A CONTROL APPROACH TO HUMAN-LIKE LOCOMOTION IN BIPED ROBOTS

with an Approach for

Numerical Simulation of Constrained Mechanical Systems

BY

DAVID J. BRAUN

Dissertation

Submitted to the Faculty of the

Graduate School of Vanderbilt University

in partial fulfillment of the requirements

for the degree of

DOCTOR OF PHILOSOPHY

in

Mechanical Engineering

December, 2009

Nashville, Tennessee

Approved:

Dr. Michael Goldfarb

Dr. George E. Cook

Dr. Alvin M. Strauss

Dr. Nilanjan Sarkar

Dr. Eric J. Barth

To my dear wife,

Krisztina

and

my parents Borbála and János

ACKNOWLEDGMENTS

I would like to express my sincere gratitude to my advisor, Dr. Michael Goldfarb, who gave me the opportunity and supported my research at Vanderbilt University - Center for Intelligent Mechatronics. Without his encouragement and advice this work would not have been possible. I would also like to thank Dr. Alvin M. Strauss for his help and support. I acknowledge the other members of my thesis committee, Dr. George Cook, Dr. Nilanjan Sarkar, Dr. Eric J. Barth, for their time and suggestions through the preparation of this dissertation.

My thanks go to Jason Mitchell who has spent time and effort to design the robot which was used in numerous experiments included in this work. Furthermore, I thank all the people in the graduate school with whom I spent time discussing research problems.

I acknowledge the education received at the University of Novi Sad, Faculty of Technical Sciences, Department of Mechanics, and the support of Dr. Božidar D. Vujanović and Dr. Teodor M. Atanacković.

Finally, I am honored with the patience and support of my dear wife Krisztina Braun, and would like to thank my parents Borbála and János Braun who helped me to come this far.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER I

INTRODUCTION

Background, Terminology and Motivation

There has been considerable research effort devoted to bipedal (legged) locomotion, a review of which can be found in [4]. While various methods are proposed to tackle the related control problem, most of the ideas are either built on the Zero Moment Point (ZMP) paradigm, or rely on the (passive) Dynamic Walking Principle.

The Zero Moment Point control approach, originally introduced by Vukobratović *et.al* [5], [6], is the most accepted and widely used paradigm for biped locomotion syntheses. When level ground walking is considered, ZMP is defined as that point on the ground at which the net moment vector of the inertial and applied forces has no component in a ground plane [7]. As long as ZMP is within the foot support area, its location coincides with the center of pressure (CoP), which is the point on the foot where the ground reaction force acts. The importance of the ZMP is that by controlling its position within a foot support region, no foot rotation will occur (i.e., the foot can be used as a base link from which a trajectory tracking motion control can be performed on the body). The ZMP idea is well discussed in literature and served to control (among others) the first dynamically balancing robot WL-10RD [8], and the state-of-the-art humanoid robot the Honda Asimo [9].

Walking with ZMP kept within the support region prevents foot rotation and is attributed as "dynamically balanced walk" [7]. In the related context the word "dynamic" is meant to indicate that along the motion the center of mass (CoM) of the robot does not need to be kept above the CoP (which would be required to maintain static balance). While keeping ZMP within the support region (to prevent foot rotation) is also referred as a "dynamically stable walk", restriction on foot rotation is not required for stable walking, broadly considered as walking without falling herein.

In particular, foot rotation is naturally utilized by humans (while walking stably) [10].

In addition to indicating a dynamically balanced walk, dynamic walking is also a paradigm in legged locomotion. In this context, dynamic walking refers to a robot with finely adjusted inertial and geometric design which, enhanced with slight control, can emulate stable walking [11]. On the end of the spectrum of this approach is the "passive dynamic walking principle" introduced by McGear [12], by showing that (with precisely tuned design) an uncontrolled legged machine can walk on a slight downward slope (powered only by gravity), see [13]. While utilization of a passive (uncontrolled) dynamics is an important objective through dynamic walking, this approach does not exclude control. Specifically, one can utilize simple control strategy on actuator assisted dynamic walkers to emulate natural looking walking [14] or even walking with human efficiency [1].

During the conducted research, we have identified two preconditions which allow *human-like dynamic walking* to be realized on actuated robots:

- The first precondition, related to the control approach, precludes enforcing a predefined reference trajectory and may also not favor enforcing state dependent kinematic constraints, or other attributes of the walking cycle (such as stride length, stepping frequency of average forward speed) with high gain control. This condition motivated us to develop a control framework which utilizes state-dependent control torques generated by low-gain spring-damper couples to provide motion coordination without prespecifying the motion of the system.

- The second precondition (not related to control) depends on joint actuation which should not suppress passive joint motion (i.e., joints should be highly back-drivable such as human joints). Utilization of back-drivable joint design allows the inertial motion of the robot to be exploited through walking rather than being suppressed by the actuation units. This condition motivated us to design a 7-link biped robot, with highly back-drivable joints, which is used in the experimental verification of the control framework.

The overall control philosophy is analytically developed, numerically investigated and ex-

perimentally realized on a 7-link biped.

Compared to approaches proposed for actuated dynamic walking, [15], [16], [17], [18], [19], [20], [21], [22], [23], [24], implementation of the present one allowed experimental demonstration of human-like dynamic walking with (partially) ballistic swing, extended knee stance support, and (preemptive) ankle push-off, on a robot with flat foot and upper body. Compared to the ZMP walking paradigm (which must prevent foot rotation to ensure dynamic balance), the presented approach allows foot rotation and as such emulation of *human-like* actuated dynamic walking. Beyond this difference, the extended knee stance support offered here allows walking which is more natural looking then the usual (ZMP-based) bent knee robot walking.

In addition to the analytical derivation, numerical evaluation and experimental implementation of the proposed control approach, the dissertation also offers a modeling and simulation method developed to support the presented control methodology.

Summary and Outline of the Dissertation

The dissertation is organized in five chapters. Chapter I presents the introduction of the work. Chapters II-IV contain three manuscripts that summarize the research completed and have been submitted for publication as journal articles. Chapter V concludes the work with the contributions and the *proposed* future direction. An overview of the manuscripts presented through Chapter II-IV is given as follows:

- Manuscript 1: D.J. Braun and M. Goldfarb, "Eliminating Constraint Drift in the Numerical Simulation of Constrained Dynamical Systems," *Computer Methods in Applied Mechanics and Engineering*, vol. 198 no. 37-40, pp. 3151–3160, 2009.

  This article provide a theoretical framework for numerical simulation of constrained dynamical system modeled with differential-algebraic equations (DAEs). Specifically, the paper offers an equation of constrained motion which, solved with a standard explicit ODE integrator (i.e., Euler, Runge-Kutta method), provides a precise motion prediction for DAEs. Beyond

the theoretical contribution, the paper presents three trajectory tracking simulations on a seven link planar anthropometric biped robot which demonstrates feasibility of the approach to simulate bipedal motion subjected to redundant (dependent) constraints. This features have been recognized important through numerical investigation of the walking controller proposed in [25] (Manuscript 2). The real-time implementation of the method was also utilized for parameter identification, free swing experiments, and the PD control experiment performed to characterize the dynamics of the seven link biped, design and control of which is discussed in [26] (Manuscript 3).

- Manuscript 2: D.J. Braun and M. Goldfarb, "A Control Approach to Actuated Dynamic Walking in Biped Robots," *IEEE Transaction on Robotics*, 2009 - accepted. A short version of this paper, [27], is presented at the IEEE/RSJ International Conference on Intelligent Robots and Systems, October 11-15, 2009, St. Louis, USA.

  This article presents a control framework for human-like actuated dynamic walking in biped robots. Instead of utilizing the ZMP (zero-moment-point) control philosophy (frequently preferred to synthesize actuated dynamic walking), we propose an alternative control method for human-like dynamic walking. The proposed approach meant to improve, the walking style (bent knee walking) and the low locomotion efficiency recognized as fundamental issues in the majority of actuated walking robots. As was recognized during the conducted research, realization of a compliant walking precludes enforcement of a prespecified reference trajectory, or may also not favor enforcing state dependent kinematic constraints or other attributes of the walking cycle (such as step length, stepping frequency or average forward speed) with high gain control. This recognition motivated us to develop a control framework which utilizes state-dependent control torques generated by low-gain spring-damper couples to provide motion coordination without prespecifying the motion of the system. As is demonstrated (through numerous simulations) in the article, the approach can provide energy-efficient human like actuated dynamic walking in biped robots.

- Manuscript 3: D.J. Braun, Jason E. Mitchell and M. Goldfarb, "Experimental Implementation of Actuated Dynamic Walking in Biped Robots," *The International Journal of Robotics Research* - submitted. A short version of this paper, [2], is accepted on the 9th IEEE-RAS International Conference on Humanoid Robots December 7-10, 2009, Paris, France.

  By utilizing the control framework proposed in [25] (Manuscript 2), this article presents an experimental realization of actuated dynamic walking in biped robots. During the development of the walking controller, an important design requirement for energy-efficient realization of dynamic walking have been identified. This requirement depends on joint actuation which should not suppress the passive joint motion (i.e., joints should be backdrivable such as human joints). Practically, utilization of backdrivable joint design allows the inertial motion of the robot to be exploited through walking rather than being suppressed by the actuation units. This recognition motivated us to design a seven link biped robot, with highly backdrivable joints, which device is used in the experimental verification of the control framework. The presented walking experiment demonstrates dynamic walking characterized with (partially) ballistic swing, extended knee stance support and (preemptive) ankle push-off which feature can also be identified during (fast) human walking.

Utilizing the framework presented in [28], a control approach for human-like actuated dynamic walking was analytically developed and numerically investigated in [25], which approach is then experimentally verified on a seven link biped robot designed for this purpose [26]. According to the authors best knowledge, the preemptive ankle push-off (which is an important qualitative attribute of a fast human walking) identified during the walking experiments, have been only demonstrated on the MIT Spring Flamingo [18] and on the actuator assisted Cornell dynamic walker [1].

CHAPTER II

MANUSCRIPT 1: ELIMINATING CONSTRAINT DRIFT IN THE NUMERICAL SIMULATION OF CONSTRAINED DYNAMICAL SYSTEMS

By

David J. Braun and Michael Goldfarb

Vanderbilt University

Nashville, TN

Published as a Regular Paper in the

Computer Methods in Applied Mechanics and Engineering,

vol. 198, no. 37-40, pp. 3151–3160, 2009

Abstract

By means of the Udwadia-Kalaba approach we propose an explicit equation of constrained motion developed to simulate constrained dynamical systems without error accumulation due to constraint drift. The basic idea is to embed a small virtual force and a small virtual impulse to the equation of motion, in order to avoid the drift typically experienced in constrained multibody simulations. The embedded correction terms are selected to minimally alter the dynamics in an acceleration and kinetic energy norm sense. The formulation allows one to use a standard ODE solver, avoiding the need for iterative constraint stabilization. The equation is based on the pseudoinverse of a constraint matrix such that it can be used under redundant constraints and kinematic singularities. The proposed method takes into account the finite word-length of the computational environment, and also accommodates possibly inconsistent initial conditions.

Introduction

Constrained dynamical systems are traditionally modeled with a Lagrangian equation of the first kind [29] where additional algebraic variables (Lagrangian multipliers) are used to incorporate the motion constraints to the equation. In addition to this classical approach, many alternative formulations have been proposed in order to model constrained dynamical systems, including: Gauss's principle of least constraint [30], Maggi's equation [31], Gibbs-Appell's formulation [32], [33], Kane's equation [34], and the Udwadia-Kalaba approach [35], [36], [37], supported with additional discussions and development presented by Pars [38], Neĭmark and Fufaev [39], Gantmacher [40], Goldstein [41], Chetaev [42] and Lurie [43].

Despite the strong theoretical foundation, direct numerical implementation of the proposed governing equations generally leads to error accumulation due to "constraint drift". Specifically, motion constraint which should be physically invariant will move in space due to error and imperfection in numerical integration. The resulting solution is not physically consistent and as such loses value with respect to practical interpretation. This issue has been addressed by many authors including:

Baumgarte [44], Gear *et.al* [45], Lötstedt and Petzold [46], Führer and Leimkuhler [47], Petzold [48], ten Dam [49], Eich [50], Bayo and Ledesma [51], Blajer [52] and Aghili [53]. The importance of having a reliable simulation tool which produces physically consistent motion prediction is motivated by many practical applications as was discussed by Schiehlen [54] and Brogliato *et.al* [55].

Our aim is to propose a formulation which enables stable numerical simulation without error accumulation in motion constraints. In order to do so, it was necessary to take the nonideal computational environment as well as the possible errors in initial data (caused by the user) into account. Following a revised constraint definition, we derive an explicit equation for constrained motion with constraint correction terms. Although these additional terms have no direct physical meaning, they can be interpreted as a set of small virtual forces and impulses and are derived by means of Gauss's principle of least constraint. After presentation of the proposed formulation, the approach is discussed in the context of prior work in the field. Finally, the approach is illustrated on and validated with several representative examples.

Constrained Multibody Dynamics

In this section, the equation of motion for a constrained dynamical system is derived. The approach is based on the explicit equation of constrained motion presented by Udwadia and Kalaba [35].

Unconstrained Multibody Dynamics

Consider an n-degree-of-freedom multibody system, the configuration of which is uniquely specified by $\mathbf{q} \in \mathbb{R}^n$ generalized coordinates. Let the equation of motion of the considered system (derived by means of the Lagrangian formalism) be represented in the following form

$$\mathbf{M}(t,\mathbf{q})\ddot{\mathbf{q}} = \mathbf{Q}(t,\mathbf{q},\dot{\mathbf{q}}). \tag{2.1}$$

Here, $t \in [0,T]$ is a time variable, $\mathbf{M} \in \mathbb{R}^{n\times n}$ is a symmetric and positive definite mass matrix while $\mathbf{Q} \in \mathbb{R}^n$ represents the generalized forces. If no constraints are applied on (2.1), the dynamical system is considered as unconstrained with respect to the chosen generalized coordinates $\mathbf{q}$.

Holonomic Constraints: Revised

Let us introduce additional *m* holonomic bilateral constraints on the system dynamics,

$$\mathbf{\Phi}(t,\mathbf{q}) = \mathbf{0}, \tag{2.2}$$

where $\mathbf{\Phi} : [0,T] \times \mathbb{R}^n \to \mathbb{R}^m$. In the forthcoming analysis, we will assume that these rheonomic (explicitly time dependent) constraints are $C^2[0,T]$, such that, (2.2) has well defined first and second partial derivatives at least.

Let us discuss the effect of (2.2) on the dynamical system (2.1). Generally speaking, each bilateral constraint adds a constraint reaction force to the system dynamics. If the constraint is ideal, it generates an "ideal reaction" which does no work on any constraint consistent virtual displacement. We assume that all constraints are ideal and as such D'Alambert's principle applies [56].

Considering the constraint equations (2.2), one can see that $\mathbf{\Phi} = \mathbf{0}$ defines position-level relations between the generalized coordinates. However, in order to avoid error accumulations along the numerical solution, the holonomic (position) constraints must also be satisfied on the velocity level $\dot{\mathbf{\Phi}} = \mathbf{0}$. In this light, by adding the aforementioned velocity level constraint to (2.2), one obtains

$$\mathbf{\Phi}(t,\mathbf{q}) = \mathbf{0}, \dot{\mathbf{\Phi}}(t,\mathbf{q},\dot{\mathbf{q}}) = \mathbf{0}. \tag{2.3}$$

If the dynamical system is restricted by (2.2), it is important to make sure that not only (2.2) but also (2.3) is satisfied. As follows, we will replace (2.3) with velocity and acceleration level constraints which are linear with respect to $\dot{\mathbf{q}}$ and $\ddot{\mathbf{q}}$, respectively. To do so, let us assume that $\mathbf{q} = \mathbf{q}(t)$ defines the positions of the constrained dynamical system. Using these functions, one can come up with

$\mathbf{\Phi}(t) = \mathbf{\Phi}(t, \mathbf{q}(t))$, and the following Taylor expansion holds

$$\mathbf{\Phi}(t+dt) = \mathbf{\Phi}(t) + (\mathbf{A}\dot{\mathbf{q}} - \mathbf{b}_q)dt + O(dt^2), \tag{2.4}$$

where, $\mathbf{A}(t, \mathbf{q}) = \partial\mathbf{\Phi}/\partial\mathbf{q}$ and $\mathbf{b}_q(t, \mathbf{q}) = -\partial\mathbf{\Phi}/\partial t$. Similarly, one can define $\dot{\mathbf{\Phi}}(t) = \dot{\mathbf{\Phi}}(t, \mathbf{q}(t), \dot{\mathbf{q}}(t))$ and expand the velocity level (nonholonomic) constraints up to the acceleration level as,

$$\dot{\mathbf{\Phi}}(t+dt) = \dot{\mathbf{\Phi}}(t) + (\mathbf{A}\ddot{\mathbf{q}} - \mathbf{b}_v)dt + O(dt^2), \tag{2.5}$$

where $\mathbf{b}_v(t, \mathbf{q}, \dot{\mathbf{q}}) = -\dot{\mathbf{q}}^T[\partial^2\mathbf{\Phi}/\partial\mathbf{q}^2]\dot{\mathbf{q}} - 2[\partial^2\mathbf{\Phi}/\partial t\partial\mathbf{q}]\dot{\mathbf{q}} - \partial^2\mathbf{\Phi}/\partial t^2$.
If dt is interpreted as a (numerical) integration step, then the first order approximation of (2.3), (as well as (2.4) and (2.5)) can be satisfied at each subsequent integration step, $\mathbf{\Phi}(t+dt) = \mathbf{0}$, $\dot{\mathbf{\Phi}}(t+dt) = \mathbf{0}$, with velocity and acceleration level constraints defined as follows

$$\begin{aligned} \mathbf{A}\dot{\mathbf{q}} &= \mathbf{b}_q - \mathbf{\Phi}/dt, \\ \mathbf{A}\ddot{\mathbf{q}} &= \mathbf{b}_v - \dot{\mathbf{\Phi}}/dt. \end{aligned} \tag{2.6}$$

Instead of (2.6), the analytical derivation of the explicit equation of constrained motion proposed by [45] and [35], [36], [37] is based on a velocity or acceleration level representation of the original constraints

$$\mathbf{A}\dot{\mathbf{q}} = \mathbf{b}_q, \mathbf{A}\ddot{\mathbf{q}} = \mathbf{b}_v. \tag{2.7}$$

However, the equation of constrained motion which is based on (2.7) does not have a numerically stable ODE implementation without additional constraint correction. Namely, in order to substitute (2.6) with (2.7), the following conditions must be met:

- The user must provide initial conditions which are constraint consistent, $\mathbf{\Phi}(0, \mathbf{q}(0)) = \mathbf{0}$, $\dot{\mathbf{\Phi}}(0, \mathbf{q}(0), \dot{\mathbf{q}}(0)) = \mathbf{0}$.
- The solution $\mathbf{q}(t), \dot{\mathbf{q}}(t)$ must not contain any integration error.

Due to numerical errors, such conditions are not realistic in a numerical (i.e., computer) simulation environment. Rather, in practice, use of (2.7) typically leads to significant error accumulation and constraint drift. This motivated us to derive an implementation of (2.6) which takes the expected numerical imperfections into account. We will show that adequate implementation of (2.6) prevents error accumulation and results in a numerical solution without constraint drift.

Constrained Multibody Dynamics

In order to incorporate position-level constraints to the dynamic equations, we recall the Lagrange multiplier approach. However, in contrast to the method traditionally used in mechanics, where the multipliers represent constraint reaction forces, our intention is to use the same idea to eliminate constraint violations.

Let us start with the traditional representation of the constrained dynamical system,

$$\mathbf{M}\ddot{\mathbf{q}} = \mathbf{Q} + \mathbf{Q}_c, \tag{2.8}$$

where $\mathbf{Q}_c = \mathbf{A}^T\boldsymbol{\lambda}$ is the generalized constraint force, while the undetermined Lagrange multipliers, $\boldsymbol{\lambda} \in \mathbb{R}^m$, represent the physical forces generated by constraints. This representation is valid for ideal constraints which do no work, $\mathbf{Q}_c^T\delta\mathbf{q} = 0$, along any admissible virtual displacement $\delta\mathbf{q} \in \{\delta\mathbf{q} : \delta\mathbf{q} \in \mathbb{R}^n\ , \mathbf{A}\delta\mathbf{q} = \mathbf{0}\}$. To formulate the equation of motion in explicit form, let us solve the constrained acceleration from (2.8) as a function of $\boldsymbol{\lambda}$,

$$\ddot{\mathbf{q}} = \mathbf{a} + \mathbf{M}^{-1}\mathbf{A}^T\boldsymbol{\lambda}, \tag{2.9}$$

where $\mathbf{a} = \mathbf{M}^{-1}\mathbf{Q}$ is the unconstrained acceleration the system would have without the imposed constraints, see (2.1). Now, substituting (2.9) back to the practical acceleration level constraints

$(2.6)_2$, one can solve for the Lagrangian multipliers by direct inversion

$$\boldsymbol{\lambda} = (\mathbf{A}\mathbf{M}^{-1}\mathbf{A}^T)^{-1}(\mathbf{b}_v - \mathbf{A}\mathbf{a} - \dot{\boldsymbol{\Phi}}/dt). \tag{2.10}$$

Although, the physical constraint reactions $\boldsymbol{\lambda}$ are not well defined if the constraint matrix $\mathbf{A}$ is rank deficient (due to kinematic singularities or constraint redundancy), the constrained acceleration as well as the generalized constraint force $\mathbf{Q}_c = \mathbf{A}^T\boldsymbol{\lambda}$ are always unique, [57], [58], [36]. Thus, we can determine $\mathbf{Q}_c$ to accommodate kinematic singularities and constraint redundancy by first defining the following matrices: $\mathbf{M}^{1/2}$, $\mathbf{M}^{-1/2}$, $\mathbf{B} = \mathbf{A}\mathbf{M}^{-1/2}$ and $\mathbf{B}^+$, where due to the positive definiteness of $\mathbf{M}$, the so called principal square root of the mass matrix $\mathbf{M}^{1/2}$ and its inverse $\mathbf{M}^{-1/2}$ are always well defined, as is the (Moore-Penrose inverse) pseudoinverse $\mathbf{B}^+$ of $\mathbf{B}$, [59]. Using the introduced notation, the generalized constraint force becomes

$$\mathbf{Q}_c = \mathbf{M}^{1/2}\mathbf{B}^+(\mathbf{b}_v - \mathbf{A}\mathbf{a} - \dot{\boldsymbol{\Phi}}/dt). \tag{2.11}$$

Substituting (2.11) into (2.8), the explicit equation of the constrained dynamics can be easily obtained. In order to further proceed, one can define the constrained acceleration, $\ddot{\mathbf{q}} = \dot{\mathbf{v}}$ and rewrite the equation of motion in the following first order form

$$\begin{aligned} \dot{\mathbf{q}} &= \mathbf{v}, \\ \dot{\mathbf{v}} &= \mathbf{a} + \mathbf{M}^{-1/2}\mathbf{B}^+(\mathbf{b}_v - \mathbf{A}\mathbf{a} - \dot{\boldsymbol{\Phi}}/dt). \end{aligned} \tag{2.12}$$

Although this formulation accounts for numerical errors on the velocity level, $\dot{\boldsymbol{\Phi}} \approx \mathbf{0}$, it cannot in general prevent error accumulation. This is because (2.12) does not yet take the numerically induced position level error given by $\boldsymbol{\Phi} \approx \mathbf{0}$ into account. With the aim of incorporating this error source, we mimic the above procedure, adding a new Lagrangian multiplier to $(2.12)_1$,

$$\dot{\mathbf{q}} = \mathbf{v} + \mathbf{M}^{-1}\mathbf{A}^T\boldsymbol{\mu}. \tag{2.13}$$

In contrast to $\boldsymbol{\lambda}$, the new multiplier $\boldsymbol{\mu}$ is not generated by the constraints but rather is introduced to compensate for numerical errors along the integration. A similar term, $\mathbf{A}^T\boldsymbol{\mu}$ was used by [45] to incorporate velocity level constraints in the equation of motion. Note that neither $\mathbf{A}^T\boldsymbol{\mu}$ nor $\mathbf{M}^{-1}\mathbf{A}^T\boldsymbol{\mu}$ has clear physical meaning. However while the former can only be considered as a kinematic correction term, the latter (introduced here) is a dynamic correction which allows us to interpret $\boldsymbol{\mu}$ as a small mechanical impulse.

Substituting (2.13) into $(2.6)_1$, one obtains

$$\boldsymbol{\mu} = (\mathbf{A}\mathbf{M}^{-1}\mathbf{A}^T)^{-1}(\mathbf{b}_q - \mathbf{A}\mathbf{v} - \boldsymbol{\Phi}/dt). \tag{2.14}$$

Once again, using the pseudoinverse notation, the compensation term becomes

$$\mathbf{M}^{-1}\mathbf{A}^T\boldsymbol{\mu} = \mathbf{M}^{-1/2}\mathbf{B}^{+}(\mathbf{b}_q - \mathbf{A}\mathbf{v} - \boldsymbol{\Phi}/dt). \tag{2.15}$$

Substituting (2.15) into (2.13), and combining with $(2.12)_2$, the complete equation of motion for the constrained dynamical system is obtained

$$\begin{aligned} \dot{\mathbf{q}} &= \mathbf{v} + \mathbf{M}^{-1/2}\mathbf{B}^{+}(\mathbf{b}_q - \mathbf{A}\mathbf{v} - \boldsymbol{\Phi}/dt), \\ \dot{\mathbf{v}} &= \mathbf{a} + \mathbf{M}^{-1/2}\mathbf{B}^{+}(\mathbf{b}_v - \mathbf{A}\mathbf{a} - \dot{\boldsymbol{\Phi}}/dt), \end{aligned} \tag{2.16}$$

where $\mathbf{a} = \mathbf{M}^{-1}\mathbf{Q}$ is the unconstrained acceleration, $\dot{\mathbf{v}}$ is the constrained acceleration, and $\dot{\mathbf{q}}$ is the constrained velocity.

It can be easily recognized that if $\boldsymbol{\Phi} = \mathbf{0}$ and $\dot{\boldsymbol{\Phi}} = \mathbf{0}$ (which also implies $\mathbf{A}\mathbf{v} = \mathbf{b}_q$) then (2.16) reduces to the well known explicit equation of motion derived by Udwadia and Kalaba (which assumes an ideal computational environment and perfect initial conditions). Taking the error sources in the real computational environment into account, we do not assume exact constraint satisfaction which, following strict mathematical derivations, produces additional error compensation terms in the equation. The new terms compensate for the numerical errors and guarantee that no error accumulation can take place.

Discussion of the Proposed Formulation

In order to discuss (2.16), let us recall the explicit equation of motion which does not contain the constraint correction terms,

$$\begin{aligned} \dot{\mathbf{q}} &= \mathbf{v}, \\ \dot{\mathbf{v}}^i &= \mathbf{a} + \mathbf{M}^{-1/2}\mathbf{B}^+(\mathbf{b}_v - \mathbf{A}\mathbf{a}), \end{aligned} \tag{2.17}$$

where $\dot{\mathbf{v}}^i$ is the acceleration which exactly satisfies $\mathbf{A}\dot{\mathbf{v}}^i = \mathbf{b}_v$, while $\mathbf{v}$ is the velocity obtained by numerical integration of $\dot{\mathbf{v}}^i$ (in an ideal computational environment one would obtain, $\mathbf{v} = \mathbf{v}^i$). Comparing $(2.16)_2$ with $(2.17)_2$, one may conclude that $\mathbf{M}^{-1/2}\mathbf{B}^+\dot{\mathbf{\Phi}}/dt$ represents a small correction of the constraint force (Lagrangian multiplier $\boldsymbol{\lambda}$, see (2.10)) which is necessary to satisfy the constraints under numerically imperfect conditions. In order to have a clearer interpretation, let us reformulate $(2.16)_2$ by means of the Gauss principle of least constraint,

$$\begin{aligned} \dot{\mathbf{v}} = min\{\mathbf{x} \in \mathbb{R}^n : (\mathbf{x} - \dot{\mathbf{v}}^i)^T\mathbf{M}(\mathbf{x} - \dot{\mathbf{v}}^i), \\ \mathbf{A}\mathbf{x} = \mathbf{b}_v - \dot{\mathbf{\Phi}}/dt\}. \end{aligned} \tag{2.18}$$

Based on this interpretation, the $\dot{\mathbf{v}}$ provided by $(2.16)_2$ is the closest acceleration to $\dot{\mathbf{v}}^i$ (in an acceleration energy sense) which satisfies the constraints $(2.6)_2$.

Similarly, comparing the first equations in (2.16) and (2.17), one might recognize that $\mathbf{M}^{-1/2}\mathbf{B}^+(\mathbf{b}_q - \mathbf{A}\mathbf{v} - \mathbf{\Phi}/dt)$, although not generated by physical constraints, is necessary to prevent error accumulation. In order to give a physical interpretation of this term, let us define an equivalent formulation of $(2.16)_1$ with the following constrained quadratic program

$$\begin{aligned} \dot{\mathbf{q}} = min\{\mathbf{x} \in \mathbb{R}^n : (\mathbf{x} - \mathbf{v})^T\mathbf{M}(\mathbf{x} - \mathbf{v}), \\ \mathbf{A}\mathbf{x} = \mathbf{b}_q - \mathbf{\Phi}/dt\}. \end{aligned} \tag{2.19}$$

Here, $\dot{\mathbf{q}}$ is the closest velocity to $\dot{\mathbf{v}}$ (in a kinetic energy sense) which satisfies the constraints $(2.6)_1$. Note that $\mathbf{v}$, obtained by time integration of the constrained acceleration, may not satisfy exactly

the kinematic constraints $\mathbf{Av} \neq \mathbf{b}_q$ (which condition is taken into account along the derivation). In a special case when $\mathbf{v} = \mathbf{v}^i$, ($\mathbf{Av} = \mathbf{b}_q$), the correction term would reduce to $-\mathbf{M}^{-1/2}\mathbf{B}^+\mathbf{\Phi}/dt$.
One can conclude now that the derivation of the acceleration level correction terms are closely related to the general principle of constrained motion formulated by Gauss [30], while the velocity level correction terms are obtained using kinetic energy minimization, and as such are also physically motivated.

Numerical Implementation

In order to simulate a constrained dynamical system, a robust and stable numerical solver for the index-3 differential algebraic equation (DAE) (2.1), (2.2), is required [60]. However, in contrast to widely available ordinary differential equation (ODE) solvers, a DAE solver which reliably prevents error accumulation and constraint drift is not trivial to implement. By means of DAE integration, DASSL [61] (and its extended version DASSLRT) offers a state-of-the-art implementation of the index-2 DAE formulation proposed by [45]. In general, different DAE integrators have been developed as research codes, overviews of which can be found in [62], [63], [64].
Our intention is to show that one can use traditional ODE integrators to solve the reformulated constrained dynamic equation (2.16), without having problems of error accumulation and constraint drift. In the remainder of this section, we discuss how to obtain such a numerical solution.
When the analytical model is derived, the system is characterized with the following quantities; $\mathbf{M}$, $\mathbf{Q}$, $\mathbf{A}$, $\mathbf{b}_q$, $\mathbf{b}_v$, $\mathbf{\Phi}$ and $\dot{\mathbf{\Phi}}$. Without any further preparation, (2.16) is ready to be solved in a standard ODE solver which utilizes a first order state-space formulation, providing we can incorporate the correction terms, $\mathbf{\Phi}/dt$ and $\dot{\mathbf{\Phi}}/dt$ where, as it was mentioned, dt is interpreted as a time step of the numerical integrator. The simplest way to incorporate the correction terms is to use a fixed step solver where dt is predefined. If however, one wants to exploit the benefits of a variable step solver, the actual time step should be used over the integration procedure.
The computational expense of the numerical implementation of (2.16) is dominated by the calculations of the principal square root of the mass matrix $\mathbf{M}^{1/2}$ and the pseudoinverse $\mathbf{B}^+$. Practically,

these computations entail an eigenvalue computation of $\mathbf{M}$ and a singular value decomposition for $\mathbf{B}$, see [65]. However, in order to incorporate the constraint forces, one can use $\mathbf{M}^{-1/2}\mathbf{B}^{+} = \mathbf{R}^{-1}\mathbf{C}^{+}$ where $\mathbf{R}$ is the upper triangular Cholesky factor of the inertia matrix $\mathbf{M} = \mathbf{R}^{T}\mathbf{R}$ while $\mathbf{C} = \mathbf{A}\mathbf{R}^{-1}$. This replacement allows one to avoid the particularly expensive eigenvalue computation which would be required in order to compute $\mathbf{M}^{1/2}$. It is also important to mention that the pseudoinverse notation utilized in (2.16) allows a compact and general representation of the equation of motion regardless of whether the constraints are independent or dependent. From a computational point of view, however, only dependent constraints require singular value decomposition to define $\mathbf{B}^{+}$ (or $\mathbf{C}^{+}$) while for independent constraints one can either 1) compute the velocities and accelerations together with the Lagrangian multipliers from (2.9), (2.10), (2.13) and (2.14), or 2) utilize the following explicit definition, $\mathbf{M}^{-1/2}\mathbf{B}^{+} = \mathbf{R}^{-1}\mathbf{C}^{T}(\mathbf{C}\mathbf{C}^{T})^{-1}$ to evaluate the right hand side of (2.16) directly.

Note, that the correction terms are obtained based on the Taylor expansion of the constraints, see (2.4), (2.5), which provides a good approximation as long as a small integration step dt is used. With this in mind, one cannot expect arbitrarily precise constraint satisfaction (i.e., due to the finite time step and numerical imprecision). Let us recognize that the correction terms in (2.16) use the same matrix $\mathbf{M}^{-1/2}\mathbf{B}^{+}$, which must in any case be computed in order to incorporate the constraint forces. In this light, beyond multiplication and addition, the correction terms do not require any additional computation.

In the following, a simple numerical implementation of (2.16) will be given with some practical comments.

Numerical Procedure

Using a small time step dt, the finite domain of integration $t \in [0,T]$ is equidistantly discretized as $0 = t_0 < t_1 < ... < t_n < t_{n+1} < ... < t_N = T$. By means of numerical solution, we seek the discrete values of all positions $\{\mathbf{q}_0, \mathbf{q}_1, ..., \mathbf{q}_n, \mathbf{q}_{n+1}, ..., \mathbf{q}_N\}$ and velocities $\{\mathbf{v}_0, \mathbf{v}_1, ..., \mathbf{v}_n, \mathbf{v}_{n+1}, ..., \mathbf{v}_N\}$ which are numerically constraint consistent. Let us assume that the initial conditions $\mathbf{q}_0 = \mathbf{q}(0)$ and

$\mathbf{v}_0 = \mathbf{v}(0)$ approximately satisfy the constraints, $\boldsymbol{\Phi}(t_0, \mathbf{q}_0) \approx \mathbf{0}$, $\dot{\boldsymbol{\Phi}}(t_0, \mathbf{q}_0, \mathbf{v}_0) \approx \mathbf{0}$. The complete solution can thus be obtained by integrating the system dynamics successively between discrete time instants. In order to illustrate salient aspects of the implementation, a numerical integration of (2.16) is presented based on a forward Euler method, as follows:

1. Based on $\mathbf{q}_n = \mathbf{q}(t_n)$ and $\mathbf{v}_n = \mathbf{v}(t_n)$, known from the previous integration step (or defined by initial conditions for the starting step), one can evaluate: $\mathbf{Q} = \mathbf{Q}(t_n, \mathbf{q}_n, \mathbf{v}_n)$, $\mathbf{M} = \mathbf{M}(t_n, \mathbf{q}_n)$, $\mathbf{A} = \mathbf{A}(t_n, \mathbf{q}_n)$, $\mathbf{b}_q = \mathbf{b}_q(t_n, \mathbf{q}_n)$, $\mathbf{b}_v = \mathbf{b}_v(t_n, \mathbf{q}_n, \mathbf{v}_n)$, $\boldsymbol{\Phi} = \boldsymbol{\Phi}(t_n, \mathbf{q}_n)$ and $\dot{\boldsymbol{\Phi}} = \dot{\boldsymbol{\Phi}}(t_n, \mathbf{q}_n, \mathbf{v}_n)$. The upper triangular Cholesky factor of the mass matrix $\mathbf{R}$ is computed, where $\mathbf{M} = \mathbf{R}^T\mathbf{R}$, and the pseudoinverse $\mathbf{C}^+$ is computed based on $\mathbf{C} = \mathbf{A}\mathbf{R}^{-1}$.

2. Using $\mathbf{Q}$ and exploiting the Cholesky factorization one can efficiently solve the unconstrained acceleration $\mathbf{a}_n$ from $\mathbf{R}^T\mathbf{R}\mathbf{a}_n = \mathbf{Q}$ with a successive forward and backward substitution.

3. The endpoint position is computed from:

$$\mathbf{q}_{n+1} = \mathbf{q}_n + \mathbf{v}_n dt + \mathbf{R}^{-1}\mathbf{C}^+[(\mathbf{b}_q - \mathbf{A}\mathbf{v}_n)dt - \boldsymbol{\Phi}], \tag{2.20}$$

   and the endpoint velocity from:

$$\mathbf{v}_{n+1} = \mathbf{v}_n + \mathbf{a}_n dt + \mathbf{R}^{-1}\mathbf{C}^+[(\mathbf{b}_v - \mathbf{A}\mathbf{a}_n)dt - \dot{\boldsymbol{\Phi}}]. \tag{2.21}$$

   At the end of an integration step, the new position and velocity $(\mathbf{q}_{n+1}, \mathbf{v}_{n+1})$ is obtained. These values are used to initialize the next integration cycle.

The presented method assures that neither use of initial conditions which do not exactly satisfy the constraints, the roundoff error (cased by imperfect arithmetics), nor the truncation error (made by discretization) can cause constraint drift along the time integration. This however, does not mean that inconsistent initialization is preferred. Namely, poorly selected $\mathbf{q}_0$ and $\mathbf{v}_0$ will cause intensive corrections at the beginning of the integration which may alter the dynamic evolution of the system

over time. To avoid this effect, it is important to take care of correct initialization using approximately (numerically) consistent initial conditions, $\mathbf{\Phi}(t_0, \mathbf{q}_0) \approx \mathbf{0}$, $\dot{\mathbf{\Phi}}(t_0, \mathbf{q}_0, \mathbf{v}_0) \approx \mathbf{0}$, [66], [67].
Let us mention that due to $1/dt$ in (2.6), the compensation terms may make the governing equation (2.16) stiff. However, this term is canceled after time discretization as it is shown in (2.20) and (2.21).

If the constraints are indepndent, the pseudoinverse used in the numerical procedure is explicitely defined $\mathbf{C}^+ = \mathbf{C}^T(\mathbf{C}\mathbf{C}^T)^{-1}$. In general hovever, $\mathbf{C}^+$ requires a singular value decomposition of $\mathbf{C}$. This computation may be relatively expensive, although it allows one to resolve kinematic singularities and generally handle dynamical systems with dependent constraints. For high degree-of-freedom multibody systems however, reducing the computational effort needed for time integration could become crucial. In this case, one may favor Cartesian or "natural" coordinates [68], [63], [69], and use topological based approaches to exploit the structure as well as the sparsity pattern of the formulation [70], [71].

In order to improve accuracy and/or numerical stability, instead of the presented simple scheme, more sophisticated explicit or implicit discretization can be used as required. Implicit integrators are computationally more expensive, but also more stable and are usually required if the equations of motion are stiff. Practically, an implicit solver would use Newton iteration to obtain the positions and velocities at each time step. This iterative process can be sped up by exploiting the sparsity pattern of the Jacobian used in Newton's method as is proposed in [72]. Note however that due to its computational expense, solving (2.16) with an implicite solver is only resonable if the constraints are independent, in which case no singular value decomposition is required to compute $\mathbf{C}^+$.

Related Simulation Methods

In this section, established DAE integration methods frequently used to prevent constraint drift are discussed with respect to the presented equation (2.16). An overall review of constraint enforcement approaches can be found in [73] and [74].

Baumgarte's constraint stabilization

The main issue in all DAE problems is ensuring that the small numerical error made at each integration step does not accumulate along the solution process. One of the most popular methods in engineering practice, which does not require iterative constraint corrections, is Baumgarte's constraint stabilization [44]. Practically, instead of using an original constraint $\Phi = 0$, Baumgarte proposes the use of a corresponding second order equation, $\ddot{\Phi} + 2\alpha\dot{\Phi} + \beta^2\Phi = 0$ which for $\alpha = \beta > 0$ has a globally asymptotically stable aperiodic solution approaching zero ($\Phi = 0$) over time. As has been frequently pointed out in the literature, the introduced parameters, α and β must be carefully selected, since the selection can make the reformulated problem stiff and also can alter the original dynamics of the system under consideration. In (2.16), the correction terms are derived to minimally alter the dynamics under numerically imperfect conditions without resulting in stiff equations. This was achieved by means of (2.18), (2.19), without introducing extraneous parameters like α and β.

Various modifications of Baumgarte's idea can be found in [75], [76], [77], [78]. Specifically, the approach proposed by Asher *et.al* [77] is based on a single correction step (Newton iteration step) towards the position and velocity constraint manifold (2.3) applied after each time step. In the present paper, the velocity and acceleration level constraint set (2.6) can also be seen as a Newton iteration algorithm for (2.3). However, in the present approach, the correction terms embedded in (2.16), are derived to minimally alter the uncorrected solution in a kinetic and acceleration energy sense according to (2.18) and (2.19), and as such they cannot be obtained by directly solving (2.6) as proposed in [77]. Moreover, instead of post-correcting the computed solution, one can recognize in (2.20), (2.21) that with explicit discretization, the correction terms derived here perform pre-correction.

Iterative correction approach

In contrast to Baumgarte's approach, a variety of other methods have been developed which eliminate constraint drift by iteratively correcting the already computed solution.

The penalty based Augmented Lagrangian (AL) formulation introduced by Bayo and Ledesman [51] works with redundant constraints and singular mass matrices. In order to deal with constraint violation, a "mass-orthogonal" projection method was formulated on the acceleration, velocity and position level. Based on results presented in [51], the method can provide numerically perfect constraint satisfaction along relatively large step integrations. The motivation of the mass-orthogonal algorithm is to have the same matrix for the dynamic equations and also for the iterative constraint correction process.

The correction approach proposed in this paper can also be seen as mass-orthogonal. However, the idea presented herein is based on Gauss's principle, and as such it is free of the auxiliary (penalty) parameter one needs to specify on the AL formulation.

An alternative two-step decoupled position and velocity level constraint correction algorithm had also been proposed by Blajer [52]. This approach is built on the geometric interpretation of the constrained motion [79]. The method assumes a full-rank constraint matrix and as such it cannot be used for simulation of dynamical systems with redundant constraints.

Aghili [53] presented an efficient formulation for the constraint motion problem introducing a "constraint inertia matrix". In order to satisfy the (possibly dependent) position constraints, he proposed a geometrically motivated correction method based on the pseudoinverse of the constraint matrix.

The approach presented in this paper does not uses the pseudoinverse of the original constraint matrix, but rather is based on the inertially weighted pseudoinverse, which allows dynamically consistent constraint correction (i.e., based on Gauss's principle).

Coordinate partitioning method

In many practical applications, using dependent coordinates $\mathbf{q}$ with differential-algebraic equations (2.1), (2.2) is a convenient and natural way to model constrained dynamical systems. However, having constraints, and as such dependent coordinates, is the primary reason for the constraint drift along a numerical integration. Theoretically, this problem can be overcome by analytical reformulation of the mathematical model, embedding all constraints in (2.1) by specially selected (independent) generalized coordinates. This kind of reformulation of the original DAE problem to a corresponding ODE is often nontrivial or even impossible in practice. In this light, Wehage and Haug [80] proposed a more practical coordinate partitioning in order to separate the dependent coordinates from the independent ones. This partitioning, although non-trivial and not unique, exactly eliminates the drift at the velocity level from the integration, and allows DAE problems to be solved accurately using a correction only on the position level.

Differential-algebraic approach

In order to incorporate the motion constraints to the governing equation, Gear [60] developed an index reduction method. Instead of solving the original index-3 DAE problem (2.1), (2.2), he proposed an alternative index-2 DAE formulation, [45], where the velocity level constraints, $\mathbf{A}\dot{\mathbf{q}} = \mathbf{b}_q$ were directly embedded in the equation of motion. In contrast to the Udwadia-Kalaba approach, Gear did not eliminate the Lagrangian multipliers but rather calculated these in each time step. The integration was based on the Backward Differentiation Formula combined with Newton iterative correction of the original constraints $\mathbf{\Phi}$. Similar methods have also been developed by [46], [47], [50].

In the presented approach, the Lagrangian multipliers are not computed, which allows us to take redundant constraints into account. On the other hand, instead of using the velocity (and acceleration) level constraints in standard form, we have used (2.6), which allows (2.16) to prevent error accumulation without iteration.

Note that Eich [50] has proposed a quadratic minimization based method which corrects the computed solution by projecting it to the position and velocity constraint (2.3) after each time step. The difference between the approach suggested in [50] and the method presented herein is twofold; first, the minimization we propose is performed in an acceleration and kinetic energy sense, and second, the velocity and acceleration level constraints utilized in this paper allows the correction approach to be explicitly incorporated in equation (2.16).

Taking the numerical error sources into account

The importance considering the finite word length of the computational platform, as well as the inconsistency of the initial data was also pointed out by ten Dam [49]. In contrast to [60], ten Dam argued that the index of the DAE is not what causes difficulties in the solution, but rather the order of steps one takes to obtain the discrete formulation. Instead of discretizing the analytically derived equations, ten Dam proposed deriving the discrete Lagrange multipliers with the primary objective of forcing the solution to satisfy the constraints at each time step. It was shown that the discrete multipliers are not equal to the discretized version of the analytically derived multipliers, which is considered the main reason for the numerical instability experienced by standard approaches.

From our viewpoint, the primary reason of the error accumulation and constraint drift lies in the standard constraint representation (2.7). Namely, it assumes $\mathbf{\Phi} = \mathbf{0}$, $\dot{\mathbf{\Phi}} = \mathbf{0}$ along the numerical solution, and as such eliminates the information from the drift. Under this assumption, correction of the constraint drift is not possible. Taking the finite word-length of the computational environment into account, we accept $\mathbf{\Phi} \approx \mathbf{0}$, $\dot{\mathbf{\Phi}} \approx \mathbf{0}$, and use the velocity and acceleration level constraints independently as is proposed with (2.6). In this way the constraint drift can naturally and automatically be eliminated.

On the presented method

In order to obtain unique Lagrange multipliers, it is traditionally assumed that the constraint matrix **A** is of full rank. In real simulation however, one cannot guarantee this property through the dynamical evolution of the constrained motion. Particularly, if the constraint set becomes dependent, **A** loses rank, and the Lagrange multipliers cannot be uniquely calculated, and as such the numerical simulation fails. By this reason, the approach presented here does not require the Lagrange multipliers to be computed, instead, it uses the generalized constraint force which is always well defined.

All the dynamical simulations which seek precise constraint satisfaction implement some type of correction algorithm. This correction process alters the dynamic evolution of the system and can make it depart from the expected natural behavior over time. In order to minimize this effect, the proposed correction terms are derived to minimally alter the motion in an acceleration and kinetic energy norm sense.

In contrast to the frequently used iterative type constraint corrections, the equation of motion (2.16) proposed here, does not require any iteration, which may be highly preferred in real time applications. The noniterative constraint correction is achieved by directly embedding (2.6) in the equation of motion. Practically, this allows us to obtain one corrective step (toward the position and the velocity level constraints) at each time step. Note, however, that because only one corrective step is allowed, using initial conditions which significantly violate the constraints will result in a low accuracy solution. In this light, it is important to use initial conditions which at least approximately satisfy the constraints and thus can be accepted as constraint consistent in a numerical sense.

Finally, it is important to mention that utilizing the idea presented in [37], (2.16) can also be generalized to accommodate nonholonomic and nonideal constraints.

Application

In this section, the proposed formulation (2.16) is first tested on relatively simple dynamical systems, then on a more complex one. Specifically, in order to assess the accuracy of the solution, examples are chosen which are sufficiently simple to allow formulation of the dynamics without explicitly imposed constraints, which in turn enables an ODE formulation of the dynamics. The ODE formulation and subsequent solution via an ODE solver, which is termed the "trusted solution", is used in these cases to compare the results of the respective DAE solution (2.16) termed as the "proposed solution" (both under the same numerical conditions). The authors also compare the DAE solutions obtained by (2.16) and (2.17) (i.e., with and without constraint error correction) for constraint drift in the presence of "perfectly" consistent and inconsistent initialization. After comparing solutions for the simple examples, a seven-link biped (in three different constraint configurations), which is too complex to be formulated as a single ODE problem, is used as a "realistic" application of the proposed method.

The presented examples are solved with a fourth order fixed step Runge-Kutta method. The solver is implemented in MATLAB and compiled to a C code. Using this code, the simulations are performed on 2.4 MHz Intel Core2 Quad PC computer with a fixed time step. The selected integration step preserves stability of the explicit Runge-Kutta integrator. The error of the reported numerical solutions are measured with respect to the "numerically exact solution" obtained using the ODE formulation integrated with a MATLAB solver (with 10^{-12} relative and absolute tolerance). All physical quantities used in the simulations have standard SI units $[kg, m, s]$.

Mathematical Pendulum

A pendulum with mass m and length l is chosen to test the proposed method over a long time period simulation $t \in [0, 1000]s$. The constrained equation of motion, (2.16), is derived using two (dependent) coordinates $\mathbf{q} = [x, y]^T$ and one constraint $\Phi = x^2 + y^2 - l^2$. Starting from the horizontal rest position, the motion is simulated using 10^{-3} time step. Compared to the "numer-

ically exact" position of the pendulum, the simulation results show the same accuracy, 3×10^{-7} (i.e., worst error magnitude), for the "proposed solution" and the "trusted solution", indicating that this error is not due to the constrained formulation but is rather generated by discretization. On the other hand, using (2.17), (with no constraint correction) results in a low accuracy solution, 10^{-1}, as expected. In order to identify the importance of the correction terms, the simulations were repeated with slightly imperfect initial conditions. The result is depicted in Figure 2.1. Note

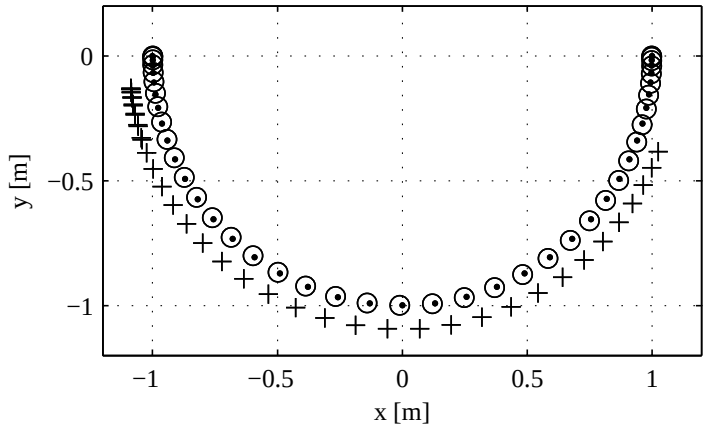

Figure 2.1 Mathematical pendulum: $l = 1m$, $m = 1kg$. Last swing in $t \in [0,1000]s$ simulation is depicted. The "trusted solution" started from the rest horizontal position $\mathbf{q}(0) = [1,0]^T$, $\mathbf{v}(0) = [0,0]^T$, is plotted with "$\cdot$". The solution obtained by (2.16), with imperfect initial conditions $\mathbf{q}(0) = [1+10^{-5}, 10^{-5}]^T$, $\mathbf{v}(0) = [10^{-4}, -10^{-4}]^T$, is depicted with "$\circ$". The solution obtained by (2.17) with the same imperfect initialization is plotted with "$+$". This solution violates the constraint and is shifted in time.

that despite the imperfect initialization, no error accumulation took place in the proposed solution. Correspondingly, (except the first few steps) the constraint is satisfied up to, 10^{-12}, such that the solution is practically free of drift. Let us mention that integration of (2.16) over $1000s$ took $13.7s$ CPU time, 26% of which was spent on constraint correction.

Since in the present context an explicit solver is utilized, the overall accuracy of the integrated solution (accuracy compared to the numerically exact solution) as well as the accuracy of the constraint

satisfaction is step-size dependent. A representative relation between the integration step-size and the mentioned accuracy measures is illustrated in Figure 2.2. As one can recognize, with a small enough time-step, the overall accuracy obtained with the proposed formulation is the same as that obtained by integration of the unconstrained formulation. The presented numerical result also verifies that with a larger time-step, the constrained formulation gives a less precise result, namely the constraint error (although steadily maintained) is not in the order of the machine precision.

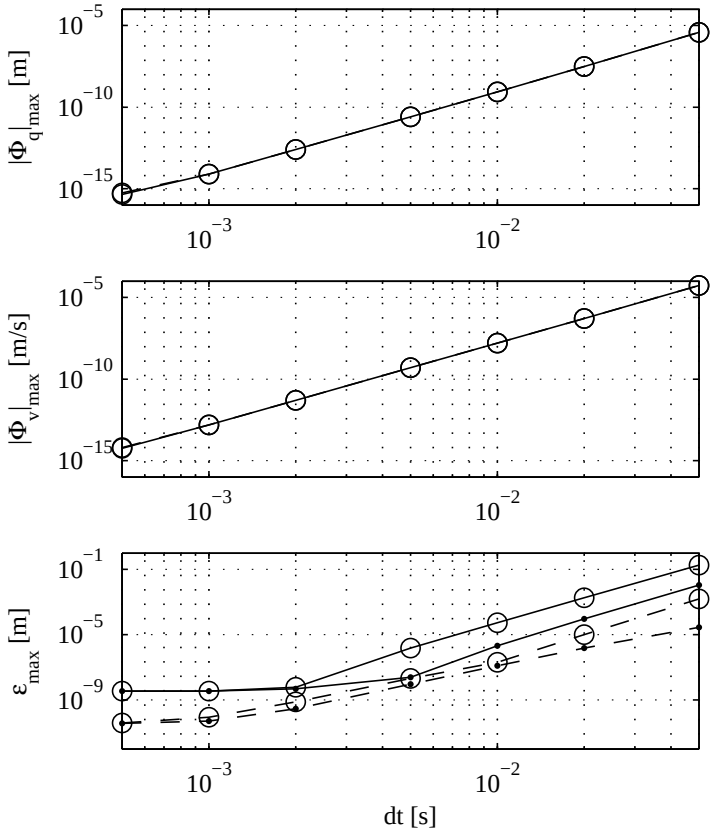

Figure 2.2 In this figure, log-log pictures depict the maximum constraint error $\left|\Phi_q\right|_{max}$, $|\Phi_v|_{max}$, and the maximum overall accuracy ε_{max}, $\varepsilon = \sqrt{dx^2 + dy^2}$ (where dx and dy are errors in the corresponding coordinates) between the "trusted solution", "$\cdot$", and the proposed solution, "$\circ$". The presented results are based on integration conducted for $t = [0,10]s$ (dashed line) and $t = [0,100]s$ (full line) with seven different time steps: $dt \in [5\times10^{-4}, 10^{-3}, 2\times10^{-3}, 5\times10^{-3}, 10^{-2}, 2\times10^{-2}, 5\times10^{-2}]$. In the first two figures, the full and the dashed lines are overlapped, indicating that the two solutions have the same accuracy on the constraint satisfaction.

Slider-crank mechanism

Consider the slider-crank mechanism consisting of two links with equal lengths l and masses m, and a horizontal slider, with mass m_s, attached to the end of the mechanism Figure 2.3. When the links are either horizontal or vertical the mechanism is in a singular configuration. Our intention is to test (2.16) under this singularity. To make the motion periodically cross the singular positions, the base link is subjected to a constant counterclockwise torque of $20Nm$ and a torsional (linear) viscous damping with damping coefficient $5Nms$. The constrained formulation is derived with four (dependent) coordinates $\mathbf{q} = [x_1, y_1, \theta_1, \theta_2]^T$ and three constraints $\mathbf{\Phi} = [x_1, y_1, y_1 + l(sin(\theta_1) + sin(\theta_2))]^T$. Here, (x_1, y_1) are coordinates of the support point while θ_1 and θ_2 are absolute angles of the links measured counterclockwise from a horizontal reference. The motion, started from a horizontal rest position, is simulated over $t \in [0, 100]s$ with 10^{-3} time step. Compared to the numerically exact solution, the motion reflected to the horizontal position of the slider $x_s = x_1 + l(cos(\theta_1) + cos(\theta_2))$, shows the same accuracy, 4.4×10^{-8}, for the proposed solution and the trusted solution. The equation (2.16) is also tested under inconsistent initialization. The simulation results are depicted in Figures 2.3 and 2.4.

Two four-bar linkages

Let us consider two four-bar linkages with links of length l and distributed masses m. When the mechanism moves through a horizontal position, its number of degrees of freedom changes instantaneously from one to three. The intention here is to test (2.16) under this constraint singularity. The equation of motion is derived with six natural coordinates, [81], which define the position of the moving joints, $\mathbf{q} = [x_1, y_1, x_2, y_2, x_3, y_3]^T$, and five constraints $\mathbf{\Phi} = [x_1^2 + y_1^2 - l^2, (x_2 - l)^2 + y_2^2 - l^2, (x_3 - 2l)^2 + y_3^2 - l^2, (x_2 - x_1)^2 + (y_2 - y_1)^2 - l^2, (x_3 - x_2)^2 + (y_3 - y_2)^2 - l^2]^T$. The motion, started from a vertical rest position of the supporting links, is simulated over $t \in [0, 100]s$ with 10^{-2} time step. Due to the relatively large time step and long simulation time we do not expect a precise solution. Accordingly, compared to the exact numerical solution, the result reflected to x_3 shows

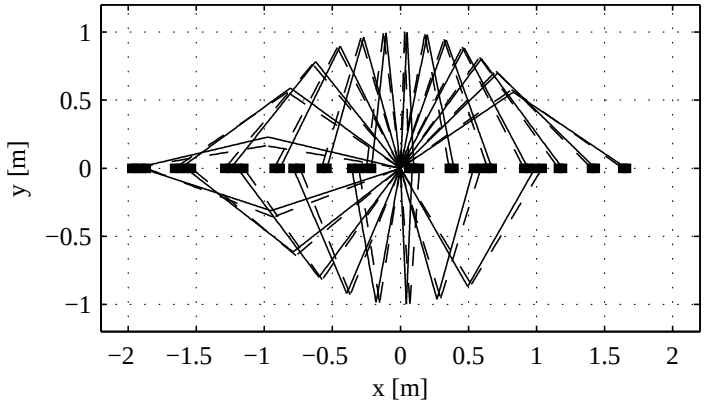

Figure 2.3 Slider-crank mechanism: $l = 1m$, $m = 1kg$, $m_s = 1kg$. Stroboscopic view of the motion for $t \in [97.5, 99]s$. The solution obtained by (2.16) with exact initialization $\mathbf{q}(0) = [0,0,0,0]^T$, $\mathbf{v}(0) = [0,0,0,0]^T$, is depicted with "$-$". The numerical integration took 18.9s CPU time, 5% of which is spent on constraint correction. The motion predicted by (2.16) under slightly inconsistent initialization $\mathbf{q}(0) = [10^{-3}, 10^{-2}, 10^{-3}, 10^{-2}]^T$, $\mathbf{v}(0) = [10^{-3}, 10^{-2}, 10^{-3}, 10^{-2}]^T$ is plotted with "$--$". The solution obtained by (2.17), (with no constraint correction), was highly inaccurate such that we decided not to present it here.

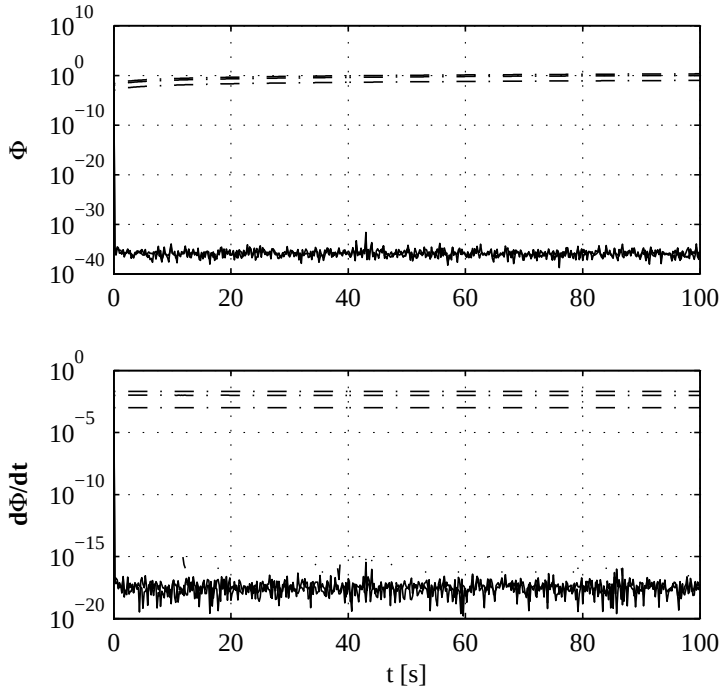

Figure 2.4 Constraint evolution along the solution obtained by (2.16) is plotted with "$-$". Note that all constraints are below the machine precision 10^{-15}. Constraint evolution along the solution obtained with no constraint correction is depicted with "$-\cdot-$". Both depicted solutions started with inconsistent initialization.

10^{-2} maximal error in the proposed solution and 2×10^{-4} error in the trusted solution. Note that, in numerical simulations with a time step, 10^{-3}, the proposed and the trusted solution possesed the same accuracy level, $\approx 10^{-7}$, as that obtained in the previous examples. With the 10^{-2} time step, integration of (2.17) with no constraint correction was unstable. The constrained equation (2.16), was also tested under inconsistent initialization. The results are depicted in Figures 2.5 and 2.6.

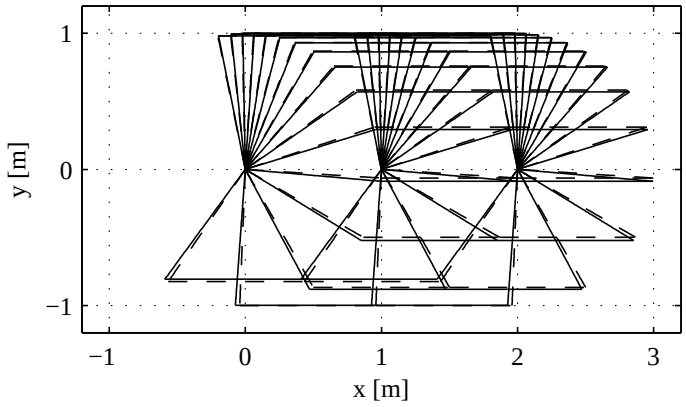

Figure 2.5 Two four-bar linkages: $l = 1m$, $m = 1kg$ for each link. Stroboscopic view of the motion for $t \in [95, 96.25]s$ obtained by (2.16) with exact initialization $\mathbf{q}(0) = [0,1,1,1,2,1]^T$, $\mathbf{v}(0) = [1,0,1,0,1,0]^T$, is depicted with "$-$". The integration took $5.2s$ CPU time. The motion predicted by (2.16) under slightly inconsistent initialization $\mathbf{q}(0) = [0, 1+10^{-2}, 1, 1+10^{-2}, 2-10^{-2}, 1]^T$, $\mathbf{v}(0) = [1-10^{-2}, 0, 1, 0, 1+10^{-2}, 0]^T$ is plotted with "$--$".

Trajectory tracking control

Motivated by a recent development in trajectory tracking control [82], our intention is to show how (2.16) can be used to simulate dynamical systems which perfectly (rather than approximately) track a predefined reference trajectory. According to the classical philosophy of tracking control, we assume that the motion of the considered dynamical system is guided with kinematic constraints interpreted as control objectives. This view will allow us to embed all predefined refer-

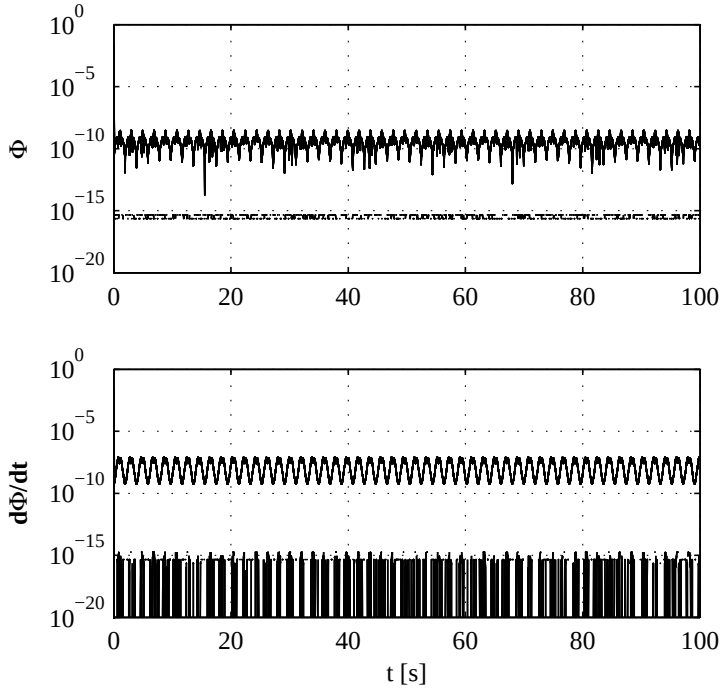

Figure 2.6 Constraint evolution along the solution obtained by (2.16) using inconsistent initialization. Due to the large integration step, 10^{-2}, some constraints are not satisfied on the order of the machine precision. Nevertheless, the constraint drift is not growing through the integration.

ence trajectories in the constraint set (2.6). In the present context, this approach will allow perfect satisfaction of the control objectives without application of the Baumgartne's constraint stabilization method as was used by [82]. The idea is presented by simulating three "exercise" motions of a planar biped robot.

Consider a 7-link planar biped robot depicted in Figure 2.7, with height $L = 1.8m$, mass $M = 75kg$ and anthropometric geometric properties and mass distribution according to Table 2.1, [3]. The configuration of the biped is defined with nine absolute coordinates $\mathbf{q} = [x, y, \theta, \theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6]^T$. This coordinate set is independent for the "flying biped" while it becomes dependent if constraints are applied on the robot.

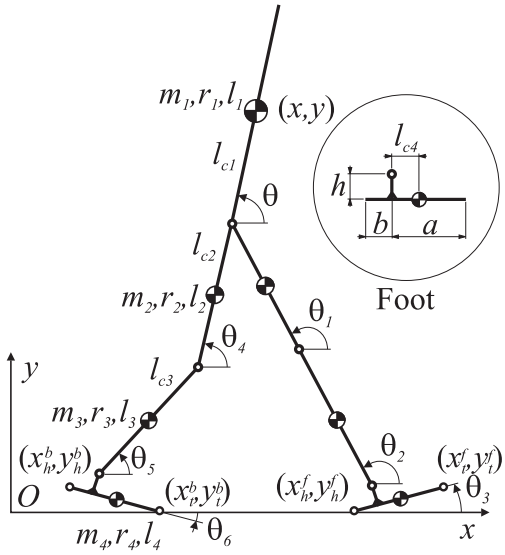

Figure 2.7 Biped model with generalized coordinates and associated geometric and inertia properties. For each segment, the moment of inertial with respect to the center of mass is calculated as $I_* = m_* r_*^2$.

First exercise

Let us consider the biped restricted with nine kinematic constraints, specifically, six physical restrictions due to the ground contact of both feet, and three additional control objectives which specify the upper body angle and also define periodic flexion of both knees. The constraint set is given by: $\mathbf{\Phi} = [x_t^f - x_{td}^f, y_t^f, y_h^f, x_t^b - x_{td}^b, y_t^b, y_h^b, \theta - \theta_d, \varphi_k^f - \varphi_{kd}^f, \varphi_k^b - \varphi_{kd}^b]^T$ where; $x_{td}^f = 0.507$, $x_{td}^b = -0.097$ define the desired horizontal position for the toes on the forward and backward foot, $\theta_d = 4\pi/9$ is the desired upper body angle, $\varphi_k^f = \theta_1 - \theta_2$ is the relative angle at the forward knee while $\varphi_{kd}^f = (\pi/6)(1 - cos(\pi t))$ defines its desired motion, $\varphi_k^b = \theta_4 - \theta_5$ is the relative angle at the backward knee with its desired motion defined by $\varphi_{kd}^b = (\pi/10)(1 - cos(\pi t))$. The simulation result is depicted in Figure 2.8.

Table 2.1 Geometric and inertial parameters of the biped.

Description	no. $(*)$	l_*/L	l_{c*}/l_*	m_*/M	r_*/l_*
Upper body	1	0.288	0.626	0.6780	0.496
Thigh	2	0.245	0.433	0.1000	0.323
Shank	3	0.246	0.433	0.0465	0.302
Foot	4	0.152	0.250	0.0145	0.475
Foot geometry			a/l_4	b/l_4	h/L
			0.75	0.25	0.039

Second exercise

Using the same biped model one can apply different constraints to generate a required motion. In this example, we apply four physical constraints which will hold the forward heel and the backward toe to remain on ground while the motion of the robot is dictated with five control constraints. Practically, we define the upper body angle $\theta_d = 4\pi/9$, the angular motion of the feet $\theta_{3d} = (\pi/18)(cos(\pi t)+1)$, $\theta_{6d} = -(\pi/18)(1-cos(\pi t))$ supported with periodic flexion of the backward knee $\varphi^b_{kd} = (\pi/10)(1-cos(\pi t))$ and full extension of the forward leg. Accordingly, the constraint set is given by: $\mathbf{\Phi} = [x^f_h - x^f_{hd}, y^f_h, x^b_t - x^b_{td}, y^b_t, \theta-\theta_d, \theta_3-\theta_{3d}, \theta_6-\theta_{6d}, \varphi^f_k, \varphi^b_k - \varphi^b_{kd}]^T$, where $x^f_{hd} = 0.262$, $x^b_{td} = -0.038$. A stroboscopic view of the simulated motion is depicted in Figure 2.9.

Third exercise

The simulated motion here represents a balancing exercise with parallel legs while only the toes are on the ground. The constraint set is given by: $\mathbf{\Phi} = [x^f_t, y^f_t, x^b_t, y^b_t, \theta-\theta_d, \varphi^f_k - \varphi^f_{kd}, \varphi^b_k - \varphi^b_{kd}, \theta_3-\theta_{3d}, \theta_6-\theta_{6d}, x_{CoM}]^T$, where $\theta_d = 4\pi/9$, $\varphi^f_{kd} = \varphi^b_{kd} = (\pi/3)(1-cos(2\pi t/5))$, $\theta_{3d} = \theta_{6d} = -(\pi/27)(1-cos(2\pi t/5))$ while the center of mass of the biped is kept above the toes, $x_{CoM} = 0$. Note that the constraint set contains ten relations while the system is described with nine coordi-

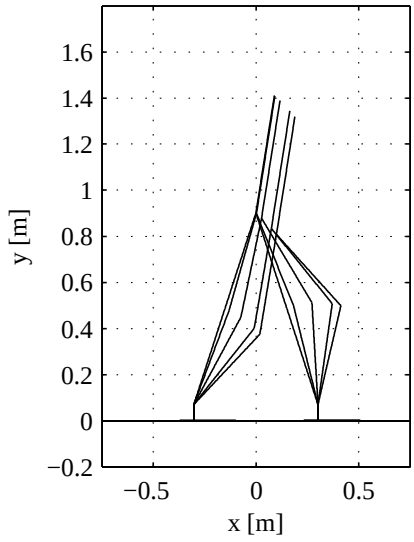

Figure 2.8 The simulation is performed with 10^{-2} time step over $t \in [0, 100]s$ with $\mathbf{q}(0) = [0.056, 1.220, 1.396, 1.920, 1.920, 0, 1.222, 1.222, 0]^T$ and $\mathbf{v}(0) = \mathbf{0}$. Along the motion, all physical and control constraints are satisfied up to 5.2×10^{-11}. The corresponding configurations in 50 successive depicted periods are overlapped.

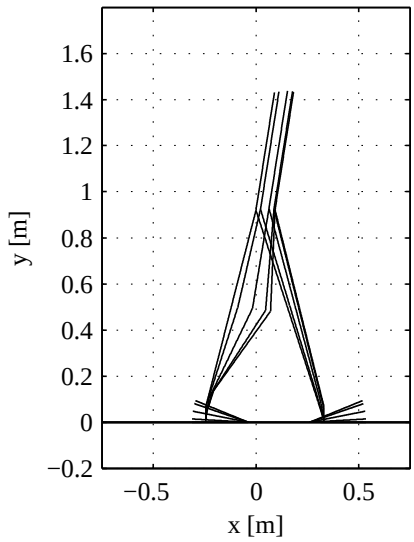

Figure 2.9 The simulation is performed with 10^{-2} time step over $t \in [0, 100]s$ with $\mathbf{q}(0) = [0.056, 1.239, 1.396, 1.920, 1.920, 0.349, 1.292, 1.292, 0]^T$ and $\mathbf{v}(0) = \mathbf{0}$. The depicted 50 motion cycles show that the corresponding configurations in successive periods are overlapped. All constraints are satisfied up to 2.7×10^{-11}.

nates. In this light, regardless of the configuration of the biped, the constraint set is redundant at each time instant. Stroboscopic view of the balancing exercise is plotted in Figure 2.10.

The reference trajectories in the above three simulations are selected such that the system cannot

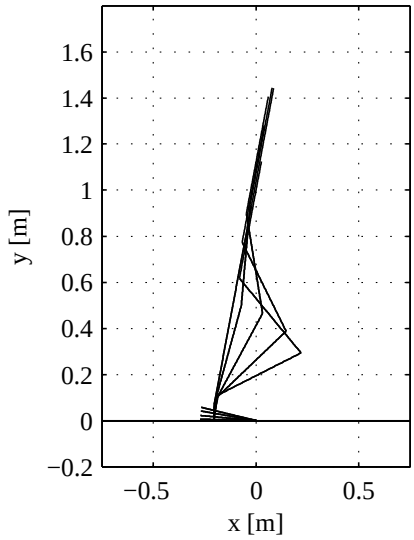

Figure 2.10 The simulation is performed with 10^{-2} time step over $t \in [0, 100]s$ with $\mathbf{q}(0) = [0.046, 1.252, 1.363, 1.363, 1.363, 0, 1.363, 1.363, 0]^T$ and $\mathbf{v}(0) = \mathbf{0}$. The stroboscopic view of the whole simulation, 20 cycles, shows that the configuration of the robot in successive periods are overlapped. All constraints are satisfied up to 1.6×10^{-12}.

experience impacts (i.e., the necessary smoothness assumption required on $\mathbf{\Phi}$ is not violated). Using an appropriate impact resolution algorithm, the presented method could be applied to simulate nonsmooth motion such as bipedal walking. However, resolving impacts for multiple constraint dynamical systems is outside of the scope of this paper [55].

Finally, we would like to point out that by embedding a reference trajectory into the constraint set, we assumed that the reference motion could be enforced with an "ideal" control force which satisfies D'Alambert's principle. This assumption, however, may generate a controllability issue on the trajectory tracking problem, as was pointed out by [83].

Conclusion

In order to simulate a constrained dynamical system, a governing equation (2.16) with embedded constraint correction terms is derived. This equation has a numerically stable implementation and allows the analyst to obtain a simulated solution over long time periods of constrained dynamical systems using simple generalized coordinates and standard ODE solvers. The presented formulation exploits the pseudoinverse of the constraint matrix, and as such, can also be used under dependent constraints and kinematic singularities. Although the idea is presented from the standpoint of classical mechanics, one can use it to simulate various physical systems modeled with differential algebraic equations.

CHAPTER III

MANUSCRIPT 2: A CONTROL APPROACH FOR ACTUATED DYNAMIC WALKING IN BIPED ROBOTS

By

David J. Braun and Michael Goldfarb

Vanderbilt University

Nashville, TN

Published as a Regular Paper in the

IEEE Transactions on Robotics, DOI: 10.1109/TRO.2009.2028762

This paper has supplementary multimedia material available at http://ieeexplore.ieee.org, provided by the authors. This material, Dynamic Walking Video.wmv, demonstrates the dynamic bipedal walk coordinated by the proposed walking controller. The video can be played with Windows Media Player. The total size is 4.5MB.

Abstract

This paper presents an approach for the closed-loop control of a fully-actuated biped robot that leverages its natural dynamics when walking. Rather than prescribing kinematic trajectories, the approach proposes a set of state-dependent torques, each of which can be constructed from a combination of low gain spring-damper couples. Accordingly, the limb motion is determined by interaction of the passive control elements and the natural dynamics of the biped, rather than being dictated by a reference trajectory. In order to implement the proposed approach, the authors develop a model-based transformation from the control torques defined in a mixed reference frame to the actuator joint torques. The proposed approach is implemented in simulation on an anthropomorphic biped. The simulated biped is shown to converge to a stable, natural-looking walk from a variety of initial configurations. Based on these simulations, the mechanical cost of transport is computed and shown to be significantly lower than trajectory tracking approaches to biped control, thus validating the ability of the proposed idea to provide efficient dynamic walking. Simulations further demonstrate walking at varying speeds and on varying ground slopes. Finally, controller robustness is demonstrated with respect to forward and backward push-type disturbances and with respect to uncertainty in model parameters.

Introduction

The zero moment point (ZMP) approach, is perhaps the most comprehensively developed in the biped locomotion control literature [5], [84], [9], [85], [86], [87]. Methods based on this approach have been shown to provide effective, robust, and versatile locomotion for biped robots. Despite their effectiveness, ZMP approaches generally result in a stiff and unnatural looking gait with low locomotive efficiency [11], [88]. The principal reason that these approaches appear stiff and have a low locomotive efficiency is that they are based on the trajectory tracking, and therefore (by definition) override the natural dynamics of the robot (i.e., position-level information is dictated by the controller, and thus integration of the inertial dynamics is not an essential part of the motion).

Such reshaping of the natural dynamics is energetically expensive. By contrast, humans (which are characterized by natural looking gait with high locomotive efficiency) have been shown to leverage the natural dynamics of their limbs when walking (e.g., [89]).

In order to achieve a more efficient and natural-looking bipedal gait, several researchers have investigated dynamic walking approaches that, like humans, leverage rather than override the limb dynamics of the robot. As defined herein, a dynamic walker is one in which the motion of the walker is not dictated substantially by the controller, but rather is influenced significantly by the gravitational and inertial characteristics of the system. As such, neither a predefined reference trajectory nor any other time or position-based attribute of the walking cycle (i.e., desired walking speed, stepping frequency or step length) can be enforced by control. Rather, all such gait characteristics are obtained indirectly by the interaction between the dynamics of the robot and environment and the influence of joint torques (i.e., from the combined influences of the joint torques and natural dynamics). Implicit in this definition is that the limb dynamics play a significant role in determining the joint angle trajectories. This definition also implies that the joints should be backdrivable such that power can flow freely and bi-directionally between the limb load and the actuator. Note that the phrase dynamic walking is also used to describe a biped gait that is dynamically (as opposed to statically) stable ([90], [10]), although that is not the meaning used herein.

Prior work on dynamic walkers includes work on both actuated and unactuated walkers. Specifically, such work describes the development of unactuated (i.e., fully passive) walkers, actuator-assisted walkers based largely on passive versions, and actuated walkers that utilize control approaches that do not dictate joint angle trajectories. Fully passive dynamic walkers do not incorporate any actuators (or control) and as such the locomotion they produce adheres to the previously given definition of dynamic walking (i.e., no motions are imposed by a controller). As such, fully passive dynamic walkers rely on precisely tuned natural dynamics of the robot, and must walk on a slight downward slope to compensate for the energetic cost of transport (i.e., they are powered by gravity). Examples of these types of walkers are described by [12], [13], [88].

Actuator-assisted dynamic walkers augment a nearly-passive walker by introducing a reduced set of actuators to overcome the energetic losses associated with gait (i.e., the walkers need not descend a slope) and to introduce some robustness to design parameter variation (via some form of feedback control). Examples of actuator-assisted walkers are described in [1], [14], and [91], the latter of which is based in part on relevant work presented in [92].

A fully actuated, partially dynamic walker is described by [93], [94], [19], [21], [20]. Specifically, the authors address a reduced order problem, which maintains balance in the walker by imposing kinematic constraints between several joint angles. In doing so, however, they violate the aforementioned definition of dynamic walking by specifying kinematic constraints. In the work described by [95], a neural network is used to learn the nominal walking trajectories generated by an impulsive control approach, then a PD controller is used to enforce these relations as state dependent constraints. Though the combination of impulsive control followed by passive dynamics is a viable approach to dynamic walking, it is not clear how much of the passive dynamics are preserved through the neural network planner and associated constraint enforcement. Pratt et al. [18] present a method that need not override the natural dynamics of the biped (depending on the choice of control parameters). The method described in [18], however, requires some limiting assumptions, namely that the biped feet remain flat on the ground and that the ankle joints remain unactuated (i.e., do not impose torque on the biped). A biologically inspired sensor and motor-neuron based approach to dynamic walking is described by [23], [24]. This approach does not utilize a trajectory tracking objective, but the extent of dynamic walking is unclear, particularly since inertial effects are largely diminished at the scale of implementation, and since the joint servos are non-backdrivable (thus they preclude bidirectional power flow in the joints, which thus precludes dynamic walking).

This paper presents a control approach that enables fully dynamic biped walking, which can provide a more efficient gait than trajectory tracking approaches. Rather than prescribing a kinematics (i.e., joint angle trajectories), the approach subjects the robot to a set of state-dependent torques. These torques are constructed from energetically passive elements (i.e., angular springs

and dampers with fixed equilibrium points), which influence the natural dynamics to generate a stable gait. Like the approach presented by [18], this work utilizes the notion of (some) control influences based in the task-space. However, the present approach relaxes all assumptions regarding robot configuration (i.e., feet need not be flat on the ground); imposes state-dependent torques generated by low gain spring-damper couples, which are constructed as strictly passive functions with fixed equilibrium points; references some of these torques to an inertial reference frame and others to the internal robot frame; and develops a model-based solution to transform the state-dependent control torques to actuator torques utilizing the Gauss principle of least constraint [30], [38].

The proposed control approach, the application of which leads to an energy efficient and natural looking dynamic walk, is described herein and subsequently demonstrated via simulation.

Biped Model

The control methodology is based upon a dynamic model of the robot introduced in this section. This model is derived by means of the Gauss principle of least constraint utilizing the Udwadia-Kalaba approach [35]. Unlike traditionally used biped models derived separately for single support, double support and flight phase, the present model offers a unified representation which is valid for all phases of gait. Compared to constrained dynamic formulations derived by means of Lagrangian equations of the first kind [96], [97], the approach presented herein provides an analytical description of the biped dynamics under redundant constraints and kinematic singularities, and as such allows the formulation of a control methodology with no restriction on biped configuration.

In order to facilitate model and controller development, both are developed in the context of a seven-link (nine degree-of-freedom) planar biped, as illustrated in Figure 3.1. The configuration of the biped is defined with the generalized coordinates, $\mathbf{q} = [x, y, \theta, \theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6]^T$, defined relative to the inertial reference frame. The biped is assumed to be actuated at each joint (i.e., right and left hip, knee, and ankle joints), such that, the dynamics of the robot are affected by the vector of actuator torques, $\mathbf{u} = [u_1, u_2, u_3, u_4, u_5, u_6]^T$, which are assumed positive in the counterclockwise

direction.

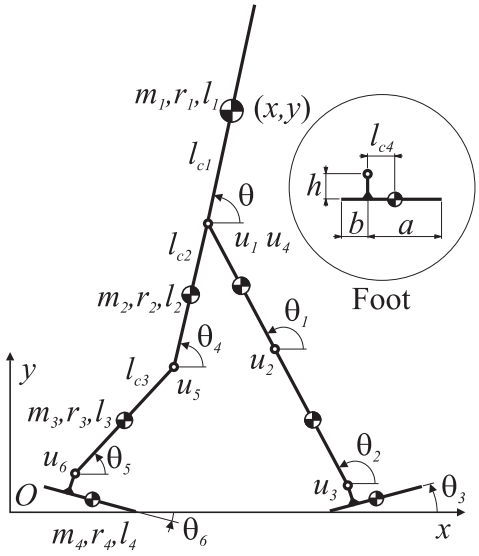

Figure 3.1 Seven-link biped with generalized coordinates and associated geometric and inertial properties. The corresponding links on both legs are geometrically and inertially identical. For each segment, the moment of inertia with respect to the center of mass of the associated link is calculated as $I_* = m_* r_*^2$.

Unconstrained Dynamics

Consider an *n*-dof autonomous multibody system, the configuration of which is uniquely specified by $\mathbf{q} \in \mathbb{R}^n$ generalized coordinates. The equations of motion, for the unconstrained "flying" biped, can be written as:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{G}(\mathbf{q}) = \mathbf{Q}_u, \tag{3.1}$$

where $\mathbf{M} \in \mathbb{R}^{n \times n}$ is a symmetric and positive definite mass matrix, $\mathbf{h} \in \mathbb{R}^n$ represents the normal and Coriolis inertial forces, $\mathbf{G} \in \mathbb{R}^n$ represents the gravitational forces, while $\mathbf{E} \in \mathbb{R}^{n \times m}$ is a matrix mapping control inputs $\mathbf{u} \in \mathbb{R}^m$ to generalized control force space $\mathbf{Q}_u = \mathbf{E}\mathbf{u}$. Note that the gener-

alized control force vector $\mathbf{Q}_u$ must belong to the range space of $\mathbf{E}$, $\mathbf{Q}_u \in \mathscr{R}(\mathbf{E})$, which indicates that, using actuator torques $\mathbf{u}$, the motion of the robot in flight phase cannot be fully prescribed. For the biped, all actuator inputs are independent, such that $rank(\mathbf{E}) = m$.

Kinematic Constraints

Along the walk, the biped is restricted with numerous physical constraints. These kinematic motion restrictions are introduced and discussed as follows.

For the biped in Figure 3.1, neither foot can penetrate the ground, the knee joints cannot extend beyond the fully straight position, and both feet are assumed not to slide when in contact with the ground. Since each toe and heel are independently characterized by non-penetration and no-slip constraint with the ground, the biped dynamics can be subject to ten (dependent) kinematic constraints. Along the walk, the kinematic constraints are "active" when imposed on the robot and "inactive" when not affecting the motion. For each (independent) active constraint, the model loses one degree of freedom. For example, when the biped is in single support phase with the stance leg foot flat on the ground, three independent constraints are active, which are the non-penetration of the toe, the non-penetration of the heel, and one no-slip condition, and as such, the biped is reduced to a six degree-of-freedom system (assuming that neither knee is fully extended). Following a general notation, the set of kinematic constraints imposed on the biped is given by:

$$\mathbf{\Phi} = [\mathbf{\Phi}_h(\mathbf{q})^T, \mathbf{\Phi}_n(\mathbf{q}, \dot{\mathbf{q}})^T]^T = \mathbf{0}, \tag{3.2}$$

where $\mathbf{\Phi}_h$ represents the holonomic constraints (e.g., the non-penetration between the toe and heel and the ground, and the full extension of the knee joint), and $\mathbf{\Phi}_n$ represents the nonholonomic constraints (i.e., the non-slip condition between each foot and the ground).

We assume that $\mathbf{\Phi}_h$ is twice and $\mathbf{\Phi}_n$ is at least once differentiable while the initial conditions

are constraint consistent. In this case, (3.2) can be equivalently represented as:

$$\mathbf{A}(\mathbf{q})\ddot{\mathbf{q}} = \mathbf{b}(\mathbf{q}, \dot{\mathbf{q}}), \tag{3.3}$$

where, $\mathbf{A} = [\mathbf{A}_h^T, \mathbf{A}_n^T]^T$ is the constraint matrix defined in terms of $\mathbf{A}_h = \partial\mathbf{\Phi}_h/\partial\mathbf{q}$ and $\mathbf{A}_n = \partial\mathbf{\Phi}_n/\partial\dot{\mathbf{q}}$, while $\mathbf{b} = \mathbf{A}\ddot{\mathbf{q}} - [\ddot{\mathbf{\Phi}}_h^T, \dot{\mathbf{\Phi}}_n^T]^T$, [36]. Note that when a constraint becomes inactive (as a function of system configuration), it is eliminated by zeroing the corresponding row in (3.3). On the other hand, when a constraint switches from inactive to active (e.g., at the ground contact events or when the knee hits a full extension stop), engagement of the constraint will impart an impact to the system dynamics. The following subsection describes the treatment of these impact events.

Modeling Impact

For the biped robot, impact occurs when the knee joint fully extends and also when each foot impacts the ground. Each impact is considered to be instantaneous and perfectly plastic. With these assumptions, and defining the pre and post-impact velocities as $\dot{\mathbf{q}}^-$ and $\dot{\mathbf{q}}^+$, respectively, the post-impact kinematic constraints can be written as:

$$\mathbf{A}_h\dot{\mathbf{q}}^+ \geq \mathbf{0}, \mathbf{A}_n\dot{\mathbf{q}}^+ = \mathbf{0}. \tag{3.4}$$

Given the no-slip assumption, we will utilize the Gauss principle of least constraint [38], [57] to formulate the effect of the impact as a constrained quadratic minimization problem as follows:

$$\begin{aligned}\dot{\mathbf{q}}^+ = min\{\dot{\mathbf{q}} \in \mathbb{R}^n : (\dot{\mathbf{q}} - \dot{\mathbf{q}}^-)^T\mathbf{M}(\dot{\mathbf{q}} - \dot{\mathbf{q}}^-),\\ \mathbf{A}_h\dot{\mathbf{q}} \geq \mathbf{0}, \mathbf{A}_n\dot{\mathbf{q}} = \mathbf{0}\}.\end{aligned} \tag{3.5}$$

Note that motion restriction in the tangential direction $\mathbf{A}_n\dot{\mathbf{q}} = \mathbf{0}$ is only active if a particular constraint does not break; however, (3.5) neglects the tangential velocity component even if a corresponding constraint breaks. This assumption, which cannot be used under "fast" impulsive re-

bound, becomes reasonable under "slow" non-impulsive constraint detachment. Practically, when bipedal walking is considered, both the knee and foot are expected to detach non-impulsively (and nearly normal to the constraint manifold), which justifies the use of (3.5). Compared to more general considerations [98], [99], [55], [100], [4], (3.5) is particularly well suited to the present context, in that it does not require computation of physical constraint forces, which may not be possible under redundant constraints and kinematic singularities.

Constrained Dynamics

Based on the Gauss principle of least constraint, [30], [101], [35], the constrained acceleration $\ddot{\mathbf{q}}$, which satisfies (3.3), can be obtained from the following quadratic programming problem:

$$\ddot{\mathbf{q}} = min\{\mathbf{x} \in \mathbb{R}^n : (\mathbf{x}-\mathbf{a})^T\mathbf{M}(\mathbf{x}-\mathbf{a}), \mathbf{A}\mathbf{x} = \mathbf{b}\}, \tag{3.6}$$

where $\mathbf{a} = \mathbf{M}^{-1}(\mathbf{Q}_u - \mathbf{h} - \mathbf{G})$ is the unconstrained acceleration (i.e., the acceleration the system would have without the imposed constraints (3.3)). According to (3.6), $\ddot{\mathbf{q}}$ minimizes the acceleration energy, $(\ddot{\mathbf{q}}-\mathbf{a})^T\mathbf{M}(\ddot{\mathbf{q}}-\mathbf{a})$, between the motion which is not restricted with the kinematic constraints and the constrained motion. Since $\mathbf{M}$ is symmetric and positive definite, the above quadratic programming problem is convex, and the solution of (3.6), $\ddot{\mathbf{q}} = \mathbf{a} + \mathbf{M}^{-1}\mathbf{A}^T(\mathbf{A}\mathbf{M}^{-1}\mathbf{A}^T)^{-1}(\mathbf{b}-\mathbf{A}\mathbf{a})$ exists and is unique. In cases in which $\mathbf{A}$ is not full rank (which is often the case in a walking biped), $(\mathbf{A}\mathbf{M}^{-1}\mathbf{A}^T)^{-1}$ will not exist. In such cases, we can find the constrained acceleration $\ddot{\mathbf{q}}$ from:

$$\ddot{\mathbf{q}} = \mathbf{a} + \mathbf{R}^{-1}\mathbf{C}^{+}(\mathbf{b}-\mathbf{A}\mathbf{a}), \tag{3.7}$$

where $\mathbf{R}$ is defined as the upper triangular Cholesky factorization of the mass matrix $\mathbf{M} = \mathbf{R}^T\mathbf{R}$, [65], $\mathbf{C} = \mathbf{A}\mathbf{R}^{-1}$, is the inertially-weighted constraint matrix, while $\mathbf{C}^+$ is the pseudoinverse (i.e., the Moore-Penrose inverse) of $\mathbf{C}$ [59]. This formulation explicitly defines the acceleration of the constrained motion, which is well defined under dependent constraints. Note that (3.7) is expressed using the Cholesky factorization of the mass matrix $\mathbf{R}$ instead of its principal square root $\mathbf{M}^{1/2}$

utilized by Udwadia and Kalaba [35], [36].

The Gauss principle of least constraint is valid for any rigid body system subjected to "ideal" constraints. Accordingly, due to the no-slip assumption, all constraints in the biped (introduced by (3.2)) are ideal, and as such, (3.7) provides a viable equation of motion which is used in the following control design.

A Control Approach for Dynamic Walking

In this section, we develop a control methodology which can be used to generate dynamic walking in legged robots.

Guideline for Control Torque Selection

Instead of directly using the actuator torques **u**, we introduce here the desired generalized control forces $\mathbf{Q}_d \in \mathbb{R}^n$ to control the biped motion. This new control element, which will be used to directly apply torques between the robot and the inertial reference frame, is shown to simplify control design and makes control parameter selection intuitive. Realization of $\mathbf{Q}_d$ using actuator torques **u** is discussed in the next subsection.

Our objective in walking is to maintain an upright body position, and also to sustain a stable oscillation in leg motion characterized by a ballistic component in swing. The first objective, to maintain an (essentially) upright body position, can be achieved by prescribing a torque that attracts the torso to a nominally vertical position (i.e., in the model coordinates of Figure 3.1, a torque that attracts θ towards an angle at or near 90^o).

In order to drive leg oscillation, the thigh segments are subjected to alternating torques, where the alternation is driven by changes in biped configuration (e.g., heel strike and heel off). Specifically, during swing phase, the prescribed torque drives hip flexion by attracting the thigh segment toward a given (flexion) angular orientation. Upon heel strike, another torque drives hip extension by attracting the thigh segment toward a given (extension) angular orientation.

During swing, the knee is not subject to a driving torque, but rather is subject only to damping. During early stance phase (i.e., heel strike to heel off) a somewhat stiff spring maintains the knee in an extended position. Note that a less stiff spring could be utilized to encourage stance knee flexion; however, walking with a straight leg in (most of) stance is described here, since doing so may reduce knee actuator torque and power requirements. Further, as is recognized through numerous simulation experiments, a "locking" knee enhances the basin of attraction for a stable gait limit cycle. The ankle is subject to a torque during swing that encourages slight flexion (to prevent stumbling), and to one during stance that generates a slight push-off before the stance leg enters swing.

Note that the torso and the thigh segment torques are defined relative to the inertial reference frame (IRF), while the knee and ankle torques are defined relative to the respective adjacent links. That the torso torque would be defined relative to the IRF is perhaps obvious, since gravity is assumed fixed with respect to the IRF, and postural stability is only relevant when defined with respect to the gravity vector. Referencing the thigh segment torques with respect to the IRF (as opposed to the torso) is less obvious, but achieving a desired (angular) dynamics with respect to the inertial frame is recognized as simpler than commanding torques with respect to the moving links in the nonlinearly coupled system.

It is important to mention that the control torques either referenced to the inertial frame or defined on the robot frame only influence the rotational dynamics of the robot. One does not need to apply forces that influence the vertical or horizontal dynamics of the torso, since the upper body will be carried atop the legs, and thus the appropriate horizontal and vertical motion will be dictated by the motion of the lower limbs.

As described, we do not specify any trajectories in time or space, but only define a single attraction point for each state. By utilizing torques defined in this manner, we are attracting the biped toward a desired configuration, but not dictating the path by which it arrives (in time or in space). Moreover, the controller does not attempt to directly maintain a desired forward speed, step frequency or stride length; rather, these motion attributes are obtained as a result of the interaction

between the natural dynamics of the robot and the low gain controller.

Transforming the Desired Control Torques to the Actuator Space

As described in the previous subsection, the control problem is made more intuitive by referencing thigh and torso torques to the inertial coordinate frame, while defining knee and ankle torques relative to adjacent links. In order to implement the presented approach, we propose a transformation between the desired generalized control forces, $\mathbf{Q}_d$ (introduced in the previous subsection), and the actuator torques, $\mathbf{u}$, as follows.

The objective of the transformation is to achieve the same constrained motion forcing the dynamics (3.7) with $\mathbf{Q}_u = \mathbf{E}\mathbf{u}$ as would be achieved with the application of $\mathbf{Q}_d$. Denoting the desired constrained acceleration as $\ddot{\mathbf{q}}_d$ (generated by $\mathbf{Q}_d$) and the constrained acceleration generated by the actuator torques as $\ddot{\mathbf{q}}$, the objective of the transformation can be stated as $\ddot{\mathbf{q}} = \ddot{\mathbf{q}}_d$. In order to consider this equivalence further, we must first consider issues of overactuation and underactuation.

Overactuation and Underactuation of the Constrained Dynamics

Due to the presence of the kinematic constraints (3.2), the biped could at times be fully actuated (i.e., same number of actuators as unconstrained degrees of freedom), overactuated (i.e., more actuators than unconstrained degrees of freedom), or underactuated (fewer actuators than unconstrained degrees of freedom). For example, the biped will be fully actuated when in single support phase the foot is flat on the ground. The biped will be overactuated in the double support configuration. Finally, the robot will be underactuated when two or fewer (independent) constraints are active, such as when in single support and only a single toe or heel (and nothing else) is in contact with the ground.

In order to address the issue of underactuation, we characterize the effect of the control force on the constrained motion of the biped. Let us first segment the unconstrained acceleration $\mathbf{a}$

as follows:

$$\mathbf{a} = \mathbf{a}_0 + \mathbf{a}_u = -\mathbf{M}^{-1}(\mathbf{G}+\mathbf{h}) + \mathbf{M}^{-1}\mathbf{Q}_u, \tag{3.8}$$

where $\mathbf{a}_0$ is the unconstrained acceleration generated by the uncontrolled dynamics and $\mathbf{a}_u = \mathbf{M}^{-1}\mathbf{Q}_u$ is the unconstrained acceleration resulting from the actuator torques. Substituting (3.8) into (3.7) provides a similar relation for the constrained accelerations:

$$\begin{aligned} \ddot{\mathbf{q}} = \ddot{\mathbf{q}}_0 + \ddot{\mathbf{q}}_u = \mathbf{R}^{-1}\mathbf{C}^{+}\mathbf{b} + \mathbf{R}^{-1}(\mathbf{I}-\mathbf{C}^{+}\mathbf{C})\mathbf{R}\mathbf{a}_0 \\ + \mathbf{R}^{-1}(\mathbf{I}-\mathbf{C}^{+}\mathbf{C})\mathbf{R}\mathbf{a}_u, \end{aligned} \tag{3.9}$$

where $\ddot{\mathbf{q}}_0$ (the first two terms on the right hand side of (3.9)) is the constrained acceleration resulting from the uncontrolled dynamics while $\ddot{\mathbf{q}}_u$ is the constrained acceleration which is directly related to the generalized control forces, and $\mathbf{I} \in \mathbb{R}^{n\times n}$ is an identity matrix. Substituting $\mathbf{a}_u$ from (3.8) into (3.9), we can obtain the explicit relation of $\ddot{\mathbf{q}}_u$ in terms of the generalized control forces $\mathbf{Q}_u$:

$$\ddot{\mathbf{q}}_u = \mathbf{R}^{-1}\mathbf{N}\mathbf{R}^{-T}\mathbf{Q}_u, \tag{3.10}$$

where $\mathbf{N} = \mathbf{I} - \mathbf{C}^{+}\mathbf{C} \in \mathbb{R}^{n\times n}$ is a symmetric projection operator to the null space of the inertially-weighted constraint matrix $\mathbf{C}$. Active constraints will reduce the biped degrees of freedom, which are given by $n_c = rank(\mathbf{N}) \leq n$. Active constraints can also reduce the effect of the control inputs. For the generalized control forces $\mathbf{Q}_u = \mathbf{E}\mathbf{u}$ in (3.10), the number of control inputs which can independently alter the constrained motion is given by $m_c = rank(\mathbf{N}\mathbf{R}^{-T}\mathbf{E}) \leq m$.

The type of actuation for the constrained dynamics can now be defined. If $n_c = m_c$, the number of degrees of freedom for the constrained motion is equal to the number of independent control actuators, and as such the system is said to be fully actuated. In this case, the transformation between the desired dynamics and achievable dynamics is exact. If $n_c < m_c$, the biped has more independent actuators than active degrees of freedom, and the system is said to be overactuated. In this case, $\mathbf{u}$ is not unique (i.e., the desired dynamics can be reproduced with different control inputs). Finally, in the case that $n_c > m_c$, the system is underactuated, and as such the desired

dynamics cannot in general be achieved.

Transformation in a Fully Actuated and Overactuated Configuration

Since the control input only effects the controlled part of the constrained acceleration (3.10), the equivalence relation between the desired and actual motion ($\ddot{\mathbf{q}} = \ddot{\mathbf{q}}_d$) can be written as:

$$\mathbf{R}^{-1}\mathbf{N}\mathbf{R}^{-T}\mathbf{E}\mathbf{u} = \mathbf{R}^{-1}\mathbf{N}\mathbf{R}^{-T}\mathbf{Q}_d. \tag{3.11}$$

Note that this relation defines n (possibly dependent) equations with m unknown control inputs $\mathbf{u}$, where the degree of dependence is a function of the constraint configuration, $\mathbf{N}$. Utilizing the generalized inverse notation [59], a particular solution to (3.11) for the actuator torque vector is given by:

$$\mathbf{u} = (\mathbf{R}^{-1}\mathbf{N}\mathbf{R}^{-T}\mathbf{E})^{+}\mathbf{R}^{-1}\mathbf{N}\mathbf{R}^{-T}\mathbf{Q}_d. \tag{3.12}$$

The solution defined by the above relation exists regardless of over or underactuation, although it does not necessarily satisfy (3.11). Practically, if the system is fully actuated, then $\mathbf{u}$ is a unique solution of (3.11). In the overactuated case, there is no unique solution of (3.11). In this case, (3.12) provides a solution of (3.11) in the minimum squared Euclidean norm sense (i.e., $\mathbf{u}^T\mathbf{u} \rightarrow min$). If however the system is underactuated, (3.11) cannot be solved exactly and as such (3.12) defines $\mathbf{u}$ which minimizes the squared Euclidean norm of the difference between the desired and the actual acceleration, $(\ddot{\mathbf{q}} - \ddot{\mathbf{q}}_d)^T(\ddot{\mathbf{q}} - \ddot{\mathbf{q}}_d) \rightarrow min$. Note however that $\mathbf{q}$ contains both translational and also rotational coordinates, and as such in the uncontrollable case, the control solution is not dimensionally consistent and does not have clear physical interpretation [102].

Transformation with Dimensional Consistency

Motivated by the Gauss principle of least constraint, we embed (3.11) in the following more general formulation:

$$\mathbf{u} = min\{\mathbf{u} \in \mathbb{R}^m : (\ddot{\mathbf{q}} - \ddot{\mathbf{q}}_d)^T\mathbf{M}(\ddot{\mathbf{q}} - \ddot{\mathbf{q}}_d)\}. \tag{3.13}$$

In contrast to (3.11), the above quadratic program will provide physically consistent actuator torque computation even through motion phases which are underactuated with the joint torque actuators.

In the present context, we expect any underactuated phases, if present, to occur only for brief periods (i.e., for periods much shorter than the characteristic times associated with the biped dynamics). As such, we assume any departure in dynamic behavior due to uncontrollability to be small. Now, using (3.9) and (3.10) one can express (3.13) as an explicit quadratic program for $\mathbf{u}$ as

$$\mathbf{u} = min\{\mathbf{u} \in \mathbb{R}^m : \frac{1}{2}\mathbf{u}^T\mathbf{A}_u^T\mathbf{A}_u\mathbf{u} - \mathbf{b}_u^T\mathbf{A}_u\mathbf{u}\}, \tag{3.14}$$

where $\mathbf{A}_u = \mathbf{N}\mathbf{R}^{-T}\mathbf{E}$ and $\mathbf{b}_u = \mathbf{N}\mathbf{R}^{-T}\mathbf{Q}_d$. Considering the fact that $\mathbf{N}$ is in general rank deficient, a particular solution to (3.14) can be defined as:

$$\mathbf{u} = (\mathbf{N}\mathbf{R}^{-T}\mathbf{E})^{+}\mathbf{N}\mathbf{R}^{-T}\mathbf{Q}_d. \tag{3.15}$$

The solution expressed by (3.15) is physically consistent for all cases of actuation. Specifically, if the biped is fully actuated (i.e., $n_c = m_c$), (3.15) yields the solution for $\mathbf{u}$ that yields $\ddot{\mathbf{q}} = \ddot{\mathbf{q}}_d$. In the overactuated case (i.e., $n_c < m_c$), the solution to (3.15) satisfies the matching dynamics criterion (i.e., $\ddot{\mathbf{q}} = \ddot{\mathbf{q}}_d$), while also minimizing the squared Euclidean norm of $\mathbf{u}$. Finally, in the case that the biped is underactuated (i.e., $n_c > m_c$), (3.15) minimizes the acceleration energy between the desired and actual motion. Using (3.15) one can transform the desired generalized control forces $\mathbf{Q}_d$ to actuator torques $\mathbf{u}$. Note that, just as in the case of human walking, there is no guarantee that the biped can recover a stable gait limit cycle from any underactuated configuration with the proposed control solution.

Works Related to the Proposed Transformation

Using quadratic programming, [103] proposed a method to modify the predefined reference trajectories to maintain balance while walking. Other works, [18] and [104], present methods that can be used to transform generalized forces to joint torques. Compared to the presented approach,

these methods do not provide a unified control force computation through changing constraints, and are restricted with respect to the robot configuration (e.g., at least one foot is assumed flat on the ground).

Implementation on a Seven-Link Biped

We illustrate and further describe the proposed approach via implementation and simulation on the seven-link biped illustrated in Figure 3.1.

Choice of Control

In order to define the control actions, we impose seven state-dependent torques which directly alter the rotational dynamics of the biped. Each of these state-dependent torques, can be constructed from energetically passive spring-damper couples with fixed equilibrium points. These include an angular torque on the torso with respect to the IRF, state-dependent alternating angular torques on both thighs (also with respect to the IRF), and state-dependent torques on knees and ankles, both with respect to the robot frame (i.e., defined with respect to adjacent links). As such, the vector of desired generalized control forces can be expressed as:

$$\mathbf{Q}_d = -\mathbf{K}_d(\boldsymbol{\phi} - \boldsymbol{\phi}_d) - \mathbf{B}_d\dot{\boldsymbol{\phi}}, \tag{3.16}$$

where

$$\mathbf{K}_d = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ k_{d1} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & k_{d2}^r & -k_{d3}^r & 0 & 0 & 0 & 0 \\ 0 & 0 & k_{d3}^r & -k_{d4}^r & 0 & 0 & 0 \\ 0 & 0 & 0 & k_{d4}^r & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & k_{d2}^l & -k_{d3}^l & 0 \\ 0 & 0 & 0 & 0 & 0 & k_{d3}^l & -k_{d4}^l \\ 0 & 0 & 0 & 0 & 0 & 0 & k_{d4}^l \end{bmatrix}, \tag{3.17}$$

is the stiffness matrix, $\mathbf{B}_d$ is the matrix of linear damping coefficients (which has the same form as (3.17)), $\boldsymbol{\phi} = [\theta, \theta_1, \theta_2 - \theta_1, \theta_3 - \theta_2 + \pi/2, \theta_4, \theta_5 - \theta_4, \theta_6 - \theta_5 + \pi/2]^T$ defines the feedback information for the control torque computation, and $\boldsymbol{\phi}_d = [\theta_b, \theta_l^r, 0, \theta_a^r, \theta_l^l, 0, \theta_a^l]^T$ defines the equilibrium point of each spring (i.e., can be considered as the attraction point of each spring). The parameters that define $\mathbf{Q}_d$ for the seven-link biped are shown schematically in Figure 3.2. Note that the right and left side parameters are indicated with superscripts.

It should be noted that the control given by (3.15), (3.16) does not guarantee a dynamic walk. Specifically, in order to meet the criteria for dynamic walking, the stiffness and damping parameters of the controller must be selected to be sufficiently low such that the control influence does not substantively prescribe the motion of the robot.

As previously mentioned, leg oscillation is generated by application of alternating torques (defined with respect to the IRF) applied to each thigh segment. This alternation is switched based on an event driven finite state structure. As follows, we describe the finite state logic along which the control parameters, $\mathbf{K}_d, \mathbf{B}_d, \boldsymbol{\phi}_d$ are changed as piecewise constant functions.

Let us start at heel strike which induces application of the state dependent torques that attract the thigh toward a hip extension configuration; initiates knee locking with a somewhat stiff spring and damper and imposes a spring-damper element at the ankle which accumulates elastic

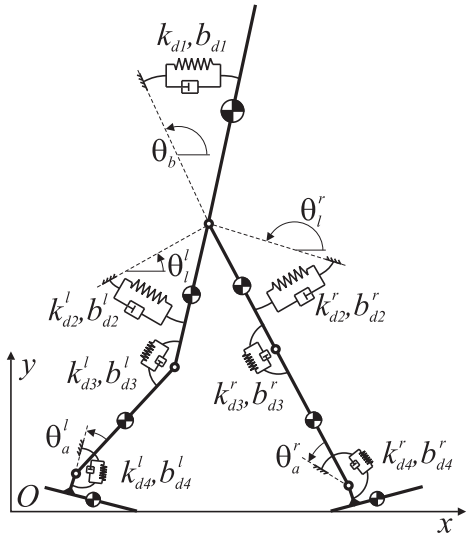

Figure 3.2 Schematic representation of the control elements.

energy during stance to provide an ankle push at late stance. The heel off event (i.e., when the heel leaves the ground) switches the hip torque (equilibrium angle) to one that attracts the thigh towards a hip flexion configuration and allows the ankle to release the energy accumulated during stance through push off. In addition to these two states, two additional states are used to facilitate stable locomotion. Specifically, following the toe-off event (when the swing foot is entirely in the air), the swing leg ankle equilibrium point (i.e., angle of attraction) is moved to a slightly flexed position, which enhances ground clearance while the swing knee is only slightly damped. The final state, defined by the knee reaching full extension, is used to retain the knee at full extension and thus prepare the (extended) swing leg for heel strike. Thus, the gait controller consists of four states, as illustrated in Figure 3.3. Note that the states apply independently to each leg, and do not apply at all to the torque acting on the torso (i.e., the control parameters for the torso are not changed during the motion). As such, (for each leg), state one consists of stance, state two is initiated by heel-off, state three initiated by toe-off, state four initiated by full knee extension, and the leg is returned to state one by heel strike. Due to external disturbances or other type of uncertainties however, the described event flow may not remain preserved along the motion of the robot. In this light, the state of each leg is identified based on its constraint configuration (i.e., state

one - both toe and heel on ground; state two - toe on ground, heel off ground; state three - both toe and heel off ground; state four - toe and heel off ground and extended knee on a forward swing leg). In particular, switching from state three (swing) to state one (stance) is important if during swing an incomplete knee extension occurs (which was recognized through the push disturbance simulations subsequently presented). Further cross switching has also been recognized to improve the robustness of the proposed control methodology.

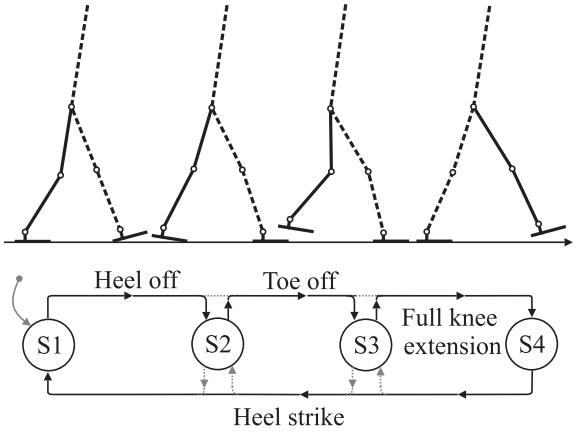

Figure 3.3 State flow diagram. The state flow presented with (solid line) corresponds to the solid leg along normal walking.

Given the independence of each leg, there is no guarantee that each leg is fully out of phase with the other. Recall, however, that the control philosophy in this work is to impose a minimum number of constraints, and thus encourage the natural dynamics of the biped rather than constraining it. This is in contrast to an implementation which utilizes time based switching, such as that described by [105]. Specifically, state switching happens along changes in constraint configuration initiated by the motion of the robot autonomously. Similar approach on a point foot robot and curved foot robot can be found in [21], [95] and [24] respectively.

Simulation

For the biped illustrated in Figure 3.1, the associated geometric and inertial parameters normalized to a body height L and mass M, as given by [3], are listed in Table 3.1. For purposes of control implementation and simulation, the biped was parameterized according to the values listed in Table 3.1 using a height $L = 1.8\ m$ and a mass $M = 75\ kg$. The simulation was conducted by utilizing the desired generalized control force described in (3.16) and (3.17), and by using the actuator torque solution (3.15). The controller was parameterized by starting with initial estimates (guided by the characteristic times that typify human gait) and using the simulation to iteratively tune parameters for a robust and human-like gait. Specifically, control parameters were considered to be a robust set when the biped would within a few steps converge to a stable, natural-looking gait after starting from rest in several different initial configurations (e.g., double support with both feet flat; double support with only forward heel and backward toe in contact; single support with foot flat).

Note that some type of automated parameter tuning could also be implemented for control gain selection. Due to the nonlinear character and nonsmooth nature of the problem, however, such automated parameter tuning is a nontrivial task which often requires additional hand tuning to provide a robust parameter set [105]. As such, for the simulations presented here, the control parameters were selected by hand tuning and intuition.

Dynamic Walking of the Biped

For the (adult) human-scale anthropomorphic biped, the control parameters used for an approximately normal walking speed are listed in Table 3.2 (where the upper index $(*) = r/l$ represents the right or left leg, respectively). A stroboscopic image of the motion results of this controller, simulated over a period of $t \in [0, 10]s$, are shown in Figure 3.4. The corresponding real-time video of the resulting gait is included in the supporting material. For the simulation shown, the initial configuration of the biped was (starting at rest) in the double-support phase with the forward heel on the ground and the backward toe on the ground. The average forward walking

speed for this simulation, after converging to a stable limit cycle, was $0.81m/s$.

As was outlined in the paper, the presented control approach is designed to leverage the natural dynamics of the biped. A direct consequence is that the simulated motions have natural human style. Beyond this qualitative characteristic, the efficiency of dynamic walking should be improved relative to a ZMP-based approach, since the former need not use significant energy to override the natural dynamics of the biped. The efficiency of gait can be characterized by the specific mechanical cost of transport, c_{mt} =(mech. energy)/(weight $\times$ distance traveled), which is adapted from the specific resistance, as presented in [106]. Based on the simulation shown in Figure 3.4, the calculated mechanical cost of transport of the proposed approach is $c_{mt} = 0.19$. Comparatively, the specific mechanical cost of transport of the ZMP-based Honda Asimo is *estimated* as $c_{mt} = 1.6$ [1], while the cost of transport of the (actuator-assisted) Cornell dynamic walker is $c_{mt} = 0.05$ [1]. As such, the walking synthesized with the proposed approach, which presumably is (as subsequently demonstrated) more robust and versatile than an actuator-assisted approach, can also be nearly an order of magnitude more efficient than walking generated by trajectory tracking approaches.

Walking with Different Speeds

In order to demonstrate versatility in the control approach, simulations were also conducted for faster and slower walking speeds. Multiple possibilities exist for varying the control parameter set to achieve stable locomotion with different walking speeds. An intuitive parameter that can be varied to influence the walking speed is the hip stiffness during stance (i.e., k_{d2} in state one) which effects the leg dynamics with respect to the inertial frame. Figures 3.5 and 3.6 show stroboscopic images of the biped walking at faster and slower walking speeds (relative to Figure 3.4), respectively, both simulated over a period of $t \in [0, 10]s$, and both of which were generated by utilizing the same control parameter set given in Table 3.2, but with different values of the stance hip stiffness. Specifically, to achieve these gaits, the corresponding stiffness value was set to $k_{d2} = 800Nm$ and $k_{d2} = 600Nm$, respectively. The faster gait, which is shown in Figure 3.5 starting from rest at an initial condition of double-support with both feet flat on the ground, is characterized by an

average walking speed of $0.92m/s$. The slower gait, which is shown in Figure 3.6 starting from rest at an initial condition of single-support with the foot flat on the ground, is characterized by an average walking speed of $0.68m/s$. Corresponding real-time videos of these simulations are included in the supporting material.

Figure 3.10 shows the respective forward velocities (of the center of mass of the torso) at each of the three walking speeds. The time evolution of the upper body angle for the three gaits are depicted in Figure 3.11. As can be seen in the figure, the torso for each case starts at an upper body posture away from the limit cycle, and in each case converges within a few steps to a stable limit cycle. Figure 3.12 shows the phase plane plots for the (right-side) hip, knee, and ankle joints, for each of the three gaits, clearly indicating that a stable limit cycle has been reached in each case. The fact that the biped achieves a stable limit cycle within a few steps for several different walking speeds from different initial conditions by varying only a single control parameter (i.e., hip stiffness during stance k_{d2}) is demonstrative of the ability of the method to generate walking with different speeds and also shows robustness with respect to variation in initial conditions. Different control parameters, as the upper body angle θ_b, hip damping at stance (b^*_{d2} in state one), and ankle stiffness in stance (k^*_{d4} in state one and two) can also be used to change the walking speed. While the proposed approach can also be used to make the robot stand, natural looking walking was obtained in a speed range of $[0.6, 1.2]m/s$.

Walking with Different Style

In order to illustrate the differing character of gait achieved with a different set of control parameters, the biped was simulated with the set of control parameters listed in Table 3.3. The stroboscopic image of walking with this controller, simulated over a period of $t \in [0, 10]s$ and corresponding to an initial condition of starting at rest in double-support with the forward heel on the ground and the backward toe on the ground, is shown in Figure 3.7. The corresponding real-time video of the resulting gait is included in the supporting material. The differing character of gait is evident by comparing the video corresponding to Figure 3.7 with the video corresponding

to the gait depicted in Figure 3.4. This motion obtained under substantial variation in control parameters also demonstrates robustness with respect to control parameter variation.

Based on our experience with simulation of the biped, stable walking is achievable with a relatively large range of control parameters. Differing sets of control parameters result in a differing character of gait, some of which appear more natural and efficient than others. Other sets of parameters generate gaits that appear either more relaxed or more deliberate. There also obviously exists a large space of parameters that fail. A video of one such failure is included in the supplemental material. This particular failure is due to a "weak" gait (caused by hip torques that do not generate sufficiently large steps) which ultimately results in a stumble.

Push Disturbance Response

In order to demonstrate robustness to push-type disturbances, the biped was simulated at the three speeds with impulsive forward and backward push-type disturbances. Specifically, an impulsive force was applied via a constant horizontal force of $200N$ for a duration of $0.2s$, applied at the center of mass of the upper body in both the forward and backward directions, respectively. Note that these disturbances are similar to those described in [107]. In the six simulations (forward and backward pushes at three different speeds), the robot recovered fully in all cases. In Figure 3.13, all six push recovery test results are depicted. The corresponding real time videos included in the supporting material demonstrate the push-type disturbance rejection of the proposed approach.

Model Parameter Uncertainties

Since the proposed approach is model based, the authors further conducted numerical experiments to explore robustness with respect to model parameter variations. Specifically, 100 simulations were conducted, in which the mass matrix $\mathbf{M}$ and constraint matrix $\mathbf{A}$ used in the controller (3.15) were simultaneously varied elementwise by an average of 10% relative to the exact values (used in the dynamic model). As depicted in Figure 3.14, the controller maintains stability with uncertainty in parameters, demonstrating a moderate degree of robustness to model parameter

uncertainty.

Walking on Slopes

The versatility and robustness of the proposed approach was also explored by walking up and down slopes. In order to walk up and down slopes, four intuitive controller parameters were modified. Specifically, relative to the fast walking set of parameters, the following changes were made: the upper body angle was selected to be $\theta_b = \{80^o, 90^o\}$ (for the uphill and downhill walk respectively), the equilibrium angle for the ankles at swing were changed to $\theta_a = 15^o$ (to prevent stumbling), the hip extension angle was $\theta_l = 128^o$, and the knee stiffness at stance was changed to $k_{d3} = 50Nm$. The corresponding simulation result for $\pm 5^o$ upward and downward slopes are shown in Figure 3.8 and Figure 3.9. Real time videos of the respective motions are included in the supporting material. Note that with the same parameters the biped can walk also on level ground.

Comment on 3D extension and parameter adaptation

It should be noted that the approach presented herein considers sagittal plane motion, although extension to three dimensional walking would neither change the structure of the model nor the control approach. Particularly, the walking controller would need to be extended with additional spring-damper elements which would apply a stabilizing torque to the (upper) body motion in the frontal plane relative to the inertial reference frame. Realization of the corresponding torques would be enabled with additional (hip and/or ankle) actuators on the robot. Finally, further implementation of parameter adaptation using learning techniques, [24], although not explored here, may improve the inherent robustness of the approach demonstrated through numerous simulation results.

Conclusion

The authors have proposed an approach for the control of biped walking that enables dynamic walking in a fully actuated biped robot. Rather than prescribe kinematic trajectories or

kinematic constraints, the approach is based on the prescription of state dependent torques obtained with low-gain spring-damper couples that "encourage" patterned movement through the natural dynamics of the biped. These simple set of torques are proposed which generate a stable gait while allowing the biped to exploit its natural dynamics. Some of the prescribed torques are referenced to the inertial reference frame, which simplifies the selection and tuning of the control parameters. Implementation of torques from a mixed set of coordinate frames is enabled by a joint torque computation (based on Gauss's principle of least constraint), which is valid for all configurations of the biped. The proposed approach is implemented in simulation on an anthropomorphic biped, motion of which is shown to quickly converge to a natural-looking gait limit cycle. Simulations are conducted with various control parameters and also different initial conditions. The mechanical cost of transport is calculated and shown to be nearly an order of magnitude lower than what would be expected from trajectory tracking approaches. The authors additionally demonstrate versatility with respect to varying walking speeds and ground slopes, and robustness with respect to push-type disturbances and uncertainty in model parameters. Future work includes experimental implementation of the proposed approach.

Table 3.1 Geometric and inertial parameters, Winter [3].

Description	no. $(*)$	l_*/L	l_{c*}/l_*	m_*/M	r_*/l_*
Upper body	1	0.288	0.626	0.6780	0.496
Thigh	2	0.245	0.433	0.1000	0.323
Shank	3	0.246	0.433	0.0465	0.302
Foot	4	0.152	0.250	0.0145	0.475
Foot geometry			a/l_4	b/l_4	h/L
			0.75	0.25	0.039

Table 3.2 Controller parameters for "normal walking"; $k^*_{d()}[Nm]$, $b^*_{d()}[Nms]$, $\theta^*_{()}[deg]$.

States	k_{d1}	k^*_{d2}	k^*_{d3}	k^*_{d4}	b_{d1}	b^*_{d2}	b^*_{d3}	b^*_{d4}
1	400	700	30	20	50	300	5	15
2	400	70	30	20	50	1	5	15
3	400	70	0	5	50	1	1	1
4	400	0	30	5	50	0	5	1

States	1	2	3	4
θ_b	87.5			
θ^*_l	68	122	122	–
θ^*_a	0	0	10	0

Table 3.3 Controller parameters; $k^*_{d()}[Nm]$, $b^*_{d()}[Nms]$, $\theta^*_{()}[deg]$.

States	k_{d1}	k^*_{d2}	k^*_{d3}	k^*_{d4}	b_{d1}	b^*_{d2}	b^*_{d3}	b^*_{d4}
1	500	750	40	10	35	250	3	10
2	500	65	40	10	35	1.5	3	10
3	500	65	0	5	35	1.5	1.25	2
4	500	0	40	5	35	0	3	2

States	1	2	3	4
θ_b	84			
θ^*_l	67	125	125	–
θ^*_a	0	0	5	0

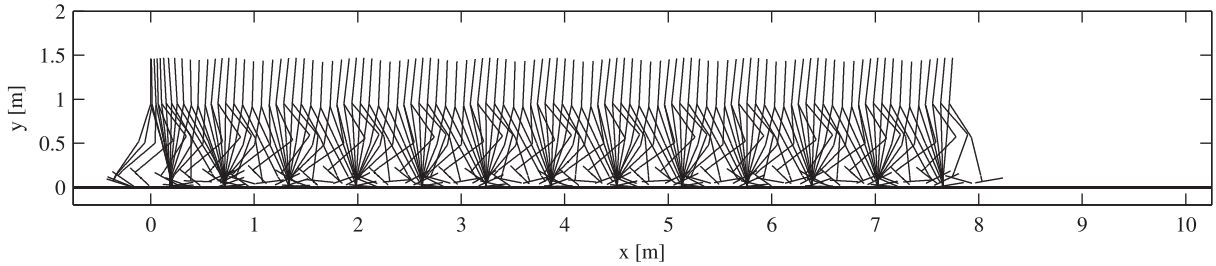

Figure 3.4 Stroboscopic view of dynamic walking with $0.81m/s$ average forward speed. The motion is started from double support phase while only the forward heel and the backward toe are on the ground, $\mathbf{q}(0) = [0, 1.27, 1.57, 1.82, 1.78, 0.2, 1.31, 1.04, -0.35]^T$, $\dot{\mathbf{q}}(0) = \mathbf{0}$. The calculated specific cost of transport is $c_{mt} = 0.19$. Within a cycle the walker spent 15.6% in double support phase.

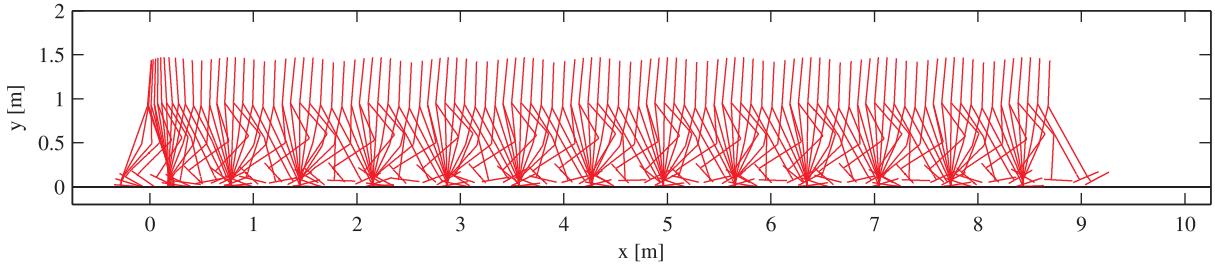

Figure 3.5 Stroboscopic view of dynamic walking with $0.92m/s$ average forward speed. The motion is started from double support with both feet flat on the ground, $\mathbf{q}(0) = [0, 1.24, 1.5, 1.86, 1.86, 0, 1.23, 1.23, 0]^T$, $\dot{\mathbf{q}}(0) = \mathbf{0}$. The calculated specific cost of transport is $c_{mt} = 0.22$. Within a cycle the walker spent 16% in double support phase.

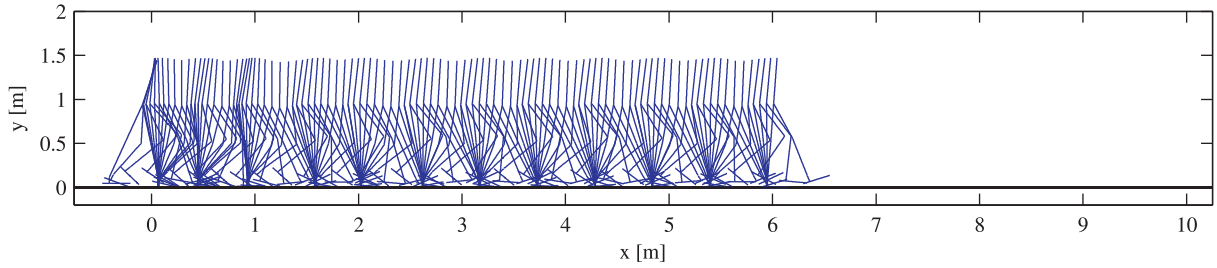

Figure 3.6 Stroboscopic view of dynamic walking with $0.68m/s$ average forward speed. The motion is started from single support with the forward foot flat on the ground, $\mathbf{q}(0) = [0, 1.25, 1.3, 1.75, 1.75, 0, 1.2, 1.2, 0]^T$, $\dot{\mathbf{q}}(0) = \mathbf{0}$. The calculated specific cost of transport is $c_{mt} = 0.17$. Within a cycle the walker spent 18.3% in double support.

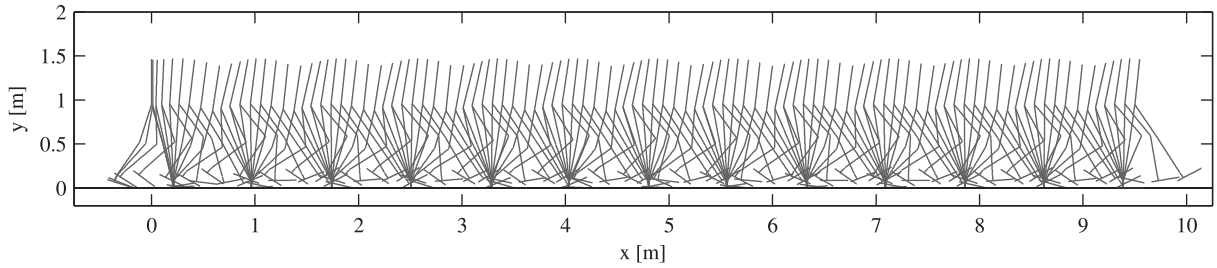

Figure 3.7 Stroboscopic view of dynamic walking with $0.97m/s$ average forward speed, simulated using the control parameters from Table 3.3. The motion is started from double support phase while only the forward heel and the backward toe are on the ground, $\mathbf{q}(0) = [0, 1.27, 1.57, 1.82, 1.78, 0.2, 1.31, 1.04, -0.35]^T$, $\dot{\mathbf{q}}(0) = \mathbf{0}$.

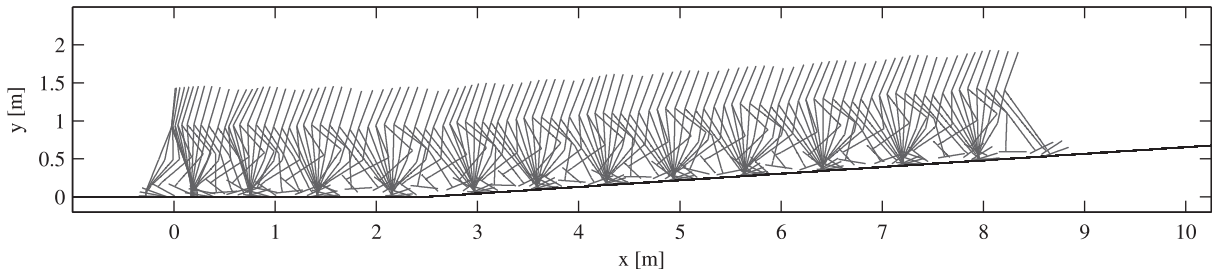

Figure 3.8 Stroboscopic view of uphill walking, simulated using the control parameters from Table 3.3. The motion is started from double support with both feet flat on the ground, $\mathbf{q}(0) = [0, 1.24, 1.5, 1.86, 1.86, 0, 1.23, 1.23, 0]^T$, $\dot{\mathbf{q}}(0) = \mathbf{0}$.

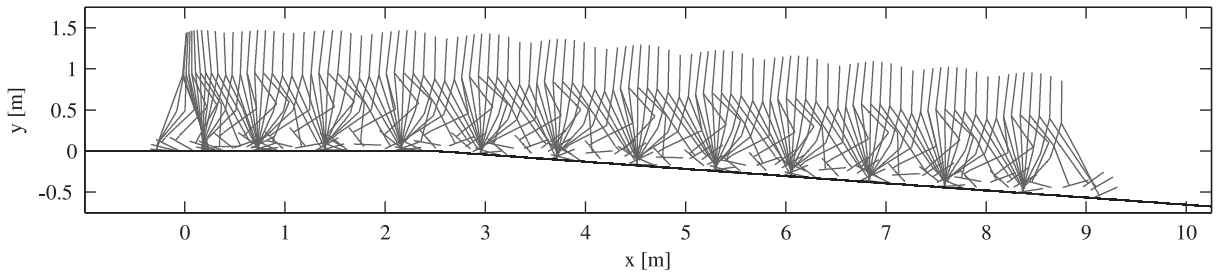

Figure 3.9 Stroboscopic view of downhill walking, simulated using the control parameters from Table 3.3. The motion is started from double support with both feet flat on the ground, $\mathbf{q}(0) = [0, 1.24, 1.5, 1.86, 1.86, 0, 1.23, 1.23, 0]^T$, $\dot{\mathbf{q}}(0) = \mathbf{0}$.

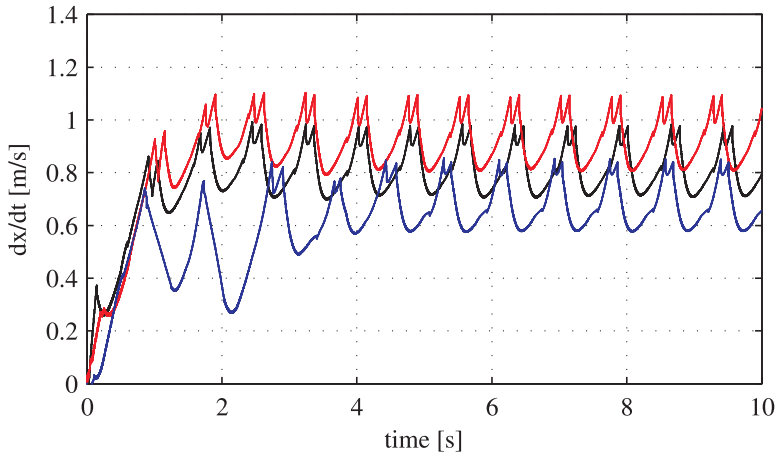

Figure 3.10 Forward velocity of the upper body CoM for walking at three different speeds. The average velocities, $\dot{x}_{avg} = [0.92, 0.81, 0.68] m/s$ are calculated on the sustained walking cycles by: $\dot{x}_{avg} = \int_{T_1}^{T_2} \dot{x}(t)dt/(T_2 - T_1)$ where $T_1 = 5s$ and $T_2 = 10s$.

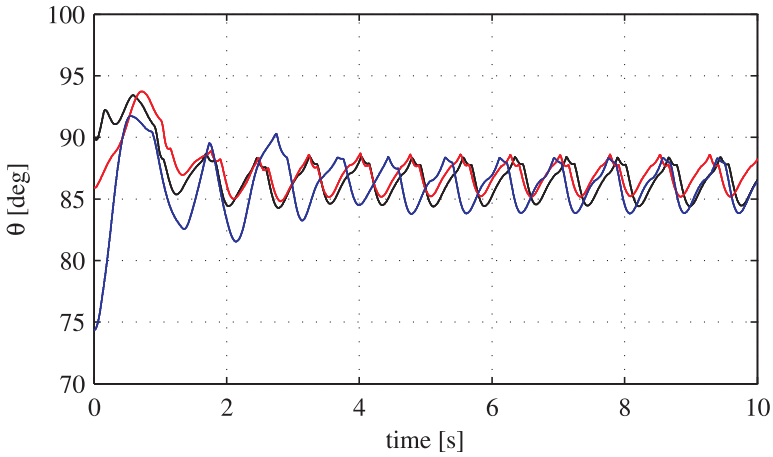

Figure 3.11 Upper body angle during walking at three different speeds. The vertical upright position corresponds to 90^o.

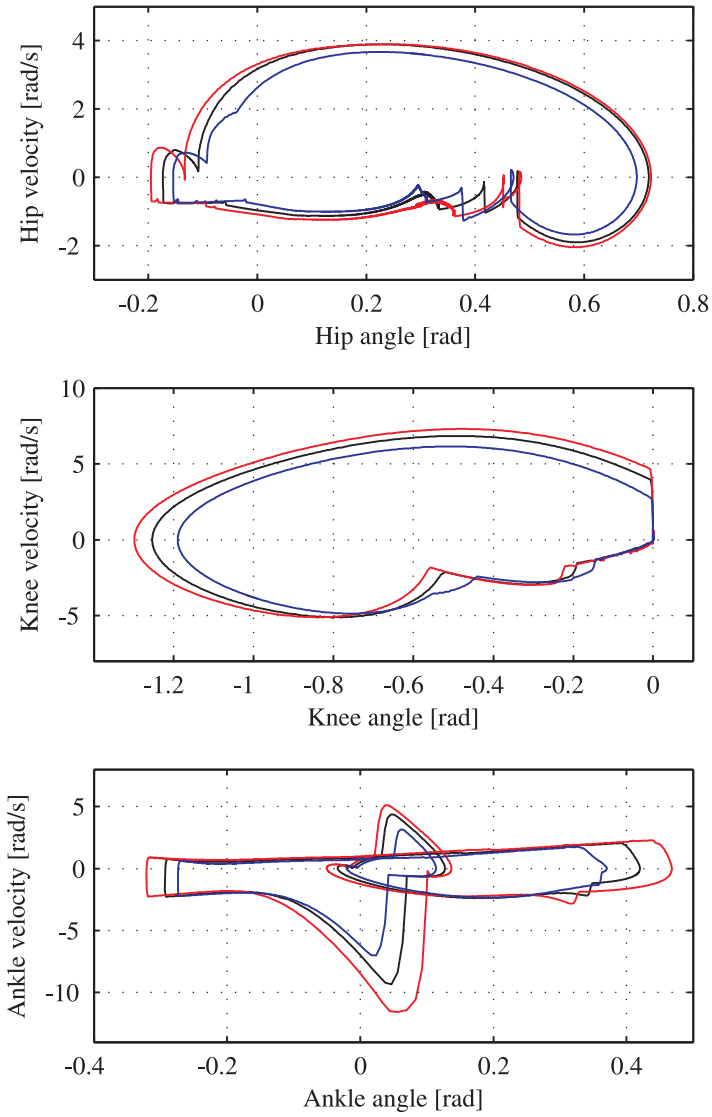

Figure 3.12 Steady walking cycle for three different speeds for the (right) hip, knee and ankle motion respectively. The joint angles are defined as: $\theta_1 - \theta$ for the hip, $\theta_2 - \theta_1$ for the knee, and $\theta_3 - \theta_2 + \pi/2$ for the ankle respectively.

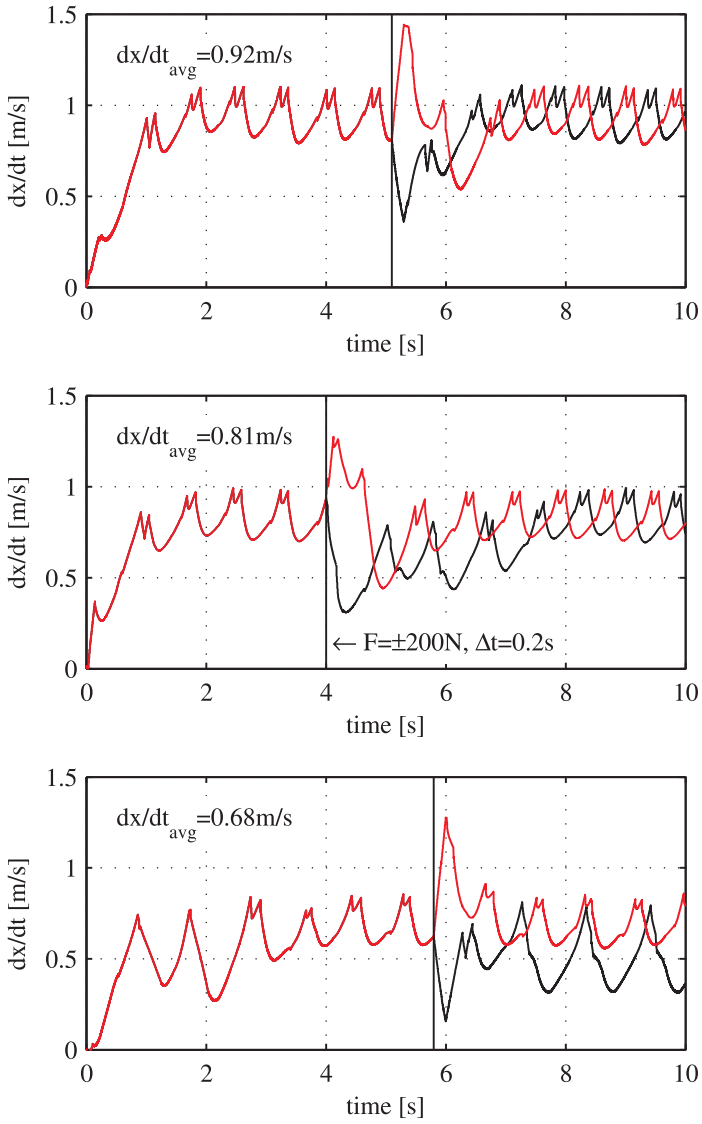

Figure 3.13 Push experiment for the walk at $[0.92, 0.81, 0.68] m/s$ average speeds. The six separate experiments shown characterize the response to forward and backward pushes (red and black lines respectively) at $[5.1, 4, 5.8] s$ with $200N$ force for a duration of $0.2s$, which act horizontally on the center of the upper body. While the walk remained stable in all six cases, at the slowest speed, the robot converged to a different cyclic trajectory after the forward push. Although, the recovery time in some cases may seem long, the corresponding real time video indicates a natural looking response.

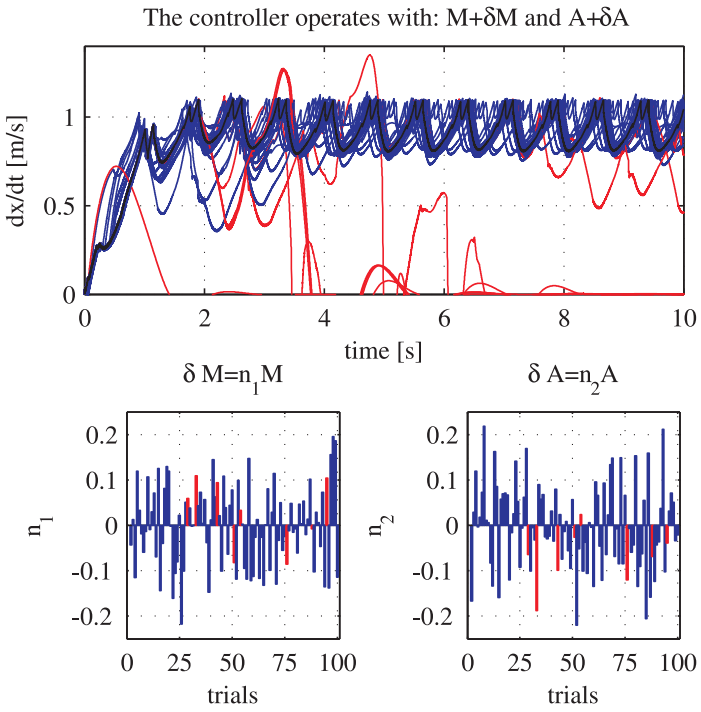

Figure 3.14 The picture depicts forward velocity versus time in 100 simulations under simultaneous variation of the mass matrix and constraint matrix. The random variables $n_{1,2}$ used to generate the parameter variation have normal distribution $N(\mu, \sigma^2)$ (with zero mean $\mu = 0$ and $\sigma = 0.1$ standard deviation). During the simulations, the robot remained stable in 92 trials, while it fell 8 times (all during the starting steps).

CHAPTER IV

MANUSCRIPT 3: EXPERIMENTAL IMPLEMENTATION OF ACTUATED DYNAMIC WALKING IN BIPED ROBOTS

By

David J. Braun, Jason E. Mitchell and Michael Goldfarb

Vanderbilt University

Nashville, TN

Submitted as an Original Paper to

The International Journal of Robotics Research

This paper has supplementary multimedia material (available upon acceptance at http://ijrr.org). The submitted video demonstrates experimentally realized dynamic walking on a seven-link biped robot. The video can be played with Windows Media Player.

Abstract

This paper presents the experimental implementation of a control method developed by the authors for actuated dynamic walking in biped robots. Rather than utilizing trajectory tracking, the control approach used herein employs state dependent control torques generated by low-gain spring-damper couples to encourage patterned motion. For the purpose of experimentation, a seven-link biped robot was designed with backdrivable joint actuators, which allows passive leg motion preferred for dynamic walking. Implementation of the control approach on this robot provided a system that emulates an energetically efficient human like locomotion. Following an overview of the control method, the paper describes the robot design, discusses the real-time control implementation, and presents experimental data (and an accompanying video) that demonstrates compliant dynamic walking with natural looking (partially ballistic) swing, extended knee stance support and human like (preemptive) ankle push-off.

Introduction

IN the recent years there has been considerable research effort devoted to bipedal locomotion. Although various approaches are proposed in the literature, see [4] for a recent review, new control ideas are often motivated by the zero moment point (ZMP) control paradigm or are closely related to the (passive) dynamic walking principle.

The ZMP approach introduced by [5], [84] is one of the most frequently used approaches to biped locomotion synthesis, see [90], [9], [10], [85], [86], [87]. Application of this method has been shown to provide effective, robust, and versatile locomotion for biped robots, [8], [9]. As is recognized through numerous implementations however, due to the characteristic bent knee stance support, flat foot constraint, and a frequently used (high-gain) trajectory tracking motion coordination, the ZMP walk may look as a careful human walk on uncertain terrain, rather than the more natural highly dynamic walking humans typically employ on level ground, [89], [11].

In order to achieve an efficient and natural-looking bipedal gait, several researchers have

investigated a dynamic walking approaches that leverage the passive (uncontrolled, natural) dynamics of the robot through walking, see [108]. On one end of this spectrum are fully passive dynamic walkers which rely on precisely tuned natural dynamics of the robot, and walk on a slight downward slope powered only by gravity, [12]. Utilizing this idea, actuator-assisted passive walkers were developed and shown to possess human-like and energy efficient gait [1], [14].

Actuated robots which are controlled to mimic some unified property of passive walking have also inspired numerous works. In this context, an energy tracking control approach was proposed in [109] and also adapted by [110], [111], [112], [113]. In these later works, a trajectory-free control approach was preferred to generate energy efficient dynamic walking. It was also shown that active feedback control could be used to remove the well known sensitivity issues of the passive walking to ground slope. These ideas are fully aligned with the control philosophy utilized in this paper.

Actuated dynamic walking which neither utilizes the ZMP method, nor requires a passive or a nearly passive robot design, have also been proposed in literature. In this context, [16] used inverse dynamics and linear optimal state feedback stabilization to control a dynamically walking robot; [15] have proposed a tracking control scheme where the reference motion was generated using Van der Pol oscillators; while [114] have introduced a control method where the reference motion was selected to be a "potential energy conserving orbit" (i.e., which is a special trajectory along which the potential energy of the system is preserved). While tracking control is a frequently used viable approach to walking, its realization with high gains (usually used for manipulators) may not be well suited with walking, where there is no ensured inertial reference (i.e., the foot cannot be considered fixed to the ground), [17].

Recently, a concept of "hybrid zero dynamics" [19] and "virtual control constraints" [20] were used to develop and experimentally verify a walking control approach by [21]. This approach while utilizes high-gain joint level control, it allows the pointed-feet underactuated robot, Rabbit, to exploit its uncontrolled rotational dynamics in the inertial frame. Instead of enforcing a predefined time dependent reference trajectory, [22] used neural networks to identify the relations between the

configuration coordinates generated by the robot while walking. These relations are then used as references to realize a robust dynamic walk. Motivated by a different idea, a biologically inspired sensor and motor-neuron based control approach, which does not utilize trajectory tracking, was proposed by [23], [24]. This method was implemented and validated with experimentally realized dynamic walking on a small size robot, RunBot. Due to the highly geared actuation unit however, the robot could not demonstrate ballistic swing leg motion, which is a major attribute of a (human-like) dynamic walk considered here.

[18] proposed "virtual model control", which was implemented on a biped with series-elastic actuators, which enabled a practical control realization that was largely free of kinematic constraints. In the mentioned work however, the control method does not employ ankle actuation which is a characteristic motion attribute utilized by humans and as such it is explicitly addressed herein.

There are two main preconditions which allow natural-looking and energy-efficient realization of actuated dynamic walking. The first, related to the control approach, precludes enforcing a predefined reference trajectory, including state dependent kinematic constraints, or other attributes of the walking cycle (such as stride length, stepping frequency or average forward speed) with high gain control. This condition motivated us to develop a control framework which utilizes state-dependent control torques (generated by low-gain spring-damper couples) to provide motion coordination without prespecifying the response of the system, [27], [25]. The second precondition (not related to control) depends on joint actuation which should not suppress passive joint motion (i.e., joints should be highly backdrivable, so that power can flow both from the actuator to the limbs, and back from the limbs to the actuator). Utilization of backdrivable joint design allows the inertial motion of the robot to be exploited through walking rather than being suppressed by the actuation units.

By means of the above arguments, the walking controller utilized here is implemented on a 7-link biped robot designed with backdrivable actuators which meets the second precondition required for realization of actuated dynamic walking. Note that such backdrivability can be sub-

stituted using closed-loop torque control of each joint through a non-backdrivable transmission, [104], but doing so largely removes the energetic advantages afforded from leveraging the bipeds passive dynamics.

In the remainder of this paper, we first recall the general idea of the control approach developed by the authors [25]. The discussion on control is followed by a section on design of a seven link biped robot developed with highly backdrivable joints. Description of design is followed by a section on real-time control implementation. Finally, we present a walking experiment, which demonstrates human-like compliant dynamic walking of a seven link robot coordinated with the proposed walking controller. The experimentally realized motion is characterized with a natural looking swing, extended knee stance support and human-like (preemptive) ankle push-off.

Model of the Biped

The control approach utilized in this paper requires information from the dynamical model of the biped. In this light, we will first introduce the model of a seven-link planar walking robot illustrated in Figure 4.1. The configuration of the biped is defined with nine coordinates, $\mathbf{q} = [x, y, \theta, \theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6]^T$, where the first two coordinates represent the translational motion of the robot in the inertial frame while the last seven angular coordinates reference the orientation of the links with respect to the inertial frame. In order to support the forthcoming discussion, we will also define the joint angles (relative angles between the links) as: $\boldsymbol{\varphi} = [\varphi_1, \varphi_2, \varphi_3, \varphi_4, \varphi_5, \varphi_6]^T = [\theta_1 - \theta, \theta_2 - \theta_1, \theta_3 - \theta_2 + \pi/2, \theta_4 - \theta, \theta_5 - \theta_4, \theta_6 - \theta_5 + \pi/2]^T$. The biped is actuated at each joint (i.e., right and left hip, knee, and ankle joints), such that, the dynamics of the robot are affected by six actuator torques, $\mathbf{u} = [u_1, u_2, u_3, u_4, u_5, u_6]^T$, which are considered positive in the same (counterclockwise) direction as the joint angles.

In the following, we derive the mathematical model by considering the biped as a constrained mechanical system, [96], [97]. This model contains the differential equations of the flight phase motion, and the (algebraic and differential) relations which define the kinematic (physical) constraints along the motion. In the present context, we will only present the basic elements of

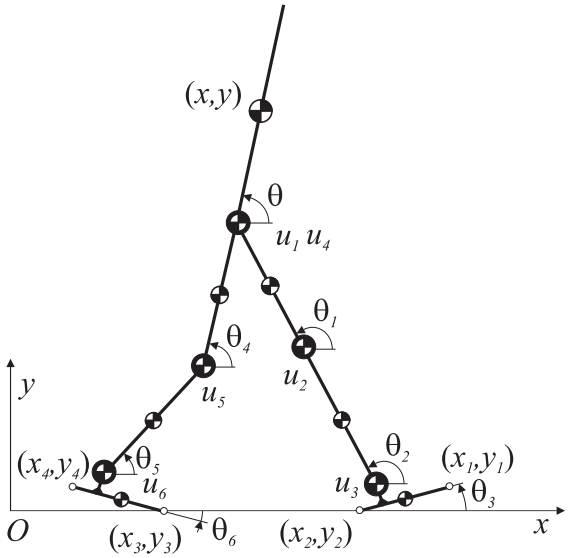

Figure 4.1 Seven-link biped with the absolute coordinates **q** and the control torques **u**. The Cartesian coordinates (x_i, y_i), $i \in \{1,2,3,4\}$, represent the position of the toe and the heel for the left and right leg.

the biped model which provides the necessary information for the closed-loop control design. For more information on the modeling approach utilized here, see [25].

Unconstrained Dynamics

The equations of motion for the 9-DoF (unconstrained) "flying" biped, can be written as

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{G}(\mathbf{q}) = \mathbf{Q}_u, \tag{4.1}$$

where $\mathbf{M} \in \mathbb{R}^{9\times 9}$ is a symmetric and positive definite mass matrix, $\mathbf{h} \in \mathbb{R}^9$ represents the inertial forces, $\mathbf{G} \in \mathbb{R}^9$ represents the gravitational forces, while $\mathbf{Q}_u = \mathbf{E}\mathbf{u}$ is a generalized control force computed using a constant matrix $\mathbf{E} \in \mathbb{R}^{9\times 6}$ which maps the control inputs $\mathbf{u} \in \mathbb{R}^6$ to the generalized control force space.

Kinematic Constraints

The unconstrained equation of motion presented above describes the flying phase motion of the biped. In normal walking however, numerous kinematic restrictions, imposed by ground-foot contact or the full knee extension stop, can restrict the motion of the robot. These kinematic motion constraints are introduced and discussed subsequently.

For the biped in Figure 4.1, neither foot can penetrate the ground, the knee joints cannot extend beyond the fully straight position, and both feet are assumed not to slide when in contact with the ground. Since each toe and heel is independently characterized by non-penetration and no-slip with the ground, the flight phase dynamics (4.1) can be subject to the following kinematic (physical) constraints,

$$\mathbf{\Phi}_h(\mathbf{q}) = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ \varphi_2 \\ \varphi_5 \end{bmatrix} = \mathbf{0}, \mathbf{\Phi}_n(\mathbf{q},\dot{\mathbf{q}}) = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \mathbf{0}, \tag{4.2}$$

where (x_i, y_i), $i \in \{1,2,3,4\}$ are the toe and heel coordinates, see Figure 4.1, while $\varphi_2 = \theta_2 - \theta_1$ and $\varphi_5 = \theta_5 - \theta_4$ are the relative angles at the knee joint. Instead of using (4.2) directly, the forthcoming control development only requires a particular information contained by the constraint matrix which is defined by

$$\mathbf{A}(\mathbf{q}) = [(\partial \mathbf{\Phi}_h / \partial \mathbf{q})^T, (\partial \mathbf{\Phi}_n / \partial \dot{\mathbf{q}})^T]^T. \tag{4.3}$$

Depending on the configuration of the robot, the constraints concatenated in (4.2) and (4.3) are active when they restrict the motion and inactive when they do not. In order to ensure that $\mathbf{A}$ only contains the active constraints, the configuration of the robot is monitored through the motion

to identify and eliminate the inactive constraints by zeroing the corresponding row in (4.3). The constraint matrix obtained in this way, carries the kinematic information from the configuration of the biped utilized in the forthcoming control development.

Control Approach

The control approach considered here can be discussed in two stages. In the first stage, the robot is acted upon by generalized control forces $\mathbf{Q}_d$ which are (partially) referenced to the inertial frame to make coordination of patterned movement intuitive. On a real robot however there is no associated control actuator which can realize the generalized control forces (referenced to the inertial frame) directly. Accordingly, in the second stage, $\mathbf{Q}_d$ is recomputed to *equivalent* joint torques, $\mathbf{u}$, which can be directly commanded through the actuators to coordinate the robot. As follows, we provide a systematic description of the outlined control idea on a seven link robot.

Generalized Control Forces

In order to generate patterned movement without trajectory tracking, the seven link robot is provided with seven control elements which are spring-damper couples with fixed equilibrium points, see Figure 4.2. Each control element can be characterized with three control parameters, a stiffness constant, a damping constant, and an equilibrium angle. These parameters are changed as piecewise constant functions through four separate states along the walk using a configuration-based switching controller.

Computing the Generalized Control Forces

For a given set of control parameters, the desired generalized control force is computed as

$$\mathbf{Q}_d = -\mathbf{K}_d(\boldsymbol{\phi} - \boldsymbol{\phi}_d) - \mathbf{B}_d\dot{\boldsymbol{\phi}}, \tag{4.4}$$

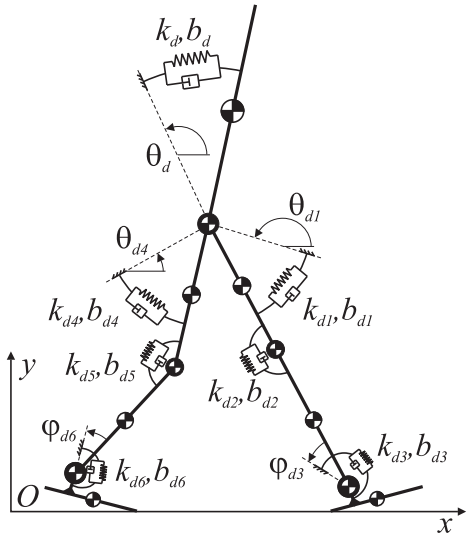

Figure 4.2 The control elements and the control parameters on a 7-link robot.

where $\boldsymbol{\phi} = [\theta, \theta_1, \varphi_2, \varphi_3, \theta_4, \varphi_5, \varphi_6]^T$ is obtained by position feedback, $\dot{\boldsymbol{\phi}}$ is known from a corresponding velocity feedback, while the control parameters concatenated in the stiffness matrix, $\mathbf{K}_d$, damping matrix $\mathbf{B}_d$ and the equilibrium angles $\boldsymbol{\phi}_d = [\theta_d, \theta_{d1}, 0, \varphi_{d3}, \theta_{d4}, 0, \varphi_{d6}]^T$ are assigned by the configuration-based switching controller as discussed in the forthcoming subsection.

Before we proceed further, let us point out that while (4.4) has the same form as a usual PD control law, the philosophy and the application of (4.4) is entirely different. Specifically, we use piecewise constant (fixed) equilibrium angles $\boldsymbol{\phi}_d$ instead of tracking a predefined desired trajectory $\boldsymbol{\phi}_d = \boldsymbol{\phi}_d(t)$. While this difference may not seem crucial, one can recognize that contrary to the precise trajectory tracking which requires high-gain PD control, the fixed angular references, Figure 4.2, make high-gains not well suited to walking control. Accordingly, utilization of low control gains is not only a preference to generate compliant motion, but also a requirement for stable gait synthesis.

Let us mention that utilizing $\boldsymbol{\phi}$ (as defined above), the generalized control force (4.4) only influences the rotational dynamics of the robot. In this light, the translational motion, (x,y), is not controlled directly but rather is allowed to be an outcome of proper postural coordination and interaction of the robot and the environment. It can also be seen that $\mathbf{Q}_d$ operates on a mixed

reference frame by utilizing absolute coordinates defined between the robot and the inertial frame, and also relative angular coordinates defined between the links. The specific coordinate choice was selected to mitigate the parameter tuning process discussed below.

Parameter Modulation based on the Robot Configuration

In order to achieve a walking motion, the control parameters are selected depending on the configuration of the robot. In this light, we define four separate states for each leg depending on whether the toe and/or the heel touch the ground and whether the leg is fully extended at the knee joint, see Figure 4.3. The configuration-based control parameter modulation is implemented with four "$if-else$" statements. In each state, the logic assigns three control parameters for each of the seven control elements from a set of user-defined desired parameters.

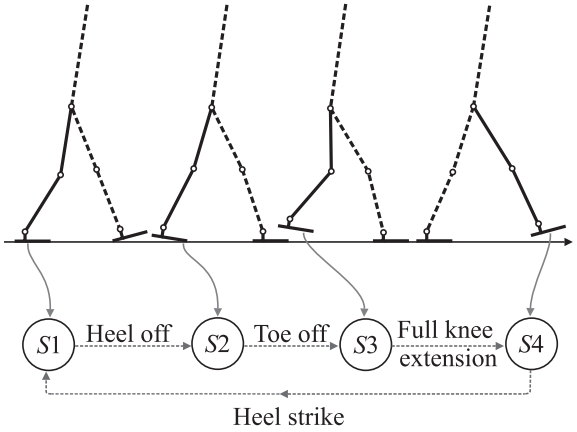

Figure 4.3 The configuration-based switching logic with the four separate states. The particular state-flow, $S1 \rightarrow S2 \rightarrow S3 \rightarrow S4 \rightarrow S1$..., together with the corresponding switching events, which correspond to normal walking, is indicated with dashed lines.

Utilizing the control elements which partially references the control influence to the inertial frame is recognized to support intuitive parameter tuning. As follows, we describe a biologically inspired approach for the parameter selection with intention to mimic muscle activation of a walk-

ing human subjects, [115], [3].

Along a walk, one of the primary objectives is to keep the upper body in an upright vertical position. Utilizing the control elements which act between the body and the inertial reference frame, one can set the stiffness parameter and the equilibrium angle to provide a near upright position for the body, and then use the associated damping parameter to influence the body dynamics. A similar idea can be used to generate leg oscillation (with respect to a fixed inertial reference) by using the control elements attached to the thigh. Specifically, in swing, a low stiffness and low damping element pulls the leg towards a fixed hip extension configuration (specified with an equilibrium angle), while in the stance phase, a higher stiffness and higher damping is assigned to the same control element which is encouraging the stance leg to move towards a fixed hip flexion angular configuration. The knee stiffness and damping is also modulated, by means of using a relatively high value in stance (to support the body with the help of the knee stop), and employing only slight damping to generate (partially) ballistic swing. Controlling the ankle is set up by mimicking the strategy taken by humans. Accordingly, the ankle stiffness is used to accumulate elastic energy from the middle stance and provide a characteristic ankle push-off at late stance. The main control parameter at the swinging ankle is an equilibrium point which should be adjusted to provide slight dorsiflexion, to avoid stumbling and scuffing during swing.

Actuator Torques

While $\mathbf{Q}_d$ is straightforward to compute, it does not represent the joint torques, and as such, it cannot be directly used to coordinate the motion of the robot. Practically, one may want to find the torques $\mathbf{u}$ which, while directly commanded through the actuators, provides the same motion the robot would have by applying the desired generalized control forces $\mathbf{Q}_d$. Since we are interested in control of the constrained motion, let us recall here the acceleration component of the constrained motion generated by the desired generalized control forces,

$$\ddot{\mathbf{q}}_d = \mathbf{R}^{-1}\mathbf{N}\mathbf{R}^{-T}\mathbf{Q}_d, \tag{4.5}$$

where $\mathbf{R}$ is the upper triangular Cholesky factorization of the mass matrix $\mathbf{M} = \mathbf{R}^T\mathbf{R}$ (where $\mathbf{M}$ is defined in (4.1)), $\mathbf{N} = \mathbf{I} - (\mathbf{A}\mathbf{R}^{-1})^+(\mathbf{A}\mathbf{R}^{-1})$ is the null-space projection operator of the inertially-weighted constraint matrix (where $\mathbf{A}$ is defined by (4.3)). The interested reader can find the derivation of (4.5) in [27]. Using the above relation, one can also define the constraint consistent accelerations generated with the actuator torques,

$$\ddot{\mathbf{q}}_u = \mathbf{R}^{-1}\mathbf{N}\mathbf{R}^{-T}\mathbf{E}\mathbf{u}. \tag{4.6}$$

Following the main objective $\ddot{\mathbf{q}}_d = \ddot{\mathbf{q}}_u$, one can equate (4.5) and (4.6) and solve the corresponding linear equation for $\mathbf{u}$. Depending on the constraint configuration of the robot however, this solution may not exist (in cases when the robot is underactuated, for example in flying phase or if only one toe or one heel is contacting the ground). In order to obtain an approximate solution even when the robot is underactuated, we propose not to solve $\mathbf{u}$ from $\ddot{\mathbf{q}}_d = \ddot{\mathbf{q}}_u$ directly, but rather to define a solution which minimizes the acceleration energy between the desired and the real motion $(\ddot{\mathbf{q}}_d - \ddot{\mathbf{q}}_u)^T\mathbf{M}(\ddot{\mathbf{q}}_d - \ddot{\mathbf{q}}_u) \rightarrow \mathit{min}$. The general solution to this problem is given with

$$\mathbf{u} = \mathbf{A}_u^+\mathbf{N}\mathbf{R}^{-T}\mathbf{Q}_d + (\mathbf{I} - \mathbf{A}_u^+\mathbf{A}_u)\mathbf{u}_0, \tag{4.7}$$

where $\mathbf{A}_u^+$ is a Moore-Penrose generalized inverse (pseudoinverse) of $\mathbf{A}_u = \mathbf{N}\mathbf{R}^{-T}\mathbf{E}$, [59], $\mathbf{I} \in \mathbb{R}^{6\times 6}$ is an identity matrix, while $\mathbf{u}_0 \in \mathbb{R}^6$ is an arbitrary (joint torque) vector. A particular solution provided with the first term in the above relation ($\mathbf{u}_0 = \mathbf{0}$) minimizes the squared Euclidean norm of the joint torques and as such, due to its optimal character, it is a preferred solution herein.

In addition to the pseudoinverse solution (first term in (4.7)), one can also utilize $\mathbf{u}_0$ (with the second term in (4.7)) to control the contact constraint forces. Although the intention here is not to maintain the grand contact constraints by force control, partial utilization of this idea is recognized as a convenient way to actively modulate the knee stiffness once the leg of the robot is fully extended.

Let us point out here that (4.7) defines a full-body control law where each joint torque

depends on the motion of the robot in whole. It is also important to recognize that no inverse dynamics is performed to cancel the gravitational and inertial forces along the motion and enforce a predefined reference trajectory on the system. Instead, the natural dynamics of the robot is allowed to substantially influence the motion of the robot which is synthesized using joint torques that mimic the effect of spring-damper forces (partially) applied between the robot and the inertial reference frame. As subsequently demonstrated, the described approach allows emulation of a human-like walking of a seven-link biped robot.

Robot Design

Practically, realization of a natural motion requires utilization of backdrivable actuation which allows passive joint motion, similar to human joints. In this light, for the purpose of validation of the proposed control method, we design a seven-link biped robot keeping in mind the mentioned design requirement.

The 7-link biped robot, depicted in Figure 4.4 is an experimental prototype which is $1.2m$ tall and $14.3kg$. The geometric parameters and mass distribution on the robot is specified in Table 4.1. Below, we discuss the upper body design, joint design, foot design, and the sensory-system on the robot.

Upper Body

The robot has an upper body which carries $4.54kg$, ($10lb$), of weights which are distanced $0.2m$ from the hip joint, see Figure 4.4. The purpose of these weights is to represent (with the rest of the robot trunk) a reasonable mass for the head, arms, and trunk of a $1.2m$ tall biped. The body is provided with a single-axis gyroscope (Analog Devices, ADXR150) which directly measures its sagittal plane angular velocity. The sensor is characterized with $\pm 150^o/s$ measurement range and a noise density of $0.05^o/s/\sqrt{Hz}$. In order to reduce the noise level, the analog signal is filtered with a first order low pass filter with a $50Hz$ roll-off frequency.

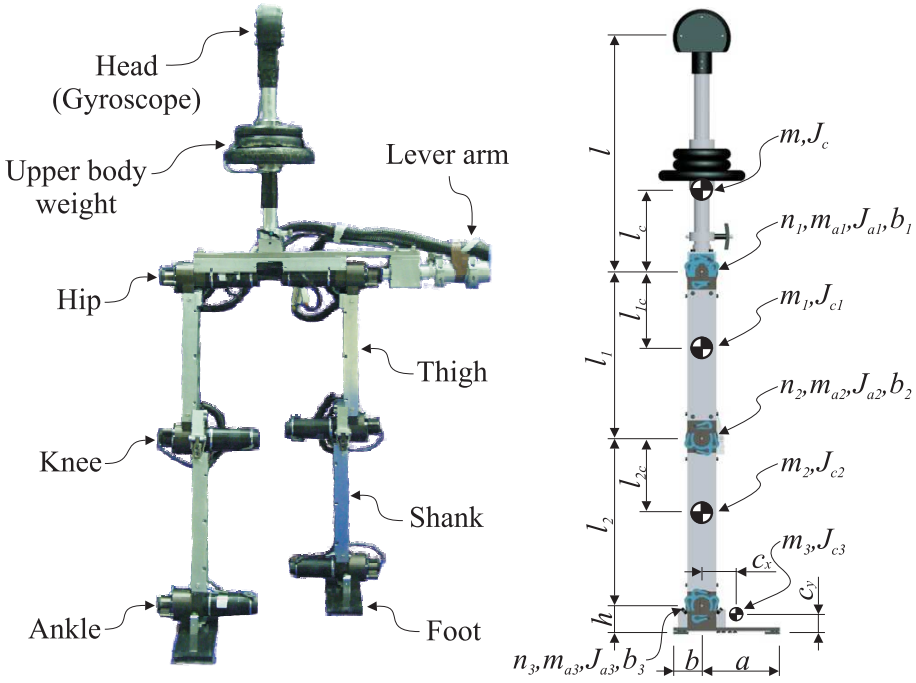

Figure 4.4 Left: Experimental prototype of a 7-link dynamic walker developed at the Vanderbilt University, Center for Intelligent Mechatronics. Right: CAD-model, side view of the 7-link biped. The values for the model parameters are reported in Table 4.1, specifically, the geometric parameters, the link masses m, m_1, m_2, m_3, link moments of inertias $J_c, J_{c1}, J_{c2}, J_{c3}$; actuator masses m_{a1}, m_{a2}, m_{a3}, actuator moments of inertias J_{a1}, J_{a2}, J_{a3}, the gear ratios on the reducers on the joints $n_1 : 1$, $n_2 : 1$, $n_3 : 1$, and the experimentally identified joint level linear viscous damping constant b_1, b_2, b_3.

Table 4.1 Geometric and inertial parameters of the robot with total mass of $M = 14.3kg$ and hight of $L = 1.2m$.

Structure	no.$(*)$	$l_*[m]$	$l_{c*}[m]$	$m_*[kg]$	$J_{c*}[kgm^2]$
Body	–	0.390	0.185	6.12	0.0210
Thigh	1	0.295	0.147	0.67	0.0096
Shank	2	0.298	0.140	0.55	0.0069
Foot	3	0.183	–	0.36	0.0007
Foot	$a[m]$	$b[m]$	$h[m]$	$c_x[m]$	$c_y[m]$
	0.137	0.046	0.055	0.014	0.035
Actuators	no.$(*)$	n_*	$m_{a*}[kg]$	$J_{a*}[kgm^2]$	$b_*[Nms]$
Hip	1	21	0.84	0.0067	≈ 0.13
Knee	2	12	0.84	0.0022	≈ 0.11
Ankle	3	21	0.84	0.0067	≈ 0.05

Joint Design

The seven link robot has an upper body, hip, knee, ankle and human-like foot. In the biped prototype, a unified design is utilized for the hip, knee and ankle joints with slight modifications made for differences in range of motion and attachment points. Figure 4.5 depicts the specific design solution of the knee joint.

Actuator Unit

The robot is actuated with six 150W brushed DC-motors (Maxon RE40) through low gear ratio planetary reducers (Maxon GP42C), specifically 21:1, 12:1, 21:1 for the hips, knees and ankles respectively. A low gear ratio drive (i.e., backdrivable joint design which allows substantial power flow between the inertial load and the actuator) allows passive motion of the joints which is a precondition to leverage natural dynamics through actuated dynamic walking without excessive energy requirement.

Unlike the backdrivable actuation unit, highly geared joints, often used for robot manipulators and also in actuated walking robots, would introduce significant joint friction and prevent power flow from the links to the actuators. While such highly geared actuation allows decoupled

motion coordination with local (joint level) control, a system with backdrivable joints becomes highly coupled and as such more difficult to control. In the present paper, however, low gear ratio actuators are utilized to meet the precondition for natural looking and energy-efficient motion.

Angular Joint Sensors

Each joint is provided with an incremental quadrature encoder (Maxon, ENC-MR-L-1024-CPT) attached to the motor shaft, Figure 4.5. The reference position for each of the six encoders is identified (in a static stance phase during initialization) using two acceleration sensors (Analog Devices, ADXL203) located on the upper body and the upper right leg. The implemented sensors provide an accurate joint angle, $\boldsymbol{\varphi}$, measurement which can be characterized with quantization step of $4.2^o \times 10^{-3}$ at the hip and ankle joints and $7.3^o \times 10^{-3}$ for the knee joints.

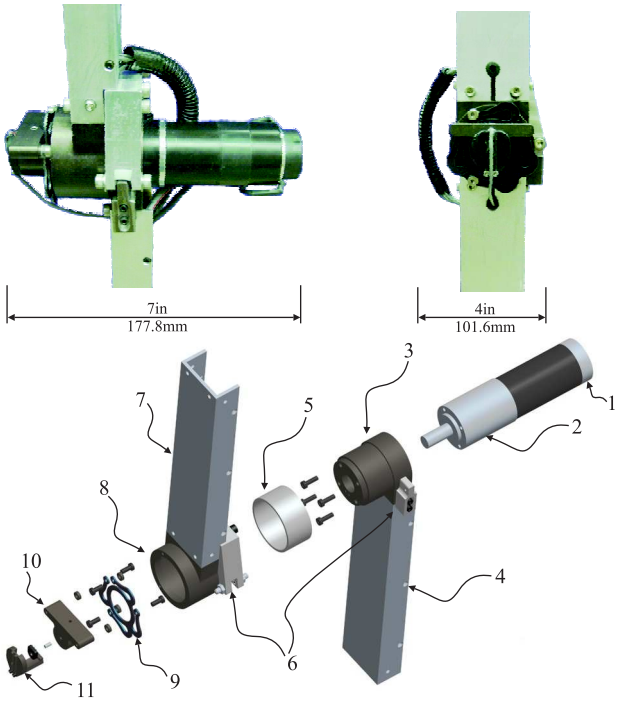

Figure 4.5 Top: Knee joint on the robot. Bottom: CAD model - exploded view of the knee joint: 1) encoder; 2) actuation unit - motor and the gearhead; 3) inner bearing housing; 4) lower leg; 5) Teflon sleeve bearing; 6) hard stop at full knee extension; 7) upper leg; 8) external bearing housing; 9) elastic coupling; 10) connecting element; 11) potentiometer and housing (not used in present implementation).

Foot Design and the Foot Sensors

The foot of the robot is constructed from ABS plastic, each of which is instrumented with four force sensing resistors (Interlink, 402 FSR), specifically, two FSR's on each toe and heel. These sensors are located between the underside of the foot and a thin foot-plate made from spring steel, Figure 4.6. When the toe and/or the heel touches the ground, the circular rubber touch-pad (located on the foot-plate) touches the foot sensors. The corresponding signal serves to identify the contact configuration between the foot and the ground. Near to the toe and the heel (which are the expected contact areas) the foot-plate is supplemented with silicon rubber pads with high frictional properties, good abrasive durability and appropriate shock absorbing capability. In the proposed control implementation, measurements of the contact forces or moments are not required. As such, the feet are not equipped with load cells.

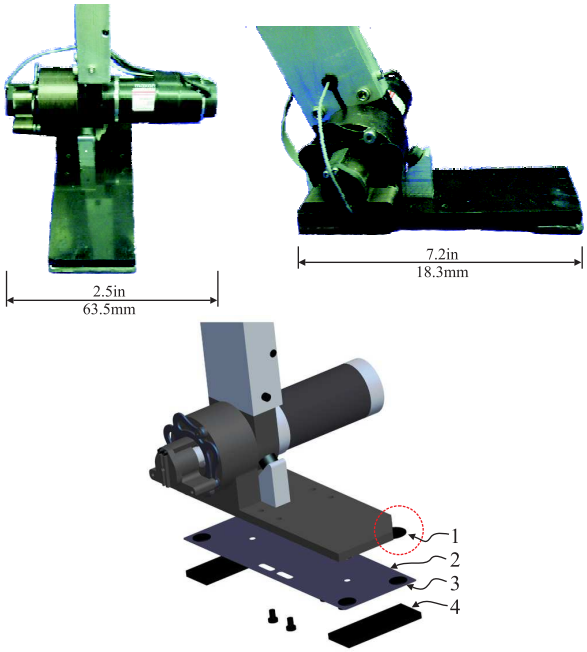

Figure 4.6 Top: Foot of the robot. Bottom: CAD model - exploded view of the foot: 1) FSR sensor; 2) foot-plate; 3) sensor touch-pad; 4) rubber foot contact-pad.

Comment on Planar Walking

For purposes of the experimental implementation of actuated dynamic walking, we consider planar motion of the robot. Specifically, the robot is attached through its hip to a lever arm which keeps the biped on a circular path, with $1.6m$ radius while walking, see Figure 4.4. This solution although not an ideal realization of a sagittal plane walk, was convenient for experimentation. Similar realization was employed for the MIT Spring Turky/Flamingo, [18], for Rabbit [21] and also for RunBot [24]. In the current realization, the lever is only used to constrain the motion of the robot but it is not instrumented or exploited in any way to provide an inertial reference for the sensory-system on the robot.

Real-Time Control Implementation

The proposed closed-loop controller was developed on desktop PC with the real-time interface provided by MATLAB / Simulink Real Time Workshop. In the following, we discuss implementation of the closed-loop controller.

Feedback Information from the Contact Configuration

In order to identify the active (and inactive) constraints imposed by foot contact with the ground we have utilized the foot sensors. Specifically, whenever the toe and/or the heel are touching the ground, the analog signal from the foot sensors are thresholded to generate an on/off type output, which is used in the control implementation. In addition to the contact condition between the foot and the environment, knowledge of full knee extension is also required for the control approach. For this purpose, the knee angle encoders are monitored to determine whether the leg is fully extended, $\varphi_{2,5} \approx 0$.

Position and Velocity Feedback

In the present paper we utilize both the joint angles $\boldsymbol{\varphi}$ (which are measured by encoders), and also absolute link angles $\boldsymbol{\theta} = [\theta, \theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6]^T$, to implement the proposed feedback-control approach. Although the absolute orientations are not measured directly, they can be calculated by $\boldsymbol{\theta} = [\theta, \theta + \varphi_1, \theta + \varphi_1 + \varphi_2, \theta + \varphi_1 + \varphi_2 + \varphi_3 - \pi/2, \theta + \varphi_4, \theta + \varphi_4 + \varphi_5, \theta + \varphi_4 + \varphi_5 + \varphi_6 - \pi/2]^T$ if the upper body angle θ is provided. Accordingly, in the following discussion we will only describe how to compute θ. Once the angular configuration is known, the velocity information is obtained by numerical differentiation.

Computing the Absolute Orientation of the Upper Body

Whenever the robot is not underactuated (at least one foot is flat on the ground or; either the backward toe or heel and the forward toe or heel is on the ground), the system has six or less degrees of freedom, and the absolute angular orientation for the upper body can be calculated using the six encoder measurements $\boldsymbol{\varphi}$ formally stated as,

$$\theta = \theta(\boldsymbol{\varphi}). \tag{4.8}$$

The related kinematic computation is performed exactly if either of the feet touches the ground in two contact points. Otherwise, if the foot touches the ground in three or more contacting points, the upper body angle is solved in a least square sense to cope with kinematic redundancy and expected inconsistency in the measurements.

Estimating the Absolute Orientation

There are two cases where (4.8) cannot be applied. This is if the robot moves through underactuated configuration (i.e., flight phase or if only one toe or one heel is contacting the ground) or if (4.8) is singular or nearly singular. Under these conditions, we have utilized the gyroscope signal to directly provide the angular velocity of the upper body, $\dot{\theta} = \dot{\theta}_g$, and an estimate of the

absolute orientation of the upper body θ by

$$\theta = \theta(t_0) + \int_{t_0}^{t} \dot{\theta}_g(\tau) d\tau, \tag{4.9}$$

where $t \in [t_0, t_1)$, t_0 is the time instant starting from which (4.8) could not be used, while t_1 is the time when (4.8) can again be used reliably. The interval $[t_0, t_1)$ for which the integration is performed is expected to be short (at most 10% per step duration). There are practical limitations which makes (4.8) preferred over (4.9). Specifically, integration of the analog gyroscope signal in longer time would cause drift in the position estimate, while filtering $\dot{\theta}_g$ to reduce the noise level will induce delay (phase-lag) on the feedback information from the motion (which may lead to stability problems in coordination), see [116]. These two issues, which are the main limitations of many inertial measurement units, are bypassed here by utilizing (4.9) only when needed for short time periods.

Concatenation of (4.8) and (4.9) allows reconstruction of the angular motion of the robot in the inertial frame, $\boldsymbol{\theta}$. Note that switching between the two computational schemes may induce discontinuities on position and also on the velocity signals. This issue is prevented with blending between the two kinematic solutions in a short time window after the switching instant.

Computing the Actuator Torques

The most specific part of the real-time control implementation is a control torque computation (4.7). Below, we provide further insight in this context.

The real-time implementation of (4.7) is developed in Matlab/Simulink environment. The model parameters required for this computation are the constant matrix **E**, the constraint matrix **A** and the mass matrix **M**. It is an intrinsic property of the model which allows computation of $\mathbf{A}(\boldsymbol{\theta})$ and $\mathbf{M}(\boldsymbol{\theta})$ based only on the angular configuration of the robot (i.e., computation of (x, y) is not required). Once the model parameters are known, a procedural way to implemented (4.7) is provided with two standard routines: the Cholesky factorization, and the pseudoinversion. Note that while

the former is not a delicate operation, the pseudoinverse calculation (which is usually based on the singular value decomposition (SVD) [65], [59] is a numerically involved operation. Despite this computational requirement, the current Matlab implementation of the complete closed-loop controller was real-time capable with 1000Hz sampling rate on an Intel Core 2 Quad 2.4Ghz desktop computer.

Experimental Characterization of the Robot

In this section, experimental and simulation results are presented to: verify the model parameter identification, characterize the passive motion of the device and validate the electronic implementation of the control actuators.

Parameter Identification

In order to compute the actuator torques, the model parameters **E**, **A** and **M** are required. For the purpose of real time control, these parameters are derived in analytical closed form. The specific geometric and inertial parameters for each link which are used to compute **A** and **M** are provided by measurements, and estimation from the CAD model. The following experiments and simulations are performed with the parameter set reported in Table 4.1.

Free Swing Experiment

The robot introduced in this paper is provided with low gear ratio drives. In order to show that the corresponding actuator unit indeed allows passive motion (necessary for ballistic swing), we have conducted free swing experiments. The experimental results are depicted in Figure 4.7.

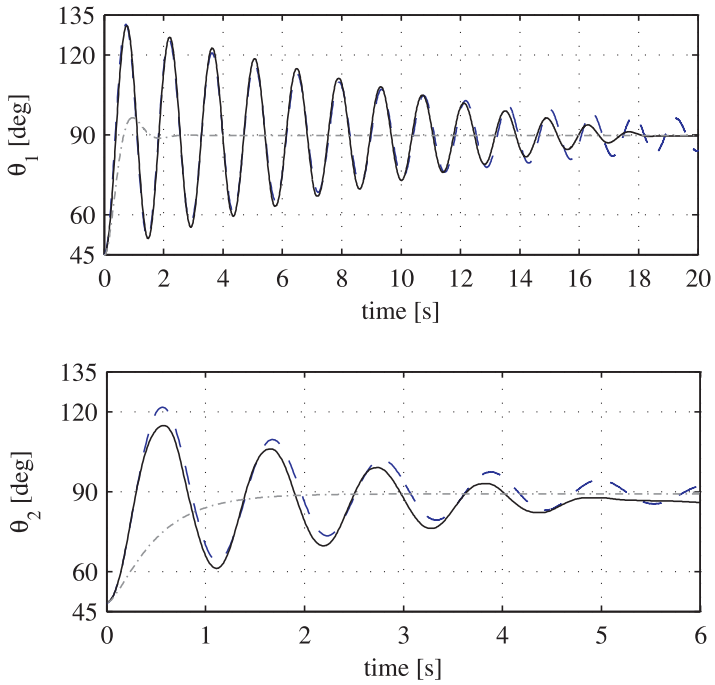

Figure 4.7 Free swing experiments which characterize the passive (uncontrolled) dynamics of the hip and the knee joint. The motion of the device is depicted with (black) solid line while the simulated response is plotted with dashed (blue) lines. The difference in the low velocity area is mainly due to the Coulomb friction and the cabling which is neglected in the simulations, (the asymmetric effect of the cabling can be seen in the knee response). In order to clearly show the difference between a backdrivable actuator unit utilized here, and a usual highly geared joint design, the dotted (gray) lines depict the model prediction of corresponding motion the robot would have with 105 : 1 and 60 : 1 gear ratio on the hip and knee respectively. Due to the low inertia of the foot, a similar free swing experiment is recognized not well suited to characterize the dynamics of the ankle joint, and as such is not conducted here.

PD Control Experiments

In order to validate the implementation of the control actuators together with the dynamic model, further experiments were conducted with PD control which operates with periodically modulated equilibrium points. The experimental response and the model prediction are both depicted in Figure 4.8. Note that using low gain PD control allows us to validate both the implementation of the actuator unit and the model parameters, namely, the response of the system in this case is not prescribed with the control force but rather substantially influenced by the natural dynamics of the robot.

Dynamic Walking

In this section we provide simulation results and experimental data for the walking biped. A corresponding video is included in the supporting multimedia material.

Simulation Result

The control approach implemented here was initially verified using numerical simulations on an anthropometric biped model, see [27], [25]. In the present paper, the authors utilized the same simulation tool to select control parameters for an energy efficient gait for the present robot. The gait resulting from a suitable set of control parameters is depicted in Figures 4.9 and 4.10. The related walking motion is characterized with average speed of $v_{avg} = 0.53m/s$ (Froude number $Fr = v_{avg}/\sqrt{gL_{leg}} = 0.21$) and mechanical cost of transport, (mech.energy)/(weight $\times$ distance traveled), of $c_{mt} = 0.15$.

While the simulation model was developed with emphasis on physical consistency, it does not incorporate all the experimental conditions which apply to the robot. Aside from the usual implementation nonidealities such as measurement noise and phase-lag on the feedback signals, other unmodelled effects were also not included in the simulation. Specifically, instead of a sagittal plane motion, the real robot walks in a circle ($1.6m$ in radius). Moreover, the lever arm used to

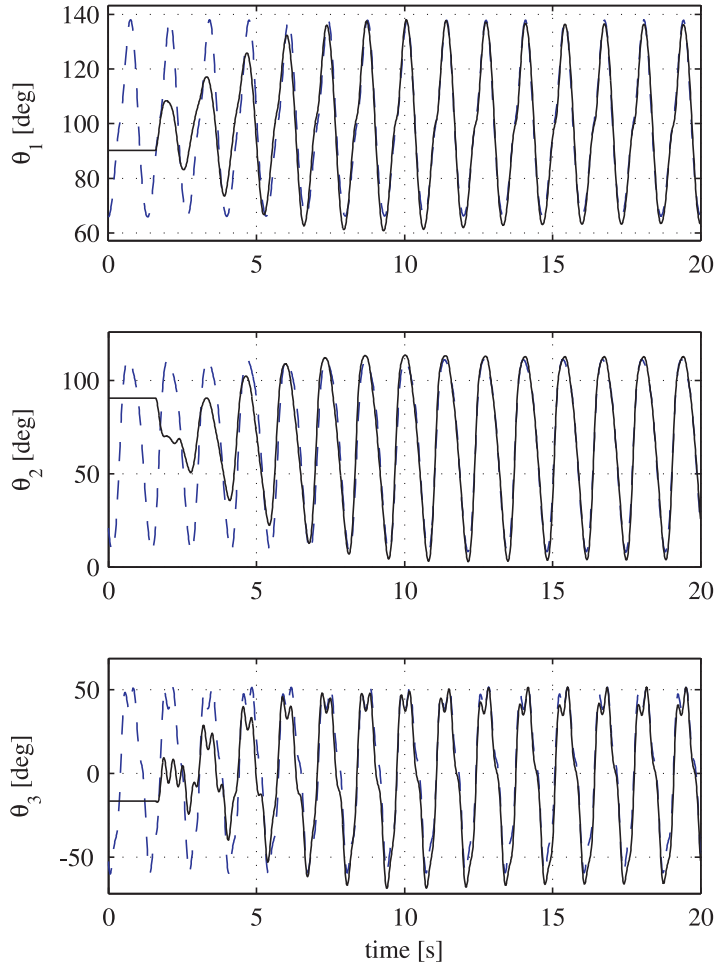

Figure 4.8 Low gain PD control experiment. Solid line (black) represents the motion of the device while the dashed line (blue) is the corresponding model response. The experiment is performed by applying a control torque vector $\mathbf{u} = -\mathbf{K}_d(\boldsymbol{\varphi} - \boldsymbol{\varphi}_d) - \mathbf{B}_d\dot{\boldsymbol{\varphi}}$, $\mathbf{K}_d = [2, 1.5, 1, 0, 0, 0]^T$, $\mathbf{B}_d = [0.2, 0.2, 0.1, 0, 0, 0]^T$, $\boldsymbol{\varphi}_d = [(\pi/3)sin(1.5\pi t) + \pi/20, (\pi/6)sin(3\pi t - pi/2) - \pi/2, (\pi/6)sin(6\pi t) + \pi/6]^T$.

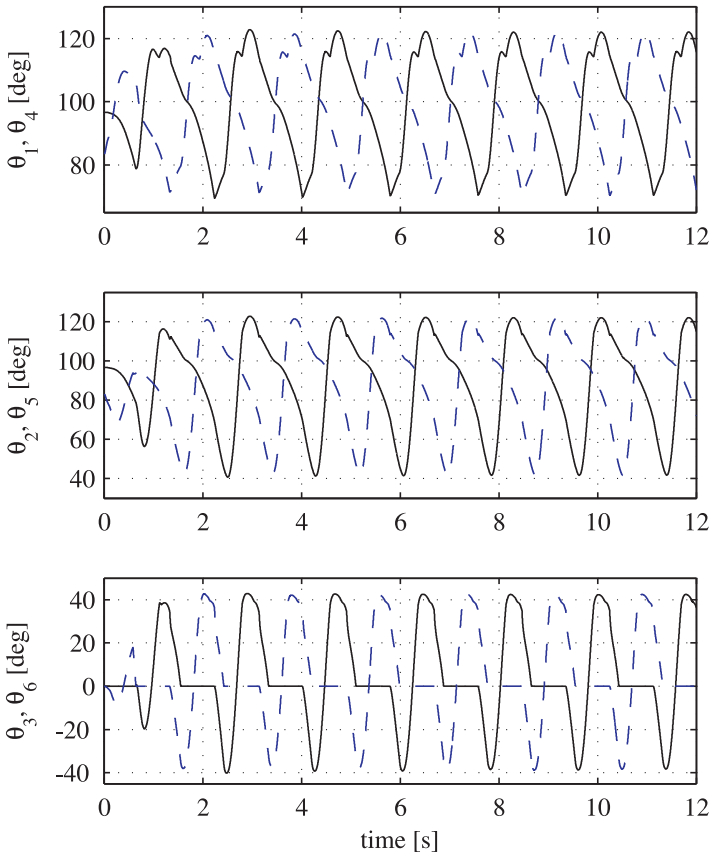

Figure 4.9 Motion plots for the simulated robot. The corresponding stroboscopic view is depicted on Figure 4.10.

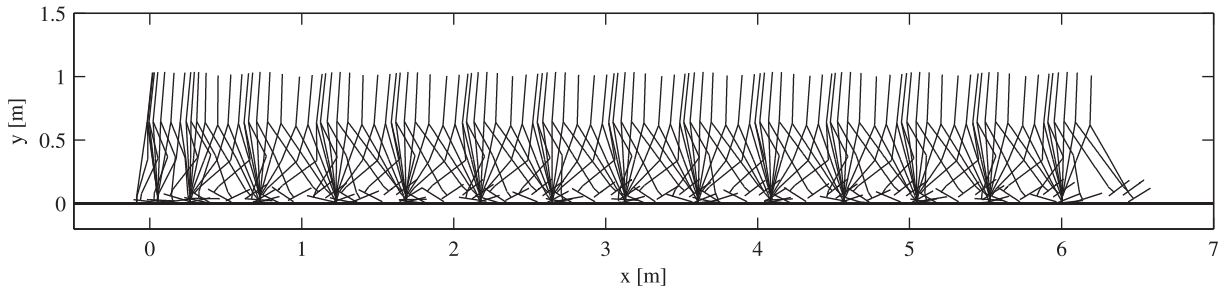

Figure 4.10 Stroboscopic view of the simulated walk over $t \in [0, 12]s$.

guide the robot is neither counterbalanced nor is it taken into account in the mass matrix used in the control computation. If high gain trajectory tracking were applied to control the robot, these effects would likely be overridden (by the control forces). In the present situation however, the experimental conditions will substantially alter the natural dynamics of the robot which is proposed to be leveraged. While the mentioned implementational nonidealities influence the walking style, they do not prevent the proposed control approach to generate coordinated motion.

Experimental Result

The control approach was implemented in real-time and used to coordinate the walking of the biped robot. A representative experimental data is shown in Figure 4.11. A frame sequence extracted from the experimental video is shown in Figure 4.12. The walking of the robot is characterized with: average forward speed of $v_{avg} \approx 0.5 m/s$ which corresponds to Froude number $Fr = v_{avg}/\sqrt{gL_{leg}} = 0.2$. According to the dynamic similarity hypothesis [117], the calculated Froud number indicates that the presented walking can be compared with the walk of the actuator-assisted Cornell dynamic walker $Fr = 0.18$ [1] and the fully actuated Honda Asimo $Fr = 0.17$ [118].

Following the concept of specific resistance, [106], we adopted the mechanical cost of transport to estimate the energy requirement of the presented walking robot. The estimated value obtained by the experimental data is $c_{mt} = 0.35$. Since the present robot is fully actuated (i.e., all joints are under continuous closed loop control), the generated walk may not be as efficient as one can obtain by actuated assisted passive dynamic walkers, see the Cornell biped $c_{mt} = 0.05$ (which has the same efficiency as humans). However, the carefully developed control approach and the backdrivable actuator design adapted herein appears to provide an energetic advantage of the presented robot over other actuated walkers which utilize highly geared joints and high-gain trajectory tracking control approaches. This claim can be supported by pointing out that the mechanical cost of transport for the Honda Asimo robot which utilizes the ZMP control paradigm is estimated to be $c_{mt} = 1.6$, see [1].

Beyond the above quantitative attributes, there are important qualitative similarities between the walking of the robot herein and a human being. Specifically, the walking experiment demonstrates a natural looking (partially ballistic) swing which is realized with passive knee (which is only slightly damped). Moreover, one can also recognize the characteristic preemptive ankle-push off with human-like flat foot. While an actuated curved foot is an accepted solution in actuator-assisted dynamic walking [1], [14], an actuated human-like (flat) foot which provides a preemptive ankle push-off to propel the robot forward, realized herein, is a major control challenge which is rarely utilized in practical implementations, see [119] for a related discussion.

Let us mention that during the preemptive ankle push-off, (when only the backward toe of the rear foot is on the ground), the robot is in underactuated configuration. Practically, underactuation characterized with foot rotation, is an indicator of dynamic disbalance (dynamic instability) [7], and as such it is not allowed through implementation of the ZMP control approach, although it is utilized by human beings [10]. Recently, [87] proposed a control solution and numerically demonstrated walking with controlled (prescribed) foot rotation. In the present paper, we have experimentally demonstrated that the proposed control approach can cope with foot rotation. Moreover, the foot rotation (preemptive ankle push-off) here is not prescribed (enforced by control), but rather is chosen by the robot only if recognized advantageous or necessary through its forward progression.

Conclusion

The authors have presented an experimental realization of a control approach which enables dynamic walking in a fully actuated robot. Rather than prescribe kinematic trajectories, state dependent control torques are utilized in motion coordination that “encourage” patterned movement. Implementation of the control methodology which, does not force the robot to follow a predefined motion, but rather allows it to chose its own walking style, step length and forward speed, is performed on a seven link biped robot which is designed with highly backdrivable joint actuation. The conducted walking experiment demonstrates human-like compliant dynamic walk-

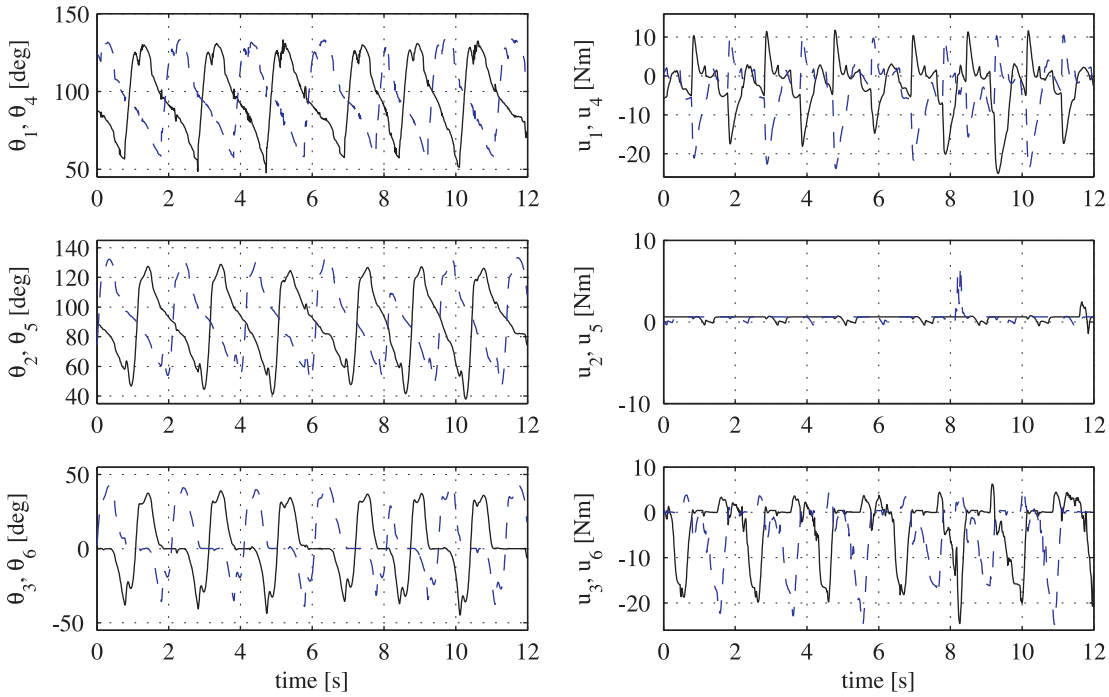

Figure 4.11 Actuated Dynamic Walking - Experimental data. Angular data from the motion of the robot. Black (full) lines depict the motion of the right leg while the blue (dashed) lines depict the motion of the left leg.

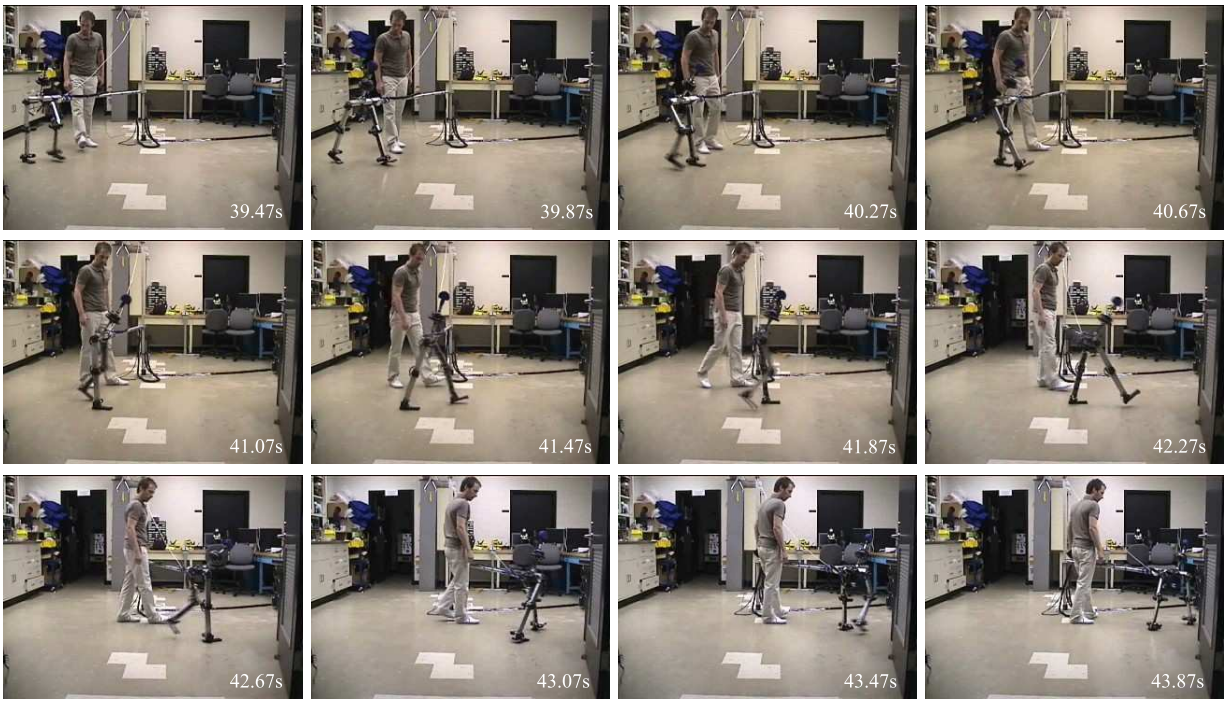

Figure 4.12 Frame sequence correspond to six subsequent walking steps extracted from the experimental video.

ing of a robot biped, characterized with natural looking (partially) ballistic swing and preemptive ankle push-off.

Appendix

This appendix contains: a description on the experimental setup; a modification on the hardware design, a description on control parameter tuning, and presents pictures and data from the conducted walking experiments.

Experimental Setup

The seven-link robot utilized in the walking experiments is designed to be a planar walker. In experimental realization, the robot is constrained with a lever arm which allows walking in a circle with $1.6m$ in radius. The lever arm, the weight of which is approximately $2kg$, is not counterbalanced but instead is considered as an unmodelled disturbance along the motion. Although the asymmetric connection of the robot to the arm also affects the motion, it does not prevent the proposed controller from generating stable walking. The experimental setup utilized in this work is depicted in Figure 4.13.

Enhancement on the Hardware Design

During the conducted experiments, failure in the joint couplings was identified on numerous occasions. In order to resolve this issue, the joints and the elastic joint couplings were redesigned and refabricated. In the new design, the potentiometers (not used during the experiments) are not included, the joint couplings are made thicker, while the overall joint assemblies became more compact and less compliant, see Figures 4.14. The new joints were tested through numerous experimental trials and have been recognized as a robust alternative to the primary design solution.

In addition to the design modification of the joints, the foot of the robot was also modified.

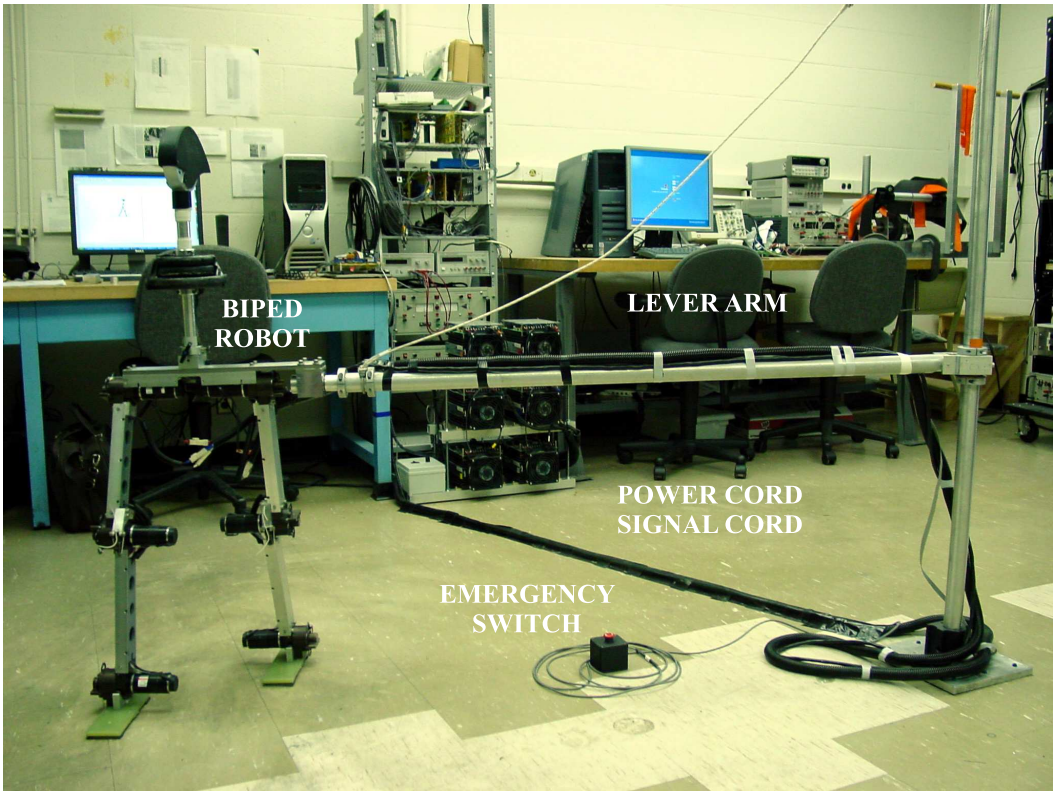

Figure 4.13 Experimental setup.

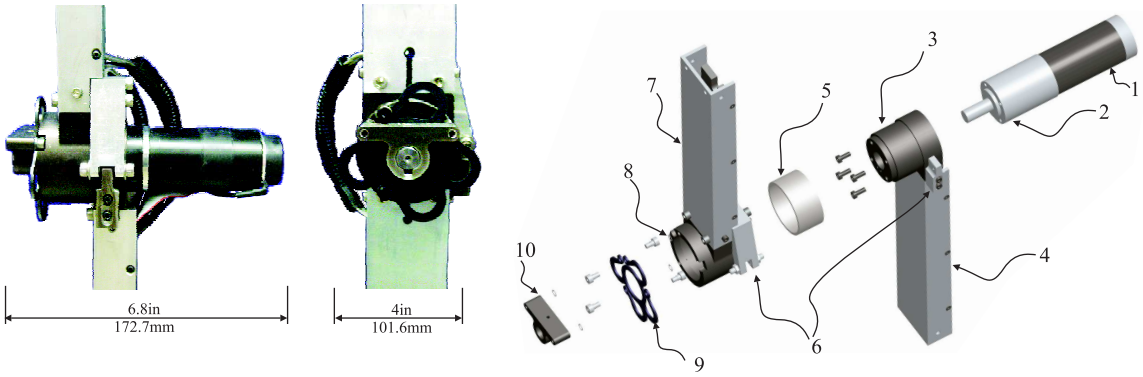

Figure 4.14 Left: Knee joint on the robot, side and frontal views. Right: CAD model - exploded view of the knee joint: 1) encoder; 2) actuation unit - motor and the gearhead; 3) inner bearing housing; 4) lower leg; 5) Teflon sleeve bearing; 6) hard stop at full knee extension; 7) upper leg; 8) external bearing housing; 9) elastic coupling; 10) connecting element.

Practically, due to the significant torsional moment applied through the foot (which can be 25 Nm) the initial design was weak in tensile strength. This issue was addressed by using Gerolite (G-10) as an alternative material for the new foot, instead of an impact resistive ABS plastic utilized on the initial foot. Due to the significantly better material properties, the new foot has been recognized as a reliable solution (no failure has been identified during the experiments).

Comment on the Sensory System

The walking experiments showed that imprecise sensory feedback can crucially affect the performance of the robot. One of the most sensitive elements in this context is the gyroscope measurement which provides the angular velocity feedback from the upper body. In addition to the gyroscope, identification of the foot contact configuration is also required to implement the proposed walking controller. For this purpose, force sensitive resistors (FSR's) are used as contact switches on the foot. The FSR's, while known to be non-reliable for force measurement, performed satisfactorily as contact switches as demonstrated through numerous walking experiments. Nonetheless, this simple design solution may be replaced with a more sophisticated foot equipped with load-cells to enhance the consistency of the feedback information from the foot contact configuration.

Comment on the Control Implementation

As any control approach, the one presented here also requires parameter tuning during its implementation. If high gain trajectory tracking control were utilized, this parameter tuning may not be a difficult task, (as long as the high gain control realization does not generate instability, due to practical issues such as noise or phase-leg on the signal measurements). The control approach proposed here does not use reference trajectories, and prefers low gains which allow compliant motion coordination on the robot. While selecting gains may not be as trivial in this case, it is made intuitive on the proposed walking controller.

The intuitive nature of the parameter tuning is provided by defining the generalized control forces on the upper body and the thighs with respect to the inertial reference frame Figure 4.2. This becomes obvious once coordination of the upper body is considered. Namely, achieving a (nearly) upright body position is trivial with the spring-damper couple located between the body and the inertial frame (i.e., one can set the equilibrium angle to nearly 90^o with a high enough stiffness parameter), while achieving the same using the hip torques (which have coupled influence on the body and the legs) would be considerably more complicated. A similar argument holds true if one considers oscillatory leg motion generation with the proposed spring-damper couples (which are referenced to the inertial frame) compared to the hip torques which could be considered referenced to the moving (oscillating) body. Utilizing this feature, a practical parameter tuning was performed independently first on the body (having the biped in double support stance phase with extended knees), then the swing leg is independently tuned (by having the biped in single support phase), and finally the stance leg parameters are tuned (again having the biped in single support phase). Since the biped is in whole coupled, experience has shown (as expected), that this preliminary parameter tuning needs to be refined with on-line tuning to achieve stable walking. This kind of parameter adjustment is made sequentially through walking experiments where the biped walked with a “little help” provided by the experimenter.

Experimental Results

A demonstrative report on the conducted walking experiments is given below. The figures depict frame sequences, extracted from walking videos, experimental data on the motion of the robot, and the computed joint torques used to coordinate the biped.

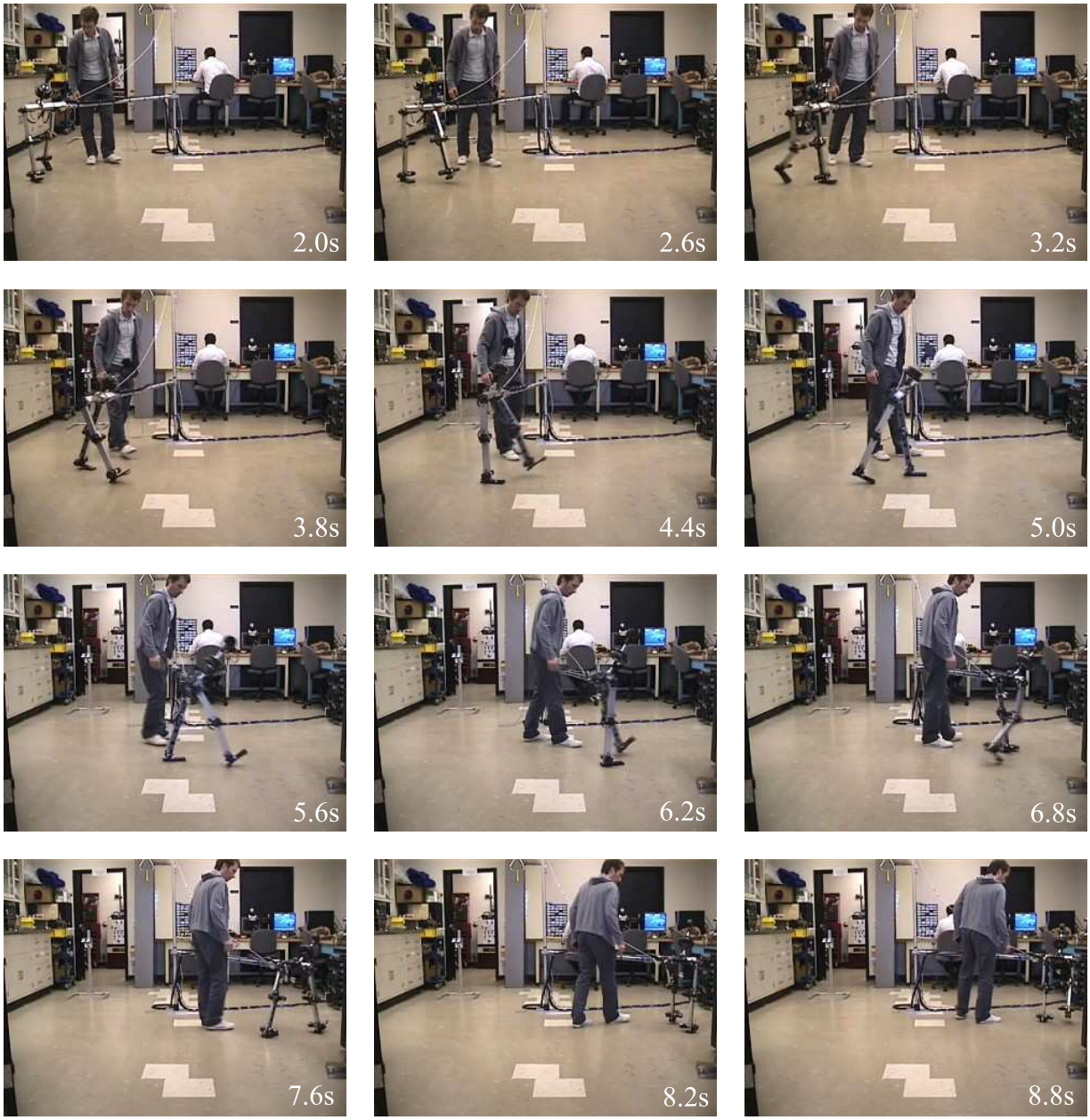

Figure 4.15 Experiment I: Frame sequence of a walking experiment. The walking is characterized with: average step length of $0.5m$, average stepping frequency of $1Hz$, average forward speed of $0.5m/s$ (Froude number $Fr = 0.2$), specific mechanical cost of transport $c_{mt} = 0.32$. Comparatively, this cost is (approximately) six times higher than human efficiency (also reproduced by the Cornell dynamic walker), while it is five times lower than the value estimated for the Asimo robot, [1]. During the walking experiment, the biped utilized a characteristic ankle push-off, preferred by humans. The experiment also verified proper coordination of the robot through short $0.1s$ underactuated motion phases (when only the forward heel was on the ground), see [2].

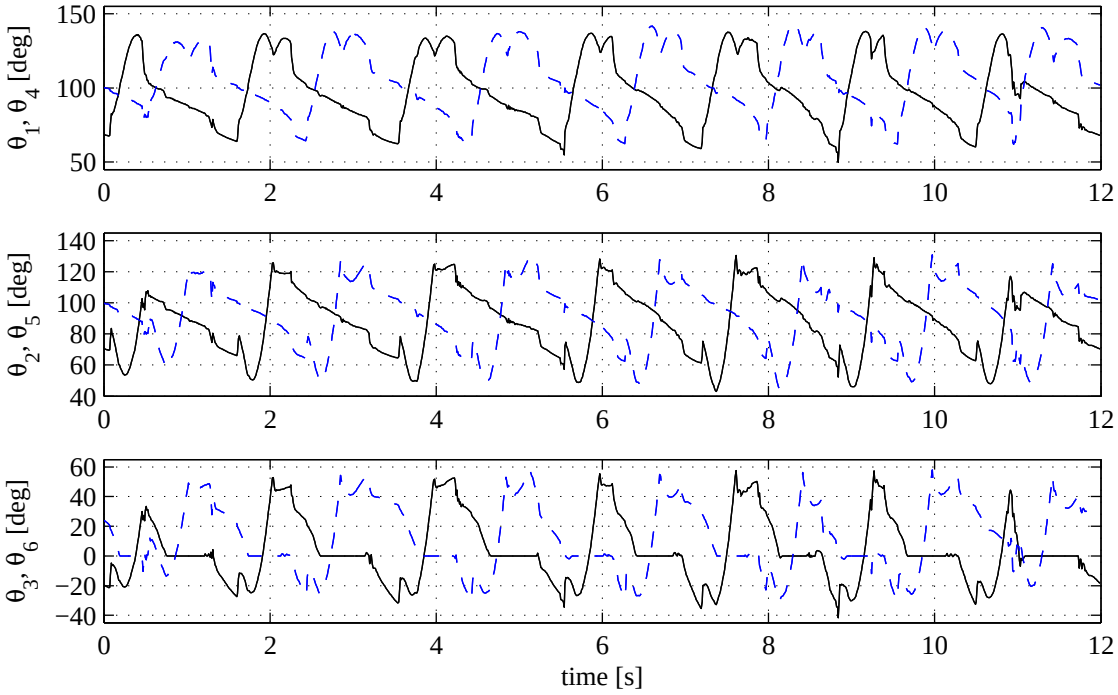

Figure 4.16 Angular motion corresponding to the frame sequence depicted on Figure 4.15.

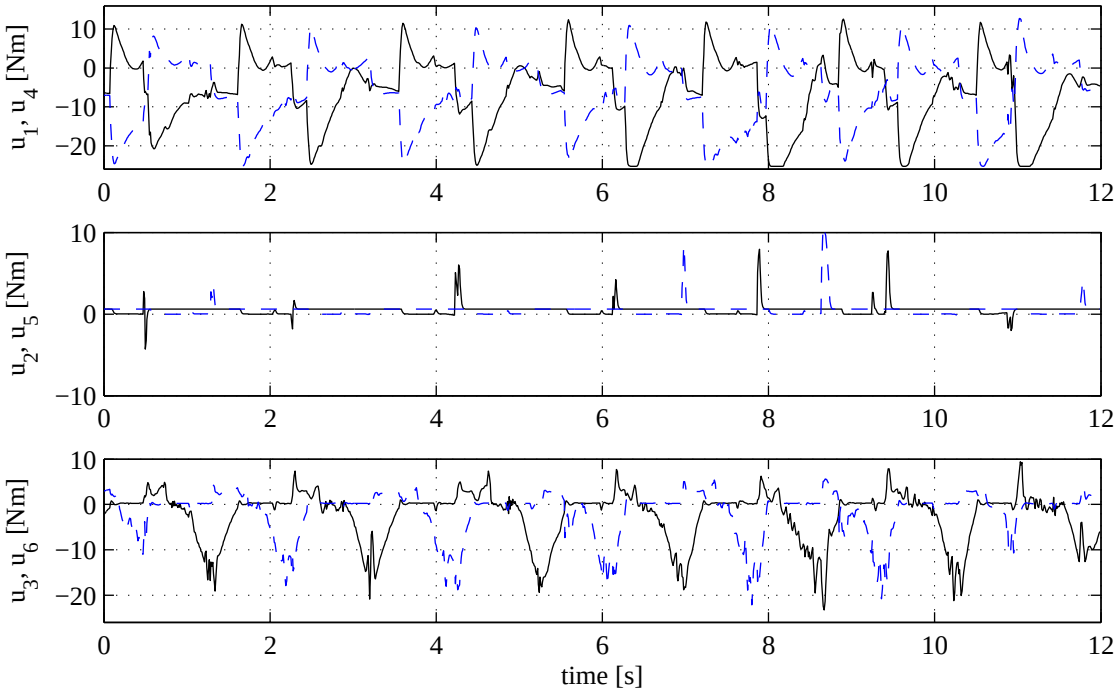

Figure 4.17 Joint torques corresponding to the frame sequence depicted on Figure 4.15.

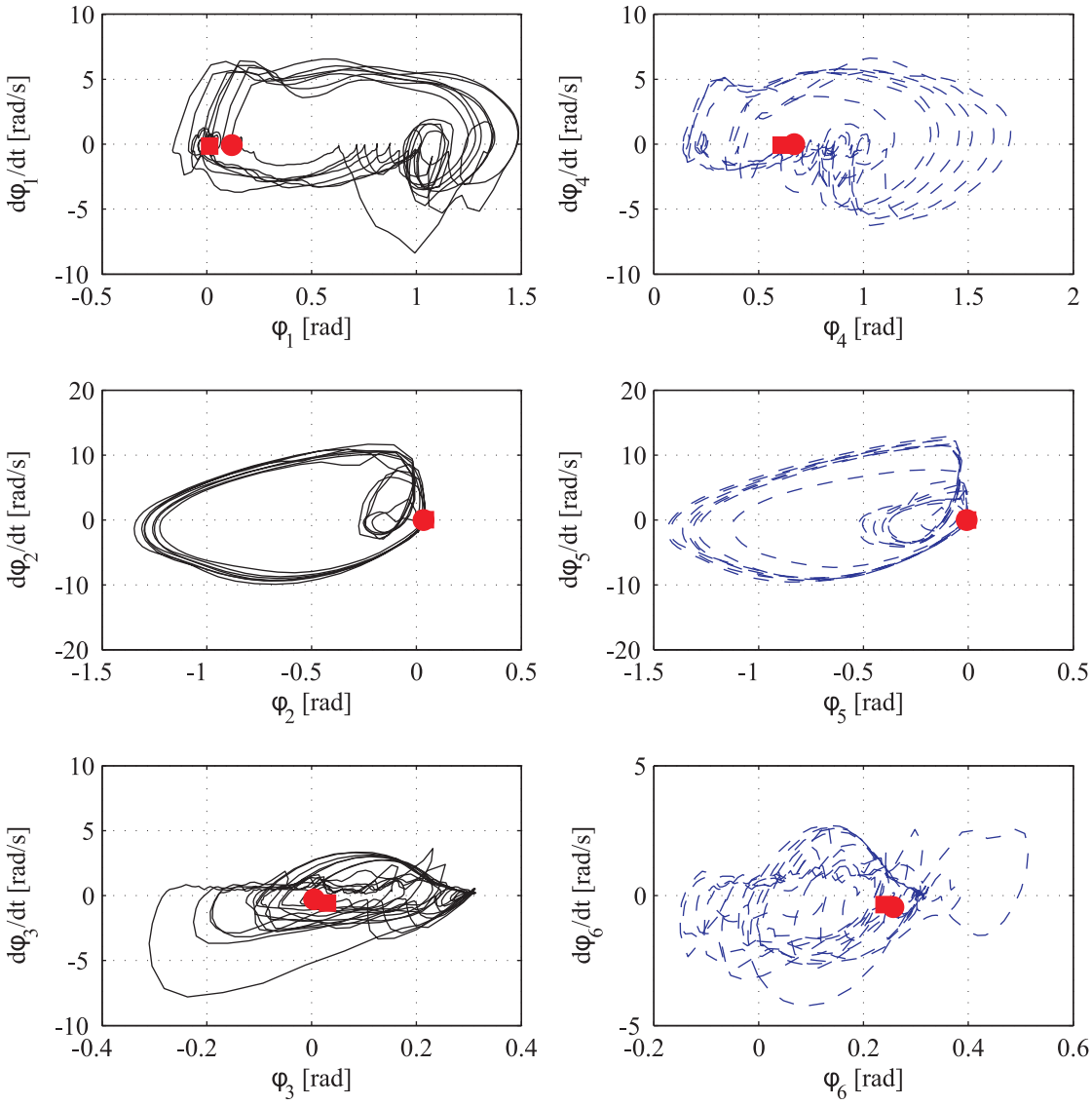

Figure 4.18 Phase plots corresponding to the hip, knee and ankle motion on the left and right legs. The data corresponds to the frame sequence depicted on Figure 4.15. The starting point of the motion is indicated with a circular dot while the end of the motion is denoted with a square mark.

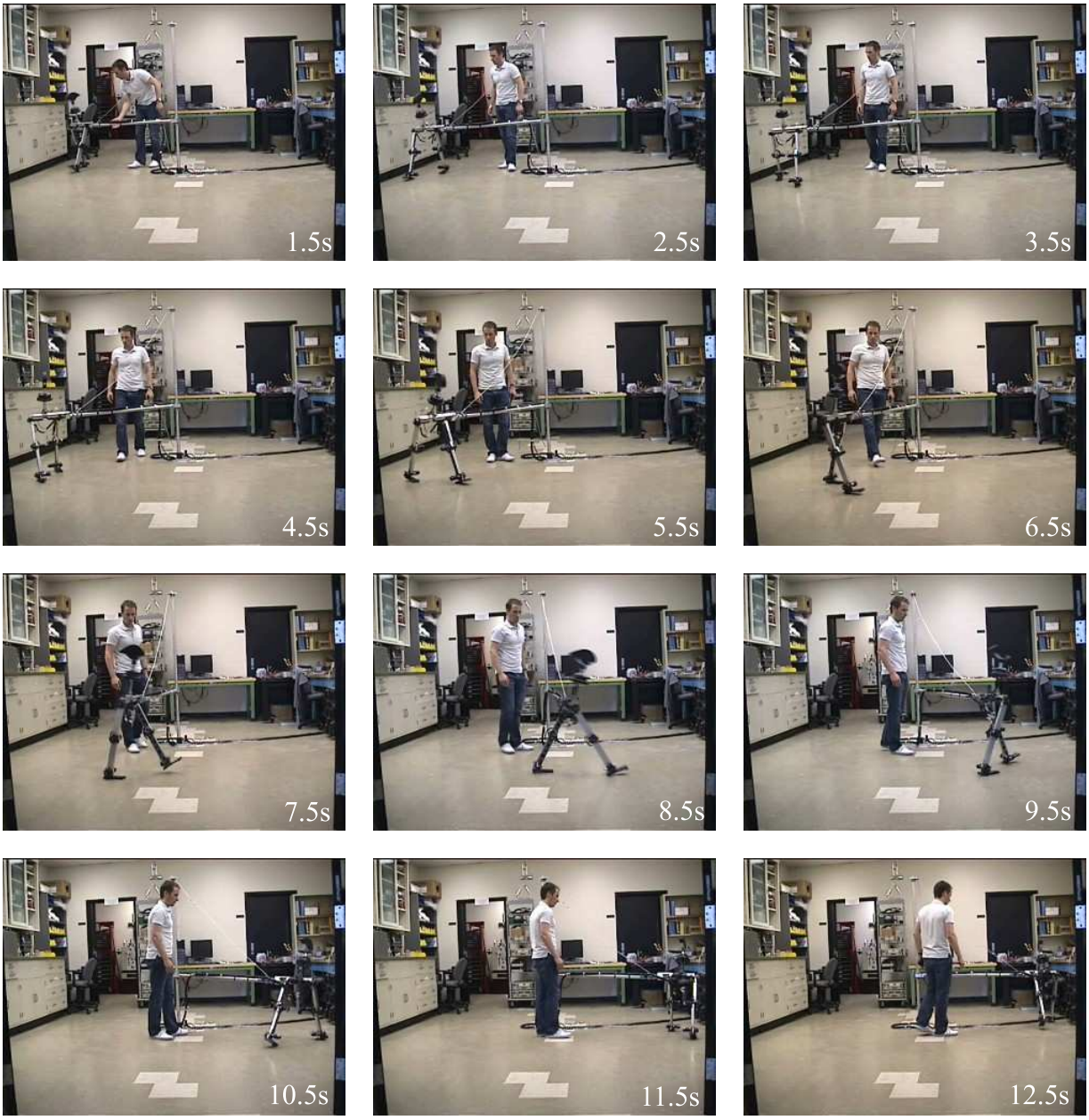

Figure 4.19 Experiment II: Frame sequence of a walking experiment. The walking is characterized with: average forward speed of $0.48 m/s$ (Froude number $Fr = 0.19$), specific mechanical cost of transport $c_{mt} = 0.32$.

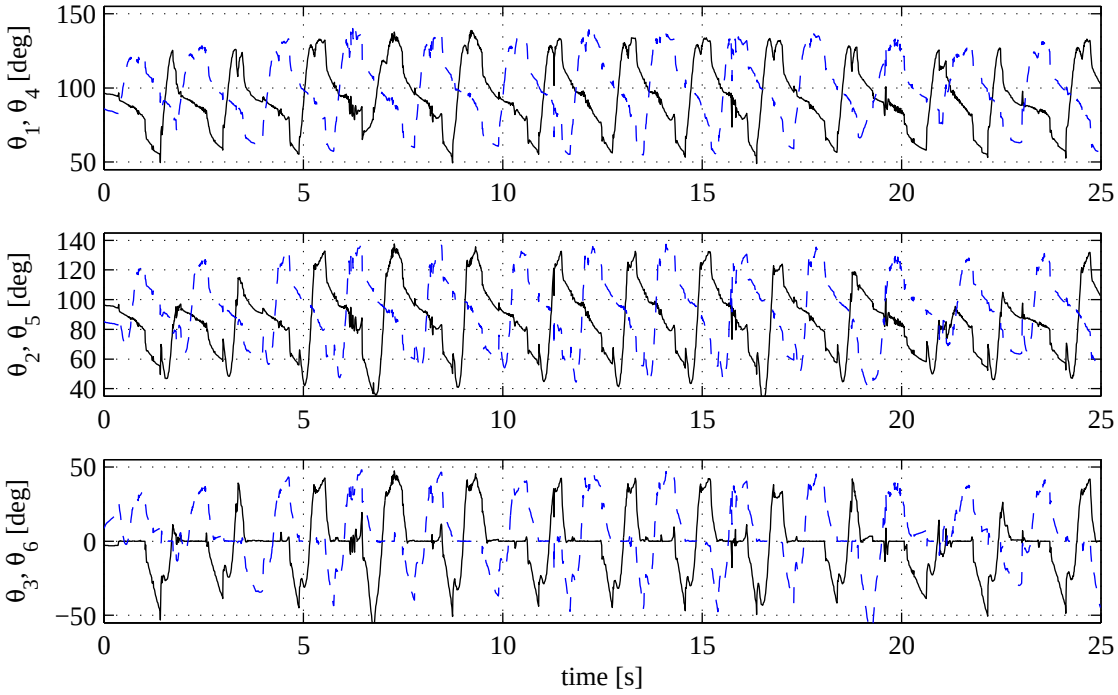

Figure 4.20 Angular motion corresponding to the frame sequence depicted on Figure 4.19.

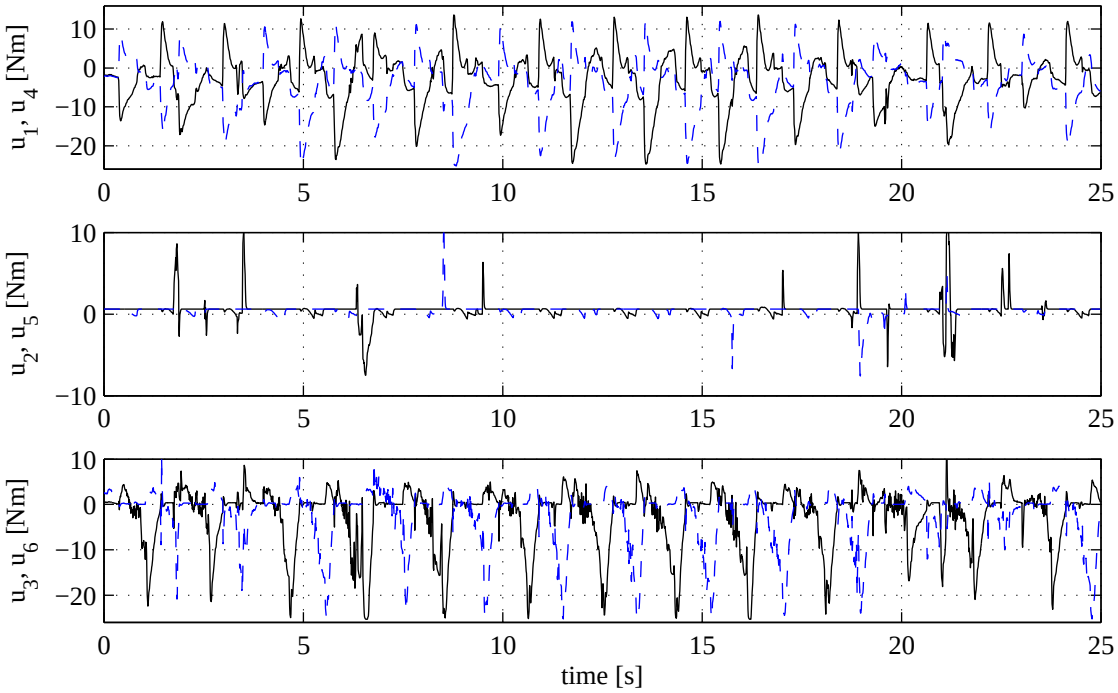

Figure 4.21 Joint torques corresponding to the frame sequence depicted on Figure 4.19.

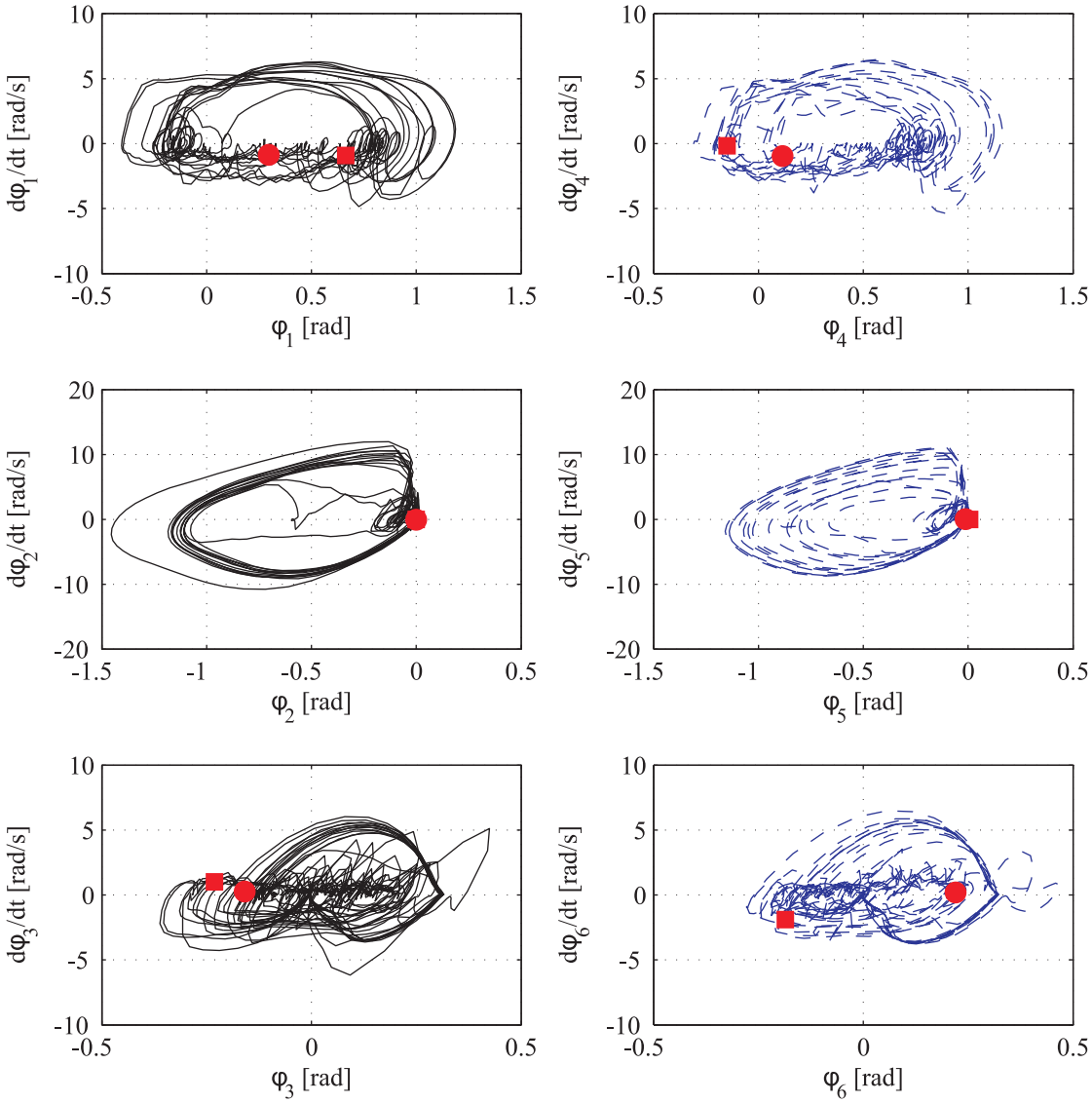

Figure 4.22 Phase plots corresponding to the hip, knee and ankle motion on the left and right legs. The data corresponds to the frame sequence depicted on Figure 4.19. The starting point of the motion is indicated with a circular dot while the end of the motion is denoted with a square mark.

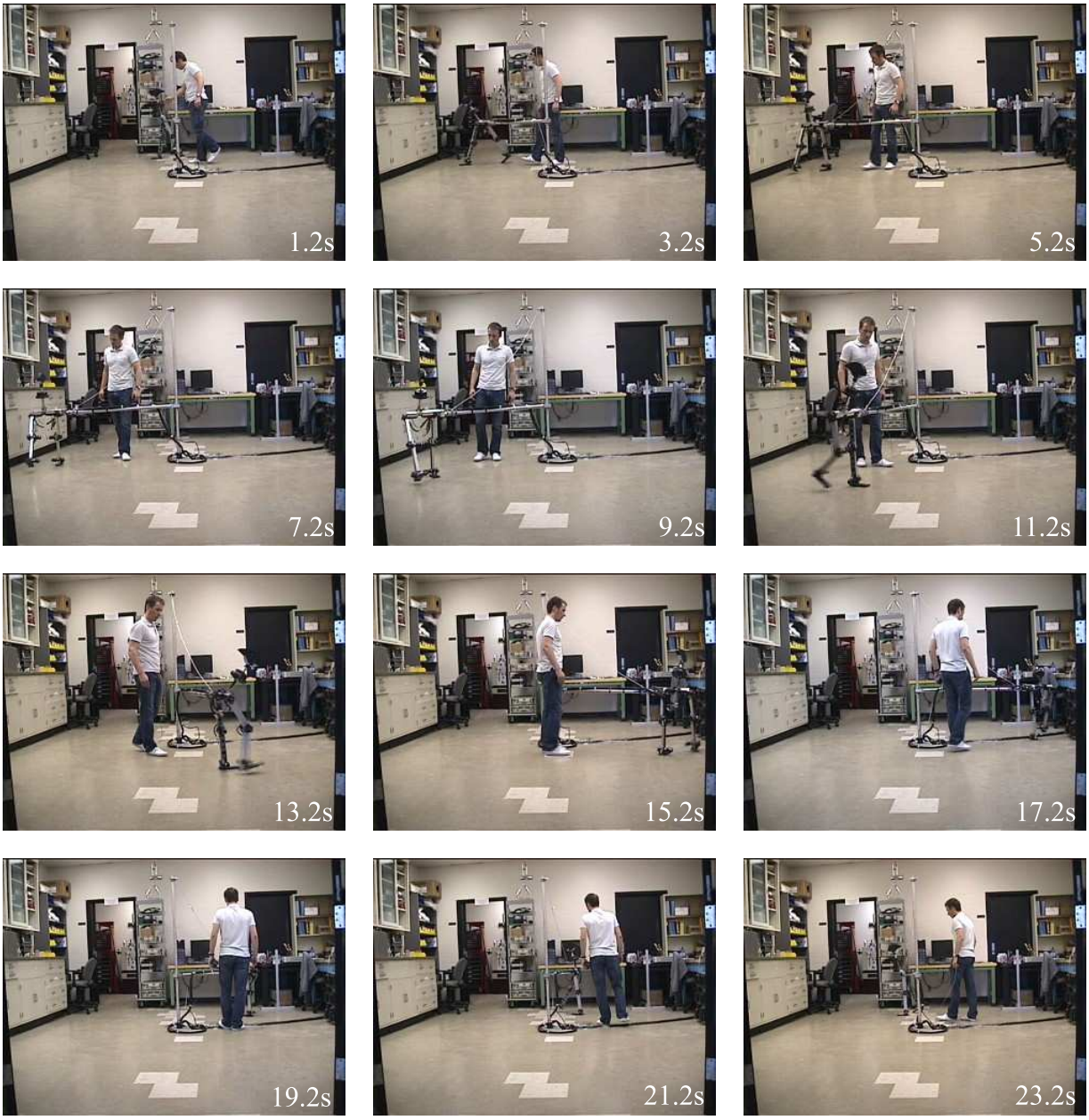

Figure 4.23 Experiment III: Frame sequence of a walking experiment. The walking is characterized with: average forward speed of $0.5m/s$ (Froude number $Fr = 0.2$), specific mechanical cost of transport $c_{mt} = 0.31$.

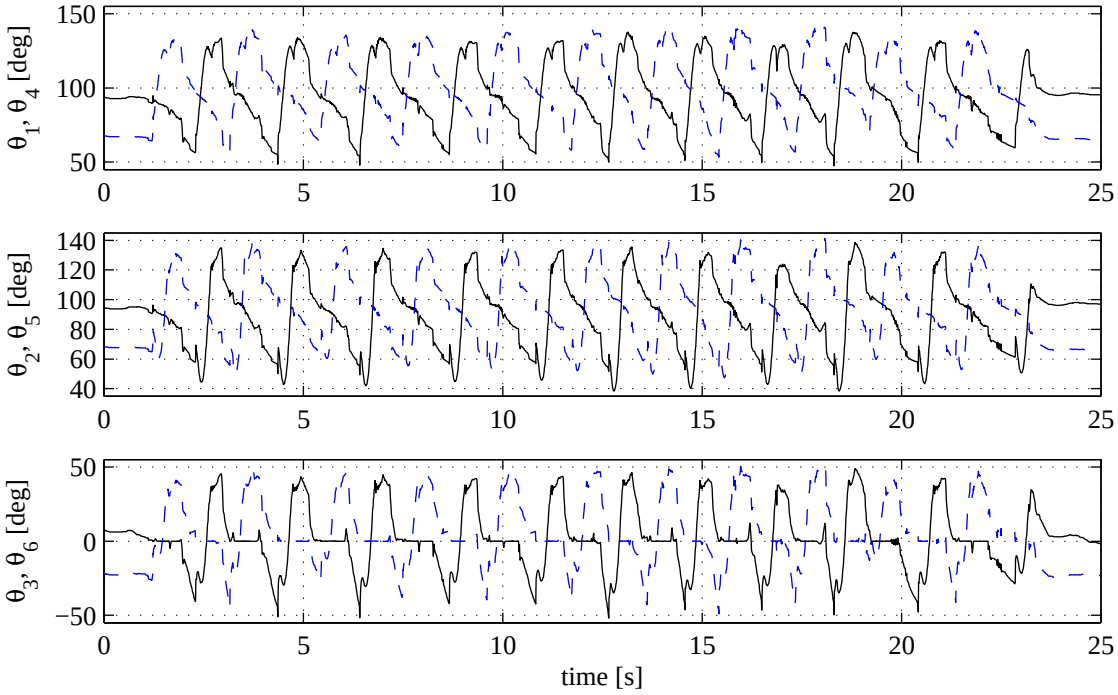

Figure 4.24 Angular motion corresponding to the frame sequence depicted on Figure 4.23.

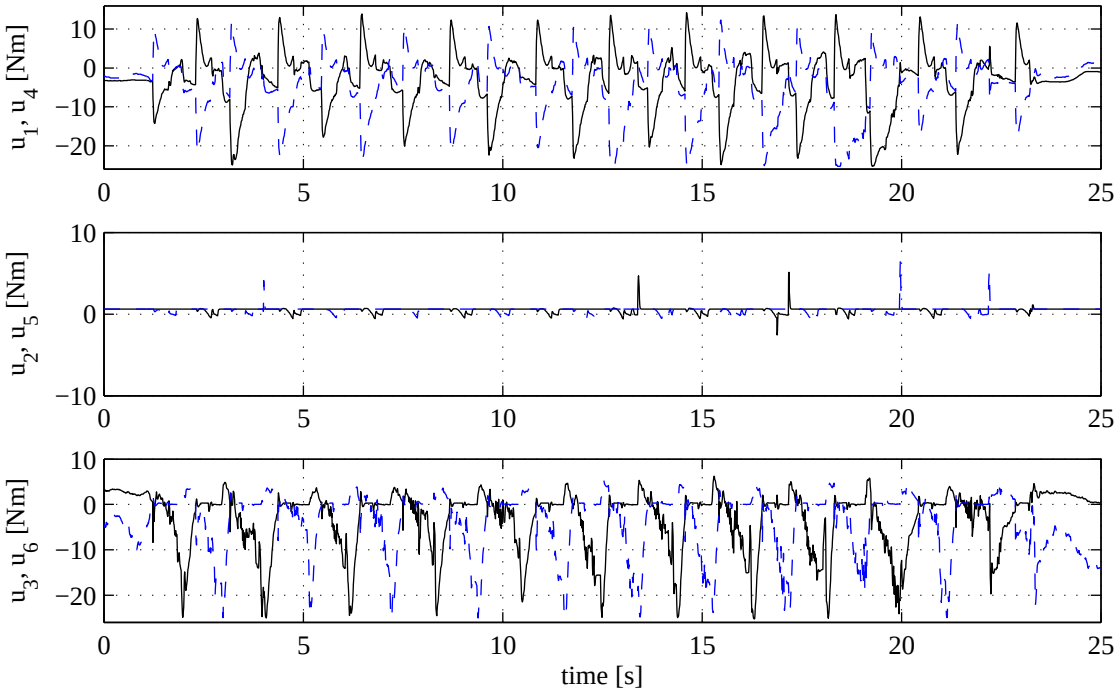

Figure 4.25 Joint torques corresponding to the frame sequence depicted on Figure 4.23.

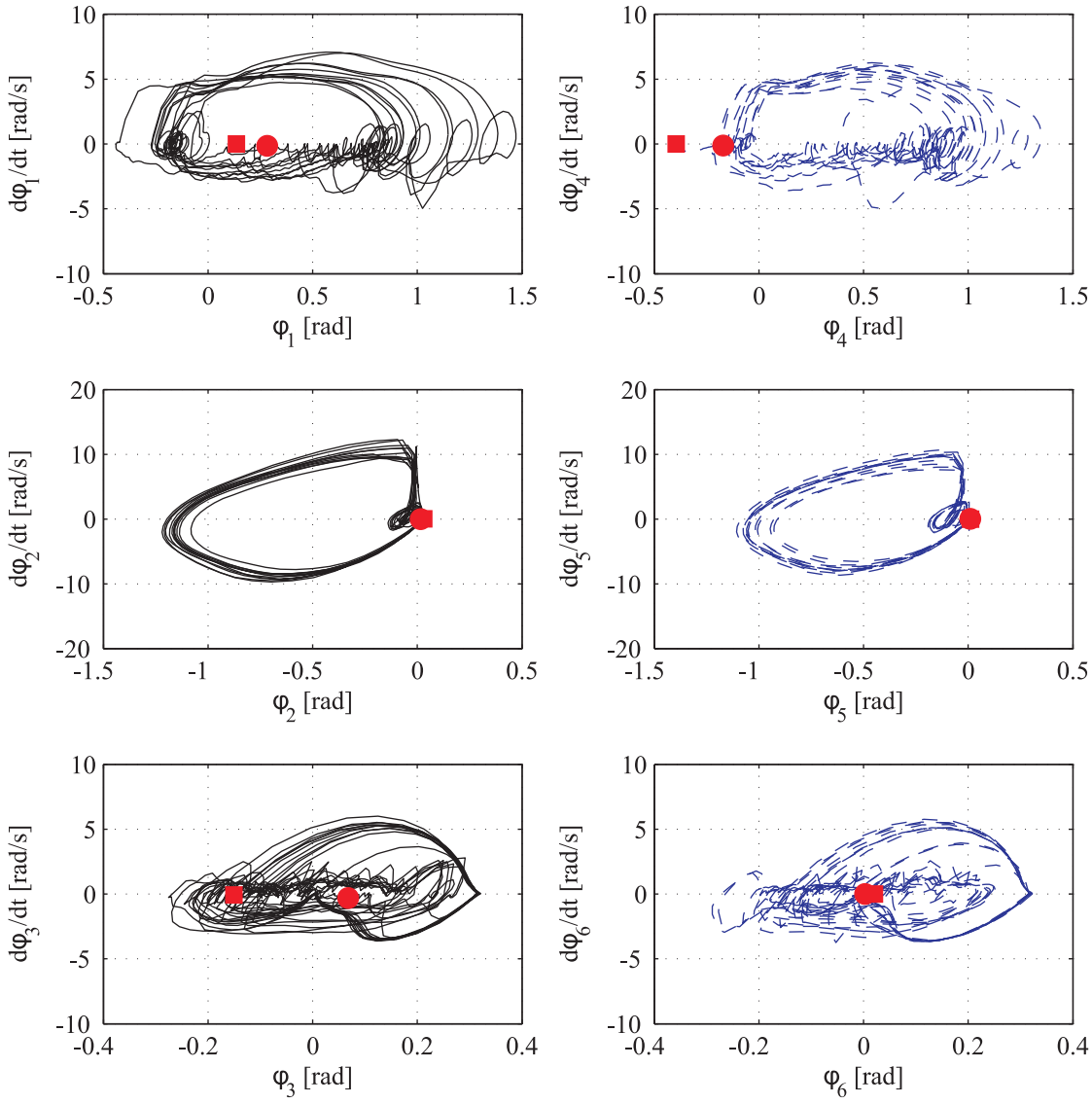

Figure 4.26 Phase plots corresponding to the hip, knee and ankle motion on the left and right legs. The data corresponds to the frame sequence depicted on Figure 4.23. The starting point of the motion is indicated with a circular dot while the end of the motion is denoted with a square mark.

CHAPTER V

CONCLUSION AND FUTURE WORK

Conclusion

This dissertation presents a dynamic modeling, analytical control development, numerical investigation and experimental realization of a human-like actuated dynamic walking in biped robots. First, a simulation approach developed for constrained dynamical system modeled with differential-algebraic equations is presented. This material provided a basis to model and simulate the biped as a constrained dynamical system. Then, a control approach for human-like actuated dynamic walking is introduced and numerically investigated. Finally, the control approach was experimentally verified on a seven-link biped robot designed (with backdrivable joints) for this purpose. The walking experiments demonstrate human-like actuated dynamic walking of a robot, as was claimed and predicted by motion simulations. Summary of the contributions is listed as follows:

1. Development of an explicit equation of motion for precise numerical simulation of constrained mechanical systems.

2. Development of a control framework which allows compliant human-like dynamic walking in biped robots.

3. Evaluation of the proposed walking controller by extensive numerical investigation. This numerical investigation addresses the ability of the controller to provide walking started from different postural configuration, walking with different speed, walking up and down slope, walking with various styles, walking under control parameter variation, walking under model parameter variations, and walking under external force disturbances.

4. Design and instrumentation of a seven-link biped robot with backdrivable joint actuators.

Beyond the description of the hardware design and the sensory system on the robot, demonstration of the passive joint motion (allowed by the backdrivable joint actuation) is specifically presented.

5. Realization of a human-like actuated dynamic walking on the seven-link robot. Experimental demonstration of a robot walking with (partially) ballistic swing leg, extended knee stance support and human like (preemptive) ankle push-off.

Future Work

Bipedal locomotion is an active research field that provides numerous possibilities for further development. Since realization of dynamic walking requires a symbiotic combination of control and design, a future research direction is natural to be discussed separated to these two categories.

Comment on the Control Approach

The control approach proposed here addresses a human-like dynamic walking on actuated biped robots. However, as presented, the approach is not restricted to walking but is general enough to be used (without any modification on its structure) to perform numerous other everyday tasks such as: standing, standing to walking, walking to standing transitions, or also sitting, sitting to standing, standing to sitting transitions for example.

In a more general view, the control approach applied to walking synthesis here, can be used to a biologically-inspired compliant coordination on robotic systems. Instead of using a desired trajectory and inverse dynamics to calculate the control forces, the presented method defines the controller on force level, which allows one to perform compliant coordination without prespecifying the motion of the system. This approach is recognized to be highly advantageous to generate a motion similar to that preferred by animals and humans while moving or locomoting efficiently.

During the development of the proposed control idea, the objective was not to use assumptions which would limit the applicability and generality of the final results. As a direct consequence, the control approach developed here is applicable to 3D walking with no structural modifications compared to its 2D (planar walking) implementation. Experimental demonstration of a 3D actuated dynamic walking however does require redesign of the current robot which investment is seen as a promising future work.

As was recognized through development, experimental realization of the proposed approach is tightly coupled with control parameter tuning. In this light, development of a machine learning algorithm for automatic parameter tuning is recognized to be a beneficial future investment.

Comment on the Robot Design

In the presented work, we have specifically pointed out that an energy efficient realization of a natural compliant motion depends on the applied control method (which should not be a high-gain trajectory tracking) and also depends on a robot design which should allow exploitation of the natural dynamics of the robot. Practically, it means that joints should be backdrivable such as human joints which allow the inertial motion of the robot to be exploited rather then being suppressed as on highly geared industrial manipulators and also on majority of actuated walking robots.

While backdrivable joints are advantageous from energetic point of view, they make the dynamics of the considered system (walking machine) coupled and as such nontrivial to control. Moreover, providing the required torque during a demanding stance phase (when the entire body needs to be supported) may not be trivial with low gear ratio (backdrivable) actuators. Development of a backdrivable but also high torque actuation unit for compliant motion/force control can be seen as a targeted future research in the present context.

A high torque actuation unit, while experimentally demonstrated not to be a requirement for level ground walking, would provide the necessary control authority under significant external

disturbances. On the present platform, experiments have verified that torque limitation prevents the robot to deal with significant disturbances, which can be addressed by further design enchancement (i.e., increasing joint torque capability).

BIBLIOGRAPHY

[1] S. Collins, A. Ruina, R. Tedrake, and M. Wisse, "Efficient bipedal robots based on passive dynamic walkers," *Science Magazine*, vol. 307, pp. 1082–1085, 2005.

[2] D. J. Braun, J. E. Mitchell, and M. Goldfarb, "Actuated dynamic walking in biped robots: Control approach, robot design and experimental validation," *The 9th IEEE-RAS International Conference on Humanoid Robots*, pp. xxx–xxx, December 7-10, 2009 Paris, France - accepted.

[3] D. A. Winter, *Biomechanics and Motor Control of Human Movement*. Wiley-Interscience, New York, 2 ed., 1990.

[4] Y. Hurmuzlu, F. Génot, and B. Brogliato, "Modeling, stability and control of biped robots - a general framework," *Automatica*, vol. 40, no. 10, pp. 1647–1664, 2004.

[5] M. Vukobratović and J. Stepanenko, "On the stability of antropometric systems," *Mathematical Bioscience*, vol. 15, no. 1, pp. 1–37, 1972.

[6] M. Vukobratović, "How to control artifical anthropomorphic systems," *IEEE Transactions on Systems, Man and Cybernetics*, vol. SMC-3, 1973.

[7] M. Vukobratović and B. Borovac, "Zero-moment point: Thirty five years of its life," *International Journal of Humanoid Robotics*, vol. 1, no. 1, pp. 157–173, 2004.

[8] A. Takanishi, M. Ishida, Y. Yamazaki, and I. Kato, "The realization of dynamic walking by the biped walking robot WL-10RD," *Proc. Intl. Conference on Advanced Robotics*, pp. 459–466, 1985.

[9] K. Hirai, M. Hirose, Y. Haikawa, and T. Takenaka, "The development of. honda humanoid robot," *Proceedings of the 1998 IEEE ICRA*, pp. 1321–1326, 1998.

[10] A. Goswami, "Postural stability of biped robots and the foot-rotation indicator (fri) point," *The International Journal of Robotics Research*, vol. 18, pp. 523–533, 1999.

[11] A. D. Kuo, "Choosing your steps carefully," *IEEE Robotics and Automation Magazine*, vol. 14, no. 2, pp. 18–29, 2007.

[12] T. McGeer, "Passive dynamic walking," *The International Journal of Robotics Research*, vol. 9, no. 2, pp. 62–82, 1990.

[13] M. J. Coleman and A. Ruina, "An uncontrolled walking toy that cannot stand still," *Physical Review Letters*, vol. 80, no. 16, pp. 3658–3661, 1998.

[14] M. Wisse, G. Feliksdal, J. van Frankenhuyzen, and B. Moyer, "Passive-based walking robot: Denis a simple efficient and lightweight biped," *IEEE Robotics and Automation Magazine*, vol. 14, no. 2, pp. 52–62, 2007.

[15] R. Kato and M. Mori, "Control method of biped locomotion giving asimptotic stability of trajectory," *Automatica*, vol. 20, no. 4, pp. 405–414, 1984.

[16] T. Mita, T. Yamaguchi, T. Kashiwase, and T. Kawase, "Realization of a high speed biped using modern control theory," *International Journal on Control*, vol. 40, no. 4, pp. 107–119, 1984.

[17] J. Furusho and A. Sano, "Sensor-based control of a nine-link robot," *International Journal of Robotics Research*, vol. 9, no. 2, pp. 83–98, 1990.

[18] J. Pratt, C. M. Chew, A. Torres, P. Dilworth, and G. Pratt, "Virtual model control: An intuitive approach for bipedal locomotion," *The International Journal of Robotics Research*, vol. 20, pp. 129–143, 2001.

[19] E. R. Westervelt, J. W. Grizzle, and D. E. Koditschek, "Hybrid zero dynamics of planar biped walkers," *IEEE Transactions on Automatic Control*, vol. 48, no. 1, pp. 42–56, 2003.

[20] C. C. de Wit, "On the concept of virtual constraints as a tool for walking robot control and balancing," *Annual Reviews in Control*, vol. 28, pp. 157–166, 2004.

[21] C. Chevallereau, G. Abba, Y. Aoustin, F. Plestan, E. Westervelt, C. de Wit, and J. Grizzle, "RABBIT: A testbed for advanced control theory," *IEEE Control Systems Magazine*, vol. 23, no. 5, pp. 57–78, 2003.

[22] C. Sabourin, O. Bruneau, and G. Buche, "Control strategy for the robust dynamic walk of a biped robot," *International Journal of Robotics Research*, vol. 25, no. 9, pp. 843–860, 2006.

[23] T. Geng, B. Porr, and F. Wörgötter, "Fast biped walking with a sensor-driven neuronal controller and real-time online learning," *International Journal of Robotics Research*, vol. 25, no. 3, pp. 243–259, 2006.

[24] P. Manoonpong, T. Geng, T. Kulvicius, B. Porr, and F. Wörgötter, "Adaptive, fast walking in a biped robot under neuronal control and learning," *PLoS Computational Biology*, vol. 3, no. 7, pp. 1305–1320, 2007.

[25] D. J. Braun and M. Goldfarb, "A control approach for actuated dynamic walking in biped robots," *IEEE Transactions on Robotics*, DOI: 10.1109/TRO.2009.2028762.

[26] D. J. Braun, J. E. Mitchell, and M. Goldfarb, "Experimental implementation of actuated dynamic walking in biped robots," *International Journal of Robotics Research*, vol. xx, no. x, pp. xxx–xxx, 2009 - submitted.

[27] D. J. Braun and M. Goldfarb, "A controller for dynamic walking in bipedal robots," *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2916–2921, October 11-15, 2009 St. Louis, USA.

[28] D. J. Braun and M. Goldfarb, "Eliminating constraint drift in the numerical simulation of constrained dynamical systems," *Computer Methods in Applied Mechanics and Engineering*, vol. 198, no. 37-40, pp. 3151–3160, 2009.

[29] J. L. Lagrange, *Mecanique Analytique*. Mme Ve Courcier, Paris, 1787.

[30] C. F. Gauss, "Über ein neues algemeines grundgesetz der mechanik," *Zeitschrift für die reine und angewandte Mathematik*, vol. 4, pp. 232–235, 1829.

[31] G. A. Maggi, *Principii della Teoria Mathematica del Movimento dei Corpi: Corso di Meccanica Razionale*. Ulrico Hoepli, Milano, 1896.

[32] J. W. Gibbs, "On the fundamental formulae of dynamics," *American Journal of Mathematics*, vol. 2, no. 1, pp. 49–64, 1879.

[33] P. Appell, "Sur une forme generale des equations de la dynamique," *C. R. Acad. Sci., Paris*, vol. 129, pp. 459–460, 1899.

[34] T. R. Kane and D. A. Levinson, *Dynamics: theory and applications*. McGraw-Hill, New York, 1985.

[35] F. E. Udwadia and R. E. Kalaba, "A new perspective on constrained motion," *Proceedings of the Royal Society of London* A, vol. 439, pp. 407–410, 1992.

[36] F. E. Udwadia and R. E. Kalaba, *Analytical Dynamics: A New Approach*. Cambridge University Press, Cambridge, England, 1996.

[37] F. E. Udwadia and R. E. Kalaba, "On the foundations of analytical dynamics," *International Journal of Non-Linear Mechanics*, vol. 37, pp. 1079–1090, 2002.

[38] L. A. Pars, *A Treatise on Analytical Dynamics*. John Wiley and Sons, New York, 1965.

[39] J. I. Neimark and N. A. Fufaev, *Dynamics of nonholonomic systems*. AMS, Providence, 1972.

[40] F. Gantmacher, *Lectures in Analytical Mechanics*. Mir, Moscow, 1975.

[41] H. Goldstein, *Classical Mechanics*. Addison-Wesley, Reading, MA, 1980.

[42] N. G. Chetaev, *Theoretical Mechanics*. Mir Publisher, Moscow, 1989.

[43] A. I. Lurie, *Analytical Mechanics*. Springer-Verlag, New York, 2002.

[44] J. Baumgarte, "Stabilization of constraints and integrals of motion in dynamical systems," *Computer Methods in Applied Mechanics and Engineering*, vol. 1, pp. 1–16, 1972.

[45] C. W. Gear, B. Leimkuhler, and G. K. Gupta, "Automatic integration of Euler-Lagrange equations with constraints," *Journal of Computational and Applied Mathematics*, vol. 12-13, pp. 77–90, 1985.

[46] P. Lötstedt and L. Petzold, "Numerical solution of nonlinear differential equations with algebraic constraints I: Convergence results for backward differentiation formulas," *Mathematics of Computation*, vol. 46, no. 174, pp. 491–516, 1986.

[47] C. Führer and B. Leimkuhler, "Numerical solution of differential-algebraic equations for constrained mechanical motion," *Numerische Mathematik*, vol. 59, pp. 55–69, 1991.

[48] L. R. Petzold, "Numerical solution of differential-algebraic equations in mechanical systems simulation," *Physica* D, vol. 60, no. 1-4, pp. 269–279, 1992.

[49] A. A. ten Dam, "Stable numerical integration of dynamical systems subject to equality state-space constraints," *Journal of Engineering Mathematics*, vol. 26, pp. 315–337, 1992.

[50] E. Eich, "Convergence results for a coordinate projection method applied to mechanical systems with algebraic constraints," *SIAM Journal on Numerical Analysis*, vol. 30, no. 5, pp. 1467–1482, 1993.

[51] E. Bayo and R. Ledesma, "Augmented lagrangian and mass-orthogonal projection methods for constrained multibody dynamics," *Nonlinear Dynamics*, vol. 9, no. 1-2, pp. 113–130, 1996.

[52] W. Blajer, "Elimination of constraint violation and accuracy aspects in numerical simulation of multibody systems," *Multibody System Dynamics*, vol. 7, pp. 265–284, 2002.

[53] F. Aghili, "A unified approach for inverse and direct dynamics of constrained multibody systems based on linear projection operator: Applications to control and simulation," *IEEE Transactions on Robotics*, vol. 21, no. 5, pp. 834–849, 2005.

[54] W. Schiehlen, "Multibody system dynamics: Roots and perspectives," *Multibody System Dynamics*, vol. 1, pp. 149–188, 1997.

[55] B. Brogliato, A. A. ten Dam, L. Paoli, F. Génot, and M. Abadie, "Numerical simulation of finite dimensional multibody nonsmooth mechanical systems," *ASME Applied Mechanics Reviews*, vol. 55, no. 2, pp. 107–150, 2002.

[56] J. d'Alembert, *Traite de Dynamique*. Paris, 1743.

[57] J. J. Moreau, "Quadratic programming in mechanics: dynamics of onesided constraints," *J. SIAM Control*, vol. 4, no. 1, pp. 153–158, 1966.

[58] P. Lötstedt, "Mechanical systems of rigid bodies subject to unilateral constraints," *SIAM Journal on Applied Mathematics*, vol. 42, no. 2, pp. 281–296, 1982.

[59] A. Ben-Israel and T. N. E. Greville, *Generalized Inverse: Theory and Applications*. Springer, 2003.

[60] C. W. Gear, "Differential-algebraic equation index transformations," *SIAM J. Sci. Statist. Comput.*, vol. 9, pp. 39–47, 1988.

[61] L. Petzold, "DASSL: A differential/algebraic system solver," *Available at http://www.netlib.org/ode/ddassl.f*, 1991.

[62] K. E. Brenan, S. L. Campbell, and L. R. Petzold, *Numerical Solutions of Initial-Value Problems in Differential-Algebraic Equations*. Elsevier Science, NY, 1989.

[63] J. G. de Jalón and E. Bayo, *Kinematic and Dynamic Simulation of Multibody Systems The Real-Time Challenge*. Springer-Verlag, New York, 1994.

[64] E. Hairer and G. Wanner, *Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems*. Springer-Verlag, Series in Computational Mathematics, 2 ed., 1996.

[65] G. Golub and C. V. Loan, *Matrix Computations*. The John Hopkins University Press, 3 ed., 1996.

[66] B. Leimkuhler, L. R. Petzold, and C. W. Gear, "Approximation methods for the consistent initialization of differential-algebraic equations," *SIAM Journal on Numerical Analysis*, vol. 28, no. 1, pp. 205–226, 1991.

[67] P. E. Nikravesh, "Initial condition correction in multibody dynamics," *Multibody System Dynamics*, vol. 18, pp. 107–115, 2008.

[68] J. G. de Jalón, J. Unda, and A. Avello, "Natural coordinates for the computer analysis of multibody systems," *Computer Methods in Applied Mechanics and Engineering*, vol. 59, pp. 309–327, 1986.

[69] C. Kraus, M. Winckler, and H. G. Bock, "Modeling mechanical DAE using natural coordinates," *Mathematical and Computer Modelling of Dynamical Systems*, vol. 7, no. 2, pp. 145–158, 2001.

[70] E. J. Haug, *Computer aided kinematics and dynamics of mechanical systems, Volume I: Basic methods*. Allyn and Bacon, Boston, 1989.

[71] R. Serban, D. Negrut, E. J. Haug, and F. A. Potra, "A topology based approach for exploiting sparsity in multibody dynamics in cartesian formulation," *Mechanics of Structures and Machines*, vol. 25, no. 3, pp. 379–396, 1997.

[72] M. Arnold, A. Fuchs, and C. Führer, "Efficient corrector iteration for DAE time integration in multibody dynamics," *Comp. Meth. Appl. Mech. Eng.*, vol. 195, no. 50-51, pp. 6958–6973, 2006.

[73] A. Laulusa and O. A. Bauchau, "Review of classical approaches for constraint enforcement in multibody systems," *Journal of Computational and Nonlinear Dynamics*, vol. 3, p. 011004, 2008.

[74] O. A. Bauchau and A. Laulusa, "Review of contemporary approaches for constraint enforcement in multibody systems," *Journal of Computational and Nonlinear Dynamics*, vol. 3, p. 011005, 2008.

[75] J. Baumgarte, "A new method of stabilization for holonomic constraints," *Journal of Applied Mechanics*, vol. 50, pp. 869–870, 1983.

[76] C. O. Chang and P. E. Nikravesh, "An adaptive constraint violation stabilization method for dynamic analysis of mechanical systems," *Journal of Mechanisms, Transmissions, and Automation in Design*, vol. 107, pp. 488–492, 1985.

[77] U. M. Ascher, H. Chin, and S. Reich, "Stabilization of DAEs and invariant manifolds," *Numerische Mathematik*, vol. 67, pp. 131–149, 1994.

[78] S. T. Lin and M. H. Hong, "Stabilization method for numerical integration of multibody mechanical systems," *Journal of Mechanical Design*, vol. 120, pp. 565–572, 1998.

[79] W. Blajer, "A geometrical interpretation and uniform matrix formulation of multibody system dynamics," *Zeitschrift für Angewandte Mathematik und Mechanik*, vol. 81, no. 4, pp. 247–259, 2001.

[80] R. A. Wehage and E. J. Haug, "Generalized coordinate partitioning for dimension reduction in analysis of constrained dynamic systems," *Journal of Mechanical Design*, vol. 104, pp. 247–255, 1982.

[81] E. Bayo and A. Avello, "Singularity free augmented lagrangian algorithms for constraint multibody dynamics," *Nonlinear Dynamics*, vol. 5, pp. 209–231, 1994.

[82] F. E. Udwadia, "A new perspective on the tracking control of nonlinear structural and mechanical systems," *Proceedings of the Royal Society of London* A, vol. 459, pp. 1783–1800, 2003.

[83] W. Blajer and K. Kolodziejczyk, "A geometric approach to solving problems of control constraints: Theory and a DAE framework," *Multibody System Dynamics*, vol. 11, pp. 343–364, 2004.

[84] M. Vukobratović, B. Borovac, D. Šurla, and D. Stokić, *Biped Locomotion*. Springer Verlag, 1990.

[85] 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*, vol. 17, no. 3, pp. 208–289, 2001.

[86] S. Kagami, T. Kitagawa, K. Nishiwaki, T. Sugihara, M. Inaba, and H. Inoue, "A fast dynamically equilibrated walking trajectory generation method of humanoid robot," *Autonomuous Robots*, vol. 12, pp. 71–82, 2002.

[87] C. Chevallereau, D. Djoudi, and J. W. Grizzle, "Stable bipedal walking with foot rotation through direct regulation of the zero moment point," *IEEE Transactions on Robotics*, vol. 24, no. 2, pp. 390–401, 2008.

[88] M. Garcia, A. Ruina, A. Chatterjee, and M. Coleman, "The simplest walking model: Stability, complexity, and scaling," *Journal of Biomechanical Engineering*, vol. 120, no. 2, pp. 281–288, 1998.

[89] S. Mochon and T. A. McMahon, "Ballistic walking: An improved model," *Mathematical Biosciences*, vol. 52, pp. 241–260, 1980.

[90] C.-L. Shih, "The dynamics and control of a biped walking robot with seven degrees of freedom," *Journal of Dynamic Systems, Measurement, and Control*, vol. 118, no. 4, pp. 683–690, 1996.

[91] F. Iida, Y. Minekawa, J. Rummel, and A. Seyfarth, "Toward a human-like biped robot with compliant legs," *Robotics and Autonomous Systems*, 2008.

[92] H. Geyer, A. Seyfarth, and R. Blickhan, "Compliant leg behaviour explains basic dynamics of walking and running," *Proceedings of the Royal Society of London* B, vol. 273, p. 28612867, 2006.

[93] J. W. Grizzle, G. Abba, and F. Plestan, "Asymptotically stable walking for biped robots: Analysis via systems with impulse effects," *IEEE Transactions on Automatic Control*, vol. 46, no. 1, pp. 51–64, 2001.

[94] 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*, vol. 19, no. 4, pp. 653–668, 2003.

[95] C. Sabourin and O. Bruneau, "Robustness of the dynamic walk of a biped robot subjected to disturbing external forces by using CMAC neural networks," *Robotics and Autonomous Systems*, vol. 51, pp. 81–99, 2005.

[96] H. Hemami and B. Wyman, "Modeling and control of constrained dynamic systems with application to biped locomotion in the frontal plane," *IEEE Transactions on Automatic Control*, vol. 24, no. 4, pp. 526–535, 1979.

[97] W. Blajer and W. Schiehlen, "Walking withouth impacts as a motion/force control problem," *Journal of Dynamic Systems, Measurement, and Control*, vol. 114, pp. 660–665, 1992.

[98] Y. Hurmuzlu and D. Marghitu, "Multi-contact collisions of kinematic chains with external surfaces," *International Journal of Robotics Research*, vol. 13, no. 1, pp. 82–92, 1994.

[99] B. Brogliato, *Nonsmooth Mechanics*. Springer Verlag London, 2 ed., 1999.

[100] F. Pfeiffer and C. Glocker, *Multibody Dynamics with Unilateral Contacts*. Wiley VCH, 1996.

[101] L. Lilov and M. Lorer, "Dynamic analysis of multirigid-body system based on the Gauss principle," *Zeitschrift für Angewandte Mathematik und Mechanik*, vol. 62, pp. 539–545, 1982.

[102] K. L. Doty, C. Melchiorri, and C. Bonivento, "A theory of generalized inverses applied to robotics," *International Journal of Robotics Research*, vol. 12, no. 1, pp. 1–19, 1993.

[103] T. Sugihara and Y. Nakamura, "Whole-body cooperative balancing of humanoid robot using cog jacobian," *IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 3, pp. 2575–2580, 2002.

[104] S. Hyon, J. G. Hale, and G. Cheng, "Full-body compliant human-humanoid interaction: Balancing in the presence of unknown external forces," *IEEE Transactions on Robotics*, vol. 23, no. 5, pp. 884–898, 2007.

[105] K. Yin, K. Loken, and M. van de Panne, "Simbicon: Simple biped locomotion control," *ACM Transactions on Graphics*, vol. 26, no. 3, pp. 105–1 – 105–10, 2007.

[106] G. Gabrielli and T. von Kármán, "What price speed?: Specific power required for propulsion of vehicles," *Mechanical Engineering*, vol. 72, no. 10, pp. 775–781, 1950.

[107] A. Takanishi, T. Takeya, H. Karaki, and I. Kato, "A control method for dynamic biped walking under unknown external force," *IEEE International Workshop on Intelligent Robots and Systems IROS '90*, vol. 2, pp. 795–801, 1990.

[108] H. Miura and I. Shimoyama, "Dynamic walk of a biped," *International Journal of Robotics Research*, vol. 3, no. 2, pp. 60–74, 1984.

[109] A. Goswami, B. Espiau, and A. Keramane, "Limit cycles in a passive compass gait biped and passivity-mimicing control laws," *Autonomuous Robotics*, vol. 4, no. 3, pp. 273–286, 1997.

[110] F. Asano, M. Yamakita, N. Kamamichi, and Z.-W. Luo, "A novel gait generation for biped walking robots based on mechanical energy constraint," *IEEE Transactions on Robotics and Automation*, vol. 20, no. 3, pp. 565 – 573, 2004.

[111] F. Asano, Z.-W. Luo, and M. Yamakita, "Biped gait generation and control based on a unified property of passive dynamic walking," *IEEE Transactions on Robotics*, vol. 21, no. 4, pp. 754 – 762, 2005.

[112] M. W. Spong and F. Bullo, "Controlled symmetries and passive walking," *IEEE Transactions on Automatic Control*, vol. 50, no. 7, pp. 1025–1031, 2005.

[113] M. W. Spong, J. K. Holm, and D. Lee, "Passive-based control of bipedal locomotion: Regulating walking by exploiting passive gaits in 2-d and 3-d bipeds," *IEEE Robotics and Automation Magazine*, vol. 14, no. 2, pp. 30–40, 2007.

[114] S. Kajita, T. Yamaura, and A. Kobayashi, "Dynamic walking control of a biped robot along a potential energy conserving orbit," *IEEE Transactions on Robotics and Automation*, vol. 8, no. 4, pp. 431–438, 1992.

[115] V. T. Inman, H. J. Ralston, and F. Todd, *Human Walking*. Baltimore, Williams and Wilkins, 1981.

[116] L. Löffler, M. Gienger, and F. Pfeiffer, "Sensors and control concept of walking "Johnnie"," *International Journal of Robotics Research*, vol. 22, pp. 229–239, 2003.

[117] R. M. Alexander and A. S. Jayes, "A dynamic similarity hypothesis for the gaits of quadrupedal mammals," *Journal of Zoology*, vol. 201, pp. 135–152, 1983.

[118] C. L. Vaughan and M. J. OMalley, "Froude and the contribution of naval architecture to our understanding of bipedal locomotion," *Gait and Posture*, vol. 21, no. 3, pp. 135–152, 2005.

[119] D. G. E. Hobbelen and M. Wisse, "Ankle actuation for limit cycle walkers," *International Journal of Robotics Research*, vol. 27, no. 6, pp. 709–735, 2008.