transition times cannot be defined in advance since transition conditions depend nonlinearly on the state and the input. One of the few successful applications of optimal control for legged robots with underactuation was presented by Fujimoto [45] using a conjugate gradient method. In this thesis a simplified problem is treated and solved for the example robots. By introduction of parameters, a boundary value problem is derived and the solution reduces to finding zeros of the boundary function. Although the problem is now simple enough to be solved in short time, some limitations have to be accepted: Planned trajectories are not optimal in energy consumption and side conditions, such as force or position constraints, are not taken into account.

A restriction in all approaches for trajectory planning of legged robots is still that the sequence of contact situations, in other words the discrete trajectory, has to be predefined and is not subject of optimization.

This chapter introduces the trajectory planning method for robotic systems with hybrid model by solution of boundary value problems (Sec. 3.2) which is illustrated for the compass gait robot in Sec. 3.3, for the monoped robot in Sec. 3.4, and for the gymnast robot in Sec. 3.5. The chapter is resumed in Sec. 3.6.

3.2 Boundary Value Problems in Trajectory Planning

Trajectory planning for legged robotic systems solves the problem of finding a dynamically feasible motion that accomplishes desired behavior like walking, jogging, hopping, etc. Dynamically feasible means that a combination of hybrid state trajectory $\boldsymbol{\zeta}$ and torque trajectory $\boldsymbol{u}$ exists that is a valid solution of the hybrid dynamical system. Therefore the result of trajectory planning is not only the trajectory in the hybrid state space but also the adequate input torque that realizes the tracking of the joint trajectory.

Periodicity is a key feature of planned motion for legged systems. One periodic motion pattern can be repeated several times, e.g. to overcome a long distance, without increasing the planning effort. Therefore the emphasis in this thesis is laid on planning periodic locomotion.

The presented method uses knowledge of the control structure in the trajectory planning formulation. If the controller inputs are desired trajectories, the trajectory design task is transformed to a calculation of feasible desired joint trajectories $\boldsymbol{\beta}^d$ and resulting hybrid

state trajectory $\boldsymbol{\zeta}$. Here a feedback linearizing controller is chosen that decouples the actuated degrees of freedom. Information on control is thus included in the trajectory planning process. See Fig. 3.1 for an overview.

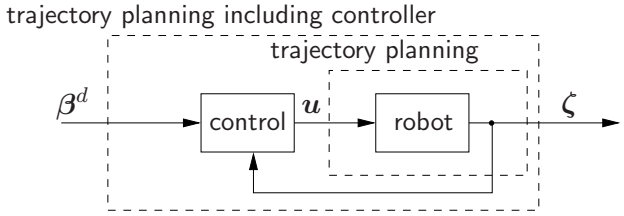

Figure 3.1: Trajectory planning and control law. In trajectory planning feasible pairs of desired trajectory $\boldsymbol{\beta}^d$ and hybrid state vector $\boldsymbol{\zeta}$ are determined.

In the following subsections, trajectory planning is described beginning with the choice of parameter-dependent desired trajectories in Sec. 3.2.1. Then the feedback linearization and the control law are presented in Sec. 3.2.2. Boundary conditions for locomotion problems are discussed in Sec. 3.2.3, and it is concluded with notes on numerical realization in Sec. 3.2.4.

3.2.1 Desired Trajectories

Desired trajectories are periodic, not only in the continuous states but also in the discrete sense demanding

$$\boldsymbol{\zeta}(t) = \boldsymbol{\zeta}(t+T),$$

where $T > 0$ is the period length of the trajectory. Periodic behavior for an actuated joint is obtained by choosing periodic desired trajectories $\boldsymbol{\beta}^d(t)$ and a controller that yields tracking. The boundary value problem has to make sure that also the nonactuated joint trajectories are periodic, as well as the discrete trajectory $x_d(t)$.

3.2.2 Feedback Linearization

For the feedback linearization, it is essential that the dynamical system is in minimal coordinates for every individual contact situation. The feedback linearization for fully actuated motion phases differs from that for underactuated motion phases by the occurrence of an internal dynamics.

Actuated Case. If the robot is fully actuated, the feedback linearizing controller reduces to a computed torque controller. Starting from the equations of motion

$$\boldsymbol{M}(\boldsymbol{\beta})\ddot{\boldsymbol{\beta}} + \boldsymbol{n}(\boldsymbol{\beta}, \dot{\boldsymbol{\beta}}) = \boldsymbol{u},$$

the choice of $\boldsymbol{u}(\boldsymbol{v}) = \boldsymbol{M}(\boldsymbol{\beta})\boldsymbol{v} + \boldsymbol{n}(\boldsymbol{\beta}, \dot{\boldsymbol{\beta}})$ yields a complete linearization

$$\ddot{\boldsymbol{\beta}} = \boldsymbol{v}. \tag{3.1}$$

Underactuated Case. If the robot is in an underactuated motion phase, an internal dynamics remains after linearization of the actuated degrees of freedom. A descriptive reason is that the dimension of the input is smaller than the dimension of the state. Here, it is started from

$$\begin{pmatrix} \boldsymbol{M}_a & \boldsymbol{M}_{ab} \\ \boldsymbol{M}_{ab} & \boldsymbol{M}_b \end{pmatrix} \begin{pmatrix} \ddot{\boldsymbol{\alpha}} \\ \ddot{\boldsymbol{\beta}} \end{pmatrix} + \begin{pmatrix} \boldsymbol{n}_a \\ \boldsymbol{n}_b \end{pmatrix} = \begin{pmatrix} \boldsymbol{0} \\ \boldsymbol{u} \end{pmatrix},$$

where $\boldsymbol{\alpha}$ collects unactuated joint angles and $\boldsymbol{\beta}$ collects actuated joint angles. The choice of a transformation

$$\boldsymbol{u}(\boldsymbol{v}) = \left(\boldsymbol{M}_b - \boldsymbol{M}_{ab}\boldsymbol{M}_a^{-1}\boldsymbol{M}_{ab}\right)\boldsymbol{v} - \boldsymbol{M}_{ab}\boldsymbol{M}_a^{-1}\boldsymbol{n}_a + \boldsymbol{n}_b$$

yields a linearization for the actuated degrees of freedom

$$\ddot{\boldsymbol{\beta}} = \boldsymbol{v},$$

but for this case the nonlinear internal dynamics

$$\ddot{\boldsymbol{\alpha}} = \boldsymbol{M}_a^{-1}\left(-\boldsymbol{n}_a - \boldsymbol{M}_b\boldsymbol{v}\right)$$

remains for the unactuated degrees of freedom. Thus, using $\boldsymbol{f}_{\text{int}}(\boldsymbol{\alpha}, \boldsymbol{\beta}, \dot{\boldsymbol{\alpha}}, \dot{\boldsymbol{\beta}}, \boldsymbol{v}) = \boldsymbol{M}_a^{-1}\left(-\boldsymbol{n}_a - \boldsymbol{M}_b\boldsymbol{v}\right)$, the complete transformed dynamics can be summarized in:

$$\begin{pmatrix} \ddot{\boldsymbol{\alpha}} \\ \ddot{\boldsymbol{\beta}} \end{pmatrix} = \begin{pmatrix} \boldsymbol{f}_{\text{int}}(\boldsymbol{\alpha}, \boldsymbol{\beta}, \dot{\boldsymbol{\alpha}}, \dot{\boldsymbol{\beta}}, \boldsymbol{v}) \\ \boldsymbol{v} \end{pmatrix} \tag{3.2}$$

Linear Controller Design. Ideal tracking for the actuated joints in the absence of disturbances is accomplished by application of the linear control law

$$\boldsymbol{v} = \ddot{\boldsymbol{\beta}}^d + \boldsymbol{K}_D(\dot{\boldsymbol{\beta}}^d - \dot{\boldsymbol{\beta}}) + \boldsymbol{K}_P(\boldsymbol{\beta}^d - \boldsymbol{\beta}). \tag{3.3}$$

The controller corrects whenever $\dot{\boldsymbol{\beta}} \neq \dot{\boldsymbol{\beta}}^d$ or $\boldsymbol{\beta} \neq \boldsymbol{\beta}^d$ and enables asymptotic convergence to the desired trajectory. The dynamic characteristics, such as transient time and transient overshoot, are determined by the eigenvalues of the characteristic polynomial of the linear control law and can be defined by diagonal gain matrices $\boldsymbol{K}_P$ and $\boldsymbol{K}_D$.

Solving for a trajectory has thus been transformed from a search for the appropriate control torque $\boldsymbol{u}$ to a search for the appropriate desired trajectory $\boldsymbol{\beta}^d$, since (3.1) and (3.2) are control systems that have $\boldsymbol{\beta}^d$ as control input by the choice of $\boldsymbol{v}$ in (3.3).

The transformation into a system with linear subsystem is essential for convergence of the numerical solution of the boundary value problem. Application of a simpler PD control law in the formulation of the problem did not result in solutions.

3.2.3 Boundary Conditions

To assure periodicity of legged locomotion, the initial configuration and the final configuration have to be equal: For most legged robots in addition the symmetric construction can be taken into account to simplify the trajectory planning problem. Then only half of the periodic cycle is planned and the second half is mirrored. Therefore the boundary condition relates the mirrored final configuration after half the period with the initial configuration. If the desired motion pattern consists of several contact situations, a boundary condition is added for every intermediate transition times. The boundary conditions also account for reinitialization rules that come from collision modeling. In the sequel, general formulations for boundary value problems are introduced.

If an initial value problem is denoted

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}) \qquad \boldsymbol{x}(t_0) = \boldsymbol{x}_0$$

with $\boldsymbol{x} \in \mathbb{R}^n$, then a corresponding (two-point) boundary value problem has the form

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}) \qquad \boldsymbol{\varrho}\left(\boldsymbol{x}(t_0), \boldsymbol{x}(t_f)\right) = \boldsymbol{0}. \tag{3.4}$$

Here, $\boldsymbol{\varrho}(\boldsymbol{x}(t_0), \boldsymbol{x}(t_f)) : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^n$ collects the boundary conditions. That means, the n conditions that constitute the behavior of the system are distributed between initial time and final time conditions and allow pure initial conditions, pure final conditions, and combinations of initial and final conditions. For locomotion systems, to achieve periodic trajectories, in particular the combined conditions are important to relate the initial and the final configuration. Obviously, it is not guaranteed that a solution for a boundary value problem exists.

Often a parameter vector $\boldsymbol{p}$ is introduced for a boundary value problem to include unknown parameters.

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{p}) \qquad \boldsymbol{\varrho}\left(\boldsymbol{x}(t_0), \boldsymbol{x}(t_f), \boldsymbol{p}\right) = \boldsymbol{0}$$

If $\boldsymbol{p} \in \mathbb{R}^{n_p}$, the mapping $\boldsymbol{\varrho}$ is allowed to be of dimension $n + n_p$, and a solution of the boundary value problem consists of state trajectories $\boldsymbol{x}(t)$ and appropriate parameters $\boldsymbol{p}$.

For trajectories with more than one contact situation, multi-point boundary value problems arise that are denoted by:

$$\dot{\boldsymbol{x}} = \begin{cases} \boldsymbol{f}_0(\boldsymbol{x}, \boldsymbol{p}) & \text{for} \quad t_0 < t < t_1 \\ \vdots & \vdots \\ \boldsymbol{f}_{m-1}(\boldsymbol{x}, \boldsymbol{p}) & \text{for} \quad t_{m-1} < t < t_m = t_f \end{cases} \qquad \boldsymbol{\varrho}(\boldsymbol{x}(t_0), \boldsymbol{x}(t_1), \ldots, \boldsymbol{x}(t_{m-1}), \boldsymbol{x}(t_f), \boldsymbol{p}) = \boldsymbol{0}$$

Here, for every time range $t_k < t < t_{k+1}$, a different dynamical description $\dot{\boldsymbol{x}} = \boldsymbol{f}_k(\boldsymbol{x}, \boldsymbol{p})$ is given. Boundary conditions are also allowed for intermediate transition times t_k with $k = 1, \ldots, m-1$. Introducing the additional possibility that transitions between the phases are discontinuous $\boldsymbol{x}(t_k^-) \neq \boldsymbol{x}(t_k^+)$, the boundary condition can be rewritten

$$\boldsymbol{\varrho}(\boldsymbol{x}(t_0), \boldsymbol{x}(t_1^-), \boldsymbol{x}(t_1^+), \ldots, \boldsymbol{x}(t_{m-1}^-), \boldsymbol{x}(t_{m-1}^+), \boldsymbol{x}(t_f), \boldsymbol{p}) = \boldsymbol{0}.$$

After scaling time from $t_k < t < t_{k+1}$ onto $0 < \tilde{t} < 1$, the multi-point boundary value problem is transformed to a two-point boundary value problem:

$$\dot{\tilde{\boldsymbol{x}}} = \begin{pmatrix} \dot{\boldsymbol{x}}_0 \\ \vdots \\ \dot{\boldsymbol{x}}_{m-1} \end{pmatrix} = \begin{pmatrix} \tilde{\boldsymbol{f}}_0(\boldsymbol{x}_0, \boldsymbol{p}) & \text{for} \quad 0 < \tilde{t} < 1 \\ \vdots & \vdots \\ \tilde{\boldsymbol{f}}_{m-1}(\boldsymbol{x}_{m-1}, \boldsymbol{p}) & \text{for} \quad 0 < \tilde{t} < 1 \end{pmatrix} \qquad \tilde{\boldsymbol{\varrho}}\left(\tilde{\boldsymbol{x}}(0), \tilde{\boldsymbol{x}}(1), \boldsymbol{p}\right) = \boldsymbol{0}$$

A solution can now be found using a standard boundary value problem solver since the transformed boundary value problem is in standard form (3.4):

$$\dot{\tilde{\boldsymbol{x}}} = \tilde{\boldsymbol{f}}(\tilde{\boldsymbol{x}}, \boldsymbol{p}) \qquad \tilde{\boldsymbol{\varrho}}\left(\tilde{\boldsymbol{x}}(0), \tilde{\boldsymbol{x}}(1), \boldsymbol{p}\right) = \boldsymbol{0} \tag{3.5}$$

Obviously, the more contact situations a planned motion pattern consists of, the higher is the problem dimension of the numerical problem. The reason is that every additional contact phase adds n differential equations to the boundary value problem in the transformation from the multi-point boundary value problem to the two-point boundary value problem. If initial, final, or any transition time for the problem is not known in advance, they enter the boundary value problem as a parameter in the parameter vector $\boldsymbol{p}$ due to time scaling.

Before motion planning for a certain robotic system, it is required to split the number of unknown parameters $\boldsymbol{p}$ into a set of input parameters $\boldsymbol{p}_{\text{in}}$ that will be defined before solution and a set of output parameters $\boldsymbol{p}_{\text{out}}$ that can be determined as solution of the boundary value problem. Assume n_{bvp} is the dimension of the vector $\tilde{\boldsymbol{x}}$ and r_{bvp} is the dimension of the boundary function $\tilde{\boldsymbol{\varrho}}$. Then $n_p = r_{\text{bvp}} - n_{\text{bvp}}$ parameters in $\boldsymbol{p}_{\text{out}}$ can be determined by solution of the boundary value problem. The remaining parameters are summarized in $\boldsymbol{p}_{\text{in}}$ and have to be specified before solution.

3.2.4 Numerical Solution

For the boundary value problem in standard form (3.5), any numerical solver for boundary value problems can be applied. Here, the Matlab solver `bvp4c` [113] is used that relies on a collocation method. Therefore, the time domain is split into subdomains where the solution is approximated by cubic splines. The cubic splines are determined such that the concatenated trajectory is smooth, fulfills the differential equation at the grid points, and meets the boundary conditions. The differential equation system is linearized using the Lobatto IIIa formula, and the solution is found iteratively along the gradient by line search. If necessary, the grid is refined and the process is repeated [113].

Simple shooting methods [104] turned out not to find a solution even for simple legged systems as for the compass gait robot. In shooting methods the differential equation is integrated for an initial value, and a new initial value is chosen based on the gradient information of the boundary condition. A possible reason for non-convergence of shooting methods is the unstable characteristics of the inverted pendulum-like contact situation where integration errors blow up resulting in useless gradient information.

Also applicable is the Newton-Raphson method for finding zeros of the boundary function $\boldsymbol{\varrho}$. A Newton-Raphson method was not investigated in this research but is used in approaches of passive walking to find the periodic passive trajectories [83].

3.3 Compass Gait Robot

Introduction. The compass gait robot, as introduced in Sec. 2.3.1, is often used as a passive walker without actuation in the hip joint [53]. Thus an optimization of trajectories is not necessary because the robot follows a periodic path that is determined by its dynamic properties and the inclination of the walking plane. Nevertheless, for investigation in simulation, possible paths can be determined numerically. Solutions are sets of initial conditions that result in periodic trajectories. A boundary value problem has to be solved that relates the initial configuration with the final configuration to achieve periodicity. Also for the actuated case, where a torque u is applied in the hip, possible trajectories can be found by solution of a similar boundary value problem.

Due to the symmetric construction of the robot, the problem of finding periodic walking trajectories has only one phase. Only the first step is planned, the second step is symmetric to the first step. Then step one and the symmetric step two are repeated. For illustration see Fig. 3.2.

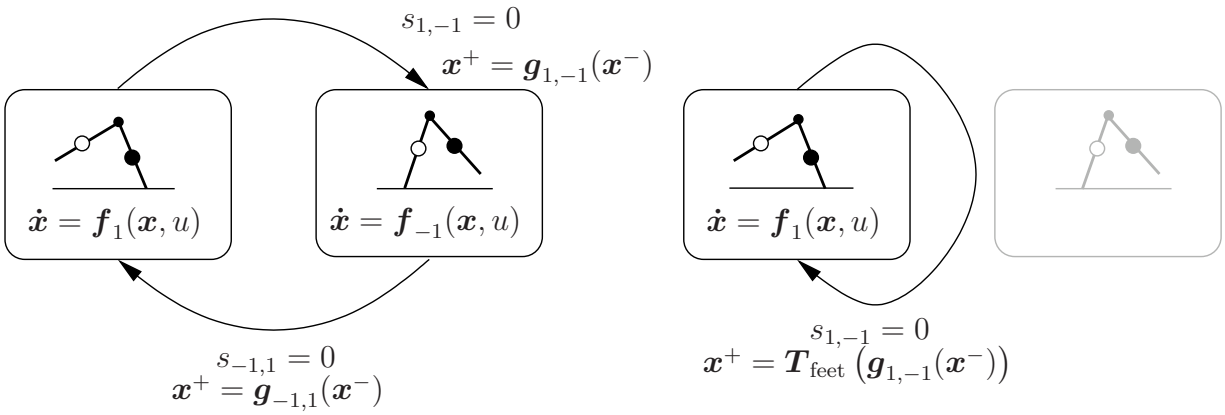

Figure 3.2: Transition graph for trajectory planning of the compass gait robot. Left: Transition graph for a complete stride consisting of two steps. Right: Transition graph accounting for the symmetric construction that makes planning necessary only for one step.

Desired Trajectories. In the following the compass gait robot is considered on a non-inclined walking plane with actuating torque u in the hip. A desired periodic trajectory β^d for the step angle is defined as

$$\beta^d(t) = \pi + A \cos \omega t, \tag{3.6}$$

where A is the maximum step width and ω is the step frequency.

Feedback Linearization. The solution of the boundary value problem answers the question when and in what configuration walking has to be initiated to obtain periodicity. Therefore, at first, a control law is chosen that yields tracking of the desired trajectory $\beta^d(t)$. A feedback linearization approach is chosen in which the motion of the nonactuated joint is the internal dynamics, see Sec. 3.2.2. With appropriate choice of $u(v)$, the

equations of motion $\boldsymbol{M}_1(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{n}_1(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \boldsymbol{u}$ from (2.28) are transformed into

$$\begin{pmatrix} \ddot{\alpha} \\ \ddot{\beta} \end{pmatrix} = \begin{pmatrix} f_{\text{int}}(\alpha, \beta, \dot{\alpha}, \dot{\beta}, v) \\ v \end{pmatrix}.$$

The linear control law $v = \ddot{\beta}^d + K_P(\beta^d - \beta) + K_D(\dot{\beta}^d - \dot{\beta})$ yields asymptotic tracking of the desired trajectory for β.

Boundary Conditions. The initial but still unknown time for the walking motion is denoted t_0, the final time t_f corresponds to half the period of the excitatory motion (3.6) and is $t_f = t_0 + \frac{\pi}{\omega}$. It is assumed that the step starts at the time where the roles of the feet exchange and both feet are in contact with the ground for a very short time. For this configuration α and β are coupled by $\alpha(t_0) = \pi - \frac{\beta(t_0)}{2}$. See Fig. 3.3 for geometrical illustration.

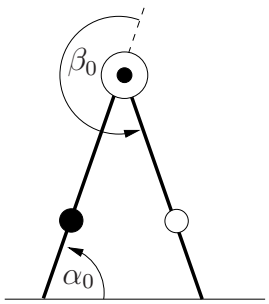

Figure 3.3: Initial configuration of the compass gait robot. The equation $(\beta_0 - \pi) + 2\alpha_0 = \pi$ that sums up the interior triangle angles relates α_0 and β_0 resulting in $\alpha_0 = \pi - \frac{\beta_0}{2}$.

The initial unknown configuration $\boldsymbol{x}_0 = \boldsymbol{x}(t_0)$ is then $\boldsymbol{x}_0 = \left(\pi - \frac{\beta_0}{2},\, \beta_0,\, \dot{\alpha}_0,\, \dot{\beta}_0\right)^T$, with unknown parameters $\beta_0 = \beta(t_0)$, $\dot{\alpha}_0 = \dot{\alpha}(t_0)$, and $\dot{\beta}_0 = \dot{\beta}(t_0)$. Additionally, the initial time t_0 is unknown. The solution of the boundary value problem is thus defined by the parameter vector $\boldsymbol{p}_{\text{out}} = \left(t_0,\, \beta_0,\, \dot{\alpha}_0,\, \dot{\beta}_0\right)$.

Then, the boundary conditions $\boldsymbol{\varrho}\left(\boldsymbol{x}(t_0), \boldsymbol{x}(t_f), \boldsymbol{p}\right) = \boldsymbol{0}$ that are necessary to achieve periodic steps are:

$\varrho_1(\boldsymbol{x}(t_0), \boldsymbol{x}(t_f), \boldsymbol{p}) = \alpha(t_0) - (\pi - \frac{\beta_0}{2})$	Initial condition for α
$\varrho_2(\boldsymbol{x}(t_0), \boldsymbol{x}(t_f), \boldsymbol{p}) = \beta(t_0) - \beta_0$	Initial condition for β
$\varrho_3(\boldsymbol{x}(t_0), \boldsymbol{x}(t_f), \boldsymbol{p}) = \dot{\alpha}(t_0) - \dot{\alpha}_0$	Initial condition for $\dot{\alpha}$
$\varrho_4(\boldsymbol{x}(t_0), \boldsymbol{x}(t_f), \boldsymbol{p}) = \dot{\beta}(t_0) - \dot{\beta}_0$	Initial condition for $\dot{\beta}$
$\varrho_5(\boldsymbol{x}(t_0), \boldsymbol{x}(t_f), \boldsymbol{p}) = l \sin \alpha(t_f) + l \sin\left(\alpha(t_f) + \beta(t_f)\right)$	Foot touches ground at final time
$\varrho_6(\boldsymbol{x}(t_0), \boldsymbol{x}(t_f), \boldsymbol{p}) = \beta(t_f) + \beta_0 - 2\pi$	Final position for β
$\varrho_7(\boldsymbol{x}(t_0), \boldsymbol{x}(t_f), \boldsymbol{p}) = \dot{\alpha}(t_f^+) + \dot{\beta}(t_f^+) - \dot{\alpha}_0$	Final velocity for α
$\varrho_8(\boldsymbol{x}(t_0), \boldsymbol{x}(t_f), \boldsymbol{p}) = \dot{\beta}(t_f^+) + \dot{\beta}_0$	Final velocity for β

Again, no explicit condition for $\alpha(t_f)$ is given since $\alpha(t_f)$ is implicitly defined from ϱ_5. In the boundary conditions $\boldsymbol{\varrho}(\boldsymbol{x}(t_0), \boldsymbol{x}(t_f), \boldsymbol{p}) = \mathbf{0}$, $\dot{\alpha}(t_f^+)$ and $\dot{\beta}(t_f^+)$ are velocities after collision. The collision transition is given by the algebraic relation $\boldsymbol{x}^+ = \boldsymbol{g}_{1,-1}(\boldsymbol{x}^-)$ from (2.33). The boundary conditions ϱ_6, ϱ_7, and ϱ_8 yield symmetry since they realize $\boldsymbol{x}_0 = \boldsymbol{T}_{\text{feet}}\left(\boldsymbol{g}_{1,-1}(\boldsymbol{x}^-)\right)$. Geometrical illustration of the boundary conditions is given in Fig. 3.4.

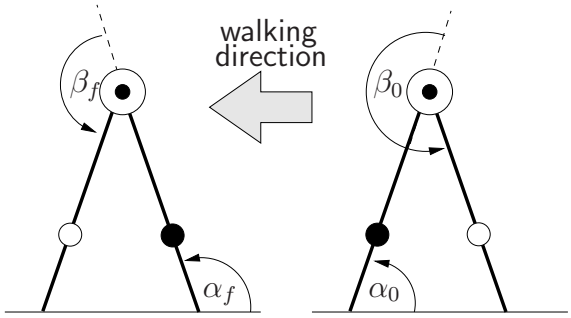

Figure 3.4: Boundary conditions for the compass gait robot. To achieve symmetry $(\pi - \alpha_f) + \alpha_0 + (\pi - \beta_f) = \pi$ has to hold which results in boundary condition ϱ_5. Furthermore $\beta_f + \beta_0 = 2\pi$ has to hold constituting boundary conditions ϱ_6.

The differential equation is of fourth order allowing for four boundary conditions. Since eight boundary conditions are defined, the four-dimensional parameter vector $\boldsymbol{p}_{\text{out}}$ can be determined. The parameters A and ω are input parameters to the boundary value problem. Thus, for every pair $\boldsymbol{p}_{\text{in}} = (A,\, \omega)$, a solution $\boldsymbol{p}_{\text{out}} = (t_0,\, \beta_0,\, \dot{\alpha}_0,\, \dot{\beta}_0)$ is determined if it exists.

Numerical Example. For illustration, three different input parameter pairs $\boldsymbol{p}_{\text{in}} = (A,\, \omega)$ are chosen. The control law $v = \ddot{\beta}^d + K_P(\beta^d - \beta) + K_D(\dot{\beta}^d - \dot{\beta})$ is parameterized by $K_P = 100$ and $K_D = 10\sqrt{2}$. Figure 3.5 depicts a snapshot series of the corresponding walking motion.

Small step amplitude:	$\boldsymbol{p}_{\text{in}} = (0.2\,\text{rad},\, 4.5\frac{\text{rad}}{\text{s}})$
Medium step amplitude:	$\boldsymbol{p}_{\text{in}} = (0.4\,\text{rad},\, 4.5\frac{\text{rad}}{\text{s}})$
Large step amplitude:	$\boldsymbol{p}_{\text{in}} = (0.6\,\text{rad},\, 4.5\frac{\text{rad}}{\text{s}})$

Figure 3.6 gives details of the solutions. In particular, it has to be checked if the following necessary constraint conditions hold since this cannot be considered by the solver for the boundary value problem: The ground contact force has to be positive. In addition, the swing foot should touch ground only at transition times. In the figures, it is seen that the swing foot height is negative for some time. Thus, the trajectories are not feasible unless foot lifting is introduced to solve this foot scuffing problem.

3.4 Monoped Robot

Introduction. The monoped robot cannot perform steps because there is only one foot, see model in Sec. 2.3.2. Nevertheless, its investigation is interesting as the single foot has

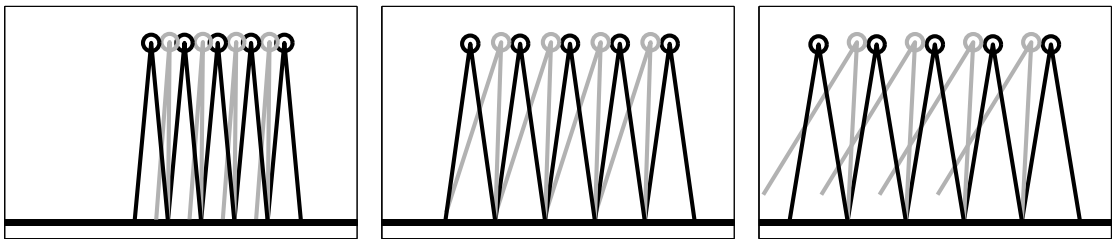

Figure 3.5: Snapshot series of compass gait walking. Left: Small step amplitude $A = 0.2\,\text{rad}$. Middle: Medium step amplitude $A = 0.4\,\text{rad}$. Right: Large step amplitude $A = 0.6\,\text{rad}$. Every picture shows four steps in $2.8\,\text{s}$. The robot is walking from right to left. Supplementary video material in [116].

various options for contacting the ground. A consecution of an underactuated contact situation and a completely actuated contact situation is possible. Since most humanoid robots have rigid feet, research on tilting around foot edges is essential to improve dynamic humanoid walking.

Due to the limited number of actuators, only few motion patterns are possible with the monoped robot. The goal to be considered is to swing the actuated link back and forth around the upright position and excite rhythmic detaching and landing of the foot plate. It is assumed that left tilting and right tilting are symmetric such that only left tilting has to be planned and the symmetry is assured by boundary conditions. So, consideration of the second half of the motion is replaced by an appropriate boundary condition. Two possibilities for motion planning arise: The excitation motion is chosen such that tilting left and tilting right are directly consecuting motion phases. Trajectory planning has to consider one contact phase only, the second half of the motion is mirrored to the first half (Sec. 3.4.1). The second scenario is that tilting left and tilting right is separated by a stable support contact phase. Then the trajectory planning algorithm has to consider two contact phases, e.g. tilting left and stable support. Tilting right is again mirrored to tilting left (Sec. 3.4.2).

Both planning scenarios still have the desired trajectory and the feedback linearization in common:

Desired Trajectories. The desired trajectory $\beta^d(t)$ for the actuated arm that is attached on the foot plate is chosen sinusoidal:

$$\beta^d(t) = \frac{\pi}{2} + A \sin \omega t \tag{3.7}$$

The constant A is the amplitude, and ω is the frequency of the motion of the arm. The goal of trajectory planning is to find appropriate values for amplitude A, frequency ω, initial time t_0, and initial value $\boldsymbol{x}_0$ such that the robot performs periodic rocking back and forth for $t > t_0$. For small values of A and ω, the foot will not detach at all; for large values of A and ω the foot detaches but does not return.

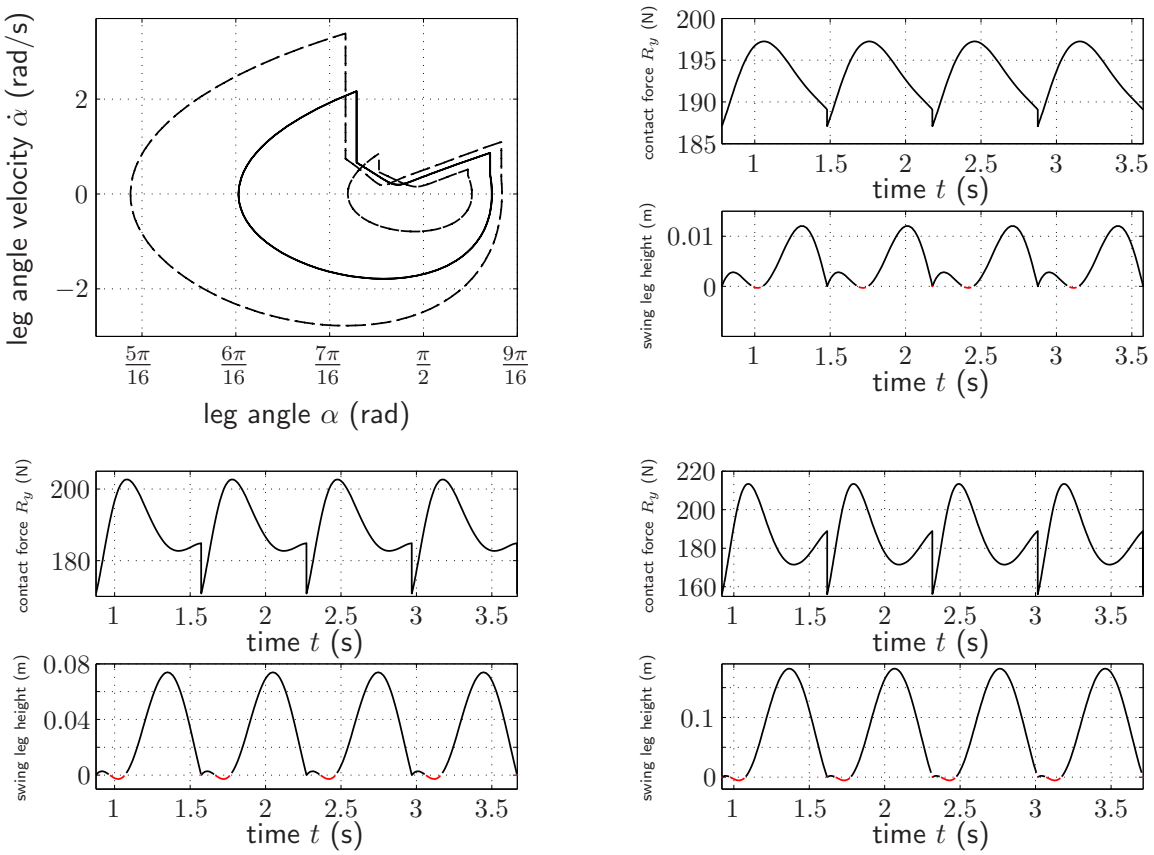

Figure 3.6: Details of compass gait walking with parameters as in Fig. 3.5. Upper Left: Phaseplots for large (dashed), medium (solid), and small (dash-dotted) step amplitudes. Upper Right: Small step amplitude. Lower Left: Medium step amplitude. Lower Right: Large step amplitude. The upper subfigures show respective contact forces R_y, the lower subfigures show respective swing leg heights.

Feedback Linearization. Trajectory planning requires knowledge on the control law that ensures tracking of the desired trajectory β^d with the actuated arm. The control is switched depending on the contact situation. For the fully actuated case ($x_d = 1$) the feedback linearizing controller reduces to a computed torque controller without internal dynamics. The equation of motion (2.37) is by the choice of an appropriate mapping $u(v)$ transformed to $\ddot{\beta} = v$. The tilt angle α is constantly zero. For the underactuated motion phases, as tilting left and tilting right ($x_d = 2, 3$), the equations of motion (2.38) are transformed to

$$\begin{pmatrix} \ddot{\alpha} \\ \ddot{\beta} \end{pmatrix} = \begin{pmatrix} f_{\text{int}}(\alpha, \beta, \dot{\alpha}, \dot{\beta}, v) \\ v \end{pmatrix}.$$

Here the internal dynamics is the dynamics of the tilt angle α. The choice $v = \ddot{\beta}^d + K_P(\beta^d - \beta) + K_D(\dot{\beta}^d - \dot{\beta})$ for the linear control law with appropriate gains K_P and K_D result in asymptotic tracking of the desired trajectory for the actuated arm.

The motion cycle begins at an unknown initial time t_0 and ends at $t_f = t_0 + \frac{\pi}{\omega}$, which corresponds to half the period of the excitatory motion (3.7). It is thereby assumed that t_0 is the time just after landing from tilted right. Thus the motion either begins with a

stable contact phase or with tilting left. In any case the planned motion ends when the foot lands from tilting left.

3.4.1 2-Point BVP for Tilting without Stable Support Phase

Boundary Conditions. At first the boundary value problem for direct consecution of tilting right and tilting left is presented. The trajectory is planned for tilting left. Tilting right is obtained as the mirrored motion. See Fig. 3.7 for illustration.

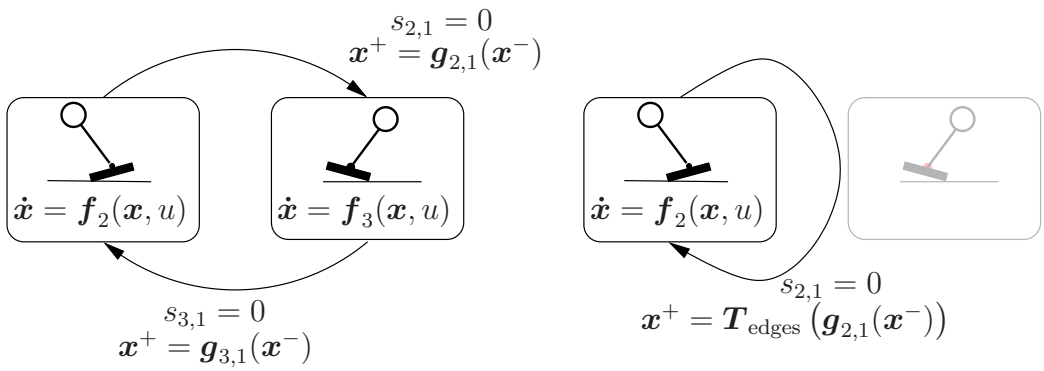

Figure 3.7: Transition graph for trajectory planning of the monoped robot. Left: Transition graph for a complete period, consisting of tilting left and tilting right. Right: Transition graph accounting for the symmetric construction that makes planning necessary only for tilting left.

The initial unknown configuration $\boldsymbol{x}_0 = \boldsymbol{x}(t_0)$ is then $\boldsymbol{x}_0 = \left(0,\, \beta_0,\, 0,\, \dot{\beta}_0\right)^T$, and the contact situation is assumed to be tilting left ($x_d = 2$). A solution is desired to account for the parameters $\boldsymbol{p}_{\text{in}} = (A,\, \omega)$. The unknown parameters are the initial time t_0, the initial angle β_0, and the initial velocity $\dot{\beta}_0$, summarized in $\boldsymbol{p}_{\text{out}} = (t_0,\, \beta_0,\, \dot{\beta}_0)$.

The boundary conditions $\boldsymbol{\varrho}\left(\boldsymbol{x}(t_0), \boldsymbol{x}(t_f), \boldsymbol{p}\right) = \boldsymbol{0}$ that are necessary to achieve periodic tilting back and forth are:

$\varrho_1(\boldsymbol{x}(t_0), \boldsymbol{x}(t_f), \boldsymbol{p}) = \alpha(t_0)$	Initial condition for α
$\varrho_2(\boldsymbol{x}(t_0), \boldsymbol{x}(t_f), \boldsymbol{p}) = \beta(t_0) - \beta_0$	Initial condition for β
$\varrho_3(\boldsymbol{x}(t_0), \boldsymbol{x}(t_f), \boldsymbol{p}) = \dot{\alpha}(t_0)$	Initial condition for $\dot{\alpha}$
$\varrho_4(\boldsymbol{x}(t_0), \boldsymbol{x}(t_f), \boldsymbol{p}) = \dot{\beta}(t_0) - \dot{\beta}_0$	Initial condition for $\dot{\beta}$
$\varrho_5(\boldsymbol{x}(t_0), \boldsymbol{x}(t_f), \boldsymbol{p}) = \alpha(t_f)$	Foot touches ground at final time
$\varrho_6(\boldsymbol{x}(t_0), \boldsymbol{x}(t_f), \boldsymbol{p}) = \beta(t_f) + \beta_0 - \pi$	Final position for β
$\varrho_7(\boldsymbol{x}(t_0), \boldsymbol{x}(t_f), \boldsymbol{p}) = \dot{\beta}(t_f^+) + \dot{\beta}_0$	Final velocity for β

The collision transition is given by the algebraic relation $\boldsymbol{x}^+ = \boldsymbol{g}_{2,1}(\boldsymbol{x}^-)$ from (2.33), and ϱ_6 and ϱ_7 follow from taking into account the symmetry $\boldsymbol{x}_0 = \boldsymbol{T}_{\text{edges}}\left(\boldsymbol{g}_{2,1}(\boldsymbol{x}^-)\right)$.

After transformation to a first order system, the differential equations are of fourth order and allow for four boundary conditions. Since seven boundary conditions are defined, the parameter vector $\boldsymbol{p}_{\text{out}}$ has three entries. The parameters A and ω are input parameters to the boundary value problem. For every pair $\boldsymbol{p}_{\text{in}} = (A, \omega)$, a solution $\boldsymbol{p}_{\text{out}} = (t_0, \beta_0, \dot{\beta}_0)$ is determined, if existing.

Numerical Results. It turns out that two solutions exist for most realizations of $\boldsymbol{p}_{\text{in}} = (A, \omega)$. The first solution is characterized by an initial time close to zero. That means, the tilting left motion corresponds to a motion of the actuated arm to the left side. The second solution family has initial values close to $\frac{\pi}{\omega}$. For this case, the tilting left motion corresponds to a deflection of the arm to the right side. In Fig. 3.8 trajectories for the tilt angle α and the actuated angle β from simulation of the planned trajectory are plotted over time. The controller for all following investigations of the monoped robot is parameterized by $K_P = 100$ and $K_D = 10\sqrt{2}$. It is in particular interesting to observe the size of the control error for the actuated degree of freedom. The control error is always large immediately after a collision to compensate for the velocity error. In the following contact phase, the control error is reduced due to the control law. Since the control law is part of the trajectory planning, the compensation motion after collisions does not result in destabilization of the trajectory.

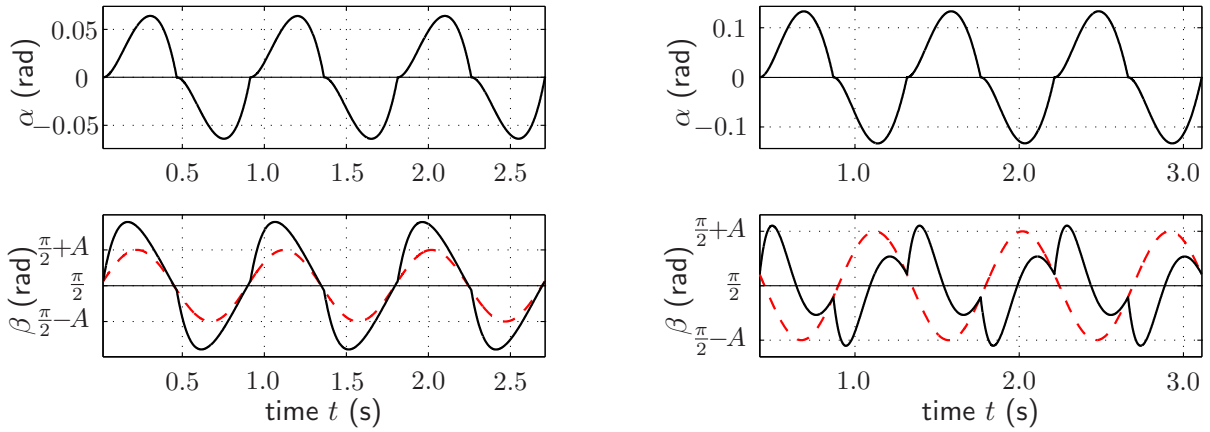

Figure 3.8: Details of monoped trajectories. Left: Initial time is close to zero ($\omega = 7\frac{\text{rad}}{\text{s}}$, $A = 0.05\,\text{rad}$, $t_0 = 0.0398\frac{\pi}{\omega}$). Right: Initial time is close to $\frac{\pi}{\omega}$ ($\omega = 7\frac{\text{rad}}{\text{s}}$, $A = 0.05\,\text{rad}$, $t_0 = 0.9341\frac{\pi}{\omega}$). Top: Tilting angle α over time. Tilting left are positive angles, tilting right are negative angles. Bottom: Actuated Angle β over time (solid) in comparison to the desired trajectory (dashed).

3.4.2 3-Point BVP for Tilting with Stable Support Phase

Boundary Conditions. If a stable contact phase is assumed to separate left tilting and right tilting, the trajectory planning problem has two phases and a multi-point boundary value problem has to be solved.

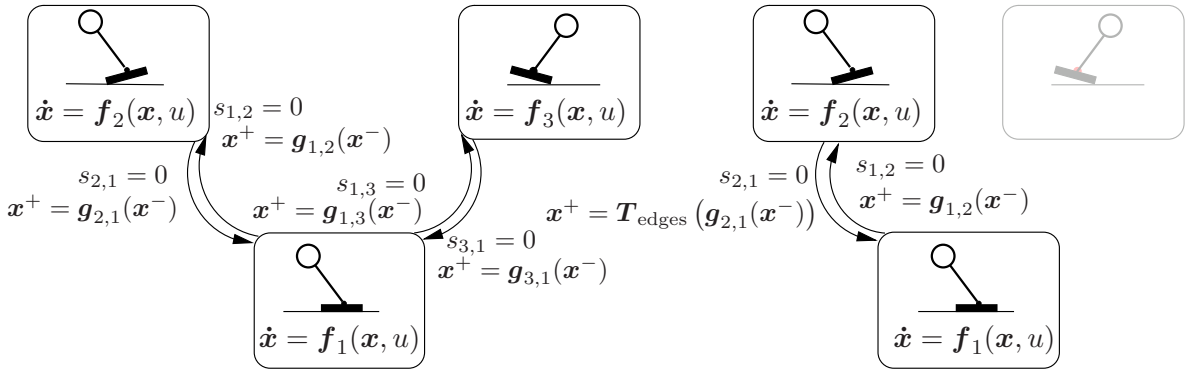

Figure 3.9: Transition graph for trajectory planning for the monoped robot. Left: Transition graph for a complete period, consisting of tilting left, stable support, and tilting right. Right: Transition graph accounting for the symmetric construction that makes planning necessary only for tilting left and stable support.

It is assumed that the first phase is a stable support phase ($x_d = 1$) until the ZMP becomes zero and a tilting left phase follows ($x_d = 2$). The initial unknown configuration $\boldsymbol{x}_0 = \boldsymbol{x}(t_0)$ for unknown initial time t_0 is again $\boldsymbol{x}_0 = \left(0,\, \beta_0,\, 0,\, \dot{\beta}_0\right)^T$, and the contact situation is assumed to be stable support contact ($x_d = 1$). Tilting time t_1 is unknown and is part of the solution. The state $\boldsymbol{x}(t_1)$ is the initial value for the tilting left contact phase with $x_d = 2$. Summarized, the multi-point boundary value problem will solve for $\boldsymbol{p}_{\text{out}} = (t_0,\, t_1,\, \beta_0,\, \dot{\beta}_0)$ for a given pair of parameters $\boldsymbol{p}_{\text{in}} = (A,\, \omega)$.

The boundary conditions $\boldsymbol{\varrho}(\boldsymbol{x}(t_0), \boldsymbol{x}(t_1), \boldsymbol{x}(t_f), \boldsymbol{p}) = \boldsymbol{0}$ that are necessary to achieve periodic tilting back and forth with intermediate stable support phase are:

$\varrho_1(\boldsymbol{x}(t_0), \boldsymbol{x}(t_1), \boldsymbol{x}(t_f), \boldsymbol{p}) = \beta(t_0) - \beta_0$	Initial condition for β
$\varrho_2(\boldsymbol{x}(t_0), \boldsymbol{x}(t_1), \boldsymbol{x}(t_f), \boldsymbol{p}) = \dot{\beta}(t_0) - \dot{\beta}_0$	Initial condition for $\dot{\beta}$
$\varrho_3(\boldsymbol{x}(t_0), \boldsymbol{x}(t_1), \boldsymbol{x}(t_f), \boldsymbol{p}) = r_{\text{zmp}}(t_1)$	Foot begins to tilt
$\varrho_4(\boldsymbol{x}(t_0), \boldsymbol{x}(t_1), \boldsymbol{x}(t_f), \boldsymbol{p}) = \alpha(t_1)$	Transition condition for α
$\varrho_5(\boldsymbol{x}(t_0), \boldsymbol{x}(t_1), \boldsymbol{x}(t_f), \boldsymbol{p}) = \dot{\alpha}(t_1)$	Transition condition for $\dot{\alpha}$
$\varrho_6(\boldsymbol{x}(t_0), \boldsymbol{x}(t_1), \boldsymbol{x}(t_f), \boldsymbol{p}) = \beta(t_1^+) - \beta(t_1^-)$	Transition condition for β
$\varrho_7(\boldsymbol{x}(t_0), \boldsymbol{x}(t_1), \boldsymbol{x}(t_f), \boldsymbol{p}) = \dot{\beta}(t_1^+) - \dot{\beta}(t_1^-)$	Transition condition for $\dot{\beta}$
$\varrho_8(\boldsymbol{x}(t_0), \boldsymbol{x}(t_1), \boldsymbol{x}(t_f), \boldsymbol{p}) = \beta(t_f) + \beta_0 - \pi$	Final position for β
$\varrho_9(\boldsymbol{x}(t_0), \boldsymbol{x}(t_1), \boldsymbol{x}(t_f), \boldsymbol{p}) = \dot{\beta}(t_f^+) + \dot{\beta}_0$	Final velocity for β
$\varrho_{10}(\boldsymbol{x}(t_0), \boldsymbol{x}(t_1), \boldsymbol{x}(t_f), \boldsymbol{p}) = \alpha(t_f)$	Foot touches ground again

The collision transition is given by the algebraic relation $\boldsymbol{x}(t_f^+) = \boldsymbol{g}_{2,1}\left(\boldsymbol{x}(t_f^-)\right)$ from (2.40). The choice of ϱ_8 and ϱ_9 ensures symmetric right tilting by $\boldsymbol{x}_0 = \boldsymbol{T}_{\text{edges}}\left(\boldsymbol{g}_{2,1}\left(\boldsymbol{x}(t_f^-)\right)\right)$.

The differential equation for the first phase (standing on the whole foot plate) is of second order and for the second phase (tilting right) of fourth order. That allows six boundary conditions. Since ten boundary conditions are specified, the parameter vector $\boldsymbol{p}_{\text{out}}$ is four-dimensional. The parameters A and ω are input parameters to the boundary value problem. For every pair $\boldsymbol{p}_{\text{in}} = (A,\, \omega)$, a solution $\boldsymbol{p}_{\text{out}} = (t_0,\, t_1,\, \beta_0,\, \dot{\beta}_0)$ is determined.

Numerical Results. Figure 3.10 depicts details on a result of the trajectory planning by solution of the boundary value problem specified above. A severe drawback of the

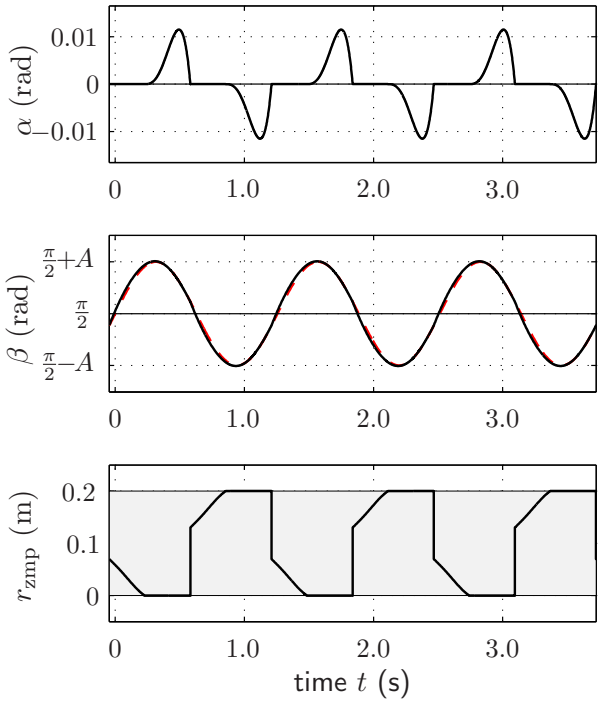

Figure 3.10: Details of monoped trajectories for $A = 0.2\,\text{rad}$, $\omega = 5.0\frac{\text{rad}}{\text{s}}$. Top: Tilting angle α over time. Middle: Actuated angle β over time (solid line) in comparison to the desired trajectory β^d (dashed line). Bottom: ZMP. The ZMP is only evaluated in stable support phases. The shaded area indicates the foot contact area.

trajectory planning procedure is that the solution of the boundary value problem does only consider the states at transition times. For the considered example, it is not assured that the ZMP does not leave the supporting area while stable support is assumed. Thus, after computation of the trajectories as solution of the boundary value problems, trajectories have to be checked on constraint compliance for every contact phase.

3.4.3 Discussion

In general, the maximal tilt angle is higher for solutions without stable support phase. For existence of a stable support phase, it is required that the ZMP does not leave the supporting area after the collisions. This is, due to control action, only possible if the disturbance from the collision is small and thus if the landing velocity is small implying a small maximal tilt angle. From Fig. 3.8 and Fig. 3.10, it is seen see that the nominal trajectory deviates much more from the desired trajectory in the case of large tilting angles. In order to allow larger tilting angles for motion patterns with stable support phase, it is possible to introduce more contact phases into the sequence of contact phases. For example allowing an additional tilting right phase immediately after tilting left to compensate the collision would lead to new feasible motion patterns.

An underlying assumption of the model is that even if tilting right follows tilting left immediately, for an arbitrarily short time, the whole foot has contact. This is considered in the model by the collision law and physically motivated by sticking effects of the foot.

Trajectory planning without consideration of the collision is much easier, compare [118]. Then it is assumed that the collision acts as a disturbance to the system and the controller compensates the disturbance. It was shown in [118] for the monoped robot that only few trajectories, preplanned disregarding collision, could be applied in simulations where collisions are considered.

3.5 Gymnast Robot

Introduction. Trajectory planning for the gymnast robot introduced in Sec. 2.3.3 is more challenging than planning for the other two example systems compass gait robot and monoped robot. Much more periodic motion patterns are possible due to the higher number of degrees of freedom. Possible patterns include walking, hopping, or even somersault. But the complexity of the boundary value problem increases not only because of the increased number of degrees of freedom. Also the increased number of consecutive contact situations that are necessary to realize locomotion, e.g. a walking motion, contributes. Another problem in motion planning is to decide which sequence of contact situations is appropriate for achieving a desired motion pattern.

As a first example a walking motion is considered. Again it is taken advantage of symmetries, and only one step is planned. A step of the walking motion of the gymnast robot comprises three contact situations: It starts with a stable support phase ($x_d = 1$) that lasts until the ZMP crosses the foot edge in walking direction. Then the foot rolls around the toe ($x_d = 2$) until the heel of the second foot touches ground. Before coming back to the stable support contact situation of the foot that was swing foot first, the robot tilts around the heel of its former swing foot ($x_d = -3$). For illustration of the contact situations considered in a walking motion, see Fig. 3.11, which is a subgraph of the transition graph for the gymnast robot in Fig. 2.15.

For the gymnast robot, two symmetries are used in trajectory planning. On the one hand, there is a symmetry between the two legs. Thus any contact situation on the non-reference foot is transformed into the corresponding contact situation of the reference foot by the transformation $\boldsymbol{T}_{\text{feet}}$ that is defined according to that of the compass gait robot (2.31). Moreover there is a symmetry between left and right foot edge. Thus any right tilting contact situation is transformed into a left tilting contact situation by the transformation $\boldsymbol{T}_{\text{edges}}$ that is defined alike for the monoped robot in (2.39). In summary, only two contact situations remain.

Desired Trajectories. Desired trajectories $\beta_1^d, \ldots, \beta_5^d$ are defined for the five actuated joints to achieve a walking motion. Thereby, constant desired trajectories β_1^d, β_5^d are defined for both ankle joints. The step width angle is defined to open and close periodically just like for the compass gait robot realized by definition of β_3^d. The definitions for the knee joint angles differ depending on which of the feet is assumed to be the supporting foot.

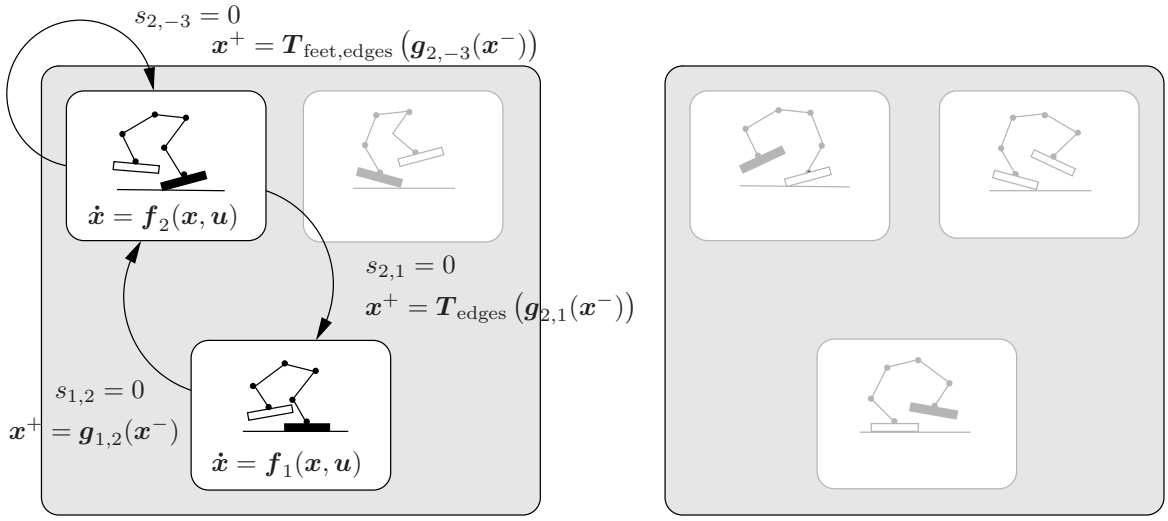

Figure 3.11: Using symmetry in trajectory planning for walking of the gymnast robot.

In the first half of the motion, the reference foot is supposed to be supporting foot, and the knee of the supporting foot is allowed to bend with a small amplitude $\frac{B}{2}$. At the same time, the knee of the swing foot bends with a larger amplitude $2B$ to avoid foot scuffing. For the second half of the period, the feet change roles.

$$
\begin{aligned}
\text{if } \; t \mod \frac{2\pi}{\omega} \leq \frac{\pi}{\omega} \qquad & \beta_1^d(t) = \frac{\pi}{2} \\
& \beta_2^d(t) = \frac{B}{2}\cos 2\omega t - \frac{B}{2} \\
& \beta_3^d(t) = \pi + A\cos\omega t \\
& \beta_4^d(t) = -2B\cos 2\omega t + 2B \\
& \beta_5^d(t) = \frac{\pi}{2} \\
\text{if } \; t \mod \frac{2\pi}{\omega} > \frac{\pi}{\omega} \qquad & \beta_1^d(t) = \frac{\pi}{2} \\
& \beta_2^d(t) = 2B\cos 2\omega t - 2B \\
& \beta_3^d(t) = \pi + A\cos\omega t \\
& \beta_4^d(t) = -\frac{B}{2}\cos 2\omega t + \frac{B}{2} \\
& \beta_5^d(t) = \frac{\pi}{2}
\end{aligned}
\tag{3.8}
$$

The concatenated desired trajectory is smooth in the positions but not smooth in the velocities. Since the discontinuity in the desired velocities results in a discontinuity in the vector field, which is the right hand side of the differential equation, the numerics have to account for this discontinuity. Another transition at $t_d = \frac{\pi}{\omega}$ has to be introduced to the problem. Thus the following boundary value problem has four phases for the numerical solution instead of three, which is the number of contact situations. The transition time $t_d = \frac{\pi}{\omega}$ is chosen to separate the toe roll phase in two phases that are connected by smooth transitions. In the following, the fourth motion phase is not further considered in the derivation of the boundary problem to avoid complicated notation.

Feedback Linearization. A control law is chosen that yields tracking of the desired trajectory $\boldsymbol{\beta}^d$ with vanishing control error in absence of disturbances. A feedback linearization approach is used, and as a consequence, the motion of the nonactuated joints is the internal dynamics, see Sec. 3.2.2. For the gymnast robot, the feedback linearization controller has to be defined for the stable support phase and for the underactuated tilting contact phase.

Boundary Conditions. The initial but still unknown time for the walking motion is denoted by t_0, the final time corresponds to half the period of the excitatory motion $t_f = t_0 + \frac{\pi}{\omega}$. At time t_1 the transition from stable support to the "tilting on toe" contact situation occurs. The ZMP has to be at the left foot edge at $r_{\text{zmp}} = 0$. Then for $t = t_2$ landing on the heel of the swing foot preludes the tilting on the heel contact phase. The transition back to stable support is triggered when the tilt angle becomes zero. The definition of transition times is illustrated in Fig. 3.12 together with the relevant transition conditions. The transition conditions are a subset of the boundary conditions for the trajectory planning boundary value problem, see below.

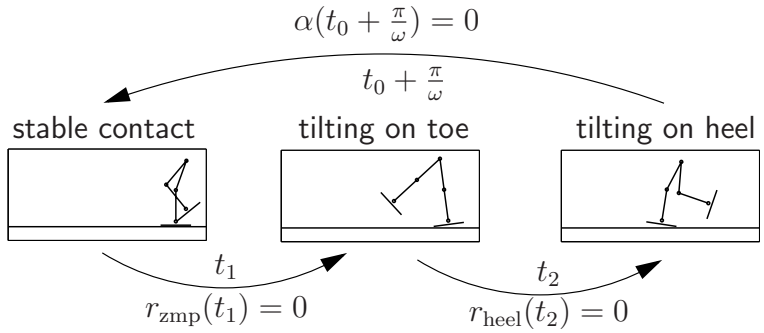

Figure 3.12: Definition of transitions for a walking trajectory of the gymnast robot.

The remaining boundary conditions are made up from ten initial conditions and assign the ten states of the stable support phase yet unknown initial conditions. Then, twelve conditions for the first transition are given that ensure that the initial configuration for the tilting on toe phase has to equal the final configuration of the stable support phase. The twelve boundary conditions for the transition from the reference foot toe to the non-reference foot heel connect the final configuration of the toe roll phase with the initial configuration of the heel roll phase and take into account the collision and symmetry transformations. Eventually, the final configuration of the heel roll phase is related to the initial configuration by ten boundary conditions, where again collision law and symmetry transformations are taken into account. See Fig. 3.11 for graphical illustration.

For the dynamics with three contact phases, the state vector of the boundary value problem has dimension 10+12+12=34. There are ten initial conditions, twelve transition conditions for the first transition, twelve transition condition for the second transition, and ten boundary conditions. Additionally for every transition one transition condition is added resulting in 47 boundary conditions. As a consequence 13 parameters are solution of the boundary value problem. Here the parameter vector $\boldsymbol{p}_{\text{out}} = \big(t_0,\, t_1,\, t_2,\, \beta_{1,0},\, \ldots,\, \beta_{5,0},\, \dot{\beta}_{1,0},\, \ldots,\, \dot{\beta}_{5,0}\big)$ is chosen to be the solution, where the parameters in $\boldsymbol{p}_{\text{in}} = \big(A,\, \omega,\, B\big)$ are specified as input to the problem.

Numerical Example. For illustration two different input parameter realizations $\boldsymbol{p}_{\text{in}} = (A,\, \omega,\, B)$ are chosen. The control law is parameterized by $K_P = 17.5^2$ and $K_D = 17.5\sqrt{2}$.

Slow walking:	$\boldsymbol{p}_{\text{in}} = (0.64\,\text{rad},\, 1.9\,\frac{\text{rad}}{\text{s}},\, 0.4\,\text{rad})$
Fast walking:	$\boldsymbol{p}_{\text{in}} = (0.92\,\text{rad},\, 3.8\,\frac{\text{rad}}{\text{s}},\, 0.4\,\text{rad})$

In Fig. 3.13 snapshot series of the corresponding walking motion are given.

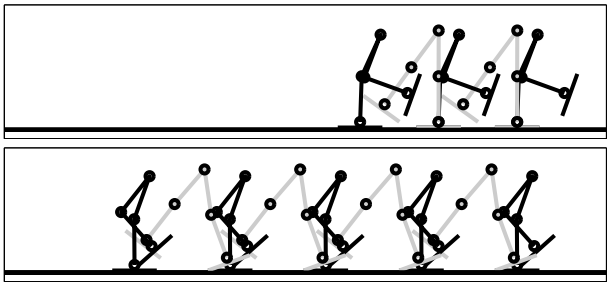

Figure 3.13: Snapshot series of gymnast robot walking for two parameterizations. Top: Slow walking. Bottom: Fast walking. In both cases 3.3 s of locomotion are depicted.

The results of the boundary value problem have to be checked if they are feasible walking trajectories. One major condition is that $t_2 > t_1 > t_0$. Then all constraints for the contact situations have to be checked. This is for example the ZMP in the stable support phase or the tilt angle in the tilting phases.

Hopping. Hopping and somersault motion is interesting because of the alternation between actuated stable support, underactuated tilting motion, and finally a ballistic phase. In Fig. 3.14 a snapshot series of a somersault trajectory is given, see [10] for reference. Also here, periodicity requires the initial posture and the final posture to be related. But for this motion, the stable support phase takes the major part of the period. Although the trajectory is solution of a boundary value problem, the depicted cycle was not found by numerical solution of the boundary value problem. For cyclic stable trajectories, there is always the possibility to find periodic solutions just by simulation. If the initial value of the integration is close enough to the periodic cycle, the trajectory converges to the cycle in simulation. Finding solutions by simulation becomes even easier if stable support contact phases exist. Then, finding an initial posture close to the periodic cycle is relatively easy if the initial condition is assumed in the stable support phase.

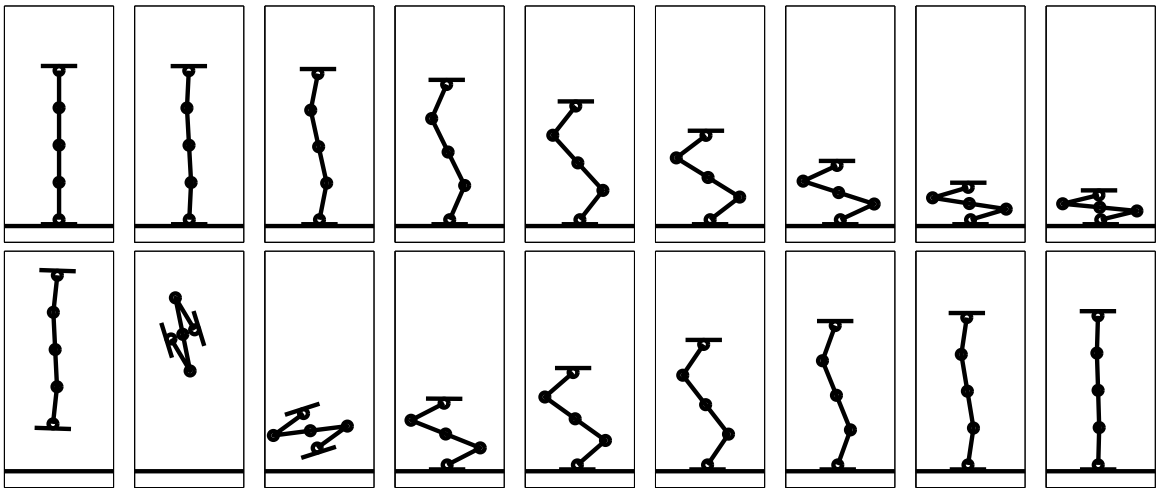

Figure 3.14: Snapshot series of gymnast robot somersault. The complete depicted motion last $4.4\,\mathrm{s}$, the snapshots are taken every $0.25\,\mathrm{s}$. Supplementary video material in [116].

3.6 Summary

Trajectory planning for dexterous motion of legged systems combines two challenges. The first one is the hybrid character of the legged system, where different ground contact situations have different dynamical descriptions and the times of transitions are variable. The second challenge is the underactuatedness in some contact situations, where the nonactuated degrees of freedom are not directly controllable and fewer actuators than degrees of freedom are available.

In this chapter, a trajectory planning method was presented that is a simplification of the optimal control problem in that it considers only the underlying boundary value problem. With this formulation, trajectories for legged robotic systems can be determined numerically without high computational cost. This is in particular important for motion cycles with a higher number of consecutive contact situations including underactuation, where optimal control still often fails. In order to formulate the boundary value problem, assumptions for the desired trajectories are made and a controller for the actuated joints is chosen. It is possible to calculate a set of trajectories with different parameterization concerning e.g. velocity or step width. The boundary value problem was established in this section for the compass gait robot, the monoped robot, and the gymnast robot. Though, the planning method does not consider optimality of the motion or physical constraints to the planning problem and the solutions that are obtained have to be checked on constraint violation prior to application in experiments.

The planning method is closely related to the approach that is used to control the legged robot RABBIT [31]. There, the desired trajectories are not defined on joint level but by the choice of output functions to the control system. The output functions are chosen heuristically and describe desired walking properties as swing foot motion or hip motion. The present approach is a generalization in the sense that an arbitrary number of consecutive contact situations can be allowed.

4 Stability of Periodic Robot Locomotion

4.1 Introduction and State of the Art

For all periodic trajectories, the initial configuration is equal to the configuration after one period. This enables locomotion by successive execution of the periodic trajectories. Nevertheless, simulation experiments give evidence that also the appropriate reaction on disturbances is important to decide on the usefulness of a particular trajectory: Only for some of it, it is possible to simulate the dynamical system for an arbitrary amount of time. For other trajectories, the simulation has to be aborted because the robot leaves the precalculated trajectory already after a tiny disturbance and finally falls. The reason for the failure is the lack of orbital stability.

Orbital stability assures that the solution returns to the precalculated periodic trajectory after small disturbances. To analyse orbital stability, one investigates if small perturbations of the initial configuration blow up or die out in the following motion cycles. Only if small perturbations decay, the periodic trajectory is said to be orbitally stable, and only then the precalculated trajectory is useful for experiments. A common analysis method for periodic orbits of ordinary differential equations uses Poincaré maps (first-return maps) [57, 60].

It was shown by Hiskens [62] that Poincaré map based stability analysis is applicable also for hybrid dynamical systems and in particular for stability analysis of periodic motion of legged robots [61]. Simić *et al.* [114] presented a method for stability analysis that only yields sufficient conditions for stability. A related method was presented by Rubensson *et al.* [108] using multiple Lyapunov functions for piecewise linear hybrid systems.

Poincaré maps for analysis of locomotion stability were used for many passive walking machines [15, 35, 46, 54, 88]. But also, they are a tool that is applied for actuated robots with nonactuated degrees of freedom, e.g. a nonactuated foot ground contact [40, 55]. Hurmuzlu demonstrated that even robots with full actuation can show orbital instability [70].

However, fast disturbance compensation does not correlate with a large region of attraction of the orbit. That means, even if it is known that small disturbances are compensated fast, nothing is known on how large acceptable disturbances are. A numerical approximation of the region of attraction is given in [112] for a simplified passive walking machine. Estimation of the region of attraction is important to realize transitions between motion patterns without additional planning of transition trajectories. If the region of attraction of a desired motion pattern is large, it may cover the final configuration of the actual motion pattern and a stable transition is possible.

A stability analysis was yet presented for the monoped robot in [120] and for the gymnast robot in [121]. In the following, results on stability of periodic orbits are presented in detail in Sec. 4.2, first for ordinary differential equations and then for hybrid systems. In

Sec. 4.3 the orbital stability of trajectories found by solution of boundary value problems in Chap. 3 is investigated for the compass gait robot, for the monoped robot, and for the gymnast robot. Finally, for gymnast robot walking, the feasibility of transitions between slow and fast walking is investigated. A summary in Sec. 4.4 concludes this chapter.

4.2 Poincaré Map Analysis for Periodic Solutions

Prerequisites for analysis methods for periodic dynamical behavior of hybrid systems are analysis tools for ordinary differential equations. These will be discussed in the following Sec. 4.2.1. Section 4.2.2 gives references on related results for hybrid dynamical systems.

4.2.1 Stability of Periodic Solutions of Ordinary Differential Equations

Preliminary Definitions. The following basic definitions and theorems can be found in textbooks from Khalil [79] or Parker *et al.* [101]. An autonomous ordinary differential equation (ODE) is considered

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}), \tag{4.1}$$

where $\boldsymbol{x} \in \mathbb{R}^n$ is the system state and $\boldsymbol{f} : \mathbb{R}^n \to \mathbb{R}^n$ is a Lipschitz-continuous vector field. Thus, there exists exactly one solution for every initial condition $\boldsymbol{x}(t_0) = \boldsymbol{x}_0$. The solution is denoted by the trajectory $\boldsymbol{x}(t)$ or by the flux $\boldsymbol{\phi}_t(\boldsymbol{x}_0)$. The flux $\boldsymbol{\phi}_t(\boldsymbol{x}_0)$ assigns a trajectory $\boldsymbol{x}(t)$ to every initial value $\boldsymbol{x}_0$.

Trajectory sensitivities provide information about the dependence of the solution $\boldsymbol{\phi}_t(\boldsymbol{x}_0)$ of (4.1) on the initial value $\boldsymbol{x}_0$.

Definition 4.2.1 (Trajectory Sensitivity) *The trajectory sensitivity $\boldsymbol{\Phi}_t(\boldsymbol{x})$ is defined as*

$$\boldsymbol{\Phi}_t(\boldsymbol{x}) = \frac{\partial \boldsymbol{\phi}_t(\boldsymbol{x})}{\partial \boldsymbol{x}} .$$

If it is presumed that $\boldsymbol{x}_1 = \boldsymbol{x}(T) = \boldsymbol{\phi}_T(\boldsymbol{x}_0)$, the trajectory sensitivity describes the trajectory deviation $\delta\boldsymbol{x}_1$ after small perturbations $\delta\boldsymbol{x}_0$ in the initial values

$$\delta\boldsymbol{x}_1 = \boldsymbol{\Phi}_T(\boldsymbol{x}_0)\delta\boldsymbol{x}_0.$$

In the same way, the trajectory sensitivity acts for the vector-fields:

$$\boldsymbol{f}(\boldsymbol{x}_1) = \boldsymbol{\Phi}_T(\boldsymbol{x}_0)\boldsymbol{f}(\boldsymbol{x}_0) \tag{4.2}$$

Definition 4.2.2 (Periodic Solution, Periodic Orbit) *A solution $\boldsymbol{\phi}_t(\boldsymbol{x}_0)$ of (4.1) is a periodic solution with period length $T > 0$ if*

$$\boldsymbol{\phi}_{T+t}(\boldsymbol{x}_0) = \boldsymbol{\phi}_t(\boldsymbol{x}_0)$$

holds for all times $t \in \mathbb{R}$. The choice of $\boldsymbol{x}_0$ is not unique, any point of the periodic solution is a valid starting value. A periodic orbit (closed orbit) is the image of a periodic solution in the phase portrait.

A periodic solution will in the following be abbreviated by γ and is a closed invariant set of the differential equation. Invariance means that for any $\boldsymbol{x}_0 \in \gamma$, it holds that $\boldsymbol{\phi}_t(\boldsymbol{x}_0) \in \gamma$ for arbitrary times t.

The definition of Lyapunov stability for invariant sets of an ODE is similar to the definition of Lyapunov stability for an equilibrium point. Since also a fixed point is an invariant set, the stability definition for invariant sets generalizes the definition for fixed points.

Definition 4.2.3 (Lyapunov Stability of Closed Invariant Sets) *Let $\gamma \subset \mathbb{R}^n$ be a closed invariant set for (4.1), and let*

$$U_\varepsilon = \{\boldsymbol{x} \in \mathbb{R}^n | \operatorname{dist}(\boldsymbol{x}, \gamma) < \varepsilon\}$$

be an ε-neighborhood of γ with

$$\operatorname{dist}(\boldsymbol{x}, \gamma) = \inf_{\boldsymbol{y} \in \gamma} \|\boldsymbol{x} - \boldsymbol{y}\|.$$

The closed invariant set γ is stable if, for each $\varepsilon > 0$ there exists a $\delta > 0$, such that if $\boldsymbol{x}(t_0) \in U_\delta$, it follows that $\boldsymbol{x}(t) \in U_\varepsilon$ for all times t. The closed invariant set γ is asymptotically stable if there exists a $\delta > 0$, such that if $\boldsymbol{x}(t_0) \in U_\delta$ it follows that $\lim_{t\to\infty} \operatorname{dist}(\boldsymbol{x}, \gamma) = 0$.

Poincaré Map. For stability analysis often an approach applying first-return maps (Poincaré maps) is used. The stability investigation of the periodic orbit is reduced to stability investigation of a fixed point of a lower-dimensional discrete mapping. The presented results can be found in the books by Guckenheimer and Holmes [57], Hirsch and Smale [60], or Parker and Chua [101].

Definition 4.2.4 (Poincaré Map) *The set γ is a periodic orbit for (4.1), and Σ is a local transversal cross section of dimension $n-1$. A cross section is defined as transversal if $\boldsymbol{n}(\boldsymbol{x})^T \boldsymbol{f}(\boldsymbol{x}) \neq 0$, where $\boldsymbol{n}$ is the normal vector to Σ. Let $\boldsymbol{x}^*$ denote the unique point of intersection between γ and Σ. Then the Poincaré map $\boldsymbol{P} : U \to \Sigma$ is defined in a neighborhood $U \subset \Sigma$ of $\boldsymbol{x}^*$ as*

$$\boldsymbol{P}(\boldsymbol{x}) = \boldsymbol{\phi}_{\tau(\boldsymbol{x})}(\boldsymbol{x}). \tag{4.3}$$

The time $\tau(\boldsymbol{x})$ denotes the first time of intersection of the orbit and the cross section after starting the integration in $\boldsymbol{x} \in U$. The mapping has a fixed point $\boldsymbol{x}^*$, where $\boldsymbol{P}(\boldsymbol{x}^*) = \boldsymbol{x}^*$.

The definition is illustrated in Fig. 4.1, and in the following the idea of the proof that orbital stability of the periodic solution can be concluded from the eigenvalues of the linearization of the Poincaré map is outlined. If all eigenvalues of the linearization around the fixed point have an absolute value smaller than one, the periodic trajectory is asymptotically stable. For details, see [60].

At first it has to be assured that the Poincaré map is well defined. It can be shown that the map $\tau(\boldsymbol{x})$ exists in a neighborhood of $\boldsymbol{x}^*$. The differentiability of $\boldsymbol{P}(\boldsymbol{x})$ is concluded from differentiability of the flow $\boldsymbol{\phi}_t(\boldsymbol{x})$.

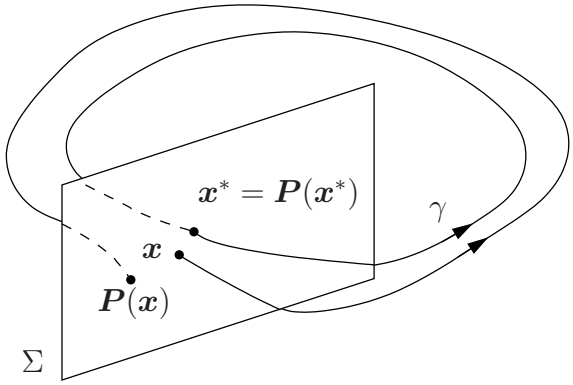

Figure 4.1: Poincaré map.

By definition, the Poincaré map has a fixed point $\boldsymbol{x}^*$, where $\boldsymbol{P}(\boldsymbol{x}^*) = \boldsymbol{x}^*$. It can be proved that asymptotic stability of the fixed point $\boldsymbol{x}^*$ implies asymptotic orbital stability of the periodic solution. The stability of the fixed point $\boldsymbol{x}^*$ of the nonlinear difference equation $\boldsymbol{x}_{k+1} = \boldsymbol{P}(\boldsymbol{x}_k)$ is related to the fixed point stability of its linearization

$$\left.\frac{\partial \boldsymbol{P}(\boldsymbol{x})}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\boldsymbol{x}^*}. \tag{4.4}$$

This procedure is usually referenced as Lyapunov's indirect method. The linearized difference equation has an asymptotically stable fixed point if all eigenvalues of the linearization (4.4) are strictly inside the unit circle. The asymptotical stability is carried forward to the fixed point of the nonlinear difference equation and finally to the periodic orbit.

An expression for (4.4) is derived from the definition of $\boldsymbol{P}(\boldsymbol{x})$ from (4.3):

$$\left.\frac{\partial \boldsymbol{P}(\boldsymbol{x})}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\boldsymbol{x}^*} = \boldsymbol{\Phi}_\tau(\boldsymbol{x}^*) + \boldsymbol{f}(\boldsymbol{x}^*)\boldsymbol{D}\tau(\boldsymbol{x}^*) \tag{4.5}$$

It is assumed that the cross section Σ is determined by an algebraic relation $s(\boldsymbol{x}) = 0$, then $\boldsymbol{n}(\boldsymbol{x})^T = \partial s(\boldsymbol{x})/\partial \boldsymbol{x}$ is normal to the cross section Σ. The derivative of the return time τ in (4.5) is obtained from evaluation of

$$\frac{\mathrm{d}}{\mathrm{d}\boldsymbol{x}} s\left(\boldsymbol{\phi}_{\tau(\boldsymbol{x})}(\boldsymbol{x})\right) = \boldsymbol{0}, \quad \text{resulting in} \quad \boldsymbol{D}\tau(\boldsymbol{x}^*) = \left.\frac{\partial \tau(\boldsymbol{x})}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\boldsymbol{x}^*} = -\frac{\boldsymbol{n}(\boldsymbol{x}^*)^T \boldsymbol{\Phi}_\tau(\boldsymbol{x}^*)}{\boldsymbol{n}(\boldsymbol{x}^*)^T \boldsymbol{f}(\boldsymbol{x}^*)}.$$

The vector field $\boldsymbol{f}(\boldsymbol{x}^*)$ is eigenvector for (4.5) with eigenvalue 0. This is seen using (4.2) and (4.5) to evaluate

$$\left.\frac{\partial \boldsymbol{P}(\boldsymbol{x})}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\boldsymbol{x}^*} \boldsymbol{f}(\boldsymbol{x}^*) = \boldsymbol{0}.$$

Chart for the Cross Section Σ. The above notation is made more precise if it is considered that the Poincaré map $\boldsymbol{P}$ maps points on the cross-section Σ to the cross-section Σ and is therefore a mapping that acts on a lower-dimensional manifold. For stability investigation, the derivative of $\boldsymbol{P}$ is restricted to the tangent space $T\Sigma$ of the manifold Σ

and denoted as $\boldsymbol{DP}$. In order to derive an expression for $\boldsymbol{DP}$, local coordinates $\boldsymbol{z}$ are introduced for Σ, and a chart $\boldsymbol{\psi}$ is used that maps an element $\boldsymbol{x} \in \Sigma \subset \mathbb{R}^n$ to the corresponding local coordinates $\boldsymbol{z} \in \mathbb{R}^{n-1}$ of the manifold Σ

$$\boldsymbol{z} = \boldsymbol{\psi}(\boldsymbol{x}). \tag{4.6}$$

In new coordinates $\boldsymbol{z}$ the iterative Poincaré map is denoted by

$$\boldsymbol{z}_{k+1} = \boldsymbol{\psi} \circ \boldsymbol{P} \circ \boldsymbol{\psi}^{-1}(\boldsymbol{z}_k).$$

Here the operator $\circ$ is used for the composition of mappings $\boldsymbol{\psi} \circ \boldsymbol{P} \circ \boldsymbol{\psi}^{-1}(\boldsymbol{z}_k) = \boldsymbol{\psi}\left(\boldsymbol{P}\left(\boldsymbol{\psi}^{-1}\left(\boldsymbol{z}_k\right)\right)\right)$. The derivative can be determined by application of the chain rule for differentiation:

$$\boldsymbol{DP}(\boldsymbol{z}_k) = \boldsymbol{D\psi}(\boldsymbol{x}_{k+1}) \left.\frac{\partial \boldsymbol{P}(\boldsymbol{x})}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\boldsymbol{x}_k} \boldsymbol{D\psi}^{-1}\left(\boldsymbol{z}_k\right)$$

Using this local coordinate representation, the eigenvalue 0 with eigenvector $\boldsymbol{f}(\boldsymbol{x}^*)$ disappears in the spectrum of $\boldsymbol{DP}(\boldsymbol{z}^*)$. The derivative $\boldsymbol{DP}(\boldsymbol{z}^*)$ in local coordinates has only $n-1$ eigenvalues.

Trajectory Sensitivities. The derivative of the Poincaré map $\boldsymbol{DP}$ is closely related to trajectory sensitivities $\boldsymbol{\Phi}_t(\boldsymbol{x})$ as it can be seen in (4.5). The eigenvalues of the linearized Poincaré map $\boldsymbol{DP}$ are a subset of the eigenvalues of the corresponding trajectory sensitivity $\boldsymbol{\Phi}_T(\boldsymbol{x}^*)$. The trajectory sensitivity $\boldsymbol{\Phi}_T(\boldsymbol{x}^*)$, or monodromy matrix, has an additional eigenvalue which is 1 with eigenvector $\boldsymbol{f}(\boldsymbol{x}^*)$, compare (4.2). The eigenvalues of the trajectory sensitivity are often termed characteristic multipliers or Floquet multipliers.

In the following section, after introduction of the necessary definitions, the Poincaré map method is generalized to hybrid systems.

4.2.2 Stability of Periodic Solutions of Hybrid Dynamical Systems

Preliminary Definitions. For the following considerations a version of the hybrid state model (2.4) is considered that neglects external inputs and outputs. In addition, the functions $\boldsymbol{f}$, $\boldsymbol{\varphi}_i$, and $\boldsymbol{s}_i$, $i \in I$, are assumed to have no direct time dependence.

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{f}(\boldsymbol{x}, x_d) && \text{if } s_i(\boldsymbol{x}, x_d) \neq 0 \text{ for all } i \in I \\ \boldsymbol{\zeta}^+ &= \boldsymbol{\varphi}_j(\boldsymbol{x}^-, x_d^-) && \text{if } s_j(\boldsymbol{x}, x_d) = 0 \text{ for } j \in I \end{aligned} \tag{4.7}$$

A solution of (4.7) is denoted by the time trajectory of the hybrid state vector $\boldsymbol{\zeta}(t)$. Sometimes, it is useful to consider the hybrid flow $\boldsymbol{\phi}_t^H(\boldsymbol{x}_0)$, in which every initial condition $\boldsymbol{x}_0$ is mapped to its trajectory $\boldsymbol{x}(t)$ like in the ODE case. For a hybrid flow, discontinuities are allowed in $\boldsymbol{x}(t)$. The discrete state is neglected in this notation. Alternatively, the corresponding switching sequence SS can be used:

$$SS = \boldsymbol{x}_0;\ (i_0, t_0),\ (i_1, t_1),\ (i_2, t_2),\ (i_3, t_3), \ldots$$

That means, the k-th discrete state is $x_d = i_k$ for $t_k \leq t < t_{k+1}$, and the continuous state $\boldsymbol{x}$ evolves according to $\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, x_d = i_k)$, where the initial value is $\boldsymbol{x}_k^+ = \boldsymbol{x}(t_k^+)$. The transition surface $s_{j_{k+1}}(\boldsymbol{x}_{k+1}^-, i_k) = 0$ is met for $t = t_{k+1}$ allowing for the jump map $\boldsymbol{\zeta}^+ = \boldsymbol{\varphi}_{j_{k+1}}(\boldsymbol{x}_{k+1}^-, i_k)$ to act, where the continuous fraction is $\boldsymbol{x}^+ = \boldsymbol{g}_{j_{k+1}}(\boldsymbol{x}_{k+1}^-, i_k)$ and the abbreviation $\boldsymbol{x}_{k+1}^- = \boldsymbol{x}(t_{k+1}^-)$ is used. See Fig. 4.2 for illustration.

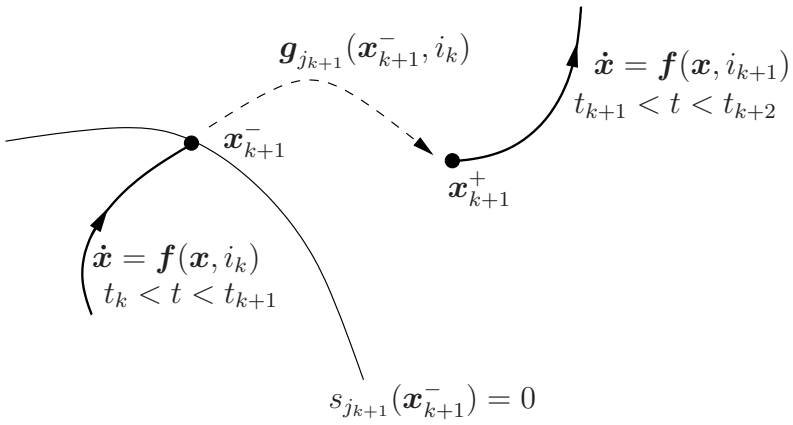

Figure 4.2: Hybrid trajectory.

To define a periodic solution for a hybrid system, the continuous and the discrete state trajectory have to be considered:

Definition 4.2.5 (Hybrid Periodic Solution, Hybrid Periodic Orbit) *A solution $\boldsymbol{\zeta}$ of (4.7) is a hybrid periodic solution with period length $T > 0$ if*

$$\boldsymbol{\zeta}(t + T) = \boldsymbol{\zeta}(t)$$

holds for all times $t \in \mathbb{R}$.

A periodic solution $\boldsymbol{\zeta}(t)$ has a corresponding periodic switching sequence, where $t_{k+N_d} = t_k + T$ and $i_{k+N_d} = i_k$.

$$SS = \boldsymbol{x}^*;\ (i_0, t_0), \ldots, (i_{N_d-1}, t_{N_d-1}),\ (i_0, t_0 + T),\ (i_1, t_1 + T), \ldots$$

The orbit passes N_d, not necessarily distinct, discrete domains. The initial value is $\boldsymbol{x}_0 = \boldsymbol{x}^*$.

A periodic orbit is an invariant set for the hybrid dynamics. Stability in the sense of Lyapunov of a hybrid periodic solution can be defined in analogy to stability for solutions of ordinary differential equations. Therefore a metric on the hybrid state space is introduced. One possible valid metric is

$$\operatorname{dist}(\boldsymbol{\zeta}_1, \boldsymbol{\zeta}_2) = \|\boldsymbol{x}_1 - \boldsymbol{x}_2\| + \delta_{x_{d,1}, x_{d,2}},$$

where $\boldsymbol{\zeta}_1 = (\boldsymbol{x}_1^T, x_{d,1})^T$ and $\boldsymbol{\zeta}_2 = (\boldsymbol{x}_2^T, x_{d,2})^T$ are points in the hybrid state space.

There are few general definitions available for stability of invariant sets of hybrid systems. The definition from Simić *et al.* [114] is based on a glued phasespace, the so called hybrifold. In most papers that deal with stability of hybrid periodic solutions, a definition is implicitly assumed by statement of stability results. In many cases, stability of the hybrid orbit is identified with stability of the underlying discrete mapping.

Hybrid Poincaré Map. Let γ be a hybrid periodic solution, denoted by

$$SS_\gamma = \boldsymbol{x}^*;\ (i_0, t_0), \ldots, (i_{N_d-1}, t_{N_d-1}),\ (i_0, t_0 + T),\ (i_1, t_1 + T), \ldots$$

The corresponding vector fields are abbreviated

$$\boldsymbol{f}_0(\boldsymbol{x}) = \boldsymbol{f}(\boldsymbol{x}, i_0),\ \boldsymbol{f}_1(\boldsymbol{x}) = \boldsymbol{f}(\boldsymbol{x}, i_1), \ldots, \boldsymbol{f}_{N_d-1}(\boldsymbol{x}) = \boldsymbol{f}(\boldsymbol{x}, i_{N_d-1}),$$

the series of crossed transition surfaces is indexed

$$s_1(\boldsymbol{x}) = s_{j_1}(\boldsymbol{x}, i_0),\ s_2(\boldsymbol{x}) = s_{j_2}(\boldsymbol{x}, i_1), \ldots, s_{N_d}(\boldsymbol{x}) = s_{j_{N_d}}(\boldsymbol{x}, i_{N_d-1}),$$

and the consecution of acting jump maps is

$$\boldsymbol{g}_1(\boldsymbol{x}) = \boldsymbol{g}_{j_1}(\boldsymbol{x}, i_0),\ \boldsymbol{g}_2(\boldsymbol{x}) = \boldsymbol{g}_{j_2}(\boldsymbol{x}, i_1), \ldots, \boldsymbol{g}_{N_d}(\boldsymbol{x}) = \boldsymbol{g}_{j_{N_d}}(\boldsymbol{x}, i_{N_d-1}).$$

At first, a cross section Σ that is transversal to the flow of the hybrid dynamics is chosen. The hybrid Poincaré map is defined to map initial values on the cross section Σ to the next occurrence of a crossing of Σ. The construction of the Poincaré map is as follows:

Assume that Σ is chosen for the orbit in the discrete state $x_d = i_0$. Then a mapping $\boldsymbol{P}_0(\boldsymbol{x}) = \boldsymbol{\phi}^{\boldsymbol{f}_0}_{\tau_0(\boldsymbol{x})}(\boldsymbol{x})$ can be defined that maps points on the Poincaré cross section Σ to the crossing of the first transition surface $S_1 : s_1(\boldsymbol{x}) = 0$ at time $\tau_0(\boldsymbol{x})$. The argument for existence and differentiability of $\boldsymbol{P}_0$ is analog to the continuous case. Then the corresponding jump map $\boldsymbol{g}_1(\boldsymbol{x})$ maps from the transition surface S_1 to the image of S_1, which is denoted Σ_1. Again, differentiability of $\boldsymbol{g}_1(\boldsymbol{x})$ has to be provided. Iteratively, flow maps $\boldsymbol{P}_{i-1}$ and jump maps $\boldsymbol{g}_i$ are repeated with appropriate first-return times τ_i until the orbit is closed and the state trajectory reaches Σ again. For illustration see Fig. 4.3.

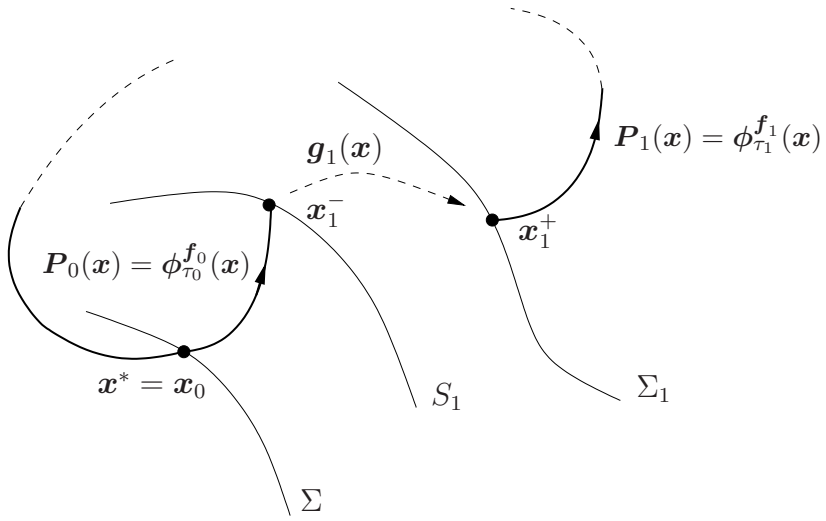

Figure 4.3: Hybrid periodic trajectory.

With maps defined as

$$\boldsymbol{P}_0 : \Sigma \to S_1,\ \boldsymbol{P}_1 : \Sigma_1 \to S_2,\ \ldots,\ \boldsymbol{P}_{N_d-1} : \Sigma_{N_d-1} \to S_{N_d},\ \boldsymbol{P}_{N_d} : \Sigma_{N_d} \to \Sigma,$$

$$\boldsymbol{g}_1 : S_1 \to \Sigma_1,\ \boldsymbol{g}_2 : S_2 \to \Sigma_2,\ \ldots,\ \boldsymbol{g}_{N_d} : S_{N_d} \to \Sigma_{N_d}$$

the Poincaré map

$$\boldsymbol{P} : \Sigma \to \Sigma$$

is constructed as

$$\boldsymbol{P}(\boldsymbol{x}) = \boldsymbol{P}_{N_d} \circ \boldsymbol{g}_{N_d} \circ \boldsymbol{P}_{N_d-1} \circ \ldots \circ \boldsymbol{g}_2 \circ \boldsymbol{P}_1 \circ \boldsymbol{g}_1 \circ \boldsymbol{P}_0(\boldsymbol{x}). \tag{4.8}$$

For practical reasons, the cross section Σ is chosen in correspondence with an event, e.g. $\Sigma = \Sigma_{N_d}$. Then $\boldsymbol{P}_{N_d}$ vanishes in (4.8) because yet $\boldsymbol{g}_{N_d}$ maps into Σ.

An expression for the derivative of $\boldsymbol{P}$ is obtained by application of the chain rule of differentiation. For the special case $N_d = 1$ with $\boldsymbol{P}(\boldsymbol{x}) = \boldsymbol{g}_1 \circ \boldsymbol{P}_0(\boldsymbol{x})$, it results:

$$\left.\frac{\partial \boldsymbol{P}(\boldsymbol{x})}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\boldsymbol{x}^*} = \boldsymbol{D}\boldsymbol{g}_1(\boldsymbol{x}_1^-)\left[\boldsymbol{\Phi}_{\tau_0}^{\boldsymbol{f}_0}(\boldsymbol{x}^*) + \boldsymbol{f}_0(\boldsymbol{x}_1^-)\boldsymbol{D}\tau_0(\boldsymbol{x}^*)\right] \tag{4.9}$$

with

$$\boldsymbol{D}\tau_0(\boldsymbol{x}^*) = -\frac{\boldsymbol{D}\boldsymbol{s}_1(\boldsymbol{x}_1^-)\,\boldsymbol{\Phi}_{\tau_0}^{\boldsymbol{f}_0}(\boldsymbol{x}^*)}{\boldsymbol{D}\boldsymbol{s}_1(\boldsymbol{x}_1^-)\,\boldsymbol{f}_0(\boldsymbol{x}_1^-)}$$

Again, an expression for $\boldsymbol{D}\boldsymbol{P}(\boldsymbol{x}^*)$ can be obtained by introducing local coordinates using a chart function $\boldsymbol{\psi}$ from (4.6) to restrict the derivative to the tangent space $T\Sigma$.

Hybrid Trajectory Sensitivities. Also for hybrid systems, trajectory sensitivity analysis is possible to investigate stability properties of periodic solution, and the result is consistent with the results presented in Sec. 4.2.1. Hiskens *et al.* [62] present a derivation for hybrid systems with differential algebraic aspect. Here a version is given that is reduced to hybrid systems of the form (4.7). Again the special case $N_d = 1$ is considered. Then only the vector fields $\boldsymbol{f}_0$ and $\boldsymbol{f}_1$, the transition surface s_1, and the jump map $\boldsymbol{g}_1$ have to be taken into account. The hybrid flow is denoted by $\boldsymbol{\phi}_t^H(\boldsymbol{x})$, and the sensitivity of the hybrid flow to perturbations in initial value is $\boldsymbol{\Phi}^H(\boldsymbol{x})$.

The following expression can be derived that relates the sensitivity after the event $\boldsymbol{\Phi}_{\tau_0^+}^H(\boldsymbol{x}^*)$ and the sensitivity before the event $\boldsymbol{\Phi}_{\tau_0^-}^H(\boldsymbol{x}^*)$

$$\boldsymbol{\Phi}_{\tau_0^+}^H(\boldsymbol{x}^*) = \boldsymbol{D}\boldsymbol{g}_1(\boldsymbol{x}_1^-)\boldsymbol{\Phi}_{\tau_0^-}^H(\boldsymbol{x}^*) - \left(\boldsymbol{f}_1(\boldsymbol{x}_1^+) - \boldsymbol{D}\boldsymbol{g}_1(\boldsymbol{x}_1^-)\boldsymbol{f}_0(\boldsymbol{x}_1^-)\right)\boldsymbol{D}\tau(\boldsymbol{x}^*). \tag{4.10}$$

Here, the abbreviations

$$\boldsymbol{\Phi}_{\tau_0^+}^H(\boldsymbol{x}^*) = \left.\frac{\partial \boldsymbol{\phi}_t^H(\boldsymbol{x})}{\partial \boldsymbol{x}}\right|_{t=\tau_0^+,\,\boldsymbol{x}=\boldsymbol{x}^*} \quad \text{and} \quad \boldsymbol{\Phi}_{\tau_0^-}^H(\boldsymbol{x}^*) = \left.\frac{\partial \boldsymbol{\phi}_t^{\boldsymbol{f}_0}(\boldsymbol{x})}{\partial \boldsymbol{x}}\right|_{t=\tau_0^-,\,\boldsymbol{x}=\boldsymbol{x}^*}$$

are used. The time τ_0 is the time of transition surface crossing, and τ_0^+ is the time immediately after the jump map was executed.

The derivation of (4.10) is as follows. The total derivatives before and after the jump map acts are connected by the derivative of the jump map $\boldsymbol{g}_1$:

$$\left.\frac{\mathrm{d}\boldsymbol{\phi}_t^H(\boldsymbol{x})}{\mathrm{d}\boldsymbol{x}}\right|_{t=\tau_0^+,\,\boldsymbol{x}=\boldsymbol{x}^*} = \boldsymbol{D}\boldsymbol{g}_1(\boldsymbol{x}_1^-)\left.\frac{\mathrm{d}\boldsymbol{\phi}_t^H(\boldsymbol{x})}{\mathrm{d}\boldsymbol{x}}\right|_{t=\tau_0^-,\,\boldsymbol{x}=\boldsymbol{x}^*} \tag{4.11}$$

The total derivative on the transition surface are calculated from partial derivatives:

$$\left.\frac{\mathrm{d}\boldsymbol{\phi}_t^H(\boldsymbol{x})}{\mathrm{d}\boldsymbol{x}}\right|_{t=\tau_0^+,\,\boldsymbol{x}=\boldsymbol{x}^*} = \boldsymbol{\Phi}_{\tau_0^+}^H(\boldsymbol{x}^*) + \boldsymbol{f}_1(\boldsymbol{x}_1^+)\boldsymbol{D}\tau(\boldsymbol{x}^*) \tag{4.12}$$

$$\left.\frac{\mathrm{d}\boldsymbol{\phi}_t^H(\boldsymbol{x})}{\mathrm{d}\boldsymbol{x}}\right|_{t=\tau_0^-,\,\boldsymbol{x}=\boldsymbol{x}^*} = \boldsymbol{\Phi}_{\tau_0^-}^H(\boldsymbol{x}^*) + \boldsymbol{f}_0(\boldsymbol{x}_1^-)\boldsymbol{D}\tau(\boldsymbol{x}^*) \tag{4.13}$$

Combination of (4.11), (4.12), and (4.13) yields the result (4.10).

Again, the eigenvalues of the trajectory sensitivity matrix indicate asymptotic stability for a closed hybrid orbit if $n-1$ eigenvalues are in the unit circle. In contrast to $\partial\boldsymbol{P}/\partial\boldsymbol{x}$ from (4.9) where the n-th eigenvalue is 0, here the n-th eigenvalue is 1 in consistency with the ordinary differential equation case. A transformation to coordinates of the manifold Σ again removes the n-th dimension and the $n-1$ eigenvalues remain that provide information on stability of the periodic orbit.

Numerical Evaluation. In the following the evaluation of the Poincaré map is done numerically. Therefore the linearized Poincaré map $\boldsymbol{DP}$ is approximated numerically by application of perturbations in independent directions of the cross section. Central differences approximate the derivative along the direction $\boldsymbol{r}$ by

$$\frac{\boldsymbol{P}(\boldsymbol{x}^* + \varepsilon\boldsymbol{r}) - \boldsymbol{P}(\boldsymbol{x}^* - \varepsilon\boldsymbol{r})}{2\varepsilon}. \tag{4.14}$$

Here, ε is a small positive scalar value, and $\boldsymbol{r}$ are directions in the tangent space of Σ.

Discussion. Although the derivative of the hybrid Poincaré map $\partial\boldsymbol{P}/\partial\boldsymbol{x} = \boldsymbol{D}\boldsymbol{g}_1\,\partial\boldsymbol{P}_0/\partial\boldsymbol{x}$ is a composition of the derivatives of the continuous Poincaré map $\partial\boldsymbol{P}_0/\partial\boldsymbol{x}$ and the jump map $\boldsymbol{D}\boldsymbol{g}_1$, compare (4.9), no conclusions can be drawn for the eigenvalues of $\partial\boldsymbol{P}/\partial\boldsymbol{x}$ from the eigenvalues of $\boldsymbol{D}\boldsymbol{g}_1$ and $\partial\boldsymbol{P}_1/\partial\boldsymbol{x}$ alone. Without loss of generality $N_d = 1$ is assumed again. In general,

$$\lambda_{\max}(\partial\boldsymbol{P}/\partial\boldsymbol{x}) \not\leq \lambda_{\max}(\boldsymbol{D}\boldsymbol{g}_1)\,\lambda_{\max}(\partial\boldsymbol{P}_0/\partial\boldsymbol{x}),$$

where $\lambda_{\max}(\boldsymbol{A}) = \max|\lambda_i(\boldsymbol{A})|$. The inequality, however, does hold for singular values σ of the maps:

$$\sigma_{\max}(\partial\boldsymbol{P}/\partial\boldsymbol{x}) \leq \sigma_{\max}(\boldsymbol{D}\boldsymbol{g}_1)\,\sigma_{\max}(\partial\boldsymbol{P}_0/\partial\boldsymbol{x})$$

The singular values are only equal to the eigenvalues if the matrices have orthonormal bases of eigenvectors. Thus, using singular values for stability analysis has severe drawbacks: Since $\lambda_{\max} \leq \sigma_{\max}$, singular-value-based stability analysis yields only necessary stability conditions and some stable solutions are not classified as stable. A second problem is seen in the valid inequality approximation

$$\sigma_{\max}(\boldsymbol{A}\boldsymbol{B}) \leq \sigma_{\max}(\boldsymbol{A})\,\sigma_{\max}(\boldsymbol{B}).$$

This inequality is not strict for most realization of matrices $\boldsymbol{A}$ and $\boldsymbol{B}$.

The difference of singular value-based stability analysis and eigenvalue-based stability analysis is illustrated in the following: A candidate discrete-time Lyapunov function for stability analysis of the periodic orbit is

$$V(\boldsymbol{x}_k) = (\boldsymbol{x}_k - \boldsymbol{x}^*)^T(\boldsymbol{x}_k - \boldsymbol{x}^*),$$

in which $\boldsymbol{x}_k$ denotes the state at the k-th time of crossing Σ and $\boldsymbol{x}^*$ is the intersection point between the periodic orbit and the cross section Σ. This discrete-time Lyapunov function is only evaluated on the cross section Σ. If the discrete-time stability condition $V(\boldsymbol{x}_{k+1}) \leq V(\boldsymbol{x}_k)$ is used with $\boldsymbol{x}_{k+1} = \boldsymbol{P}(\boldsymbol{x}_k)$, it follows that $\|\boldsymbol{P}(\boldsymbol{x}_k) - \boldsymbol{x}^*\| \leq \|\boldsymbol{x}_k - \boldsymbol{x}^*\|$ has to hold. This is only true if the operator norm $\|\boldsymbol{P}\|$ of $\boldsymbol{P}(\boldsymbol{x})$ is smaller than one. For a linear map $\boldsymbol{P}(\boldsymbol{x})$ one operator norm is equal to the largest singular value. Other operator norms can be equivalently used, but for all of them $\lambda_{\max} \leq \|\boldsymbol{P}\|$ holds. Obviously, the inequality $V(\boldsymbol{x}_{k+1}) \leq V(\boldsymbol{x}_k)$ with $V(\boldsymbol{x}_k) = (\boldsymbol{x}_k - \boldsymbol{x}^*)^T(\boldsymbol{x}_k - \boldsymbol{x}^*)$ is too restrictive to cover all solutions that are stable. Lyapunov stable solutions are allowed to violate the inequality, to allow e.g. for overshoot. Modification of the Lyapunov function

$$V(\boldsymbol{x}_k) = (\boldsymbol{x}_k - \boldsymbol{x}^*)^T\boldsymbol{W}(\boldsymbol{x}_k - \boldsymbol{x}^*).$$

may yield better results, but only for linear systems there are analytical methods to determine an appropriate choice of the weighting matrix $\boldsymbol{W}$. Rubensson *et al.* [108] use that kind of Lyapunov functions for stability analysis of periodic cycles of linear hybrid systems. In the approach of Simić *et al.* [114] operator norms are used for stability analysis resulting in above discussed results concerning necessity and sufficiency. See [50] for basic underlying results on eigenvalues and singular values.

4.3 Application for Legged Locomotion

It was demonstrated in Chap. 3 that periodic solutions of the hybrid system can be achieved by control laws for desired trajectories of the actuated joints with appropriate initial values and time. The initial values are solutions of a boundary value problem that was solved numerically. The stability of periodic trajectories for legged robots can be analyzed using hybrid Poincaré maps. A stability investigation is presented for trajectories of the compass gait robot in Sec. 4.3.1, the monoped robot in Sec. 4.3.2, and the gymnast robot in Sec. 4.3.3.

4.3.1 Compass Gait Robot

The trajectories for the compass gait robot are obtained by numerical solution of a boundary value problem according to Sec. 3.3, and their stability will be determined. For consistency a short summary repeats the model and the trajectory planning.

Model and Trajectories Revisited. Trajectory planning for periodic walking made use of the symmetric construction of the robot. It was shown how one step is planned, where

boundary conditions assure that the following step has the same initial conditions after a symmetry transformation of the state. The symmetry property is also useful for stability investigation: Only a single step has to be investigated. Figure 4.4 displays the subsystem of the hybrid system model of the compass gait robot that was basis for trajectory planning and will now be basis for stability analysis. Compare also Fig. 3.2.

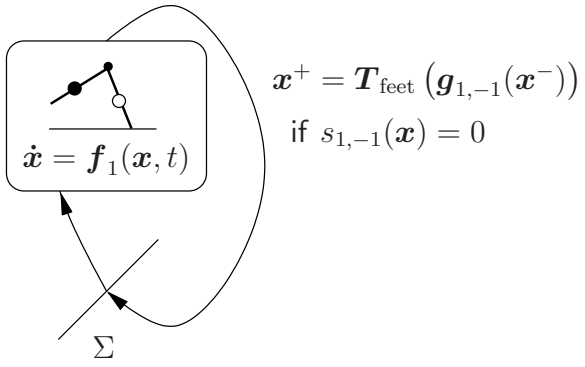

Figure 4.4: Subgraph of hybrid transition graph for the compass gait robot. Due to symmetry only one step is considered in trajectory planning and stability analysis. The transformation $\boldsymbol{T}_{\text{feet}}$ accounts for the coordinate transformation that maps coordinates of the second step on coordinates of the first step.

For trajectory planning, a desired periodic trajectory $\beta^d(t)$ for the actuated joint was defined with period length T, and a control law was chosen that yields tracking. Substituting the control into the dynamical equations for contact with the reference foot ($x_d = 1$) results in

$$\dot{\boldsymbol{x}} = \boldsymbol{f}_1(\boldsymbol{x}, t)\,.$$

The vector field $\boldsymbol{f}_1(\boldsymbol{x}, t)$ depends explicitly on time because of the time dependence of the desired trajectory $\beta^d(t)$. The jump map to consider collision modeling $\boldsymbol{g}_{1,-1}$ and the symmetry transformation $\boldsymbol{T}_{\text{feet}}$ are combined in

$$\begin{pmatrix} \boldsymbol{x}^+ \\ x_d^+ \end{pmatrix} = \begin{pmatrix} \boldsymbol{T}_{\text{feet}}\left(\boldsymbol{g}_{1,-1}(\boldsymbol{x}^-)\right) \\ x_d^- \end{pmatrix}.$$

This jump map becomes active when the swing foot touches ground:

$$s_{1,-1}(\boldsymbol{x}) = \alpha + \frac{\beta}{2} - \pi = 0. \tag{4.15}$$

For an appropriate initial value $\boldsymbol{x}_0 = \left(\pi - \frac{\beta_0}{2},\, \beta_0,\, \dot{\alpha}_0,\, \dot{\beta}_0\right)^T$ at appropriate initial time t_0, periodic behavior is achieved and can be expressed as

$$\boldsymbol{T}_{\text{feet}}\left(\boldsymbol{g}_{1,-1}\left(\boldsymbol{\phi}^{\boldsymbol{f}_1}_{\frac{T}{2}}(\boldsymbol{x}_0, t_0)\right)\right) = \boldsymbol{x}_0.$$

In summary the hybrid system trajectories rely on a hybrid system with only one discrete state $x_d = 1$. The continuous dynamics is interrupted by a jump map that acts at transition time.

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{f}_1(\boldsymbol{x}, t) && \text{if } s_{1,-1}(\boldsymbol{x}) \neq 0 \\ \begin{pmatrix} \boldsymbol{x}^+ \\ x_d^+ \end{pmatrix} &= \begin{pmatrix} \boldsymbol{T}_{\text{feet}}\left(\boldsymbol{g}_{1,-1}\left(\boldsymbol{x}^-\right)\right) \\ x_d^- \end{pmatrix} && \text{if } s_{1,-1}(\boldsymbol{x}) = 0 \end{aligned}$$

Extension for Nonautonomy. The desired trajectories are functions in time, so the ordinary differential equation system is non-autonomous. To apply Poincaré analysis as introduced above, time has to be considered as additional system state. Therefore a new periodic state variable is introduced:

$$\theta = \frac{2\pi}{T/2} t \mod 2\pi \tag{4.16}$$

For the following considerations the state vector $\tilde{\boldsymbol{x}}$ is used that includes the new auxiliary state θ:

$$\tilde{\boldsymbol{x}} = \begin{pmatrix} \boldsymbol{x} \\ \theta \end{pmatrix}$$

The jump map does not change θ when the extended state $\tilde{\boldsymbol{x}}$ crosses the transition surface:

$$\begin{pmatrix} \dot{\boldsymbol{x}} \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} \boldsymbol{f}_1(\boldsymbol{x}, t) \\ 1 \end{pmatrix} \qquad \text{if } s_{1,-1}(\boldsymbol{x}) \neq 0$$

$$\begin{pmatrix} \boldsymbol{x}^+ \\ \theta^+ \\ x_d^+ \end{pmatrix} = \begin{pmatrix} \boldsymbol{T}_{\text{feet}} \left(\boldsymbol{g}_{1,-1}(\boldsymbol{x}^-) \right) \\ \theta^- \\ x_d^- \end{pmatrix} \qquad \text{if } s_{1,-1}(\boldsymbol{x}) = 0$$

Cross Section. For the compass gait robot, the cross section is chosen when stance foot and swing foot exchange roles denoted by the transition condition

$$\Sigma : s_{1,-1}(\tilde{\boldsymbol{x}}) = \alpha + \frac{\beta}{2} - \pi = 0.$$

The surface Σ is assumed to separate the jump map from the continuous dynamics, where the jump map maps into Σ and the continuous dynamics has initial values on Σ, compare Fig. 4.4.

Poincaré Map. The Poincaré map $\tilde{\boldsymbol{x}}_{k+1} = \boldsymbol{P}(\tilde{\boldsymbol{x}}_k)$ is composed from the integration of the vector field and the execution of the jump map.

$$\boldsymbol{P}(\tilde{\boldsymbol{x}}) = \boldsymbol{T}_{\text{feet}} \left(\boldsymbol{g}_{1,-1} \left(\boldsymbol{\phi}_\tau(\tilde{\boldsymbol{x}}) \right) \right)$$

Here $\tilde{\boldsymbol{x}} = (\alpha,\, \beta,\, \dot{\alpha},\, \dot{\beta},\, \theta)^T$, and $\tau(\boldsymbol{x})$ is the time when the transitions surface Σ is reached after one cycle. The discrete dynamics that maps from cross section to cross section has a lower dimension than the hybrid dynamics: On the transition surface α is explicitly determined by the choice of β through the algebraic relation $s_{1,-1}(\tilde{\boldsymbol{x}}) = 0$, see (4.15).

Numerical Evaluation of the Poincaré Map. An analytic expression for the Poincaré map $\boldsymbol{P}(\boldsymbol{x})$ is not available due to the nonlinearity of the hybrid system. Thus the Poincaré map $\boldsymbol{P}(\boldsymbol{x})$ and its linearization $\boldsymbol{DP}(\boldsymbol{x}^*)$ have to be determined numerically.

The Poincaré map maps $\tilde{\boldsymbol{x}}_k \in \Sigma \subset \mathbb{R}^5$ to $\tilde{\boldsymbol{x}}_{k+1} \in \Sigma \subset \mathbb{R}^5$. Due to the constraint $s_{1,-1}(\tilde{\boldsymbol{x}}_k) = 0$, four independent directions can be chosen in Σ. To determine $\boldsymbol{DP} \in \mathbb{R}^{4\times 4}$ by an approximation with central differences, eight integrations of the hybrid dynamics are necessary. The stability of the considered trajectory is characterized by four eigenvalues $\lambda_i,\ i = 1, 2, 3, 4$, of the linearization $\boldsymbol{DP}(\tilde{\boldsymbol{x}})$.

For the numerical computation of eigenvalues the perturbation was chosen $\varepsilon = 10^{-5}$, see (4.14). The optimization procedure as well as the integration are performed with absolute error bounds of 10^{-12}. Thus the eigenvalues are precise in about seven digits. Smaller values for ε for the same absolute error bound reduce the accuracy of the eigenvalues. Precision in the eigenvalues can then only be enhanced if the integration accuracy is increased. Larger values for ε also decrease accuracy because the precision of the linearization is decreased. Thus $\varepsilon = 10^{-5}$ is a compromise between accuracy of the linearization concerning the methodical error and accuracy of the linearization concerning numerical errors.

Stability of Precalculated Trajectories. The planning algorithm determines trajectories for combinations of step angle A and step frequency ω. Figure 4.5 depicts a grid of (A, ω)-pairs, where for pairs labeled by crosses the corresponding periodic solution is not asymptotically stable. Only solutions for (A, ω)-pairs labeled by circles are asymptotically stable and feasible. The larger the circle, the smaller the maximum eigenvalue and the faster is the convergence to the periodic cycle after disturbances. The smallest maximum absolute value $\lambda = \max |\lambda_i| = 0.2606$ occurs for $(A, \omega) = (0.4\,\text{rad}, 4\,\frac{\text{rad}}{\text{s}})$.

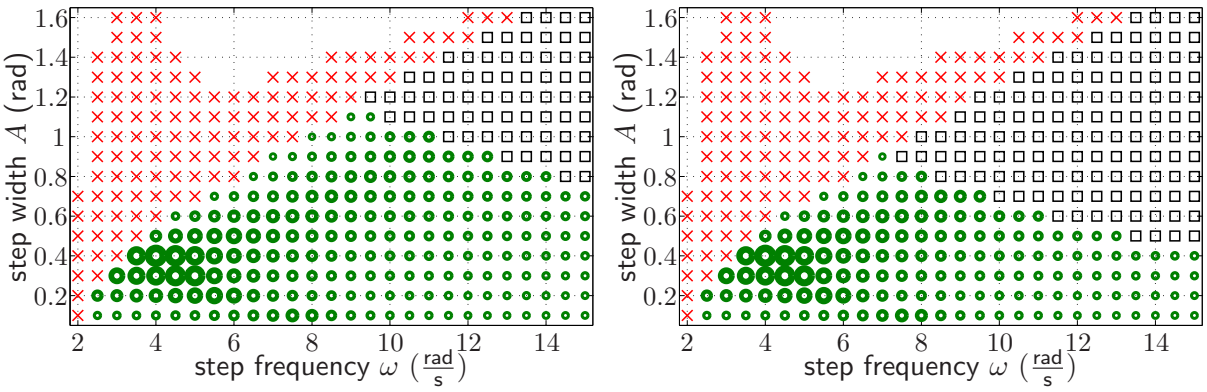

Figure 4.5: Stability and feasibility of compass gait trajectories for pairs (A, ω). Circles label asymptotically stable trajectories. Large markers correspond to a small maximum eigenvalue. Crosses label unstable trajectories. Squares label infeasible trajectories. Left: Trajectories are infeasible if the vertical contact force becomes zero. Right: Trajectories are infeasible if in addition a friction condition is violated.

Squares in Fig. 4.5 label infeasible trajectories. In the left graph, it is only checked if the ground contact force of the precalculated trajectories is positive. The set of stable trajectories is thus diminished: Fast trajectories with large step widths are not feasible. In the right graph additionally friction between ground and robot foot is considered. A robot foot sticks to the ground as long as the vertical contact force R_y and the horizontal contact force R_x fulfill $|R_x| < \nu\,|R_y|$, where ν is a friction coefficient that depends on the texture of ground and foot. Using $\nu = 0.8$ for contact between rubber and concrete, the set of feasible trajectories is further diminished.

Finally, Fig. 4.6 gives evidence that increasing the step frequency does not necessarily increase walking velocity. Therefore, to realize a desired walking velocity, several trajectories are available and a trajectory with faster disturbance compensation is to be preferred. The

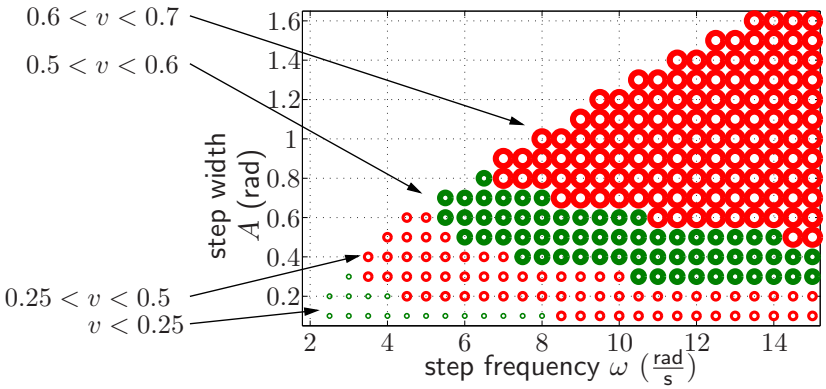

Figure 4.6: Walking velocity v (in $\frac{\text{m}}{\text{s}}$) is calculated as progression in x-direction per period. The size of the marker increases with walking velocity.

trajectory with the fastest disturbance compensation $(A, \omega) = (0.4\,\text{rad}, 4\,\frac{\text{rad}}{\text{s}})$ results in a walking velocity of $v = 0.37\,\frac{\text{m}}{\text{s}}$.

4.3.2 Monoped Robot

The periodic trajectories for the monoped robot that are basis for stability analysis in what follows are computed as solution of boundary value problems according to Sec. 3.4. A short summary of the model and of trajectories is given next.

Model and Trajectories Revisited. In Sec. 3.4 it was demonstrated that control on desired trajectories $\beta^d(t)$ with appropriate starting values results in a periodic tilting left and right of the robot foot. The starting values that yield a periodic motion are solution of a boundary value problem. Two trajectory patterns were discerned. In the first setting, tilting left and tilting right follow each other immediately. In the second setting, a stable support phase separates left and right tilting. Due to symmetry of the construction only half the motion has to be planned, the second half is mirrored. Thus, also stability is discussed for half the motion. Figure 4.7 presents the subgraphs of the transition graphs that are relevant for trajectory planning and stability analysis. Compare also Fig. 3.7 and Fig. 3.9.

In the first trajectory planning problem, trajectories are considered where the tilting to the left side ($x_d = 2$) and the tilting to the right side ($x_d = 3$) consecute directly. Due to symmetry the planning problem only comprises the left tilting phase ($x_d = 2$), where the differential equation is denoted:

$$\dot{\boldsymbol{x}} = \boldsymbol{f}_2(\boldsymbol{x}, t)$$

The time dependence of the vector field $\boldsymbol{f}_2(\boldsymbol{x}, t)$ is caused by time dependence of the control law that uses time-dependent desired trajectories. Tilting left ends when the foot makes flat ground contact:

$$s_{2,1}(\boldsymbol{x}) = \alpha = 0$$

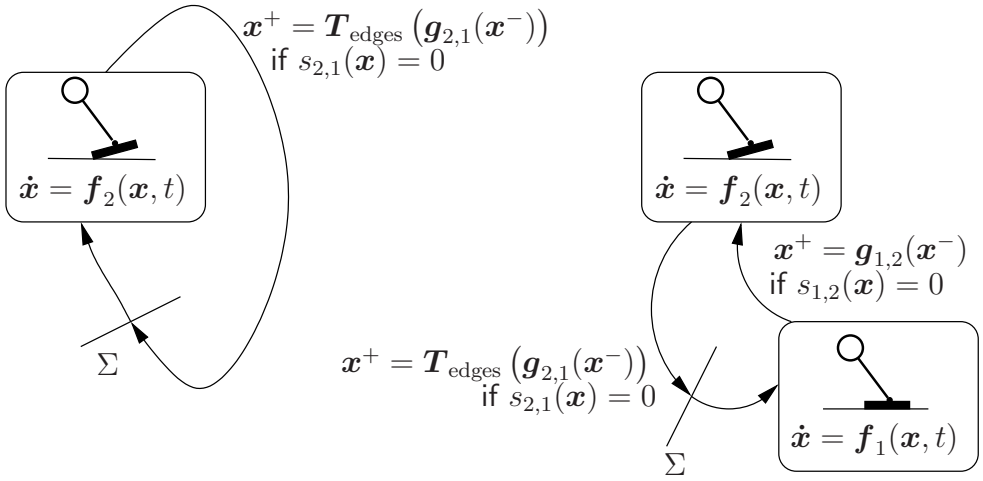

Figure 4.7: Subgraph of hybrid transition graph for the monoped robot. Left: Planning for direct consecution of left and right tilting. Right: Planning for left and right tilting separated by stable support. Due to symmetry only half of a cycle is considered in trajectory planning and stability analysis. The transformation $\boldsymbol{T}_{\text{edges}}$ accounts for the symmetry transformation.

The jump map includes the coordinate transformation $\boldsymbol{T}_{\text{edges}}$,

$$\begin{pmatrix} \boldsymbol{x}^+ \\ x_d^+ \end{pmatrix} = \begin{pmatrix} \boldsymbol{T}_{\text{edges}}\left(\boldsymbol{g}_{2,1}(\boldsymbol{x}^-)\right) \\ x_d^- \end{pmatrix}.$$

An appropriate initial value $\boldsymbol{x}_0 = \left(0,\, \beta_0,\, 0,\, \dot{\beta}_0\right)^T$ at initial time t_0 yields periodic behavior: The right tilting phase is symmetric to the left tilting phase.

$$\boldsymbol{T}_{\text{edges}}\left(\boldsymbol{g}_{2,1}\left(\boldsymbol{\phi}_{\frac{T}{2}}^{\boldsymbol{f}_2}(\boldsymbol{x}_0, t_0)\right)\right) = \boldsymbol{x}_0.$$

In the second trajectory planning approach, an additional stable support phase ($x_d = 1$) was assumed to separate left ($x_d = 2$) and right ($x_d = 3$) tilting. The differential equation is switched between

$$\dot{\boldsymbol{x}} = \boldsymbol{f}_1(\boldsymbol{x}, t) \qquad \text{and} \qquad \dot{\boldsymbol{x}} = \boldsymbol{f}_2(\boldsymbol{x}, t).$$

A transition between the stable support and tilting occurs when the ZMP leaves the foot supported area

$$s_{1,2}(\boldsymbol{x}, u) = r_{\text{zmp}}(\boldsymbol{x}, u) = 0.$$

This transition is smooth and no symmetry transformations are applied:

$$\begin{pmatrix} \boldsymbol{x}^+ \\ x_d^+ \end{pmatrix} = \begin{pmatrix} \boldsymbol{g}_{1,2}(\boldsymbol{x}^-) \\ 2 \end{pmatrix}$$

After the second transition at landing from tilted supervised by

$$s_{2,1}(\boldsymbol{x}) = \alpha = 0,$$

a discontinuous reset

$$\begin{pmatrix} \boldsymbol{x}^+ \\ x_d^+ \end{pmatrix} = \begin{pmatrix} \boldsymbol{T}_{\text{edges}} \left(\boldsymbol{g}_{2,1}(\boldsymbol{x}^-) \right) \\ 1 \end{pmatrix}$$

occurs, where $\boldsymbol{T}_{\text{edges}}$ transforms right tilting in left tilting coordinates. Periodicity in left and right tilting is summarized using $t_1 + t_2 = \frac{T}{2}$:

$$\boldsymbol{T}_{\text{edges}} \left(\boldsymbol{g}_{2,1} \left(\boldsymbol{\phi}_{t_2}^{\boldsymbol{f}_2} \left(\boldsymbol{\phi}_{t_1}^{\boldsymbol{f}_1}(\boldsymbol{x}_0, t_0), t_0 + t_1 \right) \right) \right) = \boldsymbol{x}_0.$$

Extension for Nonautonomy. Again the desired trajectories are functions of time, so the ordinary differential equation system is non-autonomous. For a Poincaré analysis as introduced above, time has to be considered as additional cyclic system state θ, like in (4.16). The state is then extended to $\tilde{\boldsymbol{x}} = (\boldsymbol{x}^T, \theta)^T$.

Cross Section. For the monoped robot in both trajectory settings, the cross section is chosen when the foot touches ground:

$$\Sigma : s_{2,1}(\tilde{\boldsymbol{x}}) = \alpha = 0$$

The choice of cross section is illustrated in Fig. 4.7.

Poincaré Map. The Poincaré map is defined to map an initial configuration on Σ to the next occurrence of landing. This is either after one left tilting phase for the first trajectory planning scenario

$$\boldsymbol{P}(\tilde{\boldsymbol{x}}) = \boldsymbol{T}_{\text{edges}} \left(\boldsymbol{g}_{2,1} \left(\boldsymbol{\phi}_{\tau}^{\boldsymbol{f}_2}(\tilde{\boldsymbol{x}}) \right) \right),$$

or after a a stable support phase and a left tilting phase for the second planning scenario:

$$\boldsymbol{P}(\tilde{\boldsymbol{x}}) = \boldsymbol{T}_{\text{edges}} \left(\boldsymbol{g}_{2,1} \left(\boldsymbol{\phi}_{\tau_2}^{\boldsymbol{f}_2} \left(\boldsymbol{\phi}_{\tau_1}^{\boldsymbol{f}_1}(\tilde{\boldsymbol{x}}) \right) \right) \right).$$

Here $\tilde{\boldsymbol{x}} = (\alpha,\, \beta,\, \dot{\alpha},\, \dot{\beta},\, \theta)^T$ and τ, τ_1, and τ_2 are the appropriate times needed to reach the next transition surface, where $\tau \approx \frac{T}{2}$ and $\tau_1 + \tau_2 \approx \frac{T}{2}$. The Poincaré map has a lower dimension than the hybrid dynamics since $\alpha = 0$ and $\dot{\alpha} = 0$ on the cross section.

Numerical Evaluation of the Poincaré Map. Again the Poincaré map and its derivative cannot be evaluated analytically. The derivative of the Poincaré map is approximated by central differences. Here $\boldsymbol{DP} \in \mathbb{R}^{3\times 3}$ because there are three independent directions in the tangent space of the cross section Σ, since for the cross-section $\alpha = 0$ and $\dot{\alpha} = 0$.

Stability of Precalculated Trajectories with one Contact Situation. For the trajectory planning problem with direct consecution of left tilting and right tilting, two types of trajectories were obtained by solution of the boundary value problem, see Sec. 3.4.1. For trajectories of the first type, the initial time t_0 is close to zero. That means tilting left and swinging left of the actuated arm are synchronous. In stability investigation, it turns out that none of those trajectories is stable. Trajectories were computed for values of the excitation amplitude A between 0.05 rad and 0.35 rad and for values of the excitation frequency ω in a range from $0.5\,\frac{\text{rad}}{\text{s}}$ to $7.0\,\frac{\text{rad}}{\text{s}}$. For most of the trajectories, instability is obvious after simulations of few cycles as the monoped turns out to fall. Often the Poincaré map cannot be evaluated numerically because the integration does not reach the cross section Σ because of the high sensitivity of the trajectory to perturbations in the initial values.

Trajectories of the second type have values of t_0 that are close to $\frac{\pi}{\omega}$. That means tilting left is in phase with swinging right of the desired motion of the actuated arm. For small values of A asymptotically stable solutions exist. For an overview of orbitally stable and orbitally unstable solutions see Fig. 4.8. Here parameter pairs (A, ω) that correspond to asymptotically stable solutions are marked by circles. Small markers indicate that the maximum eigenvalue is close to the unit circle. Crosses label solutions that are unstable.

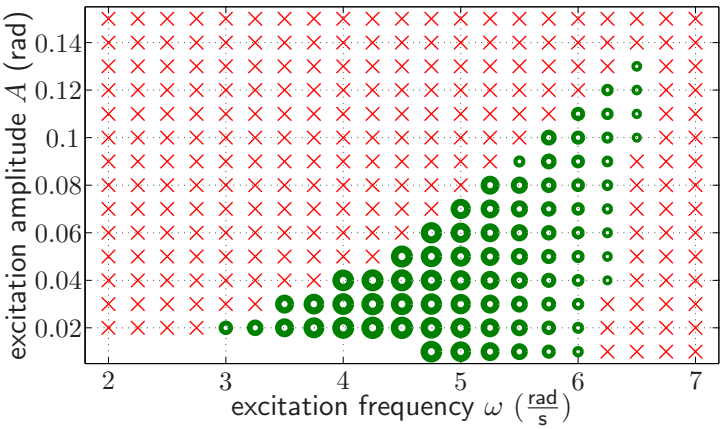

Figure 4.8: Stability of monoped trajectories of second type for the one phase problem. Circles label asymptotically stable trajectories. Huge circles correspond to small maximum eigenvalues. Crosses indicate that the maximum eigenvalue is outside the unit circle.

Figure 4.9 compares a trajectory of the first type with a trajectory of the second type to illustrate the decisive differences that could be responsible for the different stability properties. Both trajectories have excitation amplitude $A = 0.05$ rad and excitation frequency $\omega = 5.5\,\frac{\text{rad}}{\text{s}}$. In the first three graphs, the trajectories for α, $\dot{\alpha}$, and $\ddot{\alpha}$ are displayed. The trajectories for α and its derivatives do not differ much. One of the main differences is seen in the fourth plot, where β and the desired trajectory β^d is plotted. The β trajectory starts with nearly zero control error, but then the control error becomes large. This is caused by the control error in the velocity $\dot{\beta}$ that was caused by the collision and that has to be compensated for by the controller. For both trajectory types the desired trajectory for β is asymptotically reached some time before the next collision. For the type-1 trajectory the desired trajectory is still left from the upright position, so slightly negative

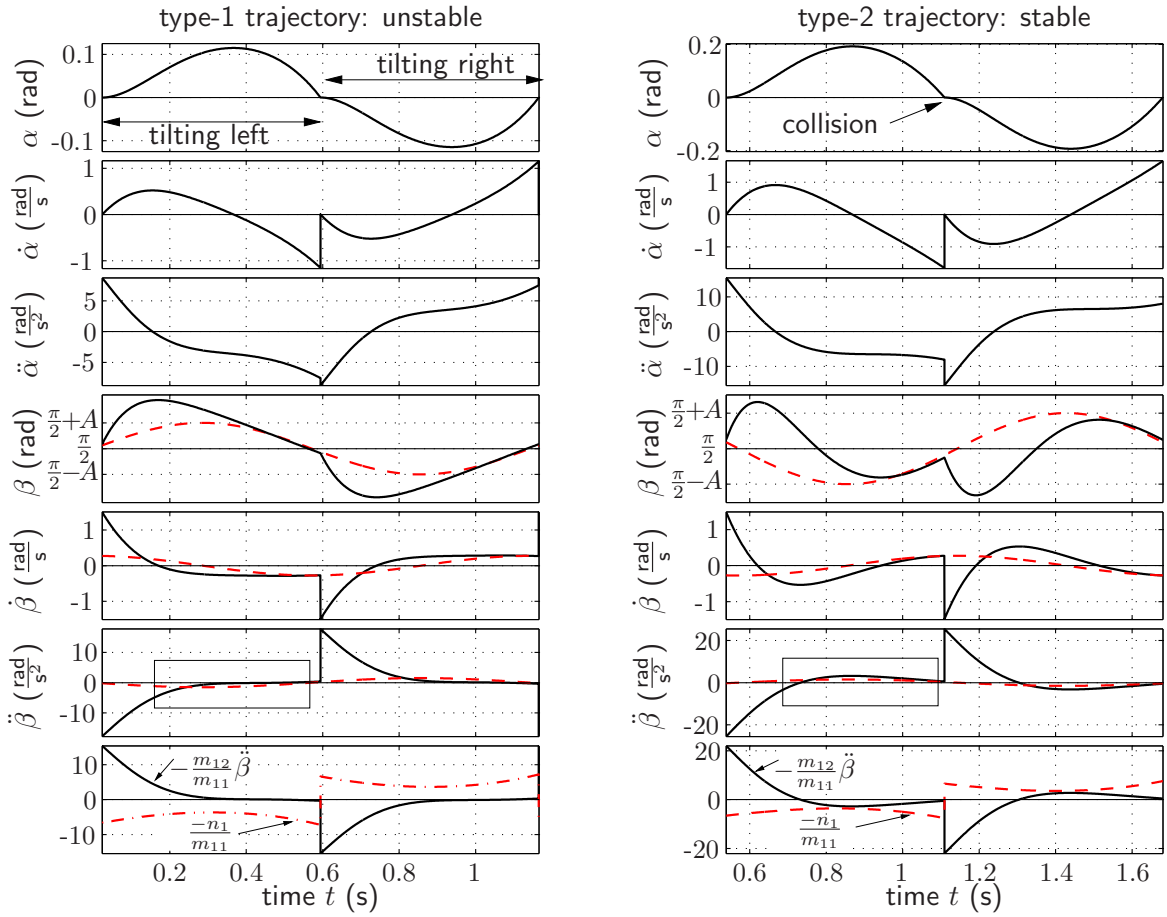

Figure 4.9: Differences in type-1 (left column) and type-2 (right column) trajectories resulting in different stability properties. Desired trajectories for the actuated joints are pictured dashed.

accelerations in β are needed to achieve tracking. For the type-2 trajectories the desired trajectory is right from the upright position of the arm, so slightly positive accelerations in β are needed for tracking. The acceleration in β influences the acceleration in α, where a negative acceleration in α is important at that time because only then landing is possible. The interaction of $\ddot{\alpha}$ and $\ddot{\beta}$ is seen in the equations of motion from (2.38)

$$\boldsymbol{M}_2 \begin{pmatrix} \ddot{\alpha} \\ \ddot{\beta} \end{pmatrix} + \boldsymbol{n}_2 = \begin{pmatrix} 0 \\ u \end{pmatrix}. \tag{4.17}$$

The first line of (4.17) can be solved for $\ddot{\alpha}$ resulting in $\ddot{\alpha} = \frac{1}{m_{11}}(-m_{12}\,\ddot{\beta} - n_1)$. It is seen that the trajectory for α is influenced by gravity, centrifugal, and coriolis effects $\frac{-n_1}{m_{11}}$ and also directly by accelerations in β via $-\frac{m_{12}}{m_{11}}\ddot{\beta}$. The last graph in Fig. 4.9 splits the acceleration $\ddot{\alpha}$ into the $\frac{-n_1}{m_{11}}$-part (dash-dotted) and the $-\frac{m_{12}}{m_{11}}\ddot{\beta}$-part (solid). For the unstable trajectory in the left column the acceleration part is in competition with the gravity/coriolis/centrifugal-part, for the stable trajectory in the right column the acceleration-part and the gravity/coriolis/centrifugal-part act in the same direction and support landing of the foot.

Stability of Precalculated Trajectories for Two Contact Situations. Now an additional stable support phase is included into the motion pattern that separates left tilting and right tilting. Of special interest is that in the stable support phase the robot is fully actuated and does thus not have an internal dynamics. The trajectories that are found by solving the boundary value problem resemble the type-1 trajectories of the problem with only one phase in the sense that tilting left and left deflection of the actuated arm are synchronous.

Figure 4.10 shows that only few trajectories exist. The set of trajectories that are found by solution of the boundary value problem is diminished since not all solutions of the boundary value problem are physically reasonable. There are trajectories where tilting time is earlier than initial time, which violates causality. Then, also trajectories where the ZMP leaves the supporting area in the stable support phase have to be taken out, though the ZMP is zero again at transition time. Furthermore, a check of non-violation of the constraint that the foot cannot penetrate ground in the tilting phase has to be done. This excludes trajectories where the tilt angle α becomes smaller then zero. Pairs of parameters (A, ω) where the boundary value problem finds an unfeasible solution are marked by squares.

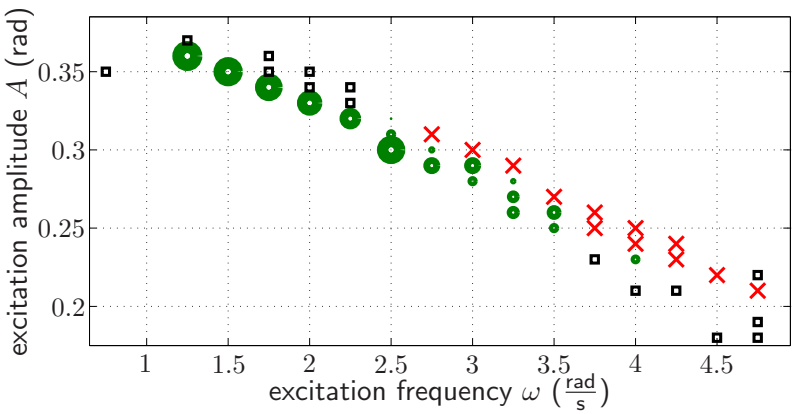

Figure 4.10: Stability of monoped trajectories for the two phase problem. Circles label asymptotically stable trajectories. A large circle corresponds to a small eigenvalue. Crosses label unstable trajectories. Squares indicate that the solution is not feasible.

Asymptotic stability is checked for the remaining valid solutions, and stable trajectories are labeled by circles at the corresponding parameter pair (A, ω). In comparison to the type-2 trajectories with only one contact phase, it turns out for this trajectory type that only trajectories that have a slow excitation frequency with large excitation amplitude are stable.

Figure 4.11 illustrates what the stable trajectories have in common and in particular how the duration of the stable support phase is related to orbital stability. From the left graph in Fig. 4.11 it cannot be concluded that a stable support phase is necessary for orbital stability. Trajectories exist with a short stable support phase that are nevertheless orbital stable. But it can be concluded that a longer stable support phase makes stability more likely. At least for the monoped robot, nearly all trajectories where the stable support phase takes more than 50 % of the period time are asymptotically stable.

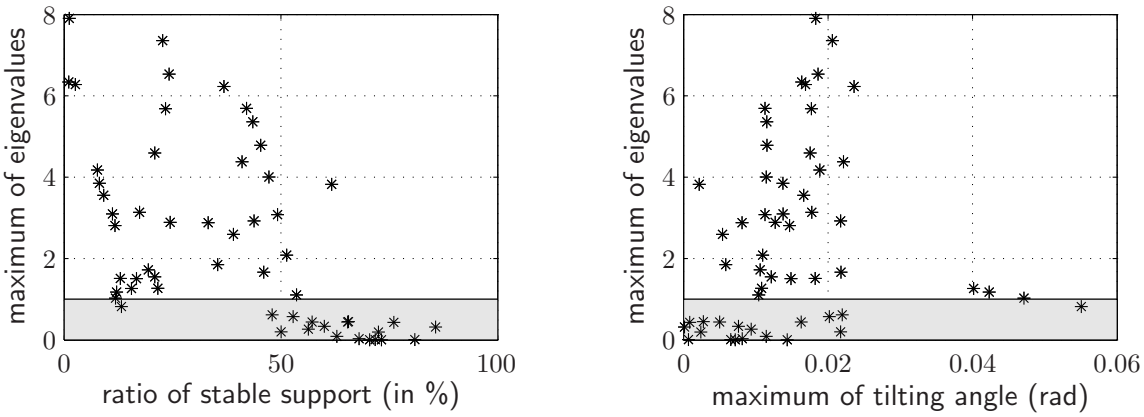

Figure 4.11: Details of stability of monoped trajectories with stable support phase. Left: Ratio of the stable support phase in percent of the complete cycle against maximum of the absolute values of the eigenvalues of the Poincaré map. Right: Maximum of tilt angle α against maximum of the absolute values of the eigenvalues of the Poincaré map.

The graph on the right hand side in Fig. 4.11 shows that orbital stability is not correlated to the size of the maximum of the tilt angle α. Trajectories with low maximum tilt angles have eigenvalues in the unit circle as well as eigenvalues outside.

4.3.3 Gymnast Robot

Walking trajectories for the gymnast robot are computed in Sec. 3.5 and will be investigated for stability in this section. A short summary of the model and the trajectories will be given in the following.

Model and Trajectories Revisited. Trajectory planning for periodic walking has made use of the symmetric construction of the robot. Thus, it was possible to plan only one step where the boundary conditions for that first step were chosen such that the following second step has the same initial conditions as the first step. The symmetry property is also used for stability investigation. Only one step has to be investigated. In Fig. 4.12 the subsystem of the hybrid system model of the gymnast robot is shown that was the basis for trajectory planning and will now be the basis for stability analysis. Compare also Fig. 3.11.

For trajectory planning, desired trajectories $\beta_1^d(t), \ldots, \beta_5^d$ were defined for the actuated joint and a control law was chosen that yields tracking. For an appropriate initial value $\boldsymbol{x}_0 = \boldsymbol{x}(t_0)$ at appropriate initial time t_0 in the stable support contact phase $x_d = 1$ periodic behavior is achieved. That means, after the stable support phase, a tilting around the toe phase, and a tilting around the heel phase the stable contact phase is reached again where the final configuration is symmetric to the initial configuration with respect to the symmetry of the two feet.

$$\boldsymbol{x}_0 = \boldsymbol{T}_{\text{edges}} \circ \boldsymbol{g}_{2,1} \circ \boldsymbol{\phi}_{t_3}^{\boldsymbol{f}_2} \circ \boldsymbol{T}_{\text{edges}} \circ \boldsymbol{T}_{\text{feet}} \circ \boldsymbol{g}_{2,-3} \circ \boldsymbol{\phi}_{t_2}^{\boldsymbol{f}_2} \circ \boldsymbol{\phi}_{t_1}^{\boldsymbol{f}_1}(\boldsymbol{x}_0)$$

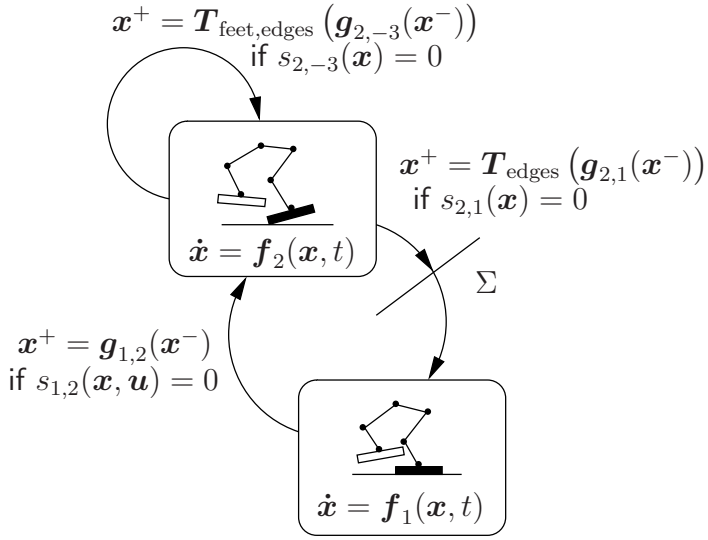

Figure 4.12: Subgraph of hybrid transition graph for gymnast robot walking.

Again, $t_1 + t_2 + t_3 = \frac{T}{2}$.

Extension for Nonautonomy. The desired trajectories are functions in time, thus the system of ordinary differential equations is non-autonomous. To apply Poincaré analysis as introduced above, time has to be considered as additional system state with dynamics $\dot{\theta} = 1$ and initial condition $t(\theta_0) = t_0$. For the following considerations the state vector $\tilde{\boldsymbol{x}} = (\boldsymbol{x}^T, \theta)^T$ is used that comprises the new state θ.

Cross Section. For the gymnast robot walking trajectories, the cross section is chosen when the heel roll motion ends and a stable support contact situation begins.

$$\Sigma : s_{2,1}(\tilde{\boldsymbol{x}}) = \alpha = 0$$

Numerical Evaluation of the Poincaré Map. The Poincaré map maps $\tilde{\boldsymbol{x}}_k \in \Sigma \subset \mathbb{R}^{13}$ to $\tilde{\boldsymbol{x}}_{k+1} \in \Sigma \subset \mathbb{R}^{13}$. The stability of the considered trajectory is characterized by eleven eigenvalues $\lambda_i,\ i = 1, ..., 11$ of the linearization $\boldsymbol{DP}$. For the numerical eigenvalue computation the perturbation was chosen $\varepsilon = 10^{-4}$. The optimization procedure as well as the integration are performed with absolute error bounds of 10^{-8}. Thus the eigenvalues are precise in about four digits.

Stability of Precalculated Trajectories. The planning algorithm determines trajectories for combinations of step angle A, knee bend angle B, and step frequency ω, see Sec. 3.5. Stability results are given for knee bend angle $B = 0.4\,\text{rad} = \text{const.}$ and varying pairs of (A, ω) in Fig. 4.13 in the left subfigure. Stable solutions exist for step frequencies ω between $1.5\,\frac{\text{rad}}{\text{s}}$ and $4\,\frac{\text{rad}}{\text{s}}$. The smallest maximum eigenvalue is obtained for $(A, \omega, B) = (0.72\,\text{rad}, 1.9\,\frac{\text{rad}}{\text{s}}, 0.4\,\text{rad})$: it is $\lambda_{\max} = \max|\lambda_i| = 0.3560 \cdot 10^{-5}$. The right subplot of Fig. 4.13 depicts the corresponding walking velocities, evaluated as advancement in x-direction per period. The particular choice of desired trajectories, see (3.8), results in

walking velocities between $0.15\,\frac{\mathrm{m}}{\mathrm{s}}$ and $0.56\,\frac{\mathrm{m}}{\mathrm{s}}$. This is slightly slower as for the compass gait robot, compare Fig. 4.6. For the gymnast robot with the chosen desired trajectory the velocity is limited because of instability of faster trajectory, whereas for the compass gait robot physical feasibility limited the velocity.

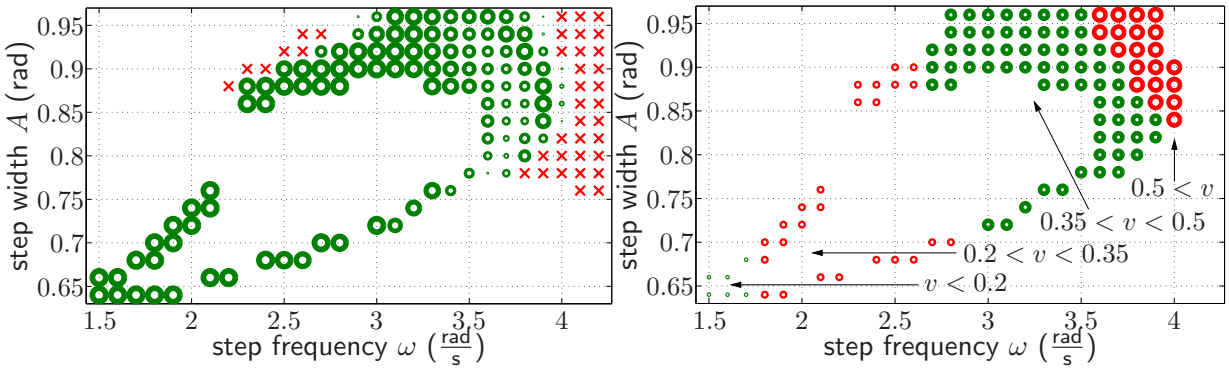

Figure 4.13: Stability and velocity v of gymnast walking trajectories for pairs (A, ω) with constant B. Left: Circles label asymptotically stable trajectories. Crosses label unstable trajectories. For pairs (A, ω) that are not labeled, either no solution was found by the numerical solver, or the solution is infeasible. Right: The size of the marker increases with the velocity v (in $\frac{\mathrm{m}}{\mathrm{s}}$) of walking.

Figure 4.14 discusses how the ratio of the stable support phase in one period is related to the eigenvalues. In the left graph, it is seen that walking with small velocity and small step width has a longer stable support phase. Here, large markers indicate a long stable support phase, small markers indicate a short stable support phase. The left graph shows the maximum eigenvalue against the ratio of stable support. It can be seen that almost all trajectories where the stable support phase takes more than 25 % are stable. In contrast, it is not necessary for stability that the stable support phase lasts long. Trajectories with stable support phases lasting between 5% and 25 % of the period can either be stable or not.

Transitions Between Preplanned Trajectories. When several trajectory primitives for different tasks are planned and saved in a trajectory database, it is important to know if and how a transition between those trajectory primitives is possible. For example if trajectories for several walking speeds are available, can walking be accelerated such that it starts with a slow preplanned trajectory and reaches a fast preplanned trajectory? A general related question is how to start-up any of the preplanned motion patterns since initial postures for a robot are usually rest positions.

Mathematically, the feasibility of a transition is a matter of the region of attraction of the target motion pattern. If the initial trajectory has an intersection with the region of attraction of the target motion, a direct transition between the trajectories is possible without calculation of special transition primitives.

The region of attraction of trajectories that were presented for the compass gait robot, the monoped robot, or the gymnast robot cannot be determined analytically, and also a

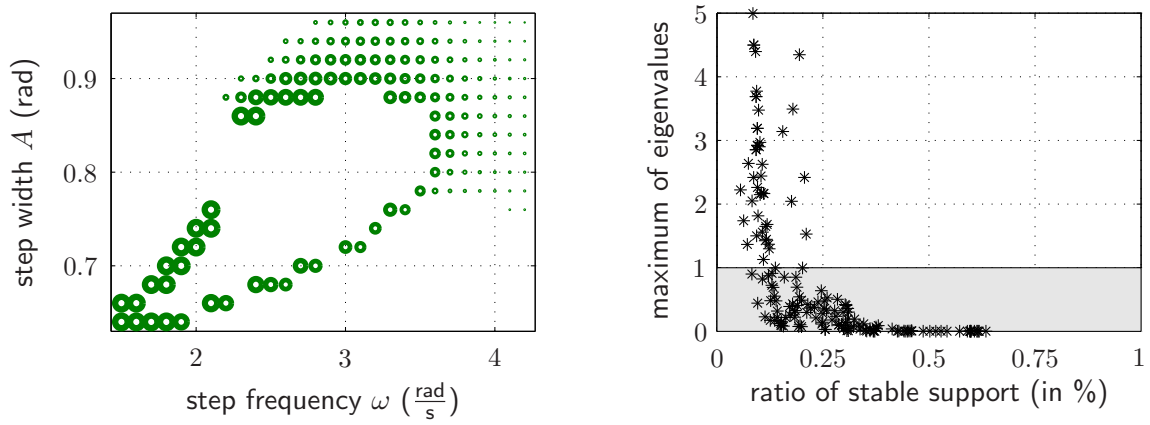

Figure 4.14: Details on stability of gymnast walking trajectories. Left: Ratio of the stable support phase. Large markers imply a long stable support phase, small markers imply a short stable support phase. Right: Ratio of the stable support phase in percent of the complete cycle vs. absolute maximum of the eigenvalues of the linearized Poincaré map.

numerical approximation is difficult due to the hybrid and nonlinear character of the dynamics. In addition, the dimension of the example systems is too high to obtain meaningful illustration. A numerical approximation of the region of attraction was calculated for a simplified compass gait robot by Schwab *et al.* [112] on the basis of the cell-to-cell mapping method [64]. For this simplified walking system, the basin of attraction was displayed in the two-dimensional configuration plane.

In the following, a switching strategy for the preplanned walking trajectories of the gymnast robot is proposed. Thereby, it is used that preplanned walking trajectories have similar desired trajectories that differ only in parameters.

Here it will be demonstrated via simulations that the walking motion of the gymnast robot is robust against changes in the parameters step width A and step frequency ω. These parameters are slightly altered while walking using the knowledge of the existence of solutions for different parameter sets. This can be used to start walking from a rest position and accelerating to maximum velocity.

The switching time for a change of the parameters (A, ω) is chosen such that discontinuities are avoided in the desired joint trajectories that cause high action of the controller. Investigation of the desired trajectories from (3.8) results in candidate switching times. Switching of A from A^- to A^+ is thus possible for

$$t_s = \frac{\pi}{2\omega} + \mathbb{Z}\frac{\pi}{\omega}.$$

This is when $\cos\omega t = 0$ and thus $\beta_d^3 = \pi + A^+ \cos\omega t = \pi + A^- \cos\omega t$. Switching of ω from ω^- to ω^+ is possible without discontinuities for

$$t_s = \mathbb{Z}\frac{\pi}{2\omega^-}.$$

For these times β_2^d and β_4^d are either maximal or minimal.

Figure 4.15 shows a snapshot series for walking, with transitions from motion pattern S0, via pattern S1 and S2, to motion pattern S3.

S0:	at rest	
S1:	slow	$(A_1, \omega_1) = (0.64\,\text{rad}, 1.9\,\frac{\text{rad}}{\text{s}})$
S2:	medium	$(A_2, \omega_2) = (0.72\,\text{rad}, 3.0\,\frac{\text{rad}}{\text{s}})$
S3:	fast	$(A_3, \omega_3) = (0.92\,\text{rad}, 3.8\,\frac{\text{rad}}{\text{s}})$

The initial time is chosen $t_0 = 0$, the first switching time is chosen $t_{s,1} = \frac{3\pi}{2\omega_1}$, then $t_{s,2} = \frac{5\pi}{2\omega_1}$, and finally to reach the fastest walking velocity $t_{s,3} = \frac{5\pi}{2\omega_1} + 2\frac{\pi}{2\omega_2}$.

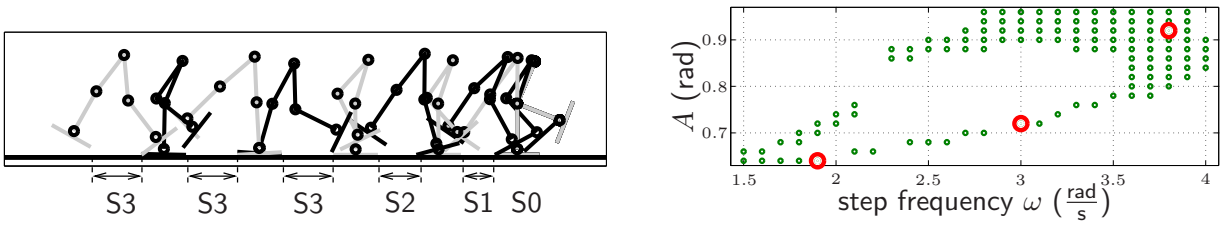

Figure 4.15: Left: Snapshot series of accelerated walking motion of the gymnast robot. Right: Participating trajectories are marked by larger circles in the (A, ω)-plane.

4.4 Summary

Periodic trajectories for legged robots with variable ground contact that are determined by trajectory planning are not necessarily orbitally stable, but orbital stability is a key necessity for any kind of application of the trajectory.

In this chapter orbital stability of trajectories for the compass gait robot, the monoped robot, and the gymnast robot was investigated. Therefore, available stability results for periodic trajectories of hybrid dynamical systems were summarized and thoroughly discussed. Poincaré maps (first-return) maps were used for stability analysis yielding conditions for orbital stability by evaluation of the eigenvalues of the linearization of the Poincaré map. Stability analysis using Poincaré maps was shown to be equivalent to sensitivity based methods. This equivalence is brought forward from the ordinary differential equation case to the hybrid system case.

Although for the compass gait robot stable trajectories were found for arbitrary step frequencies, it was demonstrated that the walking velocity does not necessarily increase if the step frequency is increased. In most cases, trajectories with low step frequencies are available that enable the same walking velocity with better stability properties. The walking velocity is bounded by physical limitations, e.g. when friction between ground and feet is insufficient. Also for the gymnast robot, not all trajectories from the database of trajectory planning are feasible. Fast walking trajectories are not feasible any more because they lack stability. The maximum possible velocity is nevertheless comparable to that of the compass gait robot. For the monoped robot and for the gymnast robot, it was analyzed how the stable support phase improves walking stability. It was concluded that a stable support phase is not necessary for orbital stability of the whole locomotion cycle but makes stability more likely. Finally the feasibility of transition between different locomotion patterns was discussed and demonstrated in a simulation experiment for gymnast robot walking. Enabling transitions between planned trajectory primitives without additional trajectory planning of transition primitives is seen as crucial for realization of dexterous locomotion. Appropriate analysis makes use of the region of attraction of hybrid periodic limit cycles.

Poincaré map based methods are a common tool for legged locomotion analysis. They are until now used mainly for passive robots or robots with point contacting feet. Then every motion cycle consists of only one contact situation and one collision. It was demonstrated in this thesis that stability analysis by Poincaré maps is a valid approach also for robots where a locomotion cycle comprises several different contact situations and multiple collisions.

5 Balance Control

5.1 Introduction and State of the Art

The Zero Moment Point (ZMP) [132, 133] is a point on ground level that provides a measure for the postural balance of legged robots. Thus, control of the ZMP is widely used for balance control. A balanced robot state is here understood to exclude underactuation in rotation around foot edges because this is often seen to initiate falling. If the ZMP is in the foot-ground contact area or inside the convex hull of the contact areas of multiple contacting feet, tilting around foot edges will not occur. Therefore, a ZMP control algorithm is necessary that yields invariance of an admissible set for the ZMP in the convex hull of the foot margins. Approaches for balance control of legged robots were presented by many groups [67, 81, 85, 99, 125] and in particular for the Honda Asimo [59] or the Toyota Partner Robots [129].

In Sec. 2.2.3, it was shown for a planar robot construction that the ZMP is calculated from the acting contact forces and moments, denoted as

$$r_{\mathrm{zmp}} = \frac{T_z}{R_y},$$

where T_z is the contact moment and R_y is the vertical contact force. This gives rise to physically motivated control laws. For example enhancing the vertical contact force shifts the ZMP towards zero, in general resulting in better stability margins, defined as distance between ZMP and foot edges [99]. In experimental studies, often a heuristic control approach is used. The reference trajectory for the ankle joint is altered proportional to the deviation of the measured ZMP from a desired ZMP [67]. Although the approach is successfully applied in experiments, it comes with a theoretical drawback: The control torque is set depending on the actual ZMP deviation where the ZMP is at the same time evaluated and also depends on the torque resulting in an algebraic loop. Thus, the approach is not applicable in simulations without solving the algebraic loop, and the validation of the control strategy is analytically not possible. Theoretically consistent formulations consider the ZMP dependence from the accelerations [81, 85] and calculate a prediction of the ZMP before applying the appropriate motor torques that either yield tracking of the desired trajectory or a correction of the ZMP. In the control method for the humanoid robot Johnnie [85], the control of the center of gravity is replaced by control of contact forces whenever a ZMP violation is about to occur, and also Kondak *et al.* [81] restrict accelerations to an admissible plane when a violation of balance is about to occur. In contrast to the experimental approaches that rely on heuristics, these two approaches are model-based and realize exact ZMP tracking.

Control of the ZMP makes the control system underactuated in the sense that the number of controlled quantities exceeds the number of control inputs [40]. Therefore a ZMP con-

trolled robotic system has an internal dynamics. This internal dynamics is unstable and the control system is termed non-minimum phase. Balance control for a legged robot is only stable if it is switched between different controllers, among them the ZMP controller.

The presented approach, see also [138], is similar to the approaches in [81, 85] since it uses model-based ZMP tracking as corrective controller that acts whenever control with the nominal task-specific controller results in violation of the inequality

$$y_l \leq r_{\text{zmp}} \leq y_u.$$

The range between the lower boundary y_l and the upper boundary y_u is the admissible set for the ZMP that is a subset of the foot supporting area. The ZMP control problem will be considered in the control theoretic framework of invariance control [137]. The ZMP is there defined as a system output that is not allowed to leave the assigned admissible set. To be applicable, the invariance control method had to be adapted to account for systems with fewer outputs than inputs. A control theoretic problem for ZMP control is that the ZMP and the motor torques are algebraically related. In control theory, an output like ZMP is said to have relative degree zero. By model refinement, a control system is obtained where the output ZMP has relative degree one, or in other words it does not depend on the input directly. This allows for a consistent formulation of ZMP control methods also for experimental studies.

In the following Sec. 5.2 the invariance control method is introduced. Then in Sec. 5.3 the method is applied for balance control of legged robots with a formulation where the ZMP has relative degree zero and another formulation where the ZMP has relative degree one. Simulation results for a humanoid robot demonstrate applicability. A summary of the chapter is given in Sec. 5.4.

5.2 Invariance Control of Control-Affine Systems

Invariance control, as proposed by Wolff *et al.* in [137], considers control-affine single-input single-output systems, denoted by

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{g}(\boldsymbol{x})u \\ y &= h(\boldsymbol{x}). \end{aligned} \tag{5.1}$$

That means, the system input u as well as the output y are scalar, the output is not algebraically related to the input, and the differential equation is linear in the input u. The invariance control method describes how to implement a switching control law that takes into account an output constraint

$$y = h(\boldsymbol{x}) \leq 0. \tag{5.2}$$

The idea of invariance control is to switch between a nominal control signal u_{nom} and a corrective control signal u_{corr}. The nominal control law is chosen specific to the application and achieves the main control task. The corrective control law and the transition time between nominal control and corrective control are chosen such that the output constraints

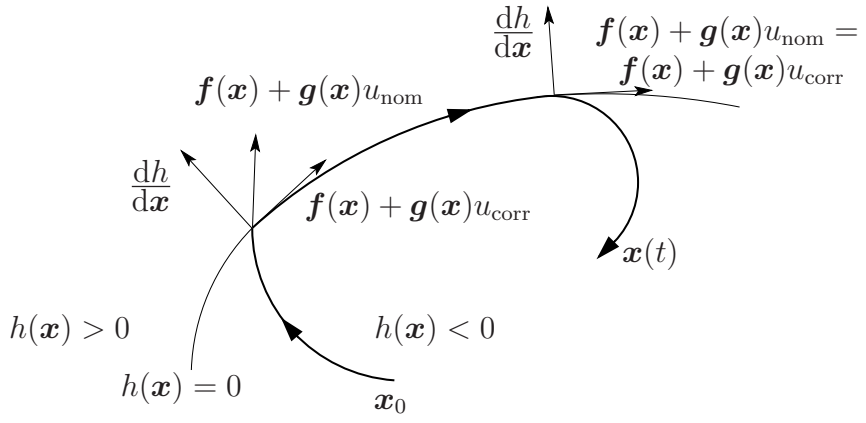

Figure 5.1: Illustration of the invariance control law for relative degree one. The trajectory starts at $\boldsymbol{x}_0$ and hits the boundary $h(\boldsymbol{x}) = 0$ of the admissible set. At this point, the nominal control signal u_{nom} would violate the constraint $y = h(\boldsymbol{x}) \leq 0$, therefore u_{nom} is replaced by u_{corr}. The vector field $\boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{g}(\boldsymbol{x})u_{\mathsf{corr}}$ is then tangential to the boundary $h(\boldsymbol{x}) = 0$. The control signal returns to u_{nom} when the vector field $\boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{g}(\boldsymbol{x})u_{\mathsf{nom}}$ begins to point inward. This is equivalent to $\dot{y}(\boldsymbol{x}, u_{\mathsf{nom}}) < 0$.

are never violated. The derivation of the corrective controller relies on input-output feedback linearization of the control system. Therefore the output function is differentiated until an algebraic relation between input and derivative of the output is obtained. The relative degree of the output is the minimum number of derivations after that the algebraic input-output relation occurs. Since the ZMP is directly related to the input torques, see (2.24), its relative degree is zero. A refinement of the model will be discussed later where the relative degree of the ZMP will be one.

Wolff *et al.* [137] outline the invariance control method for outputs with relative degrees one and higher. The idea is summarized in the following for relative degree one. In this case, the first derivative of the output is assumed to be related to the input u by

$$\dot{y} = \frac{\mathrm{d}h}{\mathrm{d}\boldsymbol{x}} \left[\boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{g}(\boldsymbol{x})\, u\right] =: \dot{y}(\boldsymbol{x}, u) \tag{5.3}$$

whenever $\frac{\mathrm{d}h}{\mathrm{d}\boldsymbol{x}}\boldsymbol{g}(\boldsymbol{x}) \neq 0$. The corrective control signal u_{corr} is obtained as solution of $\dot{y}(\boldsymbol{x}, u_{\mathrm{corr}}) = 0$, and it is used if the output y reaches the boundary of the admissible set $y = h(\boldsymbol{x}) = 0$. It is switched back from the corrective mode to the nominal mode if using the nominal controller does not violate the constraint any more. This is when $\dot{y}(\boldsymbol{x}, u_{\mathrm{nom}}) < 0$. The switching scheme is illustrated in Fig. 5.1 and summarized as:

$$u = \begin{cases} u_{\mathrm{nom}} & \text{if } h(\boldsymbol{x}) < 0 \\ u_{\mathrm{nom}} & \text{if } h(\boldsymbol{x}) \geq 0 \,\text{and}\, \dot{y}(\boldsymbol{x}, u_{\mathrm{nom}}) < 0 \\ u_{\mathrm{corr}} & \text{if } h(\boldsymbol{x}) \geq 0 \,\text{and}\, \dot{y}(\boldsymbol{x}, u_{\mathrm{nom}}) \geq 0 \end{cases} \tag{5.4}$$

The invariance control method that was described above has to be adapted to account for the special requirements of ZMP control: In the usual notation, the ZMP is directly

set by the motor torques, and the control system is said to have relative degree zero. In addition, there is a vector of control inputs available that represents the torques in the joints, and a decision has to be made how to use the numerous control inputs to manipulate the one-dimensional ZMP.

5.2.1 Adaptation for Relative Degree Zero

The output has relative degree zero if the input directly sets the output. The control system (5.1) is then modified to:

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{g}(\boldsymbol{x})\, u \\ y &= h(\boldsymbol{x}, u) \end{aligned} \tag{5.5}$$

The corrective control is obtained as solution of

$$y = h(\boldsymbol{x}, u_{\text{corr}}) = 0 \tag{5.6}$$

to account for constraints $y = h(\boldsymbol{x}, u) \leq 0$. In general, (5.6) is a nonlinear equation for u_{corr} and solvability has to be examined carefully. Also, the solution must not be unique. The corrective control signal u_{corr} is used when the output reaches the boundary of the admissible set. The nominal control signal u_{nom} is used when the output is inside the admissible set. In contrast to (5.4), the switching scheme is now:

$$u = \begin{cases} u_{\text{nom}} & \text{if } h(\boldsymbol{x}, u_{\text{nom}}) < 0 \\ u_{\text{corr}} & \text{if } h(\boldsymbol{x}, u_{\text{nom}}) \geq 0 \end{cases} \tag{5.7}$$

A problem of invariance control for constrained outputs with relative degree zero is that the control signal u has to be used to evaluate $h(\boldsymbol{x}, u)$ and, at the same time, the control signal is used in the control law. This makes it necessary to preview the behavior of the control system for the next integration step and then decide on the appropriate choice of the control signal.

5.2.2 Adaptation for Non-Scalar Inputs

If the input is not restricted to be scalar, the control system (5.1) is modified to

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{G}(\boldsymbol{x})\boldsymbol{u} \\ y &= h(\boldsymbol{x}) \end{aligned} \tag{5.8}$$

with $\boldsymbol{G}(\boldsymbol{x}) \in \mathbb{R}^{n \times m}$ to account for $\boldsymbol{u} \in \mathbb{R}^m$. If the relative degree is one, presuming $\frac{\mathrm{d}h}{\mathrm{d}\boldsymbol{x}}\boldsymbol{G}(\boldsymbol{x}) \neq \boldsymbol{0}$, the corrective control signal $\boldsymbol{u} = \boldsymbol{u}_{\text{corr}}$ solves

$$\dot{y} = \frac{\mathrm{d}h}{\mathrm{d}\boldsymbol{x}} \left[\boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{G}(\boldsymbol{x})\boldsymbol{u}\right] = 0. \tag{5.9}$$

In contrast to (5.3) that has a unique solution, (5.9) is an under-determined linear equation system for the m components of $\boldsymbol{u}_{\text{corr}}$. The equation system can be abbreviated by

$$\boldsymbol{A}\boldsymbol{u}_{\text{corr}} = b \tag{5.10}$$

with $\boldsymbol{A} = \frac{\mathrm{d}h}{\mathrm{d}\boldsymbol{x}}\boldsymbol{G}(\boldsymbol{x})$ and $b = -\frac{\mathrm{d}h}{\mathrm{d}\boldsymbol{x}}\boldsymbol{f}(\boldsymbol{x})$. In what follows, three propositions a) to c) are presented how to solve (5.10).

a.) Minimum Norm. A solution $\boldsymbol{u}_{\text{corr}}$ of (5.10) that has minimum norm $\|\boldsymbol{u}_{\text{corr}}\|^2$ is obtained by inversion using the pseudoinverse [95]:

$$\boldsymbol{u}_{\text{corr}} = \boldsymbol{A}^{+} b \tag{5.11}$$

Here $\boldsymbol{A}^{+}$ denotes the pseudoinverse matrix of $\boldsymbol{A}$. In this case, the matrix $\boldsymbol{A} = \big(a_1, \ldots, a_m\big)$ is a $1 \times m$-matrix where the pseudoinverse is a $m \times 1$ that yields $\boldsymbol{A}\boldsymbol{A}^{+} = 1$:

$$\boldsymbol{A}^{+} = \frac{1}{a_1^2 + \ldots + a_m^2} \left(a_1, \ldots, a_m\right)^T$$

It is obvious from the definition of the pseudoinverse that a solution of (5.10) exists, as long as at least one of the components a_i in $\boldsymbol{A}$ is non-zero. This requirement is equivalent to the requirement for relative degree one, where $\boldsymbol{A} = \frac{\mathrm{d}h}{\mathrm{d}\boldsymbol{x}}\boldsymbol{G}(\boldsymbol{x}) \neq \boldsymbol{0}$ was yet presumed. In the solution all input degrees of freedom are involved in constraint allowance. This is not necessarily the best choice, since while control with minimal norm is used, the nominal task cannot be taken into account, and a return to the desired task may not be possible again. Thus, this approach is modified:

b.) Minimum Distance to Nominal Control. The solution of (5.10) is chosen for the corrective control signal $\boldsymbol{u}_{\text{corr}}$ that is closest to the nominal control signal $\boldsymbol{u}_{\text{nom}}$.

$$\boldsymbol{u}_{\text{corr}} = \underset{\boldsymbol{u}}{\operatorname{argmin}} \, \|\boldsymbol{u} - \boldsymbol{u}_{\text{nom}}\|^2$$

The solution is obtained by pseudoinverse application together with a projection $(\boldsymbol{I} - \boldsymbol{A}^{+}\boldsymbol{A})\boldsymbol{u}_{\text{nom}}$ of $\boldsymbol{u}_{\text{nom}}$ into the kernel of $\boldsymbol{A}$

$$\boldsymbol{u}_{\text{corr}} = \boldsymbol{A}^{+} b + (\boldsymbol{I} - \boldsymbol{A}^{+}\boldsymbol{A})\boldsymbol{u}_{\text{nom}}, \tag{5.12}$$

where $\boldsymbol{I}$ is the identity matrix. So, if an entry a_i of the matrix $\boldsymbol{A}$ is zero, the definition in (5.12) yields $\boldsymbol{u}_{\text{corr},i} = \boldsymbol{u}_{\text{nom},i}$, and the remaining inputs are used to achieve constraint compliance.

c.) Weighted Minimum Distance to Nominal Control. The first two approaches yield constraint compliance using all available inputs. Alternatively, selected inputs can be used, or the inputs can be used with different weighting. Therefore a diagonal $m \times m$ weighting matrix $\boldsymbol{W}$ can be introduced. The $i-$th diagonal element $w_{i,i}$ weights the impact of input i in the minimization problem

$$\min \|\boldsymbol{W}^{-1}(\boldsymbol{u} - \boldsymbol{u}_{\text{nom}})\|^2$$

subject to the equality (5.10). The solution is obtained as:

$$\boldsymbol{u}_{\text{corr}} = \boldsymbol{W}(\boldsymbol{A}\boldsymbol{W})^{+} b + \left(\boldsymbol{I} - \boldsymbol{W}(\boldsymbol{A}\boldsymbol{W})^{+}\boldsymbol{A}\right)\boldsymbol{u}_{\text{nom}} \tag{5.13}$$

Thus, for small values of $w_{i,i}$ the input i has a large weight and does not contribute to the solution too much. Large values of $w_{i,i}$ give a small weight to the i-th component of the optimization problem, and high values for the corresponding input u_i are possible in the solution. If $w_{i,i} = 0$, the corresponding component u_i of the input is not used for constraint compliance, instead the control uses the nominal control signal.

5.3 Invariance Control of Zero Moment Point

The balance control task for a legged robot is based on the system dynamics from Sec. 2.2.1 of the mechanical construction subject to motor torques $\boldsymbol{\tau}$

$$\boldsymbol{M}(\boldsymbol{\beta})\ddot{\boldsymbol{\beta}} + \boldsymbol{n}(\boldsymbol{\beta}, \dot{\boldsymbol{\beta}}) = \boldsymbol{\tau}. \tag{5.14}$$

A model of the generation of the motor torques $\boldsymbol{\tau}$ can be included. The simplest assumption is that motor torques are directly set by the control input $\boldsymbol{u}$ and thus $\boldsymbol{u} = \boldsymbol{\tau}$. Another model assumption that will be investigated in the following is that motor torques are related to the input $\boldsymbol{u}$ by a first order dynamics

$$\dot{\boldsymbol{\tau}} + \boldsymbol{K}\boldsymbol{\tau} = \boldsymbol{u}. \tag{5.15}$$

The diagonal parameter matrix $\boldsymbol{K}$ characterizes the behavior of the motor dynamics. The goal of invariance control is to constrain the ZMP to an admissible set inside the convex hull of the foot contact points.

$$y_l \leq r_{\text{zmp}} \leq y_u$$

Therefore, output functions y_1 and y_2 are defined according to (5.2):

$$y_1 = r_{\text{zmp}} - y_u \leq 0 \tag{5.16}$$

$$y_2 = y_l - r_{\text{zmp}} \leq 0 \tag{5.17}$$

The ZMP is calculated from the position $\boldsymbol{\beta}$, the velocities $\dot{\boldsymbol{\beta}}$, and the accelerations $\ddot{\boldsymbol{\beta}}$ or the torques $\boldsymbol{\tau}$ respectively, see Sec. 2.2.3. Depending on what model is used for the joint actuation, the system output ZMP has either relative degree zero or relative degree one.

In what follows, an invariance control approach for the ZMP with relative degree zero is presented in Sec. 5.3.1. Then, the model is refined such that ZMP has relative degree one, see Sec. 5.3.2. After illustration of the method by simulations of a humanoid robot in Sec. 5.3.3, finally in Sec. 5.3.4 the non-minimum phase characteristics and stability of invariance ZMP control is discussed.

5.3.1 Relative Degree Zero Formulation

Control System. Basis for the controller design in this section are the equations of motion (5.14) where the torques are considered as control input $\boldsymbol{u} = \boldsymbol{\tau}$. Since the ZMP depends algebraically on the motor torques and the torques are considered as control inputs, the ZMP has relative degree zero in this case.

Before application of the invariance control method, the control system is transformed to a linear differential equation with nonlinear output by input-state linearization, compare Sec. 3.2.2. Substituting $\boldsymbol{u}$ in (5.14) by

$$\boldsymbol{u}(\boldsymbol{v}) = \boldsymbol{M}(\boldsymbol{\beta})\boldsymbol{v} + \boldsymbol{n}(\boldsymbol{\beta}, \dot{\boldsymbol{\beta}}).$$

yields

$$\dot{\boldsymbol{x}} = \underbrace{\begin{pmatrix} \dot{\boldsymbol{\beta}} \\ \boldsymbol{0} \end{pmatrix}}_{\boldsymbol{f}(\boldsymbol{x})} + \underbrace{\begin{pmatrix} \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{I} \end{pmatrix}}_{\boldsymbol{G}(\boldsymbol{x})} \boldsymbol{v}$$

$$y_1 = r_{\text{zmp}}(\boldsymbol{x}, \boldsymbol{v}) - y_u \qquad y_2 = y_l - r_{\text{zmp}}(\boldsymbol{x}, \boldsymbol{v})$$

where $\boldsymbol{v}$ is an auxiliary control input and $\boldsymbol{x}$ summarizes $\boldsymbol{\beta}$ and $\dot{\boldsymbol{\beta}}$. The notation indicates that the ZMP depends on the control input $\boldsymbol{v}$. The control system combines properties of (5.5) and (5.8), therefore the control law for the auxiliary control input $\boldsymbol{v}$ can be designed by invariance control. In the following, the nominal controller and the corrective controller are introduced.

Nominal Controller. If control is due to the nominal controller $\boldsymbol{v} = \boldsymbol{v}_{\text{nom}} = \ddot{\boldsymbol{\beta}}^d + \boldsymbol{K}_D(\dot{\boldsymbol{\beta}}^d - \dot{\boldsymbol{\beta}}) + \boldsymbol{K}_P(\boldsymbol{\beta}^d - \boldsymbol{\beta})$, the desired trajectory $\boldsymbol{\beta}^d(t)$ is tracked. Dynamical properties can be set by choice of the control parameters in $\boldsymbol{K}_D$ and $\boldsymbol{K}_P$.

Whenever the nominal control signal $\boldsymbol{v} = \boldsymbol{v}_{\text{nom}}$ is determined, $r_{\text{zmp}}(\boldsymbol{x}, \boldsymbol{v}_{\text{nom}})$ is checked for constraint compliance. If the constraint compliance is violated, which means that the ZMP reaches either the upper boundary y_u or the lower boundary y_l of the admissible set, the output of the nominal controller is discarded, and a corrective control is switched on to keep the ZMP on the boundary of the admissible set.

Corrective Controller. The corrective controller $\boldsymbol{v} = \boldsymbol{v}_{\text{corr}}$ has to yield either

$$y_1 = r_{\text{zmp}}(\boldsymbol{x}, \boldsymbol{v}_{\text{corr}}) - y_u = 0 \tag{5.18}$$

if the ZMP was about to leave at the lower boundary or

$$y_2 = y_l - r_{\text{zmp}}(\boldsymbol{x}, \boldsymbol{v}_{\text{corr}}) = 0 \tag{5.19}$$

if the ZMP was about to leave at the upper boundary.

The above equations (5.18) respectively (5.19) give one determining equation for m control inputs in $\boldsymbol{v}_{\text{corr}} = \big(v_1, \ldots, v_m\big)^T$. Thus there are multiple possibilities for the choice of $\boldsymbol{v}_{\text{corr}}$. Since the contact force R_y and contact moment T_z are linear in the accelerations $v_i = \ddot{\beta}_i$, which can be seen from (2.24), a linear equation has to be solved. For the lower boundary, it can be denoted as

$$r_{\text{zmp}}(\boldsymbol{x}, \boldsymbol{v}) = \frac{T_z}{R_y} = \frac{T_{z,0} + \sum_i T_{z,i} v_i}{R_{y,0} + \sum_i R_{y,i} v_i} = y_l \tag{5.20}$$

with appropriate coefficients $T_{z,i}$ and $R_{y,i}$. Recombination yields

$$\sum_i \underbrace{(T_{z,i} - y_l R_{y,i})}_{a_i} v_i = \underbrace{y_l R_{y,0} - T_{z,0}}_{b}.$$

Vector notation results in a linear equation system of the same structure as in (5.10):

$$\boldsymbol{A}\boldsymbol{v} = b \tag{5.21}$$

To keep the ZMP on the boundary, any of the infinitely many solutions of (5.21) can be used. In Sec. 5.2, possible choices were discussed respecting the distance to the nominal control and a weighting matrix.

The corrective controller acts as long as the nominal controller still results in violations of the output constraints. Therefore, while the corrective controller is active, the ZMP is constantly evaluated to see what would happen if it was active. The switching rules are to be found in (5.7).

5.3.2 Relative Degree One Formulation

Control System. The present formulation of invariance control of the ZMP with relative degree zero has a drawback. Since the ZMP depends on the torques and the torques are set by the control law, the decision which controller is to be used relies on knowledge of the ZMP behavior when the controller is used. That means, the ZMP must be previewed, and the control torque is rejected if violation of the constraint conditions would occur.

The present formulation of the robot dynamics is also still a simplification. It is assumed that the controller directly sets the torques. In a more detailed model, the control still has to pass the motor dynamics. In the following it is shown how the incorporation of a simple motor model (5.15) results in a control system where the ZMP is an output of relative degree one. Accounting for the motor dynamics yields that the input torques $\boldsymbol{\tau}$ are no longer a system input but a system state and $\boldsymbol{u}$ is the new input. As a consequence, the output function r_{zmp} does not depend on the input directly.

The control system is linearized by input-state feedback linearization as preparation for design of the nominal control law and the corrective control law. The state $\boldsymbol{\beta}$ of the control system has relative degree three, because neither the joint angles $\boldsymbol{\beta}$ themselves nor the first or second derivative of it depend on $\boldsymbol{u}$.

For the transformation, $\boldsymbol{\beta}$ is repeatedly differentiated, up to its relative degree.

$$\dddot{\boldsymbol{\beta}} = \boldsymbol{M}^{-1}(\boldsymbol{u} - \boldsymbol{K}\boldsymbol{\tau} - \dot{\boldsymbol{n}}) + \frac{\mathrm{d}\boldsymbol{M}^{-1}}{\mathrm{d}t}(\boldsymbol{\tau} - \boldsymbol{n}) \tag{5.22}$$

Here (5.15) is used to replace $\dot{\boldsymbol{\tau}}$. The control system is said to be input-state linearizable because the linearization does not have an internal dynamics. Demanding $\dddot{\boldsymbol{\beta}} = \boldsymbol{v}$ results in an expression $\boldsymbol{u}(\boldsymbol{v})$ that linearizes the control system:

$$\boldsymbol{u}(\boldsymbol{v}) = \boldsymbol{M}\boldsymbol{v} + \frac{\mathrm{d}\boldsymbol{M}}{\mathrm{d}t}\ddot{\boldsymbol{\beta}} + \boldsymbol{K}\boldsymbol{\tau} + \dot{\boldsymbol{n}}$$

Here, the relation

$$\frac{\mathrm{d}\boldsymbol{M}^{-1}}{\mathrm{d}t} = -\boldsymbol{M}^{-1}\frac{\mathrm{d}\boldsymbol{M}}{\mathrm{d}t}\boldsymbol{M}^{-1}$$

is used that results from $\frac{\mathrm{d}}{\mathrm{d}t}(\boldsymbol{M}\boldsymbol{M}^{-1}) = \boldsymbol{0}$.

The transformed control system is given in the coordinates $\boldsymbol{x} = \left(\boldsymbol{\beta},\, \dot{\boldsymbol{\beta}},\, \ddot{\boldsymbol{\beta}}\right)^T$ with input $\boldsymbol{v}$. It can be summarized as

$$\dot{\boldsymbol{x}} = \underbrace{\begin{pmatrix} \dot{\boldsymbol{\beta}} \\ \ddot{\boldsymbol{\beta}} \\ \boldsymbol{0} \end{pmatrix}}_{\boldsymbol{f}(\boldsymbol{x})} + \underbrace{\begin{pmatrix} \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{I} \end{pmatrix}}_{\boldsymbol{G}(\boldsymbol{x})} \boldsymbol{v}$$

$$y_1 = r_{\text{zmp}}(\boldsymbol{x}) - y_u \qquad y_2 = y_l - r_{\text{zmp}}(\boldsymbol{x})$$

to account for the formulation in (5.8). The outputs y_1 and y_2 each have relative degree one.

Nominal Controller. The input-state linearization allows a linear controller design for the nominal control $\boldsymbol{v}_{\text{nom}} = \dddot{\boldsymbol{\beta}}^d + \boldsymbol{K}_A(\ddot{\boldsymbol{\beta}}^d - \ddot{\boldsymbol{\beta}}) + \boldsymbol{K}_D(\dot{\boldsymbol{\beta}}^d - \dot{\boldsymbol{\beta}}) + \boldsymbol{K}_P(\boldsymbol{\beta}^d - \boldsymbol{\beta})$. The control parameters $\boldsymbol{K}_A$, $\boldsymbol{K}_D$, and $\boldsymbol{K}_P$ are used to parameterize the linear control law. The nominal controller acts until the ZMP reaches the boundary of the admissible set. After the model refinement, the ZMP depends on the system state only. In hardware-implementation also a measurement could be used.

Corrective Controller. Since the first derivative of the ZMP $\dot{r}_{\text{zmp}}(\boldsymbol{x})$ depends on the input $\boldsymbol{v}_{\text{corr}}$, but not the ZMP r_{zmp} itself, the system outputs y_1 from (5.16) and y_2 from (5.17) have relative degree one. The equation to determine the invariance control signal is derived from

$$\dot{y}_1 = \frac{\partial r_{\text{zmp}}}{\partial \boldsymbol{\beta}}\dot{\boldsymbol{\beta}} + \frac{\partial r_{\text{zmp}}}{\partial \dot{\boldsymbol{\beta}}}\ddot{\boldsymbol{\beta}} + \frac{\partial r_{\text{zmp}}}{\partial \ddot{\boldsymbol{\beta}}}\dddot{\boldsymbol{\beta}} = 0$$

Since $\dddot{\boldsymbol{\beta}} = \boldsymbol{v}$ is achieved by feedback linearization the invariance condition is linear in the control input $\boldsymbol{v}$, compare (5.9).

$$\boldsymbol{A}\boldsymbol{v}_{\text{corr}} = b \tag{5.23}$$

$$\text{with } \boldsymbol{A} = \frac{\partial r_{\text{zmp}}}{\partial \ddot{\boldsymbol{\beta}}} \text{ and } b = -\frac{\partial r_{\text{zmp}}}{\partial \boldsymbol{\beta}}\dot{\boldsymbol{\beta}} - \frac{\partial r_{\text{zmp}}}{\partial \dot{\boldsymbol{\beta}}}\ddot{\boldsymbol{\beta}}$$

Equation (5.23) can be solved for $\boldsymbol{v}_{\text{corr}}$ but the solution is not unique since the input dimension m is in general larger than one. Thus one of the infinitely many solutions of (5.23) is chosen for $\boldsymbol{v}_{\text{corr}}$ according to (5.11), (5.12), or (5.13). For legged robot balance control, it is important to choose a corrective motion that is as close as possible to the nominal motion. By weighting, joints can be preferred that do not interfere too much with the nominal task. For example in a walking motion, the acceleration of the arms could be used to correct the ZMP position for small disturbances of the planned motion.

The nominal controller acts until the ZMP reaches the boundary of the admissible set. The corrective control law is used after the ZMP reached the boundary of the admissible set and keeps the ZMP on the boundary. To decide, whether the corrective controller may stop acting and the nominal controller can be used again, the first derivative of the ZMP with nominal control is monitored. Only if the derivative points inward the admissible set, the nominal control can be used again. To obtain the ZMP derivative, a virtual model has to be evaluated in parallel to the integration of the real model. See (5.4) for a formalization of the switching rules.

5.3.3 Application for Balance Control of a Humanoid Robot

For the following illustration of invariance ZMP control, a robot similar to the gymnast robot in Sec. 2.3.3, is considered. Here it is extended with a trunk, a head, and one arm. The masses and lengths are adapted to suit masses and link lengths of a human being. The total weight thus is 76 kg, and the height is 1.90 m. The desired motion is balancing on one leg. Therefore an initial posture, a final posture, and a nominal controller are chosen. Then at times $t = 2.0\,\text{s}$ and $t = 4.0\,\text{s}$ the robot is pushed. The first push is applied from the front at the shoulder, acting instantaneously with 42 Ns, the second push acts from

the backside on the swing foot with 4.2 Ns. The externally applied forces $\boldsymbol{F}_{\text{push}}$ result in discontinuities in the velocities, calculated from

$$\boldsymbol{M}(\boldsymbol{q})\left(\dot{\boldsymbol{q}}^{+} - \dot{\boldsymbol{q}}^{-}\right) = \boldsymbol{J}(\boldsymbol{q})^{T}\boldsymbol{F}_{\text{push}}.$$

The matrix $\boldsymbol{J}(\boldsymbol{q})$ is the Jacobian of the working point of the attacking force. Compare with (2.17) in Sec. 2.2.2 for a derivation. The applied forces are close to the maximum possible forces that can be compensated by the proposed balance control scheme. It is seen in Fig. 5.2 that the execution of the desired motion without corrective controller implies that the ZMP leaves the admissible area, which is the foot contact set between 0 and 0.3. This happens while the desired trajectory is tracked as well as after both pushes.

Invariance control with the corrective control law (5.13) is applied, and the refined model is used, where the ZMP is an output quantity with relative degree one. For this example, the ankle joint is weighted by $w_{1,1} = 0.5$, and the shoulder and elbow joints are weighted by $w_{7,7} = w_{8,8} = 15.0$. The remaining joints have weight one. In [106] other realizations for the weighting matrix parameterization are discussed. The corrective control is applied whenever the ZMP leaves the admissible set. In Fig. 5.2 the ZMP trajectory with invariance control is compared to the ZMP trajectory without correcting controller.

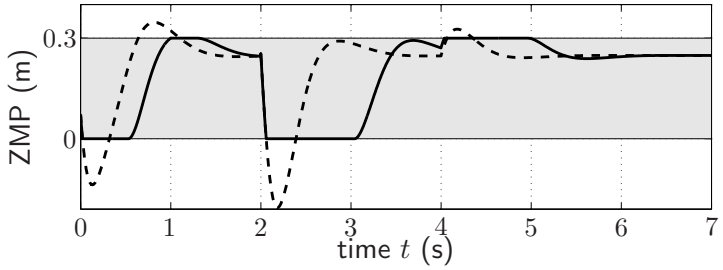

Figure 5.2: ZMP for balancing of the legged robot. Comparison of ZMP without invariance control (dashed) and with corrective control (solid).

Figure 5.3 depicts an associated snapshot series. In Fig. 5.2 it can be seen that the pushes themselves do not result in large ZMP deflections. Rather the correction motion of the nominal controller results in large ZMP amplitudes as the correction motion involves sudden high accelerations. The invariance control that keeps the ZMP on the boundary of the admissible set realizes a model-based compliancy in the robot joints.

5.3.4 Discussion

Control of the ZMP results in non-minimum phase behavior of the control system. A nonlinear control system is said to have non-minimum phase character if the zero dynamics is unstable. The zero dynamics is obtained from the internal dynamics when the output is constantly controlled to zero. The ZMP control task has an internal dynamics because the relative degree of the ZMP as output function is smaller than the system order. The instability of the zero dynamics of ZMP control is illustrated in the following considering equations for a lumped mass model and related simulation results for the monoped.

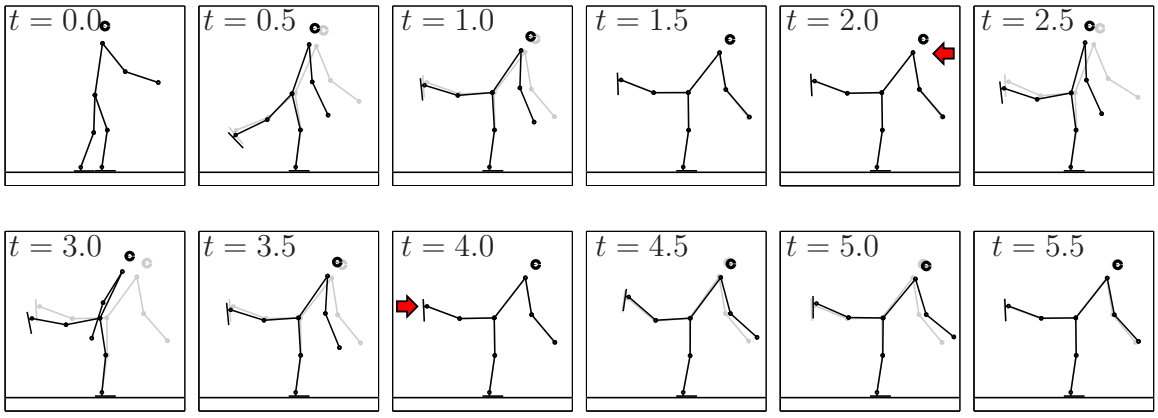

Figure 5.3: Snapshot series for balancing of the legged robot. Comparison of motion without corrective control (grey) and with corrective control (black). Arrows indicate the direction and attack point of the pushing forces at $t = 2.0\,\mathrm{s}$ and $t = 4.0\,\mathrm{s}$. Supplementary video material in [116].

The lumped (concentrated) mass at position $\boldsymbol{r}_{\mathrm{com}}$ of a legged robot and the ZMP $r_{\mathrm{zmp},x}$ are related by a differential equation

$$\ddot{r}_{\mathrm{com},x} = \frac{1}{r_{\mathrm{com},y}}(\ddot{r}_{\mathrm{com},y} + g)(r_{\mathrm{com},x} - r_{\mathrm{zmp},x})$$

that is obtained from (2.23). It can be seen that only if $r_{\mathrm{com},x} = r_{\mathrm{zmp},x}$ together with $\dot{r}_{\mathrm{com},x} = 0$, the acceleration of the center of mass (CoM) $\ddot{r}_{\mathrm{com},x}$ becomes zero and enables an equilibrium position. In all other cases, a constant ZMP $r_{\mathrm{zmp},x}$ different from the CoM $r_{\mathrm{com},x}$ results in acceleration of the CoM that drives the CoM away from the ZMP. It is thus only possible to control the ZMP to be constant if ZMP $r_{\mathrm{zmp},x}$ and CoM projection to ground level $r_{\mathrm{com},x}$ match. For other desired positions of the ZMP the control system will accelerate for all times.

The monoped robot is an example for a lumped mass robot model since it consists of two masses, one in the link and another one in the foot. The foot mass does not contribute to velocities and acceleration of the CoM of the robot since this mass is constantly at rest. For an illustration of the zero dynamics of this robot when the ZMP is controlled to a constant position see the phase portrait in Fig. 5.4. Here, the formulation with relative degree zero is used. The left picture shows the internal dynamics for ZMP control on the left foot edge of the monoped. The right plot shows the phase portrait for control of the ZMP on the right foot edge. The equilibrium point of the control system is the configuration β where the CoM projected to ground level is at the respective foot edge and thus, the ZMP is equal to the x-coordinate of the CoM. This equilibrium point is a saddle point, and most trajectories will never approach this point. Besides, there is another type of equilibrium point that correspond with the pendulum hanging down. Since these equilibrium points are circles, ZMP control at least results in bounded trajectories. For robot control, these trajectories are nevertheless useless because penetration into the ground is not feasible. Separatrices divide the periodic trajectories around the circle fixed points from the remaining phase space.

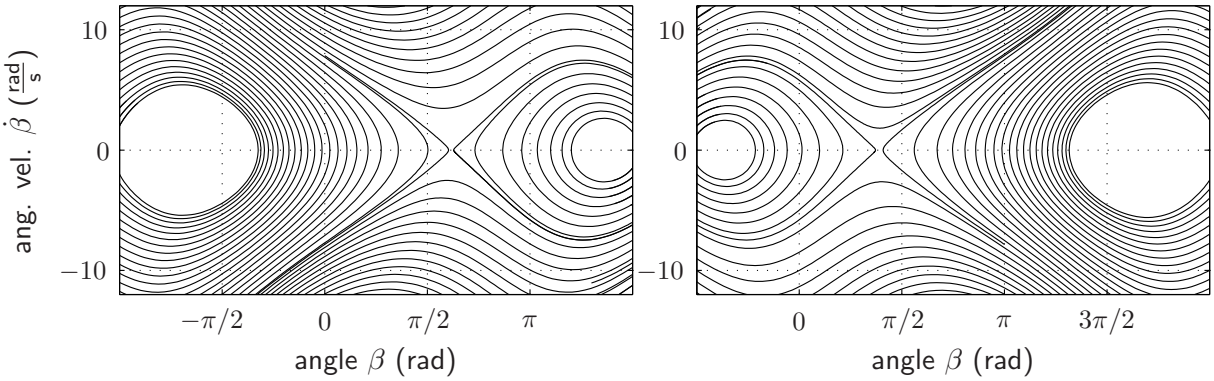

Figure 5.4: Phase portrait of the zero dynamics for monoped ZMP control. Left: ZMP is at the left foot edge. Right: ZMP is at the right foot edge.

Due to the non-minimum phase character, a robot subject to ZMP balance control can only be controlled in a stable way by switching between ZMP controllers and nominal controller. For the monoped robot, although control of the ZMP on the left foot edge and control of the ZMP on the right foot edge are each non-minimum phase and result in unbounded trajectories, switching between the two controllers at appropriate times results in bounded trajectories above ground level. See therefore in Fig. 5.5 on the left side how the two phase portraits from Fig. 5.4 intersect, giving rise to a switching surface $\beta = \pi/2$. On the right side, the resulting bounded trajectories are depicted where the two ZMP controllers alternate.

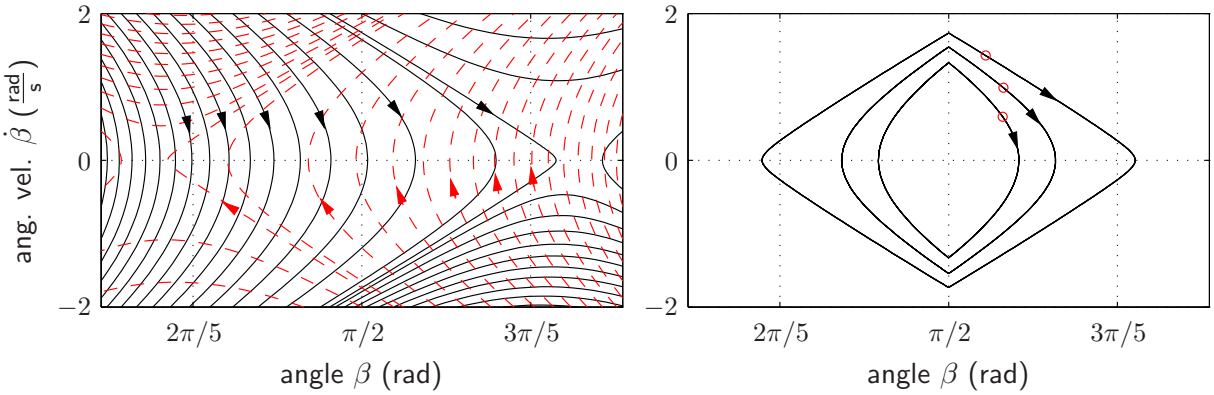

Figure 5.5: Illustration of switching ZMP control. Left: Intersection of the zero dynamics for the two ZMP control modes (solid: left foot edge, dashed: right foot edge). Right: Bounded trajectories that result for switching control with switching surface $\beta = \pi/2$.

But certainly, not every possible disturbance can be compensated for by the corrective controller. Also humans, if they experience a hard push from the front, they cannot balance, but will do a stabilizing step backwards. An investigation of the limitations of ZMP control is necessary as a basis for a comprehensive control concept for legged system stability.

One limitation of the applicability of ZMP control is obvious: For the case of relative degree zero, control is only possible when the sensitivities of the ZMP concerning the accelerations $\boldsymbol{v} = \ddot{\boldsymbol{\beta}}$ are non-zero. The sensitivities $\partial r_{\text{zmp}}/\partial v_j$ can be derived from (5.20), and it is concluded that $\partial r_{\text{zmp}}/\partial v_j = 0$ is equivalent to $a_j = 0$, where a_j is the $j-$th entry of $\boldsymbol{A}$ from (5.10). Thus, if all sensitivities vanish $\partial r_{\text{zmp}}/\partial v_j = 0$, the matrix $\boldsymbol{A}$ becomes singular, and ZMP control is unstable since small changes in the ZMP require arbitrary large accelerations $\boldsymbol{v}$.

Another reason to abort ZMP control in favor of a new strategy is when it becomes obvious that it will not be possible again for the CoM to return to the supported area without violation of the ZMP constraint. This is important for a balancing task in a rest position where disturbances are compensated by ZMP manipulation, whereas for a walking motion this must not be necessary because the CoM is desired to advance in walking direction. The upcoming failure of balance control should be detected as early as possible, and then a controller on a higher level of hierarchy may interact. Therefore, investigations to determine the admissible state space region for ZMP control are necessary. Figure 5.6 depicts the admissible state space region for balance control of the monoped robot. The admissible state space region is bounded by separatrices of the zero dynamics for ZMP control on the left foot edge and ZMP control on the right foot edge. In addition only angles for the actuated arm between 0 and π are feasible. The shape of the admissible state space region is concluded from investigation of the phase portraits when considering valid directions for any point in the state space. The obtained phase space region can be interpreted as follows: For angles of the actuated arm smaller than $\pi/2$, the velocity has to be large enough to swing the arm back to an upright position without too high acceleration that would trigger tilting. But also velocity is not allowed to exceed an upper limit because then necessary deceleration before reaching the upright position results in tilting. The reasoning is analog for angles greater than $\pi/2$. For any points in the determined state space region the CoM can return to the region above the foot-ground contact area with appropriate control. Therefore, it might be necessary to switch between nominal and corrective control signal if the nominal control law alone violates ZMP constraints.

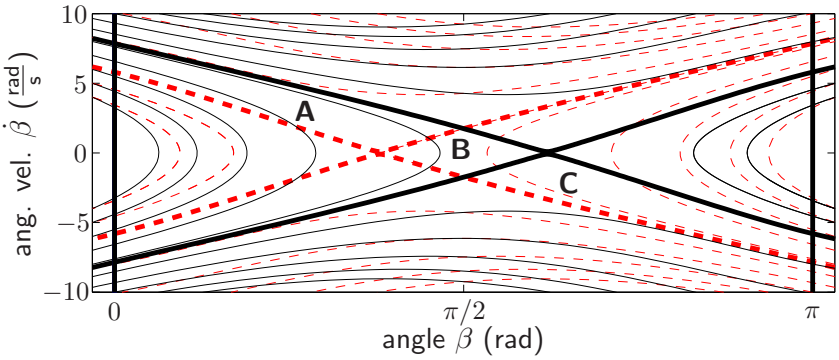

Figure 5.6: Admissible state space region for balance control of the monoped. Configurations in region A, B, and C for β between 0 and π (above ground) that are delimited by the separatrices of the internal dynamics for control on the left foot edge (solid) and control on the right foot edge (dashed) are admissible in the sense that it is possible for the center of mass to return to the supported area while the ZMP never leaves the supported area.

5.4 Summary

Balance control for legged robots is understood as control that avoids underactuation by tilting around foot edges. A common measure for postural balance is the Zero Moment Point (ZMP), and balance control methods rely on manipulation of the ZMP position. The ZMP is a system output with relative degree zero and with non-minimum phase characteristic and is thus challenging to control.

An approach to control the ZMP by invariance control was presented. Therefore, control is switched between a nominal and a corrective controller to keep the ZMP in an admissible set. The admissible set is a subset of the supporting area formed by the convex hull of the foot contact points. The control was discussed for a formulation where the relative degree of the ZMP is zero and for a formulation, after model refinement, where the relative degree of the ZMP is one. Since it is switched between a nominal and a corrective control signal, the complete trajectory is stable although the corrective controller has non-minimum phase character. It was also discussed, when the ZMP control has to be discarded because the sensitivity of the ZMP concerning joint acceleration is too small. The method is illustrated in simulation experiments for a balancing task of a humanoid robot.

The application of invariance control for balance maintainance of legged robots is considered as new concept. The involved formalization of balance control is seen as advantageous for the discussion of stability and limitations of balance control methods.

6 Conclusions and Future Directions

6.1 Concluding Remarks

This thesis provided a theoretical and simulation study on locomotion of legged robots. It is dealt with all necessary steps to realize stable locomotion on legs. This includes modeling, trajectory planning, stability analysis, and control. Especially motion patterns are considered in which the contact situation between feet and ground is variable and therefore consist of a series of different contact situations. In particular the thesis investigates the interaction of underactuated contact situations, where e.g. a foot tilts around a foot edge, with completely actuated contact situations, where a foot has flat contact. The toe and heel roll motion is still rarely considered in legged robot research. Taking into account the variable contact situations during locomotion is seen as an important step towards realization of dynamic legged robot locomotion.

The simulation based study of trajectory planning and control for legged robots relies on a model of the mechanical system. A rigid multi-body formulation for the ground contact was used. Decisive factors were advantages for simulation and availability of notation in minimal coordinates. The model was formulated in a (discrete-continuous) hybrid modeling framework where continuous dynamical descriptions interact with discrete phenomena. Every contact situation of a legged robot is described by equations of motion, and the hybrid model allows to switch between the equations whenever the contact situation changes. Collisions are considered at the times at which transitions between contact situations occur. It has been demonstrated that a hybrid modeling framework is adequate for modeling of legged locomotion. The models of three example robots, a compass gait robot, a monoped robot, and a gymnast robot were presented.

Trajectory planning for motion with variable ground contact is a challenge. One reason is the underlying hybrid dynamical description of the legged system. In addition, planning underactuated motion is difficult in general. And also, the complexity of the trajectory planning problem increases with a higher number of contact situations, and therefore fewer successful realizations of trajectory planning exist in the literature. This thesis presents a simple trajectory planning method to find trajectories for a legged robot also if there are more consecutive contact situations. Trajectories arise from the solution of boundary value problems of the underlying optimal control problems. Therefore, parameter-dependent trajectories have to be provided heuristically. A drawback of this method is that the performance depends on a skillful choice of heuristic trajectories. The advantage, however, is the fast numerical convergence to solutions. Also a parameterized family of trajectories can be obtained that provides a database for motions with different speeds, different step widths, etc. The convergence of the boundary value problem to a solution, however, does

not imply orbital stability of the periodic solutions. Orbital stability has to be shown a posteriori by stability analysis.

If after small arbitrary disturbances of the periodic behavior the system recovers and finds back to the periodic trajectory, the trajectory is said to be orbitally stable. Several methods to investigate orbital stability for hybrid systems are discussed in this thesis. Then, Poincaré maps were used to investigate stability of the trajectories for the legged robot systems. In a Poincaré map analysis the hybrid dynamics is reduced to a discrete map on a cross section to the hybrid orbit. Stability properties of the discrete map are then transfered to stability properties of the hybrid dynamics. For legged robots, the stability analysis divides the set of available trajectories into a stable set and an unstable set. Only the stable trajectories are useful in simulation and experiments. For trajectories with stable support contact phase, it was investigated how the duration of the stable support phase is related to the stability of the periodic motion. It was concluded that stable support is not necessary for stability of the whole trajectory. Nevertheless, a longer fraction of stable support in the periodic motion cycle is beneficial for its stability. Finally, the feasibility of transitions between trajectories with different properties was investigated in simulations for the gymnast robot. A transition is possible if the basin of attraction of the target motion has a nonempty intersection with the primary motion pattern. The possibility to switch between trajectories with different properties increases the dexterity of robots since it yields transition trajectories without additional planning effort. This research field is considered rather young.

The notion of balance of a legged robot is not directly related to the orbital stability of the trajectory. A robot is balanced if there is no rotation around the foot edges. Control algorithms can be applied that monitor the ZMP, which provides a balance measure, and correct when the ZMP leaves an admissible region. The balance control method that was presented in this thesis uses invariance control of the ZMP, which is a novel approach. By switching between a nominal control law and a corrective control law, the ZMP is confined to an admissible region. The method is model-based. A model refinement of the robot dynamics is presented to make the ZMP control problem better tractable for control theory. After refinement, the system output ZMP has relative degree one and is not algebraically related to the input anymore. Before model refinement, it was necessary to decide on the control input at the same time as the measurement of ZMP could be obtained. The presented approach provides a comprehensive control concept for balance control of legged robots.

6.2 Outlook

The long-term goal in legged robot research is to achieve performance comparable to human beings for legged robot systems. This includes dexterity and velocity, but also robustness against disturbances.

One line of research is optimal control for trajectory and controller design. Yet, optimal control allows a versatile formulation of the trajectory planning task. The numerical solution, however, is often not to be found because of unsuitable initial values, too slow

convergence, or an inappropriate choice of the numerical method. Legged robot trajectory planning is presumed to profit decisively from advancements of numerical optimal control algorithms for hybrid systems.

Advances in hybrid system theory are seen to be beneficial for improvement towards dexterous locomotion. The crucial point in legged robot control is to avoid that the robot falls. Stability control should thus not be limited to correction of small disturbances where the correction does not make it necessary to replan the series of contact situations. Sometimes falling can only be avoided if the planned motion is discarded for some time and a correction that is different from the planned motion is performed. A correspondence in human locomotion are stabilizing steps forward or backward or acceleration of walking after a push from behind. The hybrid system background for these control tasks are e.g. improved reachability analysis, research on regions of attraction, stability of switching control laws, etc.

With respect to experimental implementation of the control concepts even more detailed models will be necessary. The presented modeling framework presents a basis for the extension to three dimensions and the consideration of additional contact situations. The validity of the assumptions on the collision law has to be checked in experimental studies, and collision laws possibly have to be modified. The experimental implementation is essential for evaluation of the theoretical results. New questions will arise in hardware design, as the appropriate choice of components and sensors that enable application of the theoretical results. And also the experimental validation will suggest modifications in modeling and control.

The more legged robots enter human environments, the more it is necessary that robots are capable of appropriate interaction with the environment. On a basic level, self-adaptation of the robot to non-expected conditions, as ground texture or its own variable mass is desired. In advanced scenarios, it is required that the robot aptly reacts in robot-human interplay. These research areas will lead far beyond model-based motion planning and control.

Appendix Details of Hybrid Models

A.1 Model of the Compass Gait Robot

The present section provides details of the derivation of the hybrid model for the compass gait robot that was introduced in Sec. 2.3.1. Analytic expressions are specified for the equations of motion of different contact situations and for the contact forces and moments.

A.1.1 Geometry

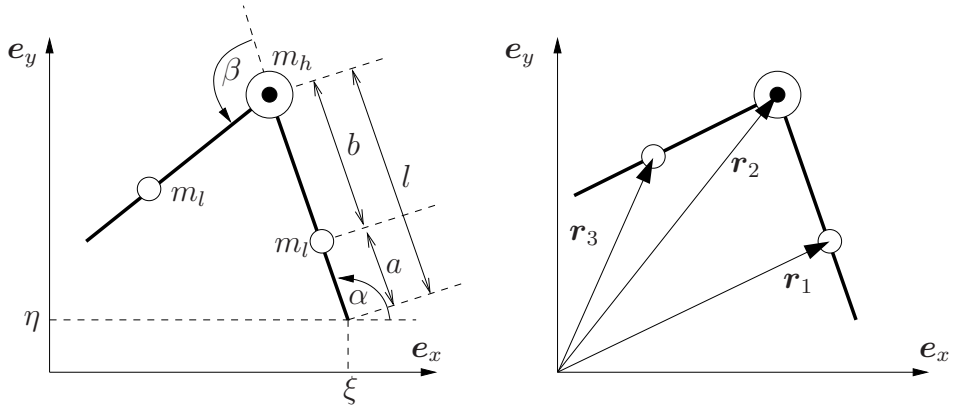

Figure A.1: Compass gait robot. Illustration of geometry, masses, and coordinate system. Leg masses are m_l, hip mass is m_h. Foot length is l, a and b are distances of leg mass centers from foot or hip. Vectors $\boldsymbol{r}_1$, $\boldsymbol{r}_2$, and $\boldsymbol{r}_3$ describe positions of mass points.

Figure A.1 is renewed from Sec. 2. For derivation of the kinetic and potential energy which are necessary to determine Lagrange functions according to (2.6) the position vectors $\boldsymbol{r}_i$ of all mass points have to be specified. Masses are assumed to be concentrated in the mass centers of the legs and in the hip. The three positions for the masses are thus obtained from geometrical considerations:

$$\text{Leg 1}: \qquad \boldsymbol{r}_1 = \begin{pmatrix} \xi + a\cos\alpha \\ \eta + a\sin\alpha \end{pmatrix} \tag{A.1}$$

$$\text{Hip}: \qquad \boldsymbol{r}_2 = \begin{pmatrix} \xi + l\cos\alpha \\ \eta + l\sin\alpha \end{pmatrix} \tag{A.2}$$

$$\text{Leg 2}: \qquad \boldsymbol{r}_3 = \begin{pmatrix} \xi + l\cos\alpha + b\cos(\alpha+\beta) \\ \eta + l\sin\alpha + b\sin(\alpha+\beta) \end{pmatrix} \tag{A.3}$$

The positions of the masses depend on the generalized coordinates $\boldsymbol{q} = (\xi,\ \eta,\ \alpha,\ \beta)^T$ that form a minimal set of variables to describe the robot posture.

A.1.2 Equations of Motion

The evaluation of Euler-Lagrange equations (2.8) yields the equations of motion for the compass gait robot (2.25). At first, it is assumed that the robot is not in contact with the environment ($x_d = 0$). The corresponding equations combine a symmetric mass matrix $\boldsymbol{M}_0$ and a vector of coupling forces and gravitational effects $\boldsymbol{n}_0$:

$$\underbrace{\begin{pmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{12} & m_{22} & m_{23} & m_{24} \\ m_{13} & m_{23} & m_{33} & m_{34} \\ m_{14} & m_{24} & m_{34} & m_{44} \end{pmatrix}}_{\boldsymbol{M}_0} \ddot{\boldsymbol{q}} + \underbrace{\begin{pmatrix} n_1 \\ n_2 \\ n_3 \\ n_4 \end{pmatrix}}_{\boldsymbol{n}_0} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ u \end{pmatrix} \tag{A.4}$$

with

$$\begin{aligned}
m_{11} &= m_h + 2\,m_l \\
m_{12} &= 0 \\
m_{13} &= -m_h\,l\sin\alpha - m_l\,[a\sin\alpha + l\sin\alpha + b\sin(\alpha+\beta)] \\
m_{14} &= -m_l\,b\sin(\alpha+\beta) \\
m_{22} &= m_h + 2\,m_l \\
m_{23} &= m_h\,l\cos\alpha + m_l\,[a\cos\alpha + l\cos\alpha + b\cos(\alpha+\beta)] \\
m_{24} &= m_l\,b\cos(\alpha+\beta) \\
m_{33} &= m_h\,l^2 + m_l\left[a^2 + b^2 + l^2 + 2\,b\,l\cos\beta\right] \\
m_{34} &= m_l\,b\,[b + l\cos\beta] \\
m_{44} &= m_l\,b^2 \\
n_1 &= -\,[m_h\,l + m_l\,a + m_l\,l]\cos\alpha\;\dot\alpha^2 - m_l\,b\cos(\alpha+\beta)\,(\dot\alpha+\dot\beta)^2 \\
n_2 &= -\,[m_h\,l + m_l\,a + m_l\,l]\sin\alpha\;\dot\alpha^2 - m_l\,b\sin(\alpha+\beta)\,(\dot\alpha+\dot\beta)^2 + m_h\,g + 2m_l\,g \\
n_3 &= -m_l\,b\,l\sin\beta\;(\dot\beta^2 + \dot\alpha\dot\beta) + [m_h\,l + m_l\,a + m_l\,l]\,g\cos\alpha + m_l\,b\,g\cos(\alpha+\beta) \\
n_4 &= m_l\,b\,l\sin\beta\;\dot\alpha^2 + m_l\,b\,g\cos(\alpha+\beta)
\end{aligned}$$

For the equations of motion for contact with the reference foot ($x_d = 1$) constraints are introduced that reduce the degrees of freedom of the dynamics, see Sec. 2.3.1. The equations of motion from (2.28) are denoted:

$$\underbrace{\begin{pmatrix} m_{11} & m_{12} \\ m_{12} & m_{22} \end{pmatrix}}_{\boldsymbol{M}_1} \ddot{\boldsymbol{q}} + \underbrace{\begin{pmatrix} n_1 \\ n_2 \end{pmatrix}}_{\boldsymbol{n}_1} = \begin{pmatrix} 0 \\ u \end{pmatrix} \tag{A.5}$$

with

$$\begin{aligned}
m_{11} &= m_h\,l^2 + m_l\left[a^2 + b^2 + l^2 + 2\,b\,l\cos\beta\right] \\
m_{12} &= m_l\,b\,[b + l\cos\beta] \\
m_{22} &= m_l\,b^2 \\
n_1 &= -m_l\,b\,l\sin\beta\;(\dot\beta^2 + \dot\alpha\dot\beta) + [m_h\,l + m_l\,a + m_l\,l]\,g\cos\alpha + m_l\,b\,g\cos(\alpha+\beta) \\
n_2 &= m_l\,b\,l\sin\beta\;\dot\alpha^2 + m_l\,b\,g\cos(\alpha+\beta)
\end{aligned}$$

A.1.3 Contact Forces and Moments

If a foot of the compass gait robot is in contact with the ground ($x_d = 1$, $x_d = -1$) contact forces act between ground and foot. The horizontal contact force R_x and the vertical contact force R_y, needed in (2.32) are:

$$\begin{aligned} R_x =& -\left[m_h\, l + m_l\, a + m_l\, l\right] \cos\alpha\ \dot{\alpha}^2 - m_l\, b \cos(\alpha+\beta)\,(\dot{\alpha}+\dot{\beta})^2 \\ & - \left[(m_h\, l - m_l\, a - m_l\, l)\sin\alpha - m_l\, b \sin(\alpha+\beta)\right] \ddot{\alpha} - m_l\, b \sin(\alpha+\beta)\ddot{\beta} \\ R_y =& \, m_h\, g + 2\, m_l\, g - \left[m_h\, l + m_l\, a + m_l\, l\right] \sin\alpha\ \dot{\alpha}^2 - m_l\, b \sin(\alpha+\beta)\,(\dot{\alpha}+\dot{\beta})^2 \\ & + \left[(m_h\, l + m\, a + m_l\, l)\cos\alpha + m_l\, b \cos(\alpha+\beta)\right] \ddot{\alpha} + m_l\, d \cos(\alpha+\beta)\ddot{\beta} \end{aligned} \tag{A.6}$$

Note that there is no contact moment, since the foot rotates freely around the contact point. The contact forces are essential to detect detaching of the foot in simulation and trajectory planning. In the present notation, the contact forces depend on position $\boldsymbol{q}$, velocities $\dot{\boldsymbol{q}}$, and accelerations $\ddot{\boldsymbol{q}}$. If the accelerations $\ddot{\boldsymbol{q}}$ are replaced using the equations of motion (A.5), the contact forces are calculated from positions $\boldsymbol{q}$, velocities $\dot{\boldsymbol{q}}$, and input torques $\boldsymbol{u}$.

A.2 Model of the Monoped Robot

In the following, analytic expressions are specified for some elements of the hybrid model of the monoped robot from Sec. 2.3.2.

A.2.1 Geometry

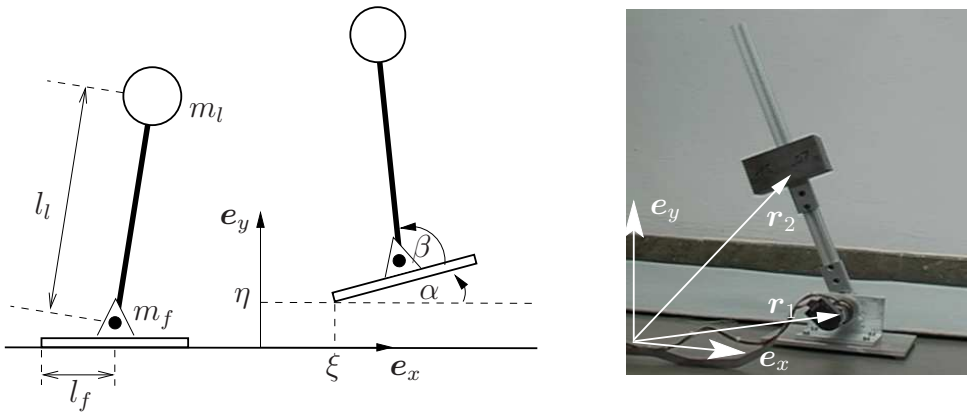

Figure A.2: Monoped robot. Illustration of geometry, masses, and coordinate system. Masses are denoted m_f and m_l. Their position vectors are denoted $\boldsymbol{r}_1$ and $\boldsymbol{r}_2$. Distances are denoted l_l, l_f, and h_f.

The position vector of the mass center of foot and actuated link can be denoted in terms of the generalized coordinate vector $\boldsymbol{q} = (\xi,\ \eta,\ \alpha,\ \beta)^T$:

$$\text{Foot}: \quad \boldsymbol{r}_1 = \begin{pmatrix} \xi + l_f \cos\alpha + h_{cm,f} \sin\alpha \\ \eta + l_f \sin\alpha + h_{cm,f} \cos\alpha \end{pmatrix} \tag{A.7}$$

$$\text{Link}: \quad \boldsymbol{r}_2 = \begin{pmatrix} \xi + l_f \cos\alpha + h_f \sin\alpha + l_l \cos(\alpha+\beta) \\ \eta + l \sin\alpha + h_f \cos\alpha + l_l \sin(\alpha+\beta) \end{pmatrix} \tag{A.8}$$

The position vectors and their derivatives are necessary to obtain the Lagrange function (2.34).

A.2.2 Equations of Motion

From Euler-Lagrange equations (2.8) applied on the Lagrange function (2.6), the equations of motion (2.35) for the monoped robot that is not subject to constraints ($x_d = 0$) are obtained as follows:

$$\underbrace{\begin{pmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{12} & m_{22} & m_{23} & m_{24} \\ m_{13} & m_{23} & m_{33} & m_{34} \\ m_{14} & m_{24} & m_{34} & m_{44} \end{pmatrix}}_{\boldsymbol{M}_0} \ddot{\boldsymbol{q}} + \underbrace{\begin{pmatrix} n_1 \\ n_2 \\ n_3 \\ n_4 \end{pmatrix}}_{\boldsymbol{n}_0} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ u \end{pmatrix} \tag{A.9}$$

with

$$\begin{aligned}
m_{11} &= m_f + m_l \\
m_{12} &= 0 \\
m_{13} &= -m_f \left[l_f \sin\alpha + h_{cm,f} \cos\alpha\right] - m_l \left[l_f \sin\alpha + h_f \cos\alpha + l_l \sin(\alpha+\beta)\right] \\
m_{14} &= -m_l\, l_l \sin(\alpha+\beta) \\
m_{22} &= m_f + m_l \\
m_{23} &= m_f \left[l_f \cos\alpha - h_{cm,f} \sin\alpha\right] - m_l \left[h_f \sin\alpha - l_f \cos\alpha - l_l \cos(\alpha+\beta)\right] \\
m_{24} &= m_l\, l_l \cos(\alpha+\beta) \\
m_{33} &= I_{f,z} + I_{l,z} + m_f \left[l_f^2 + h_{cm,f}^2\right] + m_l \left[l_f^2 + h_f^2 + l_l^2 + 2\, l_f\, l_l \cos\beta + 2\, h_f\, l_l \sin\beta\right] \\
m_{34} &= I_{l,z} + l_l m_l \left[l_l + l_f \cos\beta + h_f \sin\beta\right] \\
m_{44} &= I_{l,z} + m_l\, l_l^2 \\
n_1 &= m_l \left[h_f \sin\alpha\ \dot\alpha^2 - l_f \cos\alpha\ \dot\alpha^2 - l_l \cos(\alpha+\beta)(\dot\alpha+\dot\beta)^2\right] - m_f \left[l_f \cos\alpha - h_{cm,f} \sin\alpha\right]\ \dot\alpha^2 \\
n_2 &= \left[m_f + m_l\right] g \\
&\quad - m_f \left[l_f \sin\alpha + h_{cm,f} \cos\alpha\right]\ \dot\alpha^2 - m_l \left[l_f \sin\alpha\ \dot\alpha^2 + h_f \cos\alpha\ \dot\alpha^2 + l_l \sin(\alpha+\beta)\ (\dot\alpha+\dot\beta)^2\right] \\
n_3 &= m_f\, g \left[l_f \cos\alpha - h_{cm,f} \sin\alpha\right] - m_l\, g \left[h_f \sin\alpha - l_f \cos\alpha - l_l \cos(\alpha+\beta)\right] \\
&\quad - m_l\, l_l \left[h_f \cos\beta\ \dot\alpha^2 + l_f \sin\beta\ (\dot\alpha+\dot\beta)^2 - l_f \sin\beta\ \dot\alpha^2 - h_f \cos\beta\ (\dot\alpha+\dot\beta)^2\right] \\
n_4 &= m_l\, g\, l_l \cos(\alpha+\beta) + m_l\, l_l \left[l_f \sin\beta - h_f \cos\beta\right]\ \dot\alpha^2
\end{aligned}$$

Constraints are introduced to derive equations of motion (2.38) for the monoped robot, when it is in contact with one foot edge only ($x_d = 2, 3$). The resulting equations of motion are:

$$\underbrace{\begin{pmatrix} m_{11} & m_{12} \\ m_{12} & m_{22} \end{pmatrix}}_{\boldsymbol{M}_2} \ddot{\boldsymbol{q}} + \underbrace{\begin{pmatrix} n_1 \\ n_2 \end{pmatrix}}_{\boldsymbol{n}_2} = \begin{pmatrix} 0 \\ u \end{pmatrix} \tag{A.10}$$

with

$$\begin{aligned}
m_{11} &= I_{f3} + I_{l3} + m_f \left[h_{cm,f}^2 + l_f^2\right] + m_l \left[h_f^2 + l_f^2 + l_l^2 + 2\, h_f\, l_l \sin\beta + 2\, l_f\, l_l \cos\beta\right] \\
m_{12} &= I_{l3} + l_l\, m_l \left[l_l + h_f \sin\beta + l_f \cos\beta\right] \\
m_{22} &= I_{l3} + m_l\, l_l^2 \\
n_1 &= l_l\, m_l \left[h_f \cos\beta - l_f \sin\beta\right] (2\dot{\alpha}\dot{\beta} + \dot{\beta}^2) \\
&\quad - m_f\, g \left[h_{mc,f} \sin\alpha - l_f \cos\alpha\right] - m_l\, g \left[h_f \sin\alpha - l_f \cos\alpha - l_l \cos(\alpha + \beta)\right] \\
n_2 &= -\, l_l\, m_l \left[h_f \cos\beta - l_f \sin\beta\right] \dot{\alpha}^2 + m_l\, g\, l_l \cos(\alpha + \beta)
\end{aligned}$$

A.2.3 Contact Forces and Moments

Contact forces in vertical and horizontal direction R_y and R_x act whenever the foot is in a tilted contact phase ($x_d = 2, 3$). The analytic expression are:

$$\begin{aligned}
R_x = &\left[-m_f \left(l_f \sin\alpha + h_{cm,f} \cos\alpha\right) - m_l \left(l_f \sin\alpha + h_f \cos\alpha + l_l \sin(\alpha + \beta)\right)\right]\ \ddot{\alpha} \\
&- m_l\, l_l \sin(\alpha + \beta)\ \ddot{\beta} \\
&+ m_l \left[h_f \sin\alpha\ \dot{\alpha}^2 - l_f \cos\alpha\ \dot{\alpha}^2 - l_l \cos(\alpha + \beta)(\dot{\alpha} + \dot{\beta})^2\right] \\
&- m_f \left[l_f \cos\alpha - h_{cm,f} \sin\alpha\right]\ \dot{\alpha}^2 \\
R_y = &\left[m_f \left(l_f \cos\alpha - h_{cm,f} \sin\alpha\right) - m_l \left(h_f \sin\alpha - l_f \cos\alpha - l_l \cos(\alpha + \beta)\right)\right]\ \ddot{\alpha} \\
&+ m_l\, l_l \cos(\alpha + \beta)\ \ddot{\beta} \\
&+ \left[m_f + m_l\right] g \\
&- m_f \left[l_f \sin\alpha + h_{cm,f} \cos\alpha\right]\ \dot{\alpha}^2 \\
&- m_l \left[l_f \sin\alpha\ \dot{\alpha}^2 + h_f \cos\alpha\ \dot{\alpha}^2 + l_l \sin(\alpha + \beta)\ (\dot{\alpha} + \dot{\beta})^2\right]
\end{aligned}$$

For a stable support contact phase ($x_d = 3$), there is also a contact moment T_z between foot and ground. The contact moment T_z and the vertical contact force R_y are basis for calculation of the Zero Moment Point (2.24) that is used to monitor the onset of tilting.

$$\begin{aligned}
R_x &= -\, m_l\, l_l \sin\beta\ \ddot{\beta} - m_l\, l_l \cos\beta\ \dot{\beta}^2 \\
R_y &= m_l\, l_l \cos\beta\ \ddot{\beta} + (m_f + m_l) g - m_l\, l_l \sin\beta\ \dot{\beta}^2 \\
T_z &= \left[I_{l,z} + l_l\, m_l (l_l + l_f \cos\beta + h_f \sin\beta)\right]\ \ddot{\beta} - m_l\, l_l \left[l_f \sin\beta - h_f \cos\beta\right]\ \dot{\beta}^2 \\
&\quad + m_f\, g\, l_f + m_l\, g \left[l_f + l_l \cos\beta\right]
\end{aligned}$$

Bibliography

[1] J.V. Albro and J.E. Bobrow. Motion generation for a tumbling robot using a general contact model. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 3270–3275, New Orleans, LA, April 2004.

[2] R. Alur, T.A. Henziger, and E.D. Sontag, editors. *Hybrid Systems III*. Springer, Berlin, 1996.

[3] P. Antsaklis, editor. *Hybrid Systems IV*. Springer, Berlin, 1997.

[4] P. Antsaklis, W. Kohn, M. Lemmon, A. Nerode, and S. Sastry, editors. *Hybrid Systems V*. Springer, Berlin, 1999.

[5] P. Antsaklis, W. Kohn, A. Nerode, and S. Sastry. *Hybrid Systems II*. Springer, Berlin, 1995.

[6] P.J. Antsaklis, J.A. Stiver, and M.D. Lemmon. Hybrid system modeling and autonomous control systems. In *Hybrid Systems*, pages 366–392, London, UK, 1993. Springer-Verlag.

[7] E. Asarin, T. Dang, G. Frehse, A. Girard, C. Le Guernic, and O. Maler. Recent progress in continuous and hybrid reachability analysis. In *Proceedings of the IEEE International Symposium on Computer-Aided Control Systems Design*, Munich, Germany, 2006.

[8] K. Astrom (Chair). Session on inverted pendulum control. Proceedings of the IFAC World Congress, Prague, Czech Republic, 2005.

[9] Autolev, Online Dynamics, Inc., www.autolev.com.

[10] M. Bachmayer. *Modellierung, Trajektorienplanung und Regelung eines zweibeinigen, hybriden Roboters mit ballistischen Bewegungsphasen*. Master's thesis, Technische Universität München, 2005.

[11] A. Back, J. Guckenheimer, and M. Myers. A dynamical simulation facility for hybrid systems. In *Hybrid Systems*, pages 255–267, London, UK, 1993. Springer-Verlag.

[12] A. Balluchi, L. Benvenuti, M.D. Di Benedetto, and A.L. Sangiovanni-Vincentelli. Design of observers for hybrid systems. In *Proceedings of the International Workshop on Hybrid Systems: Computation and Control*, pages 76–89, London, UK, 2002. Springer-Verlag.

[13] A. Balluchi, L. Benvenuti, M.D. di Benedetto, C. Pinello, and A.L. Sangiovanni-Vincentelli. Automotive engine control and hybrid systems: challenges and opportunities. *Proceedings of the IEEE*, 88(7):888–912, 2000.

[14] M.D. Berkemeier and R.S. Fearing. Sliding and hopping gaits for the underactuated acrobot. *IEEE Transactions on Robotics and Automation*, 14(4):629–634, 1998.

[15] G. Bhatia and M.W. Spong. Further results on the control of the compass gait biped. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 1933 – 1938, Las Vegas, Nevada, 2003.

[16] M. Blankenburg. *Optimale Trajektorien für zweibeinige Roboter mit wechselnden Bodenkontaktzuständen*. Technical report, Technische Universität München, 2005.

[17] J. Blaya and H. Herr. Adaptive control of a variable-impedance ankle-foot orthosis to assist drop-foot gait. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 12(1):24–31, 2004.

[18] D. Botturi and P. Fiorini. Optimal control for autonomous task execution. In *Proceedings of the IEEE Conference on Decision and Control (CDC)*, pages 3525–3530, Seville, Spain, 2005.

[19] M.S. Branicky. A unified framework for hybrid control. In *Proceedings of the IEEE Conference on Decision and Control (CDC)*, pages 4228–4234, Lake Buena Vista, FL, 1994.

[20] M.S. Branicky. Multiple lyapunov functions and other analysis tools for switched and hybrid systems. *IEEE Transactions on Automatic Control*, 43(4):475–482, 1998.

[21] R.W. Brockett. Hybrid models for motion control systems. In H.L. Trentelman and J.C. Willems, editors, *Essays on Control: Perspectives in the Theory and its Applications*, pages 29–54. Birkhauser, 1993.

[22] B. Brogliato. *Nonsmooth Mechanics*. Springer, 1999.

[23] T. Buschmann, S. Lohmeier, H. Ulbrich, and F. Pfeiffer. Dynamics simulation for a biped robot: Modeling and experimental verification. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 2673–2678, Orlando, FL, 2006.

[24] M. Buss. Hybrid control of mechatronic systems. *Journal on Systems, Control and Information; Institute of Systems, Control and Information Engineers, ISCIE*, 46(3):126–137, 2002.

[25] M. Buss. *Methoden zur Regelung Hybrider Dynamischer Systeme*. VDI Verlag, Reihe 8, Nr. 970, 2002.

[26] M. Buss, M. Glocker, M. Hardt, O. von Stryk, R. Bulirsch, and G. Schmidt. Nonlinear hybrid dynamical systems: Modeling, optimal control, and applications. In S. Engell, G. Frehse, and E. Schnieder, editors, *Modeling, Analysis and Design of Hybrid Systems. Lecture Notes in Control and Information Sciences (LNCIS)*, pages 311–335. Springer, Berlin, Heidelberg, 2002.

[27] M. Buss, M. Hardt, J. Kiener, M. Sobotka, M. Stelzer, O. von Stryk, and D. Wollherr. Towards an autonomous, humanoid, and dynamically walking robot: Modeling, optimal trajectory planning, hardware architecture, and experiments. In *Proceedings of the IEEE/RAS International Conference on Humanoid Robots (HUMANOIDS)*, Munich, Karlsruhe, Germany, 2003.

[28] M. Buss, O. von Stryk, R. Bulirsch, and G. Schmidt. Towards hybrid optimal control. *at-Automatisierungstechnik*, 48(9):448–459, 2000.

[29] G.A. Cavagna and M. Kaneko. Mechanical work and efficiency in level walking and running. *J Physiol*, 268(2):467–481, 1977.

[30] J. Chestnutt, M. Lau, K.M. Cheung, J. Kuffner, J.K. Hodgins, and T. Kanade. Footstep planning for the honda ASIMO humanoid. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 631–636, Barcelona, Spain, 2005.

[31] C. Chevallereau, G. Abba, Y. Aoustin, F. Plestan, E.R. Westervelt, C. Canudas de Wit, and J.W. Grizzle. RABBIT: A testbed for advanced control theory. *IEEE Control Systems Magazine*, 23(5):57–79, 2003.

[32] C. Chevallereau, E.R. Westervelt, and J.W. Grizzle. Asymptotically stable running for a five-link, four-actuator, planar bipedal robot. *International Journal of Robotics Research*, 24(6):431–464, 2005.

[33] C. Ching-Ping and B. Hannaford. Measurement and modeling of McKibben pneumatic artificial muscles. *IEEE Transactions on Robotics and Automation*, 12(1):90–102, 1996.

[34] J.H. Choi and J.W. Grizzle. Planar bipedal walking with foot rotation. In *Proceedings of the American Control Conference (ACC)*, pages 4909–4916, Portland, OR, 2005.

[35] S. Collins, A. Ruina, R. Tedrake, and M. Wisse. Efficient bipedal robots based on passive-dynamic walkers. *Science*, 307:1082–1085, 2005.

[36] S.H. Collins and A. Ruina. A bipedal robot with efficient and human-like gait. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 1995–2000, Barcelona, Spain, 2005.

[37] S. Coombes and A.H. Osbaldestin. Period-adding bifurcations and chaos in a periodically stimulated excitable neural relaxation oscillator. *Phys. Rev. E*, 62(3):4057–4066, 2000.

[38] J. Denk. *Optimierungsbasierte Berechnung von Schrittprimitiven und Schrittsequenzen für perzeptionsgeführte zweibeinige Roboter*. PhD thesis, Technische Universität München, 2004.

[39] J. Denk and G. Schmidt. Synthesis of walking primitive databases for biped robots in 3D-environments. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 1343–1349, Taipei, Taiwan, 2003.

[40] D. Djoudi and C. Chevallereau. Stability analysis of a walk of a biped with control of the ZMP. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1801–1807, Edmonton, Canada, 2005.

[41] V. Duindam and S. Stramigioli. Port-based control of a compass-gait bipedal robot. In *Proceedings of the IFAC World Congress*, Prague, Czech Republic, July 2005.

[42] S. Engell, G. Frehse, and E. Schnieder, editors. *Modelling, Analysis, and Design of Hybrid Systems*. Springer, Berlin, Heidelberg, 2002.

[43] S. Engell, S. Kowalewski, C. Schulz, and O. Stursberg. Continuous-discrete interactions in chemical processing plants. *Proceedings of the IEEE*, 88(7):1050–1068, 2000.

[44] S. Engell and O. Stursberg. Hybrid control techniques for the design of industrial controllers. In *Proceedings of the IEEE Conference on Decision and Control (CDC)*, pages 5612–5617, Seville, Spain, 2006.

[45] Y. Fujimoto. Trajectory generation of biped running robot with minimum energy consumption. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 3803–3808, New Orleans, LA, April 2004.

[46] M. Garcia, A. Chatterjee, A. Ruina, and M. Coleman. The simplest walking model: stability, complexity, and scaling. *J Biomech Eng.*, 120(2):281–288, 1998.

[47] H. Geyer, A. Seyfarth, and R. Blickhan. Spring-mass running: simple approximate solution and application to gait stability. *J Theor Biol*, 232(3):315–328, 2005.

[48] C. Glocker. *Dynamik von Starrkörpersystemen mit Reibung und Stößen*. PhD thesis, Technische Universität München, 1995.

[49] H. Goldstein. *Classical Mechanics*. Addison Wesley, 2003.

[50] G.H. Golub and C.F. van Loan. *Matrix Computations*. The John Hopkins University Press, 1989.

[51] A. Goswami. Postural stability of biped robots and the foot rotation indicator (FRI) point. *International Journal of Robotics Research*, 18(6):523–533, 1999.

[52] A. Goswami, B. Espiau, and A. Keramane. Limit cycles and their stability in a passive bipedal gait. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 246–251, Minneapolis, Minnesota, 1996.

[53] A. Goswami, B. Espiau, and A. Keramane. Limit cycles in a passive compass gait biped and passivity-mimicking control laws. *Journal of Autonomous Robots*, 4(3):273–286, 1997.

[54] A. Goswami, B. Thuilot, and B. Espiau. A study of the passive gait of a compass-like biped robot: symmetry and chaos. *International Journal of Robotics Research*, 17(12):1282–1301, 1998.

[55] J.W. Grizzle, G. Abba, and F. Plestan. Asymptotically stable walking for biped robots: Analysis via system with impulse effects. *IEEE Transactions on Automatic Control*, 46(1):51–64, 2001.

[56] R.L. Grossman, A. Nerode, A.P. Ravn, and H. Rischel. *Hybrid Systems*. Springer, Berlin, 1993.

[57] J. Guckenheimer and P. Holmes. *Nonlinear Oscillations, Dynamical Systems, and Bifurcations of Vector Fields*. Springer-Verlag, New York, 1993.

[58] M. Hardt and O. von Stryk. Increasing stability in dynamic gaits using numerical optimization. In *Proceedings of the IFAC World Congress*, pages 1636–1641, Barcelona, Spain, 2002.

[59] K. Hirai, M. Hirose, Y. Haikawa, and T. Takenaka. The development of HONDA humanoid robot. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 1321–1326, Leuven, Belgium, 1998.

[60] M. Hirsch and S. Smale. *Differential Equations, Dynamical Systems, and Linear Algebra*. Academic Press, Inc., 1974.

[61] I.A. Hiskens. Stability of hybrid system limit cycles: Application to the compass gait biped robot. In *Proceedings of the IEEE Conference on Decision and Control (CDC)*, pages 774–779, Orlando, Florida, 2001.

[62] I.A. Hiskens and M.A. Pai. Trajectory sensitivity analysis of hybrid system. *IEEE Transactions on Circuits and Systems I*, 47(2):204–220, 2000.

[63] Honda. Asimo. http://world.honda.com/ASIMO/.

[64] C.S. Hsu. *Cell-to-cell mapping*. Springer, New York, 1987.

[65] J. Hu, J. Lygeros, and S. Sastry. Towars a theory of stochastic hybrid systems. In *Proceedings of the International Workshop on Hybrid Systems: Computation and Control (HSCC)*, pages 160–173, London, UK, 2000. Springer-Verlag.

[66] Q. Huang, K. Kaneko, K. Yokoi, S. Kajita, T. Kotoku, N. Koyachi, H. Arai, N. Imamura, K. Komoriya, and K. Tanie. Balance control of a biped robot combining off-line pattern with real-time modification. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 3346–3352, San Francisco, CA, April 2000.

[67] Q. Huang, Y. Nakamura, and T. Inamura. Humanoids walk with feedforward dynamic pattern and feedback sensory reflection. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 4220–4225, Seoul, Korea, 2001.

[68] Q. Huang, K. Yokoi, S. Kajita, K. Kaneko, H. Arai, N. Koyachi, and K. Tanie. Planning walking patterns for a biped robot. *IEEE Transactions on Robotics and Automation*, 17(3):280–289, 2001.

[69] Y. Hurmuzlu. Dynamics of bipedal gait: Part i objective functions and the contact event of planar five-link biped. *Journal of Applied Mechanics*, 60(2):331–337, 1993.

[70] Y. Hurmuzlu. Dynamics of bipedal gait: Part ii stability analysis of a planar five-link biped. *Journal of Applied Mechanics*, 60(2):337–343, 1993.

[71] Y. Hurmuzlu and T.-H. Chang. Rigid body collisions of a special class of planar kinematic chains. *IEEE Transactions on Systems, Man and Cybernetics*, 22(5):964–971, 1992.

[72] Y. Hurmuzlu, F. Genot, and B. Brogliato. Modeling, stability and control of biped robots–a general framework. *Automatica*, 40(10):1647–1664, 2004.

[73] A. Isidori. *Nonlinear Control Systems*. Springer-Verlag, Berlin, 3 edition, 1995.

[74] C. Iung and P. Riedinger. Optimal control in hybrid systems. In *Preprints of the IFAC Conference on Analysis and Design of Hybrid Systems*, pages 4–5, Alghero, Italy, 2006.

[75] K. Johansson, J. Lygeros, S. Simić, and J. Zhang. Dynamical properties of hybrid automata. *IEEE Transactions on Automatic Control*, 48(1):2–17, 2003.

[76] Johnnie Humanoid Robot. AMM, Technische Universität München. www.amm.mw.tu-muenchen.de, 1998–2006.

[77] S. Kajita, F. Kanehiro, K. Kaneko, K. Yokoi, and K. Hirukawa. The 3D linear inverted pendulum mode: A simple modeling for a biped walking pattern generation. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 239–246, Maui, Hawaii, November 2001.

[78] K. Kaneko, F. Kanehiro, S. Kajita, H. Hirukawa, T. Kawasaki, M. Hirata, K. Akachi, and T. Isozumi. Humanoid robot HRP-2. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 1083–1090, New Orleans, LA, April 2004.

[79] H.K. Khalil. *Nonlinear Systems*. Pearson Education, 2000.

[80] N. Khraief, N.K. M'Sirdi, and M.W. Spong. Nearly passive dynamic walking of a biped robot. In *Proceedings of the European Control Conference (ECC)*, Cambridge, UK, 2003.

[81] K. Kondak and G. Hommel. Control algorithm for stable walking of biped robots. In *Proceedings of the International Conference on Climbing and Walking Robots (CLAWAR)*, pages 119–126, Catania, Italy, 2003.

[82] K. Kondak and G. Hommel. Control and online computation of stable movement for biped robots. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 874–879, Las Vegas, Nevada, 2003.

[83] A.D. Kuo. Stabilization of lateral motion in passive dynamic walking. *International Journal of Robotics Research*, 18(9):917–913, 1999.

[84] B. Langhof. *Modellierung von Robotern mit variablem Umgebungskontakt durch Complementarity Systeme*. Master's thesis, Technische Universität München, 2005.

[85] K. Löffler, M. Gienger, and F. Pfeiffer. Sensor and control design of a dynamically stable biped robot. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 484–490, Taiwan, 2003.

[86] D.W. Marhefka and D.E. Orin. Simulation of contact using a nonlinear damping model. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 1662–1668, Minneapolis, Minnesota, 1996.

[87] Matlab, The Math Works, Inc., www.mathworks.com.

[88] T. McGeer. Passive dynamic walking. *International Journal of Robotics Research*, 9(2):62–82, 1990.

[89] A.N. Michel and B. Hu. Towards a stability theory of general hybrid dynamical systems. *Automatica*, 35:371–384, 1999.

[90] K.D. Mombaur, H.G. Bock, J.P. Schlöder, and R.W. Longman. Open-loop stability – a new paradigm for periodic optimal control and analysis of walking mechanisms. In *Proceedings of the IEEE Conference on Robotics, Automation and Mechatronics*, pages 704–709, Singapore, 2004.

[91] J.J. Moreau. Quadratic programming in mechanics: Dynamics of one-sided constraints. *SIAM Journal on Control and Optimization*, 4(1):163–158, 1966.

[92] E. Münz and V. Krebs. Continuous optimization approaches to the identification of piecewise affine systems. In *Proceedings of the IFAC World Congress*, Prague, Czech Republic, 2005.

[93] R.M. Murray, Z. Li, and S.S. Sastry. *A Mathematical Introduction to Robotic Manipulation*. CRC Press, 1996.

[94] K. Nagasaka, Y. Kuroki, S. Suzuki, Y. Itoh, and J. Yamaguchi. Integrated motion control for walking, jumping and running on a small bipedal entertainment robot. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 3189– 3194, New Orleans, LA, 2004.

[95] Y. Nakamura. *Advanced Robotics – Redundancy and Optimization*. Addison Wesley, 1991.

[96] N. Naksuk, C.S.G. Lee, and S. Rietdyk. Whole-body human-to-humanoid motion transfer. In *Proceedings of the IEEE/RAS International Conference on Humanoid Robots (HUMANOIDS)*, pages 104–109, Tsukuba, Japan, 2005.

[97] A. Nerode and W. Kohn. Models for hybrid systems: Automata, topologies, controllability, observability. In *Hybrid Systems*, pages 317–356, London, UK, 1993. Springer.

[98] M. Okada, T. Shinohara, T. Gotoh, S. Ban, and Y. Nakamura. Double spehrical joint and backlash clutch for lower limbs of humanoids. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 491–496, Taipei, Taiwan, 2003.

[99] J.H. Park and H.C. Cho. An online trajectory modifier for the base link of biped robots to enhance locomotion stability. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 3353–3358, San Francisco, CA, 2000.

[100] J.H. Park and H. Chung. ZMP compensation by on-line trajectory generation for biped robots. In *Proceedings of the IEEE Conference on Systems, Man and Cybernetics (SMC)*, pages 960–965, Tokyo, Japan, 1999.

[101] T.S. Parker and L.O. Chua. *Practical numerical algorithms for chaotic systems*. Springer, New York, 1989.

[102] P. Peleties and R. DeCarlo. A modeling strategy for hybrid systems based on event structures. In *Discrete Event Dynamic Systems*, volume 3, pages 39 – 69. 2003.

[103] F. Plestan, J.W. Grizzle, E.R. Westervelt, and G. Abba. Stable walking of a 7-DoF biped robot. *IEEE Transactions on Robotics and Automation*, 19(4):653–668, 2003.

[104] W.H. Press, S.A. Teukolsky, and W.T. Vetterling. *Numerical Recipes in C*. Cambridge University Press, 1993.

[105] M. Raibert. *Legged Robots that balance*. MIT Press, 1986.

[106] C. Raubitschek. *Beschränkung von Kontaktkräften und Momenten für einen zweibeingen Roboter durch Invarianzregelung*. Technical report, Technische Universität München, 2006.

[107] RoboCup. www.robocup.org.

[108] M. Rubensson and B. Lennartson. Stability of limit cycles in hybrid systems using discrete-time lyapunov techniques. In *Proceedings of the IEEE Conference on Decision and Control (CDC)*, 2000.

[109] T. Schlegl and M. Buss. Dextrous hand regrasping using hybrid system models. In *Proceedings of the IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, page 131, Tokyo, Japan, 1997.

[110] T. Schlegl, M. Buss, and G. Schmidt. Development of numerical integration methods for hybrid (discrete-continuous) dynamical systems. In *Proceedings of the IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, page 154, Tokyo, Japan, 1997.

[111] T. Schlegl, M. Buss, and G. Schmidt. Hybrid control of multi-fingered dextrous robotic hands. In S. Engell, G. Frehse, and E. Schnieder, editors, *Modelling, Analysis and Design of Hybrid Systems. Lecture Notes in Control and Information Sciences (LNCIS)*, pages 437–465. Springer, Berlin, Heidelberg, 2002.

[112] A.L. Schwab and M. Wisse. The basin of attraction of the simplest walking model. In *Proceedings of the ASME Design Engineering Technical Conferences*, Pittsburgh, Pennsylvania, 2001.

[113] L.F. Shampine, J. Kierzenka, and M.W. Reichelt. Solving boundary value problems for ordinary differential equations in Matlab with bvp4c. *available at ftp://ftp.mathworks.com/pub/doc/papers/bvp*, October 2000.

[114] S.N. Simić, K.H. Johansson, J. Lygeros, and S. Sastry. Hybrid limit cycles and hybrid Poincaré-Bendixson. In *Proceedings of the IFAC World Congress*, Barcelona, Spain, 2002.

[115] S.N. Simić, K.H. Johansson, S. Sastry, and J. Lygeros. Towards a geometric theory of hybrid systems. In N. Lynch and B. H. Krogh, editors, *Hybrid Systems: Computations and Control*, number 1790 in LNCIS, pages 421–436. Springer, 2000.

[116] M. Sobotka. Supplementary videos. www.lsr.ei.tum.de/team/sobotka/videos.html.

[117] M. Sobotka and M. Buss. Hybrid trajectory planning for a mechatronic tilting system. In *Proceedings of the IEEE Conference on Control Applications*, Taiwan, September 2004.

[118] M. Sobotka and M. Buss. Preliminary studies on the control of tilting mechatronic systems. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1142 – 1147, Sendai, Japan, 2004.

[119] M. Sobotka and M. Buss. A hybrid mechatronic tilting robot: Modeling, trajectories, and control. In *Proceedings of the IFAC World Congress*, Prague, Czech Republic, 2005.

[120] M. Sobotka and M. Buss. Hybrid modeling and control of a monoped robot with variable foot contact. *Transactions of the Society of Instruments and Control Engineers*, 44(7):477–485, 2005.

[121] M. Sobotka and M. Buss. Locomotion studies for a 5DoF gymnastic robot. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 358–363, Edmonton, Canada, 2005.

[122] M. Sobotka, D. Wollherr, and M. Buss. A jacobian method for online modification of precalculated gait trajectories. In *Proceedings of the International Conference on Climbing and Walking Robots (CLAWAR)*, Catania, Italy, 2003.

[123] G. Song and M. Zefran. A computational approach to dynamic bipedal walking. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Las Vegas, NV, 2003.

[124] R. Stengel. *Optimal Control and Estimation*. Dover Publications, 1994.

[125] T. Sugihara and Y. Nakamura. Whole body cooperative balancing of humanoid robot using CoG jacobian. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2575–2580, Lausanne, Switzerland, 2002.

[126] T. Takahashi and A. Kawamura. Posture control using foot toe and sole for biped walking robot "Ken". In *Proceedings of the IEEE Workshop on Advanced Motion Control*, pages 437–442, Slovenia, 2002.

[127] L. Tavernini. Differential automata and their discrete simulators. *Non-Linear Analysis*, 11(6), 1987.

[128] R.L. Tedrake. *Applied Optimal Control for Dynamically Stable Legged Locomotion*. PhD thesis, Massachusetts Institute of Technology, 2004.

[129] Toyota Motor Corp. Partner robots. www.toyota.co.jp/en/special/robot/.

[130] O. von Stryk. Numerical solution of optimal control problems by direct collocation. In R. Bulirsch, A. Miele, J. Stoer, and K. H. Well, editors, *Optimal Control - Calculus of Variations, Optimal Control Theory and Numerical Methods*, volume 111 of *International Series of Numerical Mathematics*, pages 129–143. Birkhäuser, Basel, 1993.

[131] O. von Stryk. *User's Guide for DIRCOL (Version 2.1): a direct collocation method for the numerical solution of optimal control problems*. Fachgebiet Simulation und Systemoptimierung (SIM), Technische Universität Darmstadt, 2000.

[132] M. Vukobratović, B. Borovac, and D. Šurdilović. Zero-Moment Point – proper interpretation and new application. In *Proceedings of the IEEE/RAS International Conference on Humanoid Robots (HUMANOIDS)*, pages 237–244, Tokyo, Japan, 2001.

[133] M. Vukobratović and D. Juričić. Contribution to the synthesis of biped gait. *IEEE Transaction on Bio-Medical Engineering*, 16:1–6, 1969.

[134] E.R. Westervelt, J.W. Grizzle, and D.E. Koditschek. Hybrid zero dynamics of planar biped walkers. *IEEE Transactions on Automatic Control*, 48(1), 2003.

[135] P.-B. Wieber and C. Chevallereau. Online adaption of reference trajectories for the control of walking systems. *Journal of Robotics and Autonomous Systems*, 54(7):513–616, 2006.

[136] H.S. Witsenhausen. A class of hybrid-state continuous-time dynamic system. *IEEE Transactions on Automatic Control*, 11:161–167, 1966.

[137] J. Wolff and M. Buss. Invariance control design for nonlinear control affine systems under hard state constraints. In *Proceedings of the NOLCOS*, pages 711–716, Stuttgart, Germany, 2004.

[138] J. Wolff, M. Sobotka, and M. Buss. Invariance controlled balance of legged robots. In *Proceedings of the European Control Conference (ECC)*, Kos, Greece, 2007.

[139] D. Wollherr. *Design and Control Aspects of Humanoid Walking Robots*. PhD thesis, Technische Universität München, 2005.

[140] D. Wollherr, F. Zonfrilli, and Y. Nakamura. Active-passive knee control for the humanoid UT-Theta. In *Proceedings of the International Conference on Advanced Robotics (ICAR)*, pages 692–697, Seattle, Washington, USA, 2005.