Katja Mombaur
Karsten Berns (Eds.)

》COSMOS 18

Modeling, Simulation and Optimization of Bipedal Walking

Springer

Cognitive Systems Monographs

Series Editors

Rüdiger Dillmann
Institute of Anthropomatics, Humanoids and Intelligence Systems Laboratories,
Faculty of Informatics, University of Karlsruhe, Kaiserstr. 12, 76131 Karlsruhe, Germany

Yoshihiko Nakamura
Dept. Mechano-Informatics, Fac. Engineering, Tokyo University, 7-3-1 Hongo, Bukyo-ku Tokyo,
113-8656, Japan

Stefan Schaal
Computational Learning & Motor Control Lab., Department Computer Science,
University of Southern California, Los Angeles, CA 90089-2905, USA

David Vernon
Department of Robotics, Brain, and Cognitive Sciences, Via Morego, 30 16163 Genoa, Italy

Advisory Board

Prof. Dr. Heinrich H. Bülthoff
MPI for Biological Cybernetics, Tübingen, Germany

Prof. Masayuki Inaba
The University of Tokyo, Japan

Prof. J.A. Scott Kelso
Florida Atlantic University, Boca Raton, FL, USA

Prof. Oussama Khatib
Stanford University, CA, USA

Prof. Yasuo Kuniyoshi
The University of Tokyo, Japan

Prof. Hiroshi G. Okuno
Kyoto University, Japan

Prof. Helge Ritter
University of Bielefeld, Germany

Prof. Giulio Sandini
University of Genova, Italy

Prof. Bruno Siciliano
University of Naples, Italy

Prof. Mark Steedman
University of Edinburgh, Scotland

Prof. Atsuo Takanishi
Waseda University, Tokyo, Japan

Katja Mombaur and Karsten Berns (Eds.)

Modeling, Simulation and Optimization of Bipedal Walking

Springer

Editors

Prof. Dr. Katja Mombaur
Universität Heidelberg
Interdisziplinäres Zentrum für
Wissenschaftliches Rechnen
Optimierung in Robotik & Biomechanik
Heidelberg
Germany

Prof. Dr. Karsten Berns
Technische Universität Kaiserslautern
Fachbereich Informatik
Arbeitsgruppe Robotersysteme
Kaiserslautern
Germany

Preface

Walking and running on two legs are extremely challenging tasks. Even though most humans learn to walk without any difficulties within the first year(s) of their life, the motion generation and control mechanisms of dynamic bipedal walking are far from being understood. This becomes obvious in situations where walking motions have to be generated from scratch or have to be restored, e.g.

- in robotics, when teaching and controlling humanoids or other bipedal robots to walk in a dynamically stable way,
- in computer graphics and virtual reality, when generating realistic walking motions for different avatars in various terrains, reacting to virtual perturbations, or
- during rehabilitation in orthopedics or other medical fields, when aiming to restore walking capabilities of patients after accidents, neurological diseases, etc. by prostheses, orthoses, functional electrical stimulation or surgery.

The study of walking motions is a truly multidisciplinary research topic. The book gives an overview of *Modeling, Simulation and Optimization of Bipedal Walking* based on contributions by authors from such different fields as Robotics, Biomechanics, Computer Graphics, Sports, Engineering Mechanics and Applied Mathematics. Methods as well as various applications are presented.

The goal of this book is to emphasize the importance of mathematical modeling, simulation and optimization, i.e. classical tools of Scientific Computing, for the study of walking motions. Model-based simulation and optimization complements experimental studies of human walking motions in biomechanics or medical applications and gives additional insights. In robotics, this approach allows to pre-test robot motions in the computer and helps to save hardware costs. Of course no model is ever perfect, and therefore no simulation and optimization result is a 100% prediction of reality, but if properly done the will result in good approximations and excellent starting points for practical experiments. The topic of Model-based Optimization for Robotics is also promoted in a newly founded technical committee of the IEEE Robotics and Automation Society.

This book goes back to a workshop with the same title organized by us at the IEEE Humanoids Conference in Paris in December 2009. The workshop consisted of 16 oral presentations and ten poster presentations. Later, all authors were invited to submit articles about their work. The papers went through a careful peer-review process aimed at improving the quality of the papers. In total, 22 papers are included in this book, representing the whole variety of research in modeling, simulation and optimization of bipedal walking.

Topics covered in this book include:

- Modeling techniques for anthropomorphic bipedal walking systems
- Optimized walking motions for different objective functions
- Identification of objective functions from measurements
- Simulation and optimization approaches for humanoid robots
- Biologically inspired control algorithms for bipedal walking
- Generation and deformation of natural walking in computer graphics
- Imitation of human motions on humanoids
- Emotional body language during walking
- Simulation of biologically inspired actuators for bipedal walking machines
- Modeling and simulation techniques for the development of prostheses
- Functional electrical stimulation of walking.

We hope that you will find the articles in this book as interesting and stimulating as we do!

Heidelberg and Kaiserslautern, Germany   Katja Mombaur
December 2012   Karsten Berns

Table of Contents

Trajectory-Based Dynamic Programming

Christopher G. Atkeson and Chenggang Liu

Abstract. We informally review our approach to using trajectory optimization to accelerate dynamic programming. Dynamic programming provides a way to design globally optimal control laws for nonlinear systems. However, the curse of dimensionality, the exponential dependence of memory and computation resources needed on the dimensionality of the state and control, limits the application of dynamic programming in practice. We explore trajectory-based dynamic programming, which combines many local optimizations to accelerate the global optimization of dynamic programming. We are able to solve problems with less resources than grid-based approaches, and to solve problems we couldn't solve before using tabular or global function approximation approaches.

1 What Is Dynamic Programming?

Dynamic programming provides a way to find globally optimal control laws (policies), $\mathbf{u} = \mathbf{u}(\mathbf{x})$, which give the appropriate action $\mathbf{u}$ for any state $\mathbf{x}$ [1, 2]. Dynamic programming takes as input a one step cost (a.k.a. "reward" or "loss") function and the dynamics of the problem to be optimized. This paper focuses on offline planning of nonlinear control laws for control problems with continuous states and actions, deterministic time invariant discrete time dynamics $\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k)$, and a time invariant one step cost function $L(\mathbf{x}, \mathbf{u})$, so we use discrete time dynamic programming. We are focusing on steady state policies and thus an infinite time horizon. Action vectors are typically limited to a finite volume set.

Christopher G. Atkeson
Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, USA
e-mail: cga@cmu.edu

Chenggang Liu
Department of Automation, Shanghai Jiao Tong University, Shanghai, China
e-mail: cgliu2008@gmail.com

One approach to dynamic programming is to approximate the value function $V(\mathbf{x})$ (the optimal total future cost from each state $V(\mathbf{x}) = \min_{\mathbf{u}_k} \sum_{k=0}^{\infty} L(\mathbf{x}_k, \mathbf{u}_k))$, by repeatedly solving the Bellman equation $V(\mathbf{x}) = \min_{\mathbf{u}}(L(\mathbf{x}, \mathbf{u}) + V(\mathbf{f}(\mathbf{x}, \mathbf{u})))$ at sampled states $\mathbf{x}_j$ until the value function estimates have converged. Typically the value function and control law are represented on a regular grid. Some type of interpolation is used to approximate these functions within each grid cell. If each dimension of the state and action is represented with a resolution R, and the dimensionality of the state is d_x and that of the action is d_u, the computational cost of the conventional approach is proportional to $R^{d_x} \times R^{d_u}$ and the memory cost is proportional to R^{d_x}. This exponential dependence of cost on dimensionality is known as the Curse of Dimensionality [1].

An example problem: We use one link pendulum swingup as an example problem to provide the reader with a visualizable example of a nonlinear control law and corresponding value function. In one link pendulum swingup a motor at the base of the pendulum swings a rigid arm from the downward stable equilibrium to the upright unstable equilibrium and balances the arm there (Fig. 1). What makes this challenging is that a one step cost function penalizes the amount of torque used and the deviation of the current angle from the goal. The controller must try to minimize the total cost of the trajectory. The one step cost function for this example is a weighted sum of the squared angle errors (θ: difference between current angle and the goal angle) and the squared torques τ: $L(\mathbf{x}, \mathbf{u}) = 0.1\theta^2 + \tau^2$ where 0.1 weights the angle error relative to the torque penalty. There are no costs associated with the joint velocity. The uniform density link has a mass m of 1kg, length l of 1m, and width of 0.1m. The dynamics are given by:

$$\ddot{\theta} = \frac{(\tau + 0.5m \cdot g \cdot l \cdot \sin(\theta))}{\mathbf{I}} \tag{1}$$

where g is the gravitational constant 9.81 and $\mathbf{I}$ is the moment of inertia about the hinge. The continuous time dynamics are discretized with a time step of 0.01s using Euler's method as discrete time dynamics are more convenient for system identification and computer-based discrete time control. Because the dynamics and cost function are time invariant, there is a steady state control law and value function (Fig. 2). Because we keep track of the direction of the error and multiple rotations around the hinge, there is a unique optimal trajectory. In general there may be multiple solutions with equal optimal costs. Dynamic programming converges to one of the globally optimal solutions.
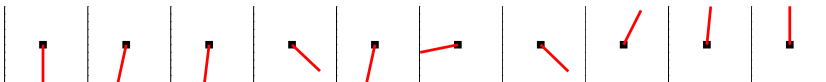
Fig. 1 Configurations from the simulated one link pendulum swingup optimal trajectory every half second and at the end of the trajectory. The pendulum starts in the downward position (left) and swings up in rightward configurations.
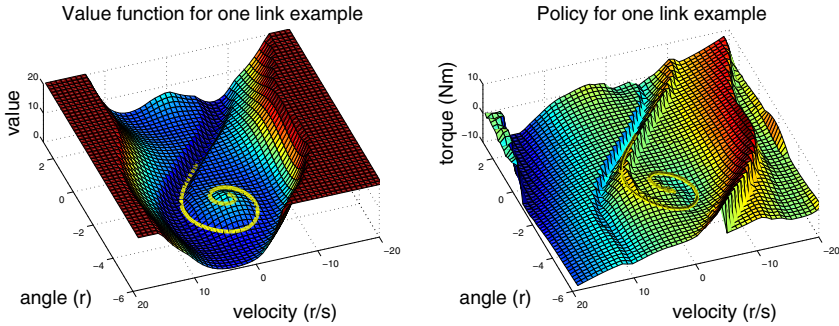
Fig. 2 The value function and policy for a one link pendulum swingup. The optimal trajectory is shown as a line in the value function and policy plots. The value function is cut off above 20 so we can see the details of the part of the value function that determines the optimal trajectory. The goal is the state (0,0), upright and not moving.

Representing trajectories explicitly to achieve representational sparseness:
A technique to accelerate dynamic programming is to optimize more than one step at a time. Larson proposed modifying the Bellman equation to allow multiple time steps and multiple evaluations of the one step cost and dynamics before evaluating the value function on the right hand side [3]:

$$V(\mathbf{x}_0) = \min_{\mathbf{u}_{0,N-1}} ((\sum_0^{N-1} L(\mathbf{x}_i, \mathbf{u}_i)) + V(\mathbf{x}_N)) \qquad (2)$$

In a grid-based approximation with multilinear interpolation, $V(\mathbf{x})$ depends on the value estimates at all the surrounding nodes. Larson's goal was to ensure that $V(\mathbf{x}_N)$ on the right hand side of the Bellman equation did not depend on the value being updated ($V(\mathbf{x}_0)$) by ensuring that the trajectory ended far enough away from its start in his State Increment Dynamic Programming. We have extended this idea by running trajectories a variety of distances including all the way to the goal. To help show that representing trajectories explicitly allows greater sparseness in dynamic programming, we show its effect on the one link swingup task. Fig. 3-top-left shows Larson's State Increment Dynamic Programming procedure on a 10x10 grid applied to this problem. In Larson's approach trajectories are run until they exit a 2x2 volume and the start value has no effect on the end value when multi-linear interpolation is used on the grid of values. Fig. 3-top-right shows a set of optimized trajectories that run all the way to the goal from a similar grid. The flow from state to state is clearly indicated. When the resolution is greatly reduced, the State Increment Dynamic Programming approach fails (Fig. 3-bottom-left), while the full trajectory-based approach is more robust to the sparse representation (Fig. 3-bottom-right) and still generates globally optimal trajectories. This work raises the question: "What should the length of the trajectory be?" Larson used a distance threshold. We used reaching the goal (attaining a point with zero future costs) as a threshold. A time
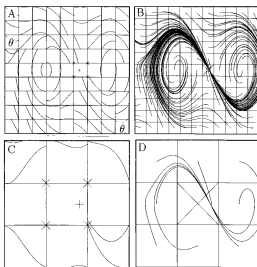
Fig. 3 Right: Different approaches to computing and representing the value function for one link swingup. On the left is the State Increment Dynamic Programming Approach of Larson. On the right trajectories are run all the way to the goal. The plots are of phase space with angles on the x axis and angular velocities on the y axis.

threshold could also be used. What distance or time threshold value should be used? Should it be the same throughout the space? Another question is how to efficiently optimize the sequence of actions in Eq. 2. We use local trajectory optimization to find an optimal sequence of actions.

2   Trajectory-Based Dynamic Programming

Our approach modifies (and complements) existing approximate dynamic programming approaches in a number of ways: 1) We approximate the value function and policy using many local models (quadratic for the value function, linear for the policy) as shown in Fig. 4. These local models, located at sampled states, help our function approximators handle sparsely sampled states. A nearest neighbor approach is taken to determine which local model should be used to predict the value and policy for a particular state. 2) We use trajectory segments rather than single time steps to perform Bellman updates (black lines in Fig. 4-Right). 3) After using either the approximated policy or value function to initialize the trajectory segment, we use trajectory optimization to directly optimize the sequence of actions $\mathbf{u}_{0,N-1}$ and the corresponding states $\mathbf{x}_{1,N}$. 4) Local models of the value function and policy are created as a byproduct of our trajectory optimization process. 5) Local models exchange information to ensure the Bellman equation is satisfied everywhere and the value function and policy are globally optimal. 6) We also use trajectory optimization on each query to refine the predicted values and actions. 7) We are exploring using adaptive grids. Fig. 4-Right shows a randomly generated set of states superimposed on a contour plot of the value function for one link swingup, and the optimized trajectories used to generate locally quadratic value function models.

Local models of the value function and policy: We need to represent value functions and policies sparsely. We use a hybrid tabular and parametric approach: parametric local models of the value function and policy are represented at sampled locations. This representation is similar to using many Taylor series approximations
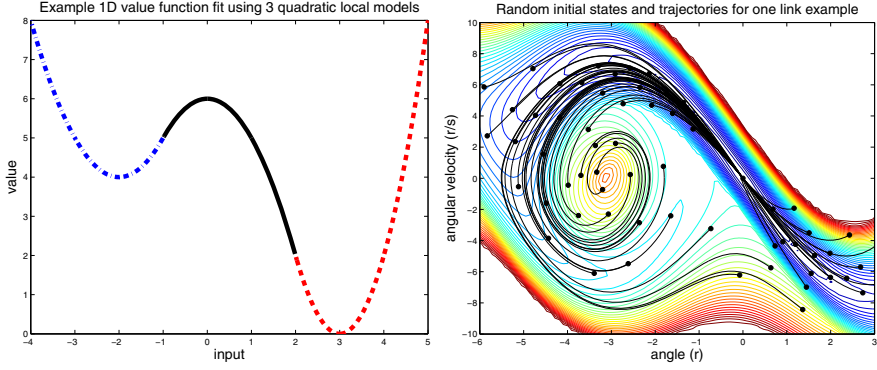
Fig. 4 Left: Example of a local approximation of a 1D value function using three quadratic models. **Right:** Random states (dots) used to plan one link swingup, superimposed on a contour map of the value function. Optimized trajectories (black lines) are shown starting from the random states.

of a function at different points. At each sampled state $\mathbf{x}^p$ the local quadratic model for the value function is:

$$V^p(\mathbf{x}) = V_0^p + \mathbf{V}_x^p \hat{\mathbf{x}} + \frac{1}{2}\hat{\mathbf{x}}^{\mathrm{T}} \mathbf{V}_{xx}^p \hat{\mathbf{x}} \tag{3}$$

where $\hat{\mathbf{x}} = \mathbf{x} - \mathbf{x}^p$ is the vector from the sampled state $\mathbf{x}^p$ to the query $\mathbf{x}$, V_0^p is the constant term, $\mathbf{V}_x^p$ is the first derivative with respect to state at $\mathbf{x}^p$, and $\mathbf{V}_{xx}^p$ is the second spatial derivative at $\mathbf{x}^p$. The local linear model for the policy is:

$$\mathbf{u}^p(\mathbf{x}) = \mathbf{u}_0^p - \mathbf{K}^p \hat{\mathbf{x}} \tag{4}$$

where $\mathbf{u}_0^p$ is the constant term, and $\mathbf{K}^p$ is the first derivative of the local policy with respect to state at $\mathbf{x}^p$ and also the gain matrix for a local linear controller. V_0, $\mathbf{V}_x$, $\mathbf{V}_{xx}$, and $\mathbf{K}$ are stored with each sampled state.

Creating the local models: These local models are created using Differential Dynamic Programming (DDP) [4, 5, 6, 7]. This local trajectory optimization process is similar to linear quadratic regulator design in that a value function and policy is produced. In DDP, value function and policy models are produced at each point along a trajectory. Suppose at a time step i we have 1) a local second order Taylor series approximation of the optimal value function: $V^i(\mathbf{x}) = V_0^i + \mathbf{V}_x^i \hat{\mathbf{x}} + \frac{1}{2}\hat{\mathbf{x}}^{\mathrm{T}} \mathbf{V}_{xx}^i \hat{\mathbf{x}}$ where $\hat{\mathbf{x}} = \mathbf{x} - \mathbf{x}^i$. 2) a local second order Taylor series approximation of the robot dynamics ($\mathbf{f}_x^i$ and $\mathbf{f}_u^i$ correspond to $\mathbf{A}$ and $\mathbf{B}$ of the linear plant model used in linear quadratic regulator (LQR) design): $\mathbf{f}^i(\mathbf{x}, \mathbf{u}) = \mathbf{f}_0^i + \mathbf{f}_x^i \hat{\mathbf{x}} + \mathbf{f}_u^i \hat{\mathbf{u}} + \frac{1}{2}\hat{\mathbf{x}}^{\mathrm{T}} \mathbf{f}_{xx}^i \hat{\mathbf{x}} + \hat{\mathbf{x}}^{\mathrm{T}} \mathbf{f}_{xu}^i \hat{\mathbf{u}} + \frac{1}{2}\hat{\mathbf{u}}^{\mathrm{T}} \mathbf{f}_{uu}^i \hat{\mathbf{u}}$ where $\hat{\mathbf{u}} = \mathbf{u} - \mathbf{u}^i$, and 3) a local second order Taylor series approximation of the one step cost, which is often known analytically for human specified criteria ($\mathbf{L}_{xx}$ and $\mathbf{L}_{uu}$ correspond to $\mathbf{Q}$ and $\mathbf{R}$ of LQR design): $L^i(\mathbf{x}, \mathbf{u}) = L_0^i + \mathbf{L}_x^i \hat{\mathbf{x}} + \mathbf{L}_u^i \hat{\mathbf{u}} + \frac{1}{2}\hat{\mathbf{x}}^{\mathrm{T}} \mathbf{L}_{xx}^i \hat{\mathbf{x}} + \hat{\mathbf{x}}^{\mathrm{T}} \mathbf{L}_{xu}^i \hat{\mathbf{u}} + \frac{1}{2}\hat{\mathbf{u}}^{\mathrm{T}} \mathbf{L}_{uu}^i \hat{\mathbf{u}}$

Given a trajectory, one can integrate the value function and its first and second spatial derivatives backwards in time to compute an improved value function and policy. We utilize the "Q function" notation [35] from reinforcement learning: $Q(\mathbf{x}, \mathbf{u}) = L(\mathbf{x}, \mathbf{u}) + V(\mathbf{f}(\mathbf{x}, \mathbf{u}))$. The backward sweep takes the following form (in discrete time):

$$\mathbf{Q}_x^i = \mathbf{L}_x^i + \mathbf{V}_x^i \mathbf{f}_x^i; \quad \mathbf{Q}_u^i = \mathbf{L}_u^i + \mathbf{V}_x^i \mathbf{f}_u^i \tag{5}$$

$$\mathbf{Q}_{xx}^i = \mathbf{L}_{xx}^i + \mathbf{V}_x^i \mathbf{f}_{xx}^i + (\mathbf{f}_x^i)^{\mathrm{T}} \mathbf{V}_{xx}^i \mathbf{f}_x^i \tag{6}$$

$$\mathbf{Q}_{ux}^i = \mathbf{L}_{ux}^i + \mathbf{V}_x^i \mathbf{f}_{ux}^i + (\mathbf{f}_u^i)^{\mathrm{T}} \mathbf{V}_{xx}^i \mathbf{f}_x^i \tag{7}$$

$$\mathbf{Q}_{uu}^i = \mathbf{L}_{uu}^i + \mathbf{V}_x^i \mathbf{f}_{uu}^i + (\mathbf{f}_u^i)^{\mathrm{T}} \mathbf{V}_{xx}^i \mathbf{f}_u^i \tag{8}$$

$$\Delta \mathbf{u}^i = (\mathbf{Q}_{uu}^i)^{-1} \mathbf{Q}_u^i; \quad \mathbf{K}^i = (\mathbf{Q}_{uu}^i)^{-1} \mathbf{Q}_{ux}^i \tag{9}$$

$$\mathbf{V}_x^{i-1} = \mathbf{Q}_x^i - \mathbf{Q}_u^i \mathbf{K}^i; \quad \mathbf{V}_{xx}^{i-1} = \mathbf{Q}_{xx}^i - \mathbf{Q}_{xu}^i \mathbf{K}^i \tag{10}$$

where subscripts indicate derivatives and superscripts indicate the trajectory index. After the backward sweep, forward integration can be used to update the trajectory itself: $\mathbf{u}_{new}^i = \mathbf{u}^i - \Delta \mathbf{u}^i - \mathbf{K}^i(\mathbf{x}_{new}^i - \mathbf{x}^i)$. We note that the cost of this approach grows at most cubically rather than exponentially with respect to the dimensionality of the state. We formulate the trajectory optimization with an infinite time horizon so that the value functions and control laws are time invariant and functions only of state.

Combining greedy local optimizers to perform global optimization: As currently described, the algorithm finds a locally optimal policy, but not necessarily a globally optimal policy. However, if the combination of local value function models generate a global value function that satisfies the Bellman equation everywhere, the resulting policy and value function are globally optimal [1, 2]. We will refer to violations of the Bellman equation as "Bellman errors". We can reduce one step Bellman errors

$$e = V(\mathbf{x}) - \min_{\mathbf{u}}(L(\mathbf{x}, \mathbf{u}) + V(\mathbf{f}(\mathbf{x}, \mathbf{u}))) \tag{11}$$

and multi-step Bellman errors

$$e = V(\mathbf{x}_0) - \min_{\mathbf{u}_{0,N-1}} ((\sum_0^{N-1} L(\mathbf{x}_i, \mathbf{u}_i)) + V(\mathbf{x}_N)) \tag{12}$$

by 1) re-optimizing local models that disagree using policies from neighboring local models, and 2) adding additional local models in the area of the discrepancies until Bellman errors are reduced below a threshold everywhere (up to a sampling resolution). This process does require globally optimizing the one step action $\mathbf{u}$ or multi-step action sequence $\mathbf{u}_{0,N-1}$ for each test. The Bellman error approach becomes similar to a standard dynamic programming approach as the resolution becomes infinite, and thus inherits the convergence properties of grid-based dynamic programming [1, 2]. A weaker test which verifies that the value function matches the current policy assesses the Bellman error for $\mathbf{u}(\mathbf{x})$ at each selected state, so no global minimization is necessary. This test is useful in policy iteration.

A useful heuristic to detect local optima that does not require a global optimization on each test is to enforce continuity of the value function and the policy. This heuristic often works because a switch from a global optimum to a local optimum in a policy often shows up as a discontinuity in the policy or value function. Unfortunately, often optimal policies and value functions have true discontinuities. As Fig. 2 shows, value functions can have derivative discontinuities (discontinuities of the spatial derivatives of the value, see the creases in the figure) at policy discontinuities. In addition, value functions can have discontinuities of the value itself in complex situations such as when there are multiple goals (zero velocity states that require no cost to maintain) and it is not possible to reach all goals from each state. A second heuristic is that optimal trajectories should not normally cross any policy or value function discontinuities given smooth dynamics and one step cost functions. However, there are exceptions to this heuristic as well.

Discrepancies between predictions of local value functions can also be used to guide computational effort and allocate local models. Discrepancies of local policies can be considered by using the local policies to generate trajectory segments, and seeing if the cost of the trajectory is accurately predicted by local value function models. We can enforce continuity of local models by 1) using the policy of one state of a pair to reoptimize the trajectory of the other state of the pair and vice versa, and 2) adding more local models in between nearest neighbors that continue to disagree until the discontinuity is confirmed or eliminated [6]. We also periodically reoptimize each local model using the policies of other local models. As more neighboring policies are considered in optimizing any given local model, a wide range of actions are considered for each state. There are several ways to perform reoptimization. Each local model could use the policy of a nearest neighbor, or a randomly chosen neighbor with the distribution being distance dependent, or just choosing another local model randomly with no consideration of distance. [6] describes how to follow a policy of another sampled state if its trajectory is stored, or can be recomputed as needed. We have also explored a different approach that does not require each sampled state to save its trajectory or recompute it. To "follow" the policy of another state, we follow the locally linear policy for that state until the trajectory begins to go away from the state. At that point we switch to following the globally approximated policy. Since we apply this reoptimization process periodically with different randomly selected local models, over time we explore using a wide range of actions from each state. This process is analogously to exploration in learning and to the global minimization with respect to actions found in the Bellman equation. This approach is similar to using the method of characteristics to solve partial differential equations [8] and finding value functions for games [9, 10, 11]. We note that value functions that are discontinuous in known locations, with known patterns, or in a relatively small area can also be handled with approaches that partition the space into regions with no discontinuities.

Adaptive grids — constant value contours: We have explored a number of adaptive grid techniques for trajectory-based dynamic programming. Adaptive grid techniques for solving partial differential equations are useful for dynamic programming as well [12]. Fig. 5 shows a trajectory-based approach being used to compute a
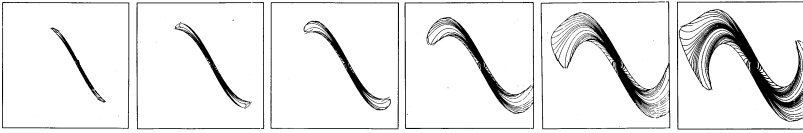
Fig. 5 Computing a 1D swingup value function using an adaptive grid. The plots are of phase space with angles on the x axis and angular velocities on the y axis.
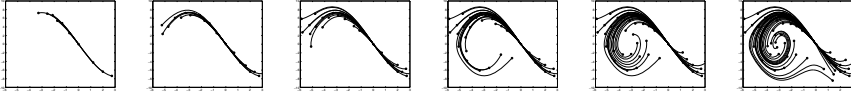
Fig. 6 Randomly sampled states and trajectories for the one link swingup problem after 10, 20, 30, 40, 50, and 60 states are stored. These figures correspond to Figs. 4:right and 5, with angle on the x axis and angular velocity on the y axis.

global value function [6, 7]. An adaptive grid of initial conditions are maintained on a "frontier" of constant value $V(\mathbf{x})$ or cost-to-go. This "frontier" is one dimension less than the dimensionality of $\mathbf{x}$. Trajectories are optimized from each sample of the frontier and local models are maintained at each sample. The value function at each frontier sample is compared with that of nearby points, using the local models for the value functions and policies. At discrepancies the trajectories are re-optimized using the value function from the neighboring frontier point. If this fails to resolve the discrepancy, new frontier points are added at the discrepancy until the discrepancy is below a threshold. Fig. 5 shows the frontier being gradually expanded. Since each trajectory optimization is independent, these approaches are "embarrassingly" parallel.

Adaptive grids — randomly sampling states: Fig. 6 shows an adaptive grid approach based on randomly sampling states, similar to Fig. 5. In this case states are randomly sampled. If the predicted value V (using the nearest local model) for a state is too high, it is rejected. If the predicted value is too similar to the cost of an optimized trajectory, it is rejected. Otherwise it is added to the database of sampled states, with its local value function and policy models. To generate the initial trajectory for optimization the current approximated policy is used until the goal or a time limit is reached. In the current implementation this involves finding the sampled state nearest to the current state in the trajectory and using its locally linear policy to compute the action on each time step. The trajectory is then locally optimized.

We solve a series of problems by gradually increasing the cost of trajectories we consider. Each cost threshold generates a volume we consider, and in the most conservative version of our algorithms, we completely solve each volume before increasing the cost threshold. More aggressive versions only partially solve each volume before increasing the cost threshold, and continue to update lower cost nodes throughout execution.
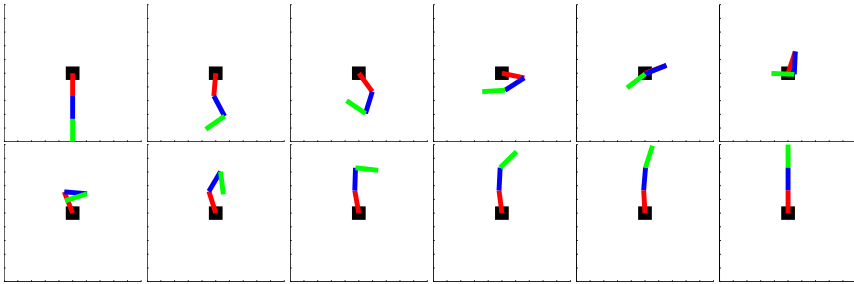
Fig. 7 Configurations from the simulated three link pendulum optimal swingup trajectory every tenth of a second and at the end of the trajectory

We expect the locally optimal policies to be fairly good because we 1) gradually increase the solved volume (Fig. 6) and 2) use local optimizers. Given local optimization of actions, gradually increasing the solved volume defined by a constant value contour will result in a globally optimal policy if the boundary of this volume never touches a non-adjacent section of itself, given reasonable dynamics and one step cost functions. Fig. 2 and 4 show the creases in the value function (discontinuities in the spatial derivative) and corresponding discontinuities in the policy that typically result when the constant value contour touches a non-adjacent section of itself as the limit on acceptable values is increased.

3   Results

In addition to the one link swingup example presented in the introduction, we present results on two link swingup (4 dimensional state), three link swingup (6 dimensional state), four link balance (8 dimensional state), and 5 link bipedal walking (10 dimensional state). In the first four cases we used a random adaptive grid approach [13]. For the one link swingup case, the random state approach found a globally optimal trajectory (the same trajectory found by our grid based approaches [14]) after adding only 63 random states. Fig. 4 shows the distribution of states and their trajectories superimposed on a contour map of the value function for one link swingup and Fig. 6 shows how the solved volume represented by the sampled states grows. For the two link swingup case, the random state approach finds what we believe is a globally optimal trajectory (the same trajectory found by our tabular approaches [14]) after storing an average of 12000 random states, compared to 100 million states needed by a tabular approach. For the three link swingup case, the random state approach found a good trajectory after storing less than 22000 random states (Fig. 7). We were not able to solve this problem using regular grid-based approaches with a 4 gigabyte table.
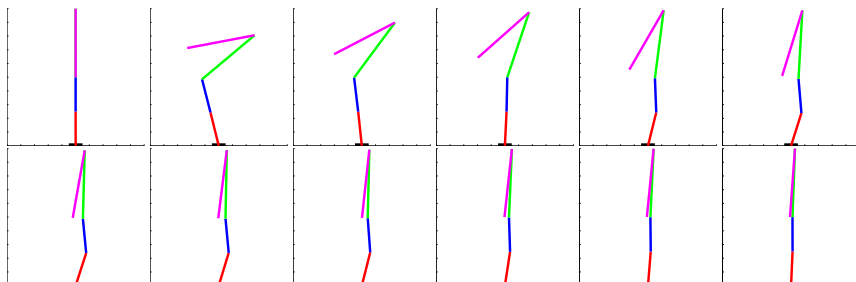
Fig. 8 Configurations every quarter second from a simulated response to a forward push (to the right) of 22.5 Newton-seconds. The lower black rectangle indicates the extent of the symmetric foot.

A simple model of standing balance: We provide results on a standing robot balancer that is pushed (Fig. 8), to demonstrate that we can apply the approach to systems with eight dimensional states. This problem is hard because the ankle torque is quite limited to prevent the foot from tilting and the robot falling. We created a four link model that included a knee, shoulder, and arm. Each link is modeled as a thin rod. We model perturbations as horizontal impulses applied to the middle of the torso. The perturbations instantaneously change the joint velocities from zero to values appropriate for the perturbation. We assume no slipping or other change of contact state during the perturbation. Both the allowable states and possible torques are limited. The one step optimization criterion is a combination of quadratic penalties on the deviations of the joint angles from their desired positions (straight up with the arm hanging down), the joint velocities, and the joint torques:
$L(\mathbf{x},\mathbf{u}) = (\theta_a^2 + \theta_k^2 + \theta_h^2 + \theta_s^2) + (\dot{\theta}_a^2 + \dot{\theta}_k^2 + \dot{\theta}_h^2 + \dot{\theta}_s^2) + 0.002(\tau_a^2 + \tau_k^2 + \tau_h^2 + \tau_s^2)$
where 0.002 weights the torque penalty relative to the position and velocity errors. The penalty on joint velocities reduces knee and shoulder oscillations. After dynamic programming based on approximately 60,000 sampled states, Fig. 8 shows the response to the largest perturbations that could be handled in the forward direction. We have designed a linear quadratic regulator (LQR) controller that optimizes the same criterion on the four link model, using a linearized dynamic model. For perturbations of 17.5 Newton-seconds and higher, the LQR controller falls down, while the controller presented here is able to handle larger perturbations of 22.5 Newton-seconds. We were able to generate behavior using optimization that matched human responses for large perturbations [15, 16]. Interestingly, we found that a single optimization criterion generated multiple strategies (both an ankle and hip strategy, for example).

We explored trajectory-based control of bipedal walking. We simulated a 5 link planar robot (2 legs and a torso). We optimized a periodic steady state trajectory (solid line) and 12 additional optimal trajectory segments starting just after -4 and 10 Newton-seconds perturbations at the hip at different times (Figure 9-left). The trajectory library was evaluated using perturbations of -10, -6, 6, 16, and 20 Newton-seconds at the hip (Figure 9-right). The robot successfully recovered from these
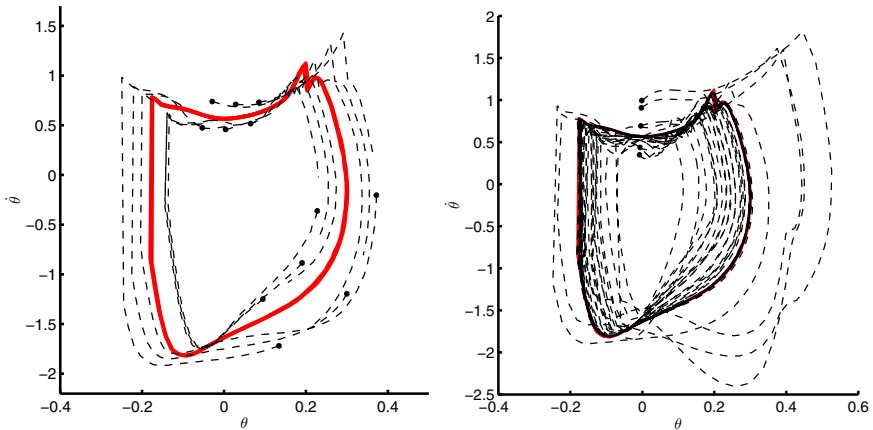
Fig. 9 Trajectory-based dynamic programming applied to bipedal walking. On the left we show the entries in a trajectory library, and on the right we show trajectories generated from the trajectory library in response to perturbations. The solid curve is the periodic steady state trajectory. 2D phase portraits are shown which are projections of the actual 10D trajectories. We plot the angle (x axis) and angular velocity (y axis) of a line from the hip to a foot.

perturbations. The simulated robot could also walk up and down 5 degree inclines using this trajectory-based policy generated by optimizing walking on level ground.

4 Related Work

Trajectories: In our approach we use trajectories to provide a more accurate estimate of the value of a state. In reinforcement learning "rollout" or simulated trajectories are often used to provide training data for approximating value functions [17, 18], as well as evaluating expectations in stochastic dynamic programming. Murray et. al. used trajectories to provide estimates of values of a set of initial states [19]. A number of efforts have been made to use collections of trajectories to represent policies [3, 6, 7, 20, 21, 22, 23, 24, 25, 26, 27]. [21] created sets of locally optimized trajectories to handle changes to the system dynamics. NTG uses trajectory optimization based on trajectory libraries for nonlinear control [28]. [6] and [7] used information transfer between stored trajectories to form sets of globally optimized trajectories for control.

Local models: We use local models of the value function and policy. Werbos proposed using local quadratic models of the value function [29]. The use of trajectories and a second order gradient-based trajectory optimization procedure such as Differential Dynamic Programming (DDP) allows us to use Taylor series-like local models of the value function and policy [4, 5]. Similar trajectory optimization approaches could have been used [30], including robust trajectory optimization

approaches [31, 32, 33]. An alternative to local value function and policy models are global parametric models, for example [17, 34, 35]. A difficult problem is choosing a set of basis functions or features for a global representation. Usually this has to be done by hand. An advantage of local models is that the choice of basis functions or features is not as important.

5 Discussion

On what problems will our approach work well? We believe our approach can discover underlying simplicity in many typical problems. An example of a problem that appears complex but is actually simple is a problem with linear dynamics and a quadratic one step cost function. Dynamic programming can be done for such linear quadratic regulator (LQR) problems even with hundreds of dimensions and it is not necessary to build a grid of states [36]. The cost of representing the value function is quadratic in the dimensionality of the state. The cost of performing a "sweep" or update of the value function is at most cubic in the state dimensionality. Continuous states and actions are easy to handle. Perhaps many problems, such as the examples in this paper, have local simplifying characteristics similar to LQR problems. For example, problems that are only "slightly" nonlinear and have a locally quadratic cost function may be solvable with quite sparse representations. One goal of our work is to develop methods that do not immediately build a hugely expensive representation if it is not necessary, and attempt to harness simple and inexpensive parallel local planning to solve complex planning problems. Another goal of our work is to develop methods that can take advantage of situations where only a small amount of global interaction is necessary to enable local planners capable of solving local problems to find globally optimal solutions.

Why dynamic programming? To generate a control law or policy, trajectory optimization can be applied to many initial conditions, and the resulting actions can be interpolated as needed. If trajectory optimization is fast enough it can be done online, as in Receding Horizon Control/Model Predictive Control (RHC/MPC). Why do we need to deal with dynamic programming and the curse of dimensionality? Dynamic programming is a global optimizer, while trajectory optimization alone finds local optima. Often, the local optima found using just trajectory optimization are not acceptable.

What about state estimation, learning models, and robust policies? We assume we know the dynamics and one step cost function, and have accurate state estimates. Future work will address simultaneously learning a dynamic model, finding a robust policy, and performing state estimation with an erroneous partially learned model [37, 38, 39].

Aren't there better trajectory optimization methods than DDP? DDP, invented in the 1960s, is useful because it produces local models of value functions and policies. It may be the case that newer methods can optimize trajectories faster than

DDP, and that we can use a combination of methods to achieve our goals. Parametric trajectory optimization based on sequential quadratic programming (SQP) dominates work in aerospace and animation. We have used SQP methods to initially optimize trajectories, and a final pass of DDP to produce local models of value functions and policies.

6 Future Work

Future work will optimize aspects and variants of this approach and do a thorough comparison with alternative approaches. More extensive experimentation will lead to a clearer understanding of when this approach works well, and how much storage and computation costs are reduced in general. An interesting but difficult research question is how sacrificing global optimality would enable finding useful solutions to bigger problems. Another interesting question is how to combine Receding Horizon Control/Model Predictive Control with a pre-computed value function [40, 41].

From our point of view, the most important question is whether model-based optimal control of this form can be usefully applied to humanoid robots, where the dynamics and thus the model depend on a poorly characterized environment as well as a well characterized robot.

7 Conclusion

We have combined local models and local trajectory optimization to create a promising approach to practical dynamic programming for robot control problems. New elements in our work relative to other trajectory library approaches include variable-length trajectories including trajectories all the way to a goal, using local models of the value function and policy, and maintaining consistency across local models of the value function. We are able to solve problems with less resources than grid-based approaches, and to solve problems we couldn't solve before using tabular or global function approximation approaches.

References

1. Bellman, R.: Dynamic Programming (1957); reprinted by Dover 2003
2. Bertsekas, D.P.: Dynamic Programming and Optimal Control. Athena Scientific (1995)
3. Larson, R.L.: State Increment Dynamic Programming. Elsevier, New York (1968)

4. Dyer, P., McReynolds, S.R.: The Computation and Theory of Optimal Control. Academic Press, New York (1970)
5. Jacobson, D.H., Mayne, D.Q.: Differential Dynamic Programming. Elsevier, New York (1970)
6. Atkeson, C.G.: Using local trajectory optimizers to speed up global optimization in dynamic programming. In: Cowan, J.D., Tesauro, G., Alspector, J. (eds.) Advances in Neural Information Processing Systems, vol. 6, pp. 663–670. Morgan Kaufmann Publishers, Inc. (1994)
7. Atkeson, C.G., Morimoto, J.: Non-parametric representation of a policies and value functions: A trajectory-based approach. In: Advances in Neural Information Processing Systems, vol. 15. MIT Press (2003)
8. Abbott, M.B.: An Introduction to the Method of Characteristics. Thames & Hudson (1966)
9. Isaacs, R.: Differential Games. Dover (1965)
10. Lewin, J.: Differential Games. Spinger (1994)
11. Breitner, M.: Robust optimal on-board reentry guidance of a European space shuttle: Dynamic game approach and guidance synthesis with neural networks. In: Reithmeier, E. (ed.) Complex Dynamical Processes with Incomplete Information. Birkhauser, Basel (1999)
12. Munos, R.: Munos home, http://www.researchers.lille.inria.fr/~munos/ (2006)
13. Atkeson, C.G., Stephens, B.: Random sampling of states in dynamic programming. IEEE Transactions on Systems, Man, and Cybernetics, Part B 38(4), 924–929 (2008)
14. Atkeson, C.G.: Randomly sampling actions in dynamic programming. In: IEEE International Symposium on Approximate Dynamic Programming and Reinforcement Learning, ADPRL (2007)
15. Atkeson, C.G., Stephens, B.: Multiple balance strategies from one optimization criterion. In: IEEE-RAS International Conference on Humanoid Robots, Humanoids (2007)
16. Stephens, B.: Integral control of humanoid balance. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS (2007)
17. Boyan, J.A., Moore, A.W.: Generalization in reinforcement learning: Safely approximating the value function. In: Tesauro, G., Touretzky, D.S., Leen, T.K. (eds.) Advances in Neural Information Processing Systems, vol. 7, pp. 369–376. The MIT Press, Cambridge (1995)
18. Tsitsiklis, J.N., Van Roy, B.: Regression methods for pricing complex American-style options. IEEE-NN 12, 694–703 (2001)
19. Murray, J.J., Cox, C., Lendaris, G.G., Saeks, R.: Adaptive dynamic programming. IEEE Transactions on Systems, Man. and Cybernetics, Part C: Applications and Reviews 32(2), 140–153 (2002)
20. Grossman, R.L., Valsamis, D., Qin, X.: Persistent stores and hybrid systems. In: Proceedings of the 32nd Conference on Decision and Control, pp. 2298–2302 (1993)
21. Schierman, J.D., Ward, D.G., Hull, J.R., Gandhi, N., Oppenheimer, M.W., Doman, D.B.: Integrated adaptive guidance and control for re-entry vehicles with flight test results. Journal of Guidance, Control, and Dynamics 27(6), 975–988 (2004)
22. Frazzoli, E., Dahleh, M.A., Feron, E.: Maneuver-based motion planning for nonlinear systems with symmetries. IEEE Transactions on Robotics 21(6), 1077–1091 (2005)
23. Ramamoorthy, S., Kuipers, B.J.: Qualitative hybrid control of dynamic bipedal walking. In: Proceedings of the Robotics: Science and Systems Conference, pp. 89–96. MIT Press, Cambridge (2006)

24. Stolle, M., Tappeiner, H., Chestnutt, J., Atkeson, C.G.: Transfer of policies based on trajectory libraries. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS (2007)
25. Safonova, A., Hodgins, J.K.: Construction and optimal search of interpolated motion graphs. In: SIGGRAPH (2007)
26. Tedrake, R.: LQR-Trees: Feedback motion planning on sparse randomized trees. In: Proceedings of Robotics: Science and Systems (RSS), p. 8 (2009)
27. Reist, P., Tedrake, R.: Simulation-based LQR-trees with input and state constraints. In: IEEE International Conference on Robotics and Automation, ICRA (2010)
28. Milam, M., Mushambi, K., Murray, R.: NTG - a library for real-time trajectory generation (2002), http://www.cds.caltech.edu/murray/software/2002antg.html
29. Werbos, P.: Personal communication (2007)
30. Todorov, E., Tassa, Y.: Iterative local dynamic programming. In: 2nd IEEE International Symposium on Approximate Dynamic Programming and Reinforcement Learning (AD-PRL), pp. 90–95 (2009)
31. Altamimi, A., Abu-Khalaf, M., Lewis, F.L.: Adaptive critic designs for discrete-time zero-sum games with application to H-infinity control. IEEE Trans. Systems, Man, and Cybernetics, Part B: Cybernetics 37(1), 240–247 (2007)
32. Altamimi, A., Lewis, F.L., Abu-Khalaf, M.: Model-free Q-learning designs for linear discrete-time zero-sum games with application to H-infinity control. Automatica 43, 473–481 (2007)
33. Morimoto, J., Zeglin, G., Atkeson, C.G.: Minmax differential dynamic programming: Application to a biped walking robot. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (2003)
34. Si, J., Barto, A.G., Powell, W.B., Wunsch II, D.: Handbook of Learning and Approximate Dynamic Programming. Wiley-IEEE Press (2004)
35. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction. MIT Press, Cambridge (1998)
36. Lewis, F.L., Syrmos, V.L.: Optimal Control, 2nd edn. Wiley Interscience (1995)
37. Atkeson, C.G., Schaal, S.: Learning tasks from a single demonstration. In: Proceedings of the 1997 IEEE International Conference on Robotics and Automation (ICRA 1997), pp. 1706–1712 (1997)
38. Atkeson, C.G., Schaal, S.: Robot learning from demonstration. In: Proc. 14th International Conference on Machine Learning, pp. 12–20. Morgan Kaufmann (1997)
39. Atkeson, C.G.: Nonparametric model-based reinforcement learning. In: Advances in Neural Information Processing Systems, vol. 10, pp. 1008–1014. MIT Press, Cambridge (1998)
40. Liu, C., Su, J.: Biped walking control using offline and online optimization. In: 30th Chinese Control Conference (2011)
41. Tassa, Y., Erez, T., Todorov, E.: Synthesis and stabilization of complex behaviors through online trajectory optimization. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS (2012)

Use of Compliant Actuators in Prosthetic Feet and the Design of the AMP-Foot 2.0

Pierre Cherelle, Victor Grosu, Michael Van Damme,
Bram Vanderborght, and Dirk Lefeber

Abstract. From robotic prostheses, to automated gait trainers, rehabilitation robots have one thing in common: they need actuation. The use of compliant actuators is currently growing in importance and has applications in a variety of robotic technologies where accurate trajectory tracking is not required like assistive technology or rehabilitation training. In this chapter, the authors presents the current state-of-the-art in trans-tibial (TT) prosthetic devices using compliant actuation. After that, a detailed description is given of a new energy efficient below-knee prosthesis, the AMP-Foot 2.0.

1   Introduction

Experience in clinical and laboratory environments indicates that many trans-tibial (TT) amputees using a completely passive prosthesis suffer from non-symmetrical gait, a high measure of perceived effort and a lack of endurance while walking at a self-selected speed [28, 20, 3]. Using a passive prosthesis means that the patient's remaining musculature has to compensate for the absence of propulsive ankle torques. Therefore, adding an actuator to an ankle-foot prosthesis has the potential to enhance a subjects mobility by providing the missing propulsive forces of locomotion. In the growing field of rehabilitation robotics, prosthetics and wearable robotics, the use of compliant actuators is becoming a standard where accurate trajectory tracking is not required. Their ability to safely interact with the user and to absorb large forces due to shocks makes them particularly attractive in applications based on physical human-robot interactions. The approach based on compliance on a mechanical level (i.e. passive compliance), compared to introduced compliance on the control level (i.e. active compliance), ensures intrinsic compliance of the device

Pierre Cherelle · Victor Grosu · Michael Van Damme · Bram Vanderborght · Dirk Lefeber
Vrije Universiteit Brussel, Pleinlaan 2, 1050 Brussels, Belgium
e-mail: pcherell@vub.ac.be

at all time, enhancing hereby system safety. Therefore, this type of actuator is preferred in novel rehabilitation robots where safe human-robot interaction is required. In the particular case of trans-tibial (TT) prostheses, compliance of the actuation provides even more advantages. Besides shock absorption in case of collision with objects during walking, energy provided by the actuator (e.g. electric motor) can be stored into its elastic element (e.g. spring in series). This energy can be kept for a moment and released when needed to provide propulsion of the subject [7]. As a result of this, the electric drive can be downsized so as the overall weight and inertia of the prosthetic device to improve the so-called 3C-level, i.e. comfort, control and cosmetics.

Compliant actuators can be divided into actuators with fixed or variable compliance. Examples of fixed compliance actuators are the various types of series elastic actuators (SEA) [19], the bowden cable SEA [22] and the Robotic Tendon Actuator [14] to name a few. On the other hand the PPAM (Pleated Pneumatic Artificial Muscles) [25], the MACCEPA (Mechanically Adjustable Compliance and Controllable Equilibrium Position Actuator) [6, 8] and the Robotic Tendon with Jack Spring actuator [15, 16] are examples of variable stiffness actuators. For a complete state-of-the-art in compliant actuation, the authors refer to [9].

In this chapter, the authors present the current state-of-the-art in powered transtibial prostheses using compliant actuation and a brief analysis of their working principles. A description of the author's latest actuated prosthetic foot design will then be given, i.e. the AMP-Foot 2.0. Conlusions and future work will be outlined at the end of the chapter.

2 Powered Prosthetic Feet

In this section, the authors present the current state-of-the-art in powered ankle-foot prostheses, better known as "bionic feet", in which the generated power and torques serve for propulsion of the amputee. The focus is placed on devices using compliant actuators. For a complete state-of-the-art review of passive TT prosthesis comprising "Conventional Feet" and "Energy Storing and Returning" (ESR) feet, the authors refer to [24].

2.1 Pneumatically Actuated Devices

Pneumatic actuators are also known as "antagonistically controlled stiffness" actuators [9] since two actuators with non-adaptable compliance and non-linear force displacement characteristics are coupled antagonistically. By controlling both actuators, the compliance and equilibrium position can be set.

Klute et al. [17] have designed an artificial musclo-tendon actuator to power a below-knee prosthesis. To meet the performance requirements of an artificial *triceps surae* and *Achilles* tendon, an artificial muscle, consisting of two flexible

pneumatic actuators in parallel with a hydraulic damper, and placed in series with a bi-linear, two-spring implementation of an artificial tendon, was build into the ankle-foot prosthesis.

Goldfarb et al. [21] at Vanderbilt University have developed a powered transfemoral prosthesis using knee and ankle pneumatic actuation.

Developed within the Robotics & Multibody Mechanics Research Group at Vrije Universiteit Brussel, Belgium, the Pleated Pneumatic Artificial Muscle (PPAM) [23] was originally intended to be used in bipedal walking robots. It is a lightweight, air-powered, muscle-like actuator consisting of a pleated airtight membrane. Its advantage compared to other artificial muscle comes from the unfolding of the pleated membrane. Because of this there is virtually no threshold pressure, hysteresis is reduced when compared to other types of muscles, and contractions of over 40% of the initial length are possible. Whithin the IPAM (Intelligent Prosthesis using Artificial Muscles) Project [25], a TT prosthesis using Pleated Pneumatic Artificial Muscles was developed to demonstrate the importance of push-off during gait [25].

In general, drawbacks of pneumatic systems are the high cost of pressurized air production and supply requirements for autonomy. Therefore, electric actuators are preferred in novel prosthetic designs.

2.2   *Electrically Actuated Devices*

At the Massachusetts Institute of Technology (MIT), the Powered Foot Prosthesis [1] has been developed using a combination of a spring and a high power series elastic actuator. Its working principle consists of loading a spring during the controlled dorsiflexion phase and to activate a torque source (SEA) in parallel when peak power is needed. As a result of this, energy is added to the system to provide push-off. A peak output torque of 140 Nm and power output of 350W is applied with a torque bandwidth up to 3.5Hz. This prosthetic device has shown its effectiveness by improving metabolic economy of walking individuals with TT amputation [2], on average by 14% compared to evaluated conventional prostheses. Further research at the MIT led to the developement of the Powerfoot BiOM sold by iWalk [10]. The BiOM is a Bionic lower-leg system to replace lost Muscle function that approximates the action of the ankle, Achilles tendon and calf muscles by propelling the amputee upwards and forwards during walking.

At the Arizona State University, the SPARKy project (Spring Ankle with Regenerative Kinetics) [12] uses a Robotic Tendon actuator (including a 150W DC motor) [14] to provide 100% of the push-off power required for walking while maintaining intact gait kinematics. The first prototype (SPARKy 1) [11] was shown to store and release approximately 16J of energy per step while an intact ankle of a 80Kg subject at 0.8Hz walking rate needs approximately 36J [13]. A second prototype was built (SPARKy 2) with a lighter and more powerfull roller screw transmission and brushless DC motor. Both design's working principle rely on a SEA attached between the heel and the leg. This robotic tendon is controlled to provide the ankle torque and power necessary for propulsion during gait. The third prototype (SPARKy 3) [4]

was designed to actively control both inversion and eversion as well as plantarflexion and dorsiflexion while providing high power for running and jumping.

At the Vrije Universiteit Brussel, a compact, low-weight and energy efficient trans-tibial prosthesis powered by electric drives was proposed to improve the amputee's gait [26]. The challenge was to design a device respecting the ankle-foot requirements that mimics a natural ankle behavior during walking. It was shown that by incorporating a modified MACCEPA [8] into the design, an acceptable approximation of the ankle characteristic is obtained. The prosthesis contains two unidirectional springs in parallel, connected to two lever arms. By connecting one of the lever arms to a locking mechanism it is shown that the energy efficiency is greatly improved. The actuation comprises a 150W motor with gearhead transmission connected to a ball screw mechanism through a timing belt. The Powered Below-knee Prosthesis's behavior is adjustable depending on amputee's gait speed by regulating the pretension of the springs. It is capable of providing 100% of the required push-off power, consuming only 22.19J per step for a 75Kg subject walking at normal cadence on level ground.

Further research at the Robotics & Multibody Mechanics Research Group [5] led to the design and development of the Ankle Mimicking Prosthetic Foot (AMP-Foot) 2.0. Fig. 1 shows some of the named prosthetic devices.
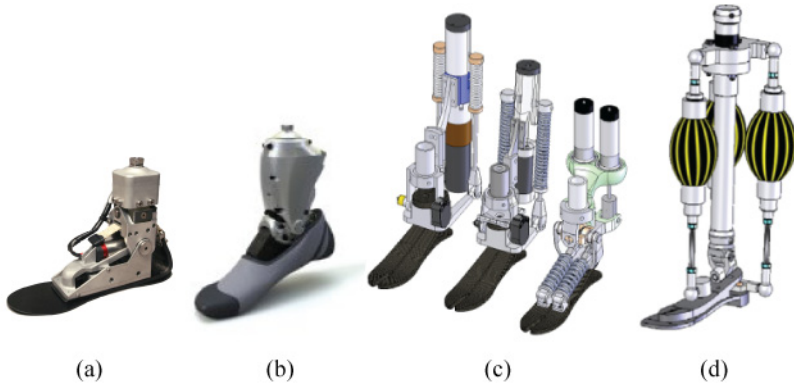
Fig. 1 (a) MIT Power Foot Prosthesis. (b) The BiOM from iWalk. (c) SPARKy 1, 2 and 3 (from left to right). (d) Trans-tibial Prosthesis using Pleated Pneumatic Artificial Muscles.

3   The Amp-Foot 2.0: A New Energy Efficient Concept

The main objective of this research is to harvest as much energy as possible from the gait and to implement an electric actuator with minimized power consumption. The concept of the AMP-Foot 2.0 relies on the use of a "plantar flexion (PF)" spring, to store energy from the controlled dorsiflexion phase of stance while an electric actuator is loading a "push-off (PO)" spring during the complete stance phase. Due to the use of a locking mechanism, the energy injected into the PO spring can be
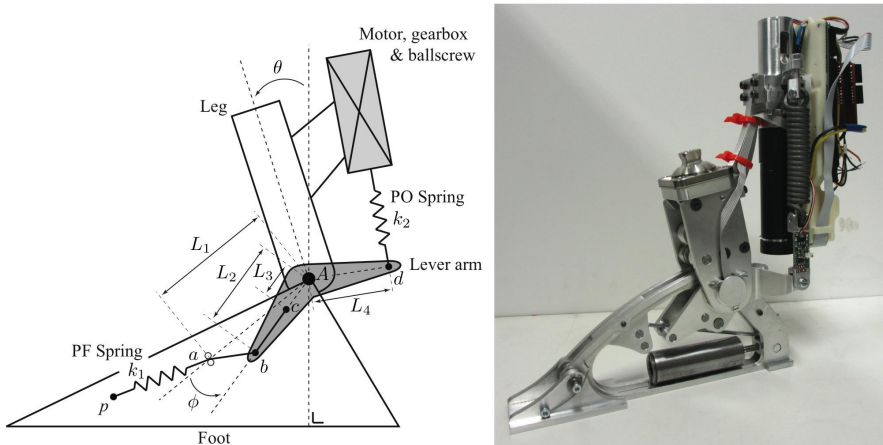
Fig. 2 Schematics and picture of the AMP-Foot 2.0

delayed and released at push-off. This way, the actuator's power is significantly reduced and so is its size and weight while still providing the full torque and power needed for locomotion.

In Fig. 2, the essential parts of the AMP-Foot 2.0 are represented. The device consists of 3 bodies pivoting around a common axis (the ankle), i.e. the leg, the foot and a lever arm. As mentioned before, the system comprises 2 spring sets: a PF and a PO spring set. The PF spring is placed between a fixed point p on the foot and a cable that runs over a pulley a to the lever arm at point b and is attached to the lever arm at point c, while the PO spring is placed between the motor-ballscrew assembly and a fixed point d on the lever arm. Not drawn in Fig. 2 is the locking mechanism which provides a rigid connection between the leg and the lever arm when energy is injected into the system. Its working principle is discussed further in the text.

To maintain a consistent notation through the chapter, symbols used in the schematics are described:

L_1 = distance between ankle axis (A) and point a.

L_2 = distance between ankle axis (A) and point b.

L_3 = distance between ankle axis (A) and point c.

L_4 = distance between ankle axis (A) and point d.

θ = angle between foot and leg.

ϕ = angle between foot and lever arm.

k_1 = Plantar Flexion spring stiffness.

k_2 = Push-Off spring stiffness.
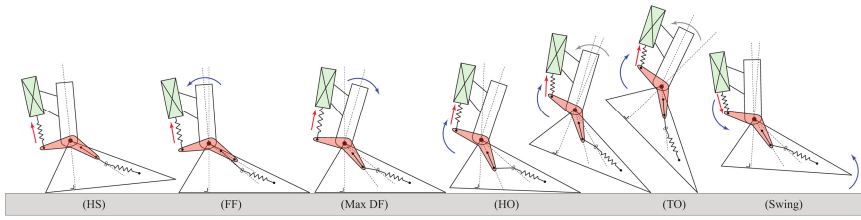
Fig. 3 Behavior of the AMP-Foot 2.0 during a complete stride

To illustrate the behavior of the AMP-Foot 2.0, one complete gait cycle is divided
into several phases, shown in Fig. 3, and the working principle of the prosthetic
device during each phase is explained.

3.1 Principle of Optimal Power Distribution

As mentioned before, the gait cycle is divided in 5 phases starting with a *controlled
plantarflexion* from heel strike (HS) to foot flat (FF) produced by muscles as the *Tib-
ialis Anterior*. This is followed by a *controlled dorsiflexion* phase ending in *push-off*
at heel off (HO) during which propulsive forces are generated mainly by the *Soleus*
and *Gastrocnemius* muscle groups. In the *late stance* phase, the torque produced by
the ankle decreases until the leg enters the so-called *swing* phase at toe off (TO).
Once the leg is engaged in the swing phase, the foot resets and prepares for the
next step.

From heel strike (HS) to foot flat (FF):
A step is initiated by touching the ground with the heel. During this phase the
foot rotates with respect to the leg, until θ $(=\phi)$ reaches approximatly $-5°$.
Since the lever arm is fixed to the leg, the PF spring is elongated and generates a
dorsiflexing torque at the ankle which is calculated as

$$T_1 = k_1(l_1 - l_0 + V_{0,1})\frac{L_1L_3}{l_1}sin\phi \tag{1}$$

in which

T_1 = Torque applied by the PF spring to the lever arm and thus to the ankle.
k_1 = Spring constant of the PF spring.
l_0 = Distance between the fixed points a and c when $\phi = 0$ i.e. $l_0 = L_1 - L_3$.
$V_{0,1}$ = Pretension of the PF spring.
l_1 = Distance between the fixed point a and c i.e.

$$l_1 = \sqrt{L_1^2 + L_3^2 - 2L_1L_3cos(\phi)} \tag{2}$$

During this period the electrical drive pulls the PO spring. Since the motor is attached to the leg and lever arm is locked to the leg, the PO spring is loaded without delivering torque to the ankle joint. Therefore the prosthesis is not affected by the forces generated by the actuator.

From (FF) to heel off (HO):
When the foot stabilizes at FF, the leg moves from $\theta = -5°$ to $\theta = +10°$. Until the leg reaches $\theta = 0°$ the torque of the system is given by Equation (1). From $\theta = 0°$ to $\theta = +10°$ the lever arm length is adjusted and thus the torque becomes:

$$T_1 = k_1(l_1^* - l_0^* + V_{0,1})\frac{L_1 L_2}{l_1^*}\sin\phi \qquad (3)$$

in which

$l_0^* =$ Distance between the fixed points a and b when $\phi = 0$ i.e. $l_0^* = L_1 - L_2$.
$l_1^* =$ Distance between the fixed point a and b i.e.

$$l_1^* = \sqrt{L_1^2 + L_2^2 - 2L_1 L_2 cos(\phi)} \qquad (4)$$

This is done by using two different connection points b and c (Fig. 2), on the lever arm, which are respectively active when $\theta > 0$ and $\theta < 0$. This way it is possible to mimic the change in stiffness of a sound ankle. During this phase the motor is still injecting energy into the system by loading the PO spring.

At heel off (HO):
Because the angle between the PO spring and the lever arm is fixed at $\pi/2$, the torque excerted by the spring (no pretension) on the lever arm is given by

$$T_2 = k_2 l_2 L_4 \qquad (5)$$

with

$T_2 =$ Torque applied to the lever arm by the PO spring.
$k_2 =$ Spring constant of the PO spring.
$l_2 =$ Elongation of the PO spring.

The torque T_1 excerted by the PF spring on the lever arm is given by Equation (3). At the moment of HO, the locking mechanism is unlocked and as a result of this, all the energy which is stored into the PO spring is fed to the system. Since $T_1 \leq T_2$ both PF and HO springs tend to rotate the lever arm with an angle ψ to a new equilibrium position. In other words, T_1 and T_2 respectively evolves to new values T_1' and T_2' such that $T_1' = T_2' = T'$ with $T' \geq T_1$ and $T' \leq T_2$. The torque at the ankle becomes

$$T' = k_1(l_1' - l_0' + V_{0,1})\frac{L_1L_2}{l_1'}sin(\phi + \psi) \tag{6}$$

in which

$$l_0' = l_0^* = L_1 - L_2$$

$$l_1' = \sqrt{L_1^2 + L_2^2 - 2L_1L_2cos(\phi + \psi)}$$

The effect of this is a virtually instantaneous increase in torque and decrease in stiffness of the ankle joint as depicted in Fig. 4.

from HO to toe off (TO):
In the last phase of stance, the torque is decreasing until toe off (TO) occurs at $\theta = -20°$. Since the two springs are now connected in series, the rest position of the system has changed according to the elongation of the PO spring. As a result of this a new equilibrium position is set to approximately $\theta = -20°$. The actuator is still working during this phase.

Swing phase:
After TO, the leg enters into the so called swing phase in which the whole system is resetted. While the motor turns in the opposite direction to bring the ballscrew mechanism back to its initial position, return springs are used to set θ back to $0°$ and to close the locking mechanism. At this moment, the device is ready to undertake new step.
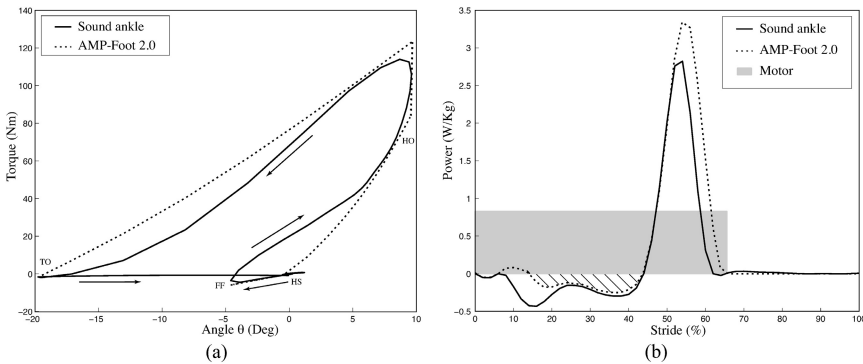
Fig. 4 (a) Torque-Angle characteristic of the AMP-Foot 2.0 compared to abled-bodied ankle-feet according to gait analysis conducted by D. Winter [27]. (b) Ankle power during one stride. The solid line represents the power generation of a sound ankle while the dotted line represnts the resulting power of the AMP-Foot 2.0. The gray rectangle shows how the actuator power is spread over one gait cycle while the shade area represents the energy gathered from the controlled dorsiflexion with the PF spring.

Table 1 Lever arm dimensions

$L_1 = 80\ mm$	$L_2 = 60\ mm$
$L_3 = 30\ mm$	$L_4 = 60\ mm$

3.2 Mechanics and Design

According to Winter [27] a 75 *kg* subject walking at normal cadence (ground level) produces a maximum joint torque of 120 *Nm* at the ankle. This has been taken as a criterion. Moreover, an ankle articulation has a moving range from approximatly $+10°$ at maximal dorsiflexion to $-20°$ at maximal plantarflexion. Therefore a moving range of $-30°$ to $+15°$ has been chosen for the joint to fulfill the requirements of the ankle anatomy. The length of the lever arms named in Fig. 2 are given in TABLE 1. The foot is made to match a European size 43 with a ankle height of approximately 8 cm. The largest part of the prosthesis has a width of 5 cm and is located at the toes to enhance stability. This way the prosthesis fits in a shoe which is significantly more comfortable for the amputee. A description of the elements used in the prosthesis is given.

Spring Sets:
As described in the previous section, the AMP-Foot 2.0 uses two spring sets. For the PF spring (k_1), a belleville spring assembly, which is shown in Fig. 5, is used because of its compactness en ability to provide extremely high forces. This assembly consists of a tube in which a slider is moving to compress the disc springs. To achieve the desired, as linear as possible, spring characteristic, 29 belleville springs are stacked in series. The PF spring has a stiffness of approximately 300 N/mm. For the PO spring (k_2), two tension springs with each a stiffness of 60 N/mm are used.

Actuation:
To achieve the requirements of a able-bodied ankle-foot complex, an actuator with a good "power and strength to weight" ratio, high mechanical efficiency is needed. A Maxon Brushed DC motor (60 W) has been chosen in combination with a gearbox and ballscrew assembly, which is described in TABLE 2. The positioning of the motor and other hardware have been chosen in view of the range of motion and optimized for compactness of the system.

Locking Mechanism:
As mentioned before, a critical part of this mechanical system is the locking mechanism. This locking must be able to withstand high forces while being as compact and lightweight as possible. The crucial and challenging part is that the system must be unlocked when bearing its maximum load and last but not least,
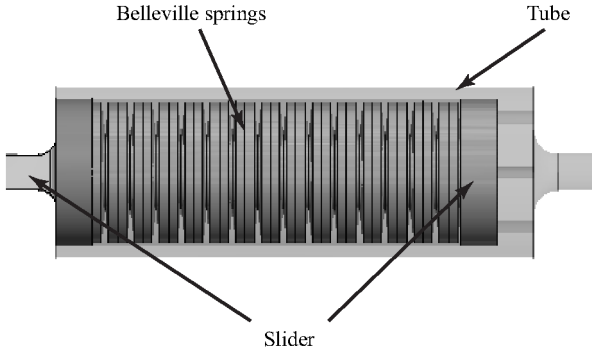
Fig. 5 Section representation of a disc spring assembly. 29 disc springs are stacked in series on a slider which moves into a tube.

Table 2 Motor and Transmissions

Motor	Maxon RE 30 - 60 W $T_{cont.} = 51.7\ mNm$ $T_{peak} = 150\ mNm$
Transmission stage 1	Maxon GP32BZ i = 5.8:1
Transmission stage 2	Maxon ballscrew GP32S $\phi 10x2$ $\eta_{transmission1\&2} = +/-75\%$

this unlocking must require a minimum of energy. Fortunately, the lever arm has to be locked to the leg at a fixed angle. These requirements have been taken as criteria and to achieve this, it has been chosen to work with a four bar linkage moving in and out of its singular position. This principle has already proved its effectiveness in [18], where it is used to lock the knee joint of a walking robot. Fig. 6 shows the schematics of the four bar linkage when locked (a) and opened (b). When the four bar linkage is set in its singular position, it is in unstable equilibrium. Therefore to ensure locking, the system is allowed to move a bit further than its singular position. When the singular position is past, the load forces the mechanism to continue moving in the same direction. To keep it in equilibrium, a mechanical stop blocks the system. A solenoid (Mecalectro, 12VDC, 5W) is

then used to push the mechanism back past its singular position when triggered. Because close to its singular position, the transmission coefficient of the four bar linkage tends to infinity, the resulting force (or torque) which has to be applied to unlock the system is greatly reduced. Fig. 7 shows the transmission coefficient and the resulting force necessary for unlocking under maximal load in function of the lever arm angle.

It can be estimated that the maximum resulting load which can be applied to the lever arm, e.g. when PF spring and PO spring are fully extended (at maximal dorsiflexion), is more or less 40 Nm. In this case, and if the four bar mechanism is past its singular position of a few degrees, the resulting force needed for unlocking is estimated to be less than 10 N. Of course, this is a worst case senario. Having the PO spring completely extended at maximal dorsiflexion is certainly not optimal. This would mean the motor has to stop moving between HO and TO. A better control strategy is to make the motor move during the complete stance phase as shown in Fig. 4. Therefore, depending on the way the motor is controlled, the resulting force needed to unlock the four bar linkage will be reduced.

Sensors:
The two spring assemblies are equipped with custom made loadcells which allows a force measurement with a resolution of ± 1.5 N. To measure the position of the lever arm, and the leg with respect to the foot, two absolute magnetic encoders (Austria Micro Systems AS5055) are used with a resolution of $\pm 0.08°$. While the magnets of the encoders are glued to the ankle axis (which is fixed to the foot), the two hall sensors are fixed on the lever arm, respectively on the leg.

(a)   (b)
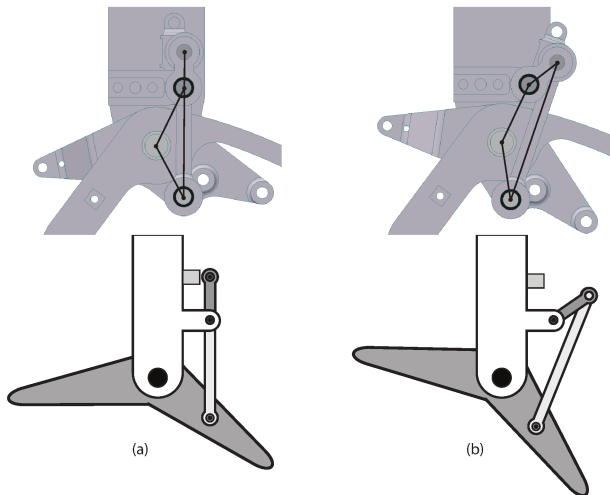
Fig. 6 CAD representation and schematics of the four bar mechanism in locked (a) and unlocked (b) position
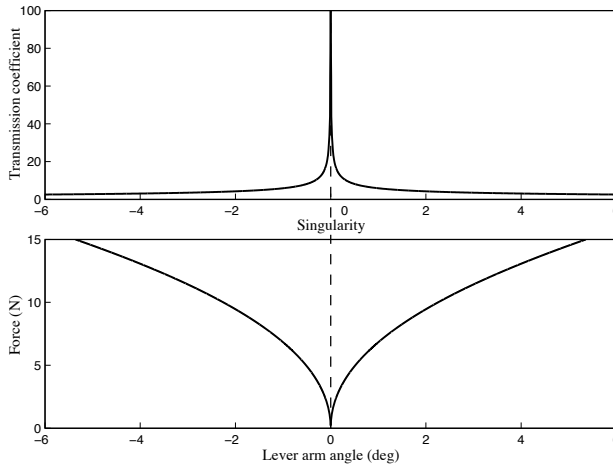
Fig. 7 Transmission Coefficient and resulting force of the four bar linkage mechanism close to its singular position (0°)

As a result of this, the resulting torque at the ankle can be calculated using the mathematical model of the mechanical system which has been discussed before. To detect the important triggers during the stance phase (IC, FF, HO, TO), two Force Sensing Resistors (FSR) are placed on the foot sole: one at the heel and one at the toes. These triggers will be used to control the motor and to lock or unlock the locking mechanism.

4 Conclusions

In this chapter, the authors propose a new design of an energy efficient powered transtibial prosthesis mimicking able-bodied ankle behavior, the AMP-Foot 2.0. The inovation of this study is to gather energy from motion during the controlled dorsiflexion with a PF spring while storing energy produced by a low power electric motor into a PO spring. This energy is then released with a delay at a favourable time for push-off thanks to the use of a locking system. The prosthesis is designed to provide a peak output torque of 120 Nm with a range of motion of approximately 45° to fullfill the requirements of a 75 kg subject walking on level ground at normal cadence. Its total weight is $\pm$ 2.5 kg which corresponds to the requirements of an intact foot. The prototype is completely built and hardware and control are currently being tested. Experiments with amputees will follow.

References

1. Au, S.K., Herr, H.M.: Powered ankle-foot prosthesis. IEEE Robotics & Automation Magazine (2008)
2. Au, S.K., Weber, J., Herr, H.: Powered ankle-foot prosthesis improves walking metabolic economy. IEEE Trans. Robot. 25(1), 1–16 (2009)
3. Bateni, H., Olney, S.: Kinematic and kinetic variations of below-knee amputee gait. Journal of Prosthetics and Orthotics 14(1), 2–13 (2002)
4. Bellman, R., Holgate, M., Sugar, T.: Sparky 3: Design of an active robotic ankle prosthesis with two actuated degrees of freedom using regenerative kinetics. In: Proceedings of the IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics, pp. 1–6 (2009)
5. Brackx, B., Damme, M.V., Matthys, A., Vanderborght, B., lefeber, D.: Passive ankle-foot prosthesis prototype with extended push-off. Advanced Robotics (2011) (in review)
6. Cherelle, P., Grosu, V., Matthys, A., Ham, R.V., Damme, M.V., Vanderborght, B., Lefeber, D.: The maccepa actuation system as torque actuator in the gait the maccepa actuation system as torque actuator in the gait rehabilitation robot altacro. In: IEEE RAS and EMBS International Conference on Biomedical Robotics and Biomechatronics, pp. 27–32 (2010)
7. Cherelle, P., Matthys, A., Grosu, V., Vanderborght, B., Lefeber, D.: The amp-foot 2.0: Mimicking intact ankle behavior with a powered transtibial prosthesis. In: Proceedings of the IEEE International Conference on Biomedical Robotics and Biomechatronics (2012)
8. Ham, R.V., Damme, M.V., Verrelst, B., Vanderborght, B., Lefeber, D.: Maccepa, the mechanically adjustable compliance and controllable equilibrium position actuator: A 3d of joint with 2 independent compliances. International Journal of Applied Mechanics 4, 130–142 (2007)
9. Ham, R.V., Sugar, T., Vanderborght, B., Hollander, K., Lefeber, D.: Compliant actuator designs. IEEE Robotics & Automation Magazine 16(3), 81–94 (2009)
10. Herr, H.: iwalkpro, http://www.iwalkpro.com/home.html
11. Hitt, J., Bellman, R., Holgate, M., Sugar, T., Hollander, K.: The sparky project: Design and analysis of a robotic transtibial prosthesis with regenerative kinetics. In: Proceedins of the ASME International Design Engineering Technical Conference & Computers and Information in Engineering Conference, pp. 1–10 (2007)
12. Hitt, J.K., Sugar, T.G., Holgate, M., Bellman, R.: An active foot-ankle prosthesis with biomechanical energy regeneration. Journal of Medical Devices 4(1) (2010)
13. Holgate, M., Hitt, J., Bellman, R.: The sparky (spring ankle with regenerative kinetics) project: Choosing a dc motor based actuation method. In: Proceedings of the IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics, pp. 163–168 (2008)
14. Hollander, K.W., Ilg, R., Sugar, T.G.: Design of the robotic tendon. In: Design of Medical Devices Conference, pp. 1–6 (2005)
15. Hollander, K.W., Sugar, T.G., Herring, D.E.: Adjustable robotic tendon using a jack spring. In: Proceedings of the IEEE International Conference on Rehabilitation Robotics, pp. 1–6 (2005)
16. Hollander, K.W., Sugar, T.G., Herring, D.E.: A robotic jack spring for ankle gait assistance. In: Proceedings of IDETC/CIE ASME International Design Engineering Technical Conferences & Computers and Information in Engineering Conference, pp. 1–10 (2005)

17. Klute, G.K., Czerniecki, J.M., Hannaford, B.: Muscle-like pneumatic actuators for below-knee prostheses. In: International Conference on New Actuators (2000)
18. van Oort, G., Carloni, R., Borgerink, D.J., Stramigioli, S.: An energy efficient knee locking mechanism for a dynamically walking robot. In: IEEE International Conference on Robotics and Automation, pp. 9–13 (2011)
19. Pratt, G.A., Williamson, M.M.: Series elastic actuators. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, vol. 1, pp. 399–406 (1995)
20. Skinner, H.B., Effeney, D.J.: Gait analysis in amputees. American Journal of Physical Medicine 64, 82–89 (1985)
21. Sup, F., Bohara, A., Goldfarb, M.: Design and control of a powered knee and ankle prosthesis. In: IEEE International Conference on Robotics and Automation (2007)
22. Veneman, J., Ekkelenkamp, R., Kruidhof, R., van der Helm, F., van der kooij, H.: Design of a series elastic- and bowdencable-based actuation system for use as torque-actuator in exoskeleton-type training. In: Proceedings of the IEEE International Conference on Rehabilitation Robotics, pp. 1–4 (2005)
23. Verrelst, B., Ham, R.V., Vanderborght, B., Daerden, F., Damme, M.V., Lefeber, D.: Second generation pleated pneumatic artificial muscle and its robotic applications. Advanced Robotics 20(7), 783–805 (2006)
24. Versluys, R., Beyl, P., Damme, M.V., Desomer, A., Ham, R.V., Lefeber, D.: Prosthetic feet: State-of-the-art review and the importance of mimicking human ankle–foot biomechanics. Disability and Rehabilitation: Assistive Technology 4(2), 65–75 (2009), doi:10.1080/17483100802715092
25. Versluys, R., Desomer, A., Lenaerts, G., Pareit, O., Vanderborght, B., der Perre, G.V., Peeraer, L., Lefeber, D.: A biomechatronical transtibial prosthesis powered by pleated pneumatic artificial muscles. International Journal of Modelling, Identification and Control 4(4), 1–12 (2008)
26. Versluys, R., Matthys, A., Ham, R.V., Vanderniepen, I., Lefeber, D.: Powered ankle-foot system that mimics intact human ankle behavior: Proposal of a new concept. In: Proceedings of the IEEE International Conference on Rehabilitation Robotics, pp. 658–662 (2009)
27. Winter, D.A.: The biomechanics and motor control of human gait: Normal, elderly and pathological. Waterloo Biomechancs 2 (1991)
28. Winter, D.A., Sienko, S.E.: Biomechanics of below-knee amputee gait. Journal of Biomechanics 21(5), 361–367 (1988)

Modeling and Optimization of Human Walking

Martin Felis and Katja Mombaur

Abstract. In this paper we show how optimal control techniques can be used to generate natural human walking motions in 3D. Our approach has potential applications in humanoid robotics, biomechanics and computer graphics. It has the advantage that it does not require any previous knowledge about walking motions from experiments. In this study we consider symmetric walking along a straight line, but the same techniques can be used to generate walking motions along curved paths or asymetric motions. We establish a multibody model of the human body with twelve segments including a head, a three-segment trunk, and arms and legs with two segments each. An optimal control problem is formulated that minimizes joint torques head movement, and the impulse on touch-down in a combined criterion. The dynamics of the multi-body system are considered as constraints to the optimal control problem to guarantee physically feasible motions. The optimal control problem is solved using an efficient direct multiple-shooting method. A skeletal animation library is used to present the results of the optimized motion.

1 Introduction

Our anatomy is highly optimized for bipedal locomotion, which makes it very easy, for most of us, to walk on different terrain even under disturbances. Also, the environment we live in has been greatly influenced by our locomotion mode (e.g. stairs).

Martin Felis
Interdisciplinary Center for Scientific Computing (IWR), University of Heidelberg, INF 368, D - 69120 Heidelberg, Germany
e-mail: martin.felis@iwr.uni-heidelberg.de

Katja Mombaur
Interdisciplinary Center for Scientific Computing (IWR), Heidelberg University, INF 368, 69120 Heidelberg, Germany, Associate Researcher at LAAS-CNRS, Toulouse
e-mail: kmombaur@uni-hd.de, kmombaur@laas.fr

Research of bipedal walking motions is of great interest in many areas. In robotics, the aim is to create humanoids and other bipedal robots with a human-like capability to walk in flexible environments. In computer graphics, creating realistic motions for virtual characters in games or movies presents a big challenge since our perception of motion is very specialized and easily recognizes unphysical motions. In biomechanics, models for human walking are required to gain a better understanding of the human locomotor system.

In the different fields, a large variety of models exists to analyze or generate motions. Models in biomechanics range from simple mass spring systems [4] to complex multibody systems with simulated muscles [2]. These are primarily used to describe or investigate forces that act within the body, but not to generate motions.

In computer graphics a lot of research is being done to synthesize plausible motions. Some authors use optimization techniques to compute or find a transition from one pre-recorded motion to another such as in [13] or [12]. Other works in this area incorporate dynamics simulations to generate more realistic motions. Witkin et. al. [17] used a dynamical model and optimization to animate a lamp figure with six degrees of freedom. They used a two boundary value formulation to generate an optimal motion minimizing power consumption. Hodgins et al. [5] used a finite state machine and proportional-derivative controllers to compute torques that generate variuous motions such as running, cycling and vaulting. A robust controller for virtual humans that also allows modification of the generated walking style is described in [18]. The controller allows the model to walk on uneven terrain in both 2D and 3D.

The zero-moment-point (ZMP) [15] is frequently used in humanoid robotics, where the controller aims to keep the ZMP within the polygon of support (see e.g. [6], [7]). ZMP-based control leads to a safe and conservative motion for humanoid robots. However, the gait is very different from human walking. The human gait is both faster and in general more energy efficient, since robots mainly control the precise joint angles instead of exploiting its dynamics.

Another approach that is inspired by biology is to use central pattern generators (CPG) that also allow the robot to adapt to the environment [11]. CPGs generate rhythmic motor signals and have to be trained, e.g. by reinforcement learning or neural networks, to generate walking patterns.

In this paper we want to generate physically valid and natural human walking motions by using a dynamic model and optimal control techniques. The same approach has already been successfully used for human running [14], [10]. Its advantage is that it does not require the prescription of exact trajectories or fixed keyframes for the degrees of freedom of the walking system, so no previous knowledge from experiments is needed. Also, it does not require previous information about the driving torques of the walking motion. Instead, trajectories, as well as torques, that best satisfy the optimization criterion are determined simultanously by the optimization process. Walking differs from running with respect to the sequence of foot contacts: while running involves alternating single–foot contact and flight phases, walking is characterized by a change between single– an double–support contact phases. The double support phase has frequently been ignored in simpler models, but it is

considered in our walking model. We present a biollogically inspired objective function, which is a combination of different factors and leads to realistic walking motions. The computations of this paper are performed for human geometry and mass distribution using standard biomechanical data. However, the same type of computations could be done for robot-specific parameters to determine best-possible input torques for a humanoid robot.

In the next section we describe the dynamic modeling of a human gait as a multi–phase problem based on a rigid multibody model. We then present the formulation of natural gaits as an optimal control problem and how this problem can be solved numerically. Finally we describe the optimal solution and show visualization snapshots.

2 Modeling of the Gait and the Human Model

In this section we describe the formulation of human walking motions as a multi–phase problem based on a rigid multibody system. We consider regular forward walking along a straight line, which is charaterized by:

i) Identical left and right steps (bilateral symmetry);
ii) Periodicity constraints on the pose and the velocities.

Additionally, we chose a moderate walking velocity of $1.1 m/s$ and a step length of $0.5m$. In our problem velocity and step length are only input variables but could also be used as optimization variables.

We focus on this most dominant mode of human locomotion, but the same techniques could be used to study more irregular forms of walking. This allows us to focus on the optimization of a single step gait cycle by formulating appropriate periodicity constraints including a shift of sides. The gait cycle we are considering is

(a) The human walk cycle for a single step.

(b) Overview of our model and its degrees of freedom.
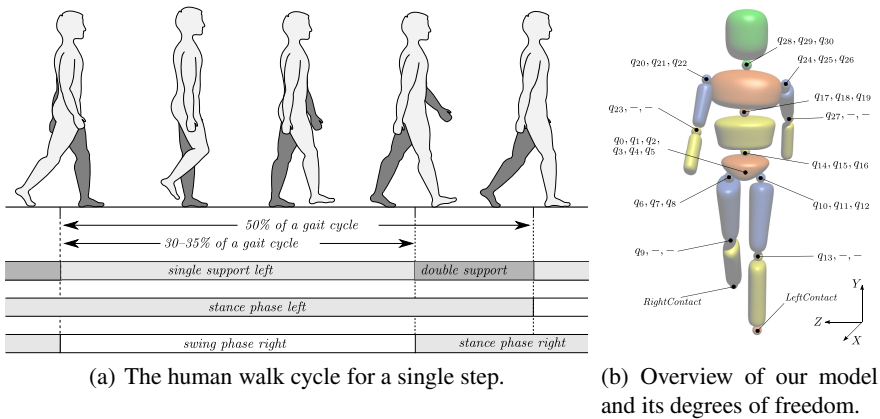
Fig. 1 Gait cycle and model overview

shown in Figure 1(a). It starts at the moment of right foot toe–off with a left leg stance phase. After touchdown of the right leg there is a double support phase and the cycle ends at the instant of left toe-off.

The multi–body model we created consists of 12 bodies, including legs, upper body, arms, head and a three–segmented trunk. An overview of the model configuration and its resulting $n_{dof} = 31$ degrees of freedom can be found in Figure 1(b). We use a right–handed coordinate system in which, seen from the model, X points forward (saggital direction), Y up (longitudonal direction), and Z towards the left (transversal direction). Rotations are described by ZYX–Euler angles. The dynamical parameters for the model (segment geometry, masses, inertia) were taken from [9] to represent an adult human. The posture of the model is described by using minimal coordinates $q \in \mathbb{R}^{n_{dof}}$.

In the present model version we use point feet which means there is no distinction between initial heel, flat foot, and toe contact. Instead, for each foot, there is only a single point of contact at positions *RightContact* and *LeftContact* at the distal ends of the lower leg segments (see Figure 1(b)). We assume nonsliding ground contact and therefore use algebraic equations to fix the translation of the feet when in contact with the ground. This results in a system of differential algebraic equations (DAE) with differential index 3. Since the single– and double–support phases are described by different algebraic equations we have two distinct DAE model equations. These equations are then reformulated to index 1 systems that can be solved with standard ordinary differential equation (ODE) methods, each time solving the underlying system of algebraic constraints of the form:

$$\begin{pmatrix} M & G^T \\ G & 0 \end{pmatrix} \begin{pmatrix} \ddot{q} \\ -\lambda \end{pmatrix} = \begin{pmatrix} \tau - N \\ -\gamma \end{pmatrix}. \tag{1}$$

The matrix of the linear system consists of the joint–space inertia matrix M and the Jacobian of the algebraic constraint G. On the right–hand side we have the applied torques τ, the vector of non–linear forces (e.g. coriolis forces) N, and γ containing second derivatives of the algebraic constraint. Solving this system provides us with the joint accelerations $\ddot{q}$ and the ground reaction forces λ of the contact points. Source code for quantities M, G, N, γ were established using the HUMANS Toolbox [16].

The transition from single–support to double–support occurs when the Y–value of the right foot point is 0 and the foot point moves along the negative Y–axis. The collision of the foot point with the ground is assumed to be instantaneous and fully inelastic. The successive double–support phase ends when the vertical component of the ground–reaction force acting on the left foot vanishes.

The change of the generalized velocity vector $\dot{q}$ due to the collision is computed by solving the system:

$$\begin{pmatrix} M & G_{new}^T \\ G_{new} & 0 \end{pmatrix} \begin{pmatrix} \dot{q}_c^+ \\ \Lambda \end{pmatrix} = \begin{pmatrix} M\dot{q}_t^- \\ 0 \end{pmatrix}. \tag{2}$$

where G_{new} describes the new constraint Jacobian of the double support constraint, $\dot{q}^-$ the generalized velocity right before the collision, $\dot{q}^+$ the generalized velocity right after the collision and Λ the impulsive force in cartesian coordinates that act at the contact points.

We do not include muscles in the model, but instead use torques to power all internal DOF. In addition, we insert linear spring damper elements to mimic the compliance and damping properties of muscles, ligaments and passive tissues. The parameters of these spring-damper elements are also left free for the optimization.

3 Optimal Control Problem of a Human Gait

By using optimal control methods we can simultaneously optimize the motion $x(t) = [q(t), \dot{q}(t)]^T$ which consists of the positional variables $q(t) \in \mathbb{R}^{n_{dof}}$ and the velocities $\dot{q}(t) \in \mathbb{R}^{n_{dof}}$ of the generalized coordinates of our model. The torques at the actuated joints are described by $u(t) \in \mathbb{R}^{n_u}$, with $n_u = 25$. Additional model parameters, such as spring damper constants or step length and velocity, are in the vector p. As we have two model equations (one for single– and one for double–support) we have a two–phase optimal control problem.

The complete optimal control problem can be written for $i = 1, 2$ as:

$$\min_{x(\cdot), u(\cdot), p, t_i} \int_0^{t_f} \Phi_L(t, x(t), u(t), p) dt + \Phi_M(t_f) \tag{3}$$

subject to:

$$\dot{x}(t) = f_i(t, x(t), u(t), p) \tag{4}$$
$$x(t_{i+1}^+) = h_i(x(t_{i+1}^-), p), \tag{5}$$
$$g_i(t, x(t), u(t), p) \geq 0, \tag{6}$$
$$r^{eq}(x(0), \ldots, x(t_f), p) = 0, \tag{7}$$
$$r^{ineq}(x(0), \ldots, x(t_f), p) \geq 0. \tag{8}$$

The Lagrange term Φ_L in the objective function (3) is of the form:

$$\Phi_L(x(t), u(t), p) = c_u ||Wu(t)||_2^2 + c_h ||v_{head}(t)||_2^2 \tag{9}$$

which minimizes both the torques $u(t)$ applied to the system and the motion of the head $v_{head}(t)$. The Mayer term $\Phi_M(t_f) = c_m ||\Delta p_{td}(t_f)||_2^2$ contains the impulse at the foot on touch-down. The weight matrix W as well as the constants $c_u = 1.0$, $c_h = 5.0$, and $c_m = 0.1$ are used to scale the objective function components, taking into account their different dimensions and different strengths of joint actuators. The weighting coefficients of matrix W are shown in Table 1.

Single– and double–support phases are described by (4) as distinct model equations and the transition between them is modeled by the phase transition function

Table 1 Weighting coefficients of torque contributions in the Lagrange term Φ_L. The same weighting factors were used for joints of the left and right side of the model.

W_{HipZ}	74.2	$W_{UpperTrunkZ}$	1.6	W_{HeadZ}	2.2
W_{HipY}	8.6	$W_{UpperTrunkY}$	1.3	W_{HeadY}	0.8
W_{HipX}	74.1	$W_{UpperTrunkX}$	1.7	W_{HeadX}	2.1
W_{KneeZ}	74.2	$W_{ShoulderZ}$	3.3		
$W_{MiddleTrunkZ}$	4.1	$W_{ShoulderY}$	0.7		
$W_{MiddleTrunkY}$	1.5	$W_{ShoulderX}$	8.0		
$W_{MiddleTrunkX}$	60.8	W_{ElbowZ}	4.0		

(5). The touch–down is handled instantaneously by calculating and using an impulse applied at the contact point which is then propagated through the multi–body system in function (5). General state and control boundaries such as joint and torque limits are described by (6). Posture conditions (e.g. foot positions), periodicity at given time points and phase switches are modeled by (7) and (8).

The resulting multi–phase optimal control problem was solved by using the software package MUSCOD-II [8] which uses a *direct multiple–shooting method*.

It solves the optimization problem *directly*, which means that the controls in the continuous formulation (3)-(8) are approximated by a finite dimensional discretization. In our case the time–horizon $[t_0, t_f]$ was divided into N equidistant intervals with:

$$t_0 < t_1 < \cdots < t_{N\text{-}1} < t_N = t_f.$$

On each interval $I_j = [t_{j-1}, t_j]$ we define finite dimensional base functions $\varphi(t, w_j)$ with parameters $w_j \in \mathbb{R}^{n_w}$ such that the controls can be written as: $u(t) = \varphi(t, w_j)$ for all I_j. Thus the controls are solely discretized by w_j with $j = 0, \ldots, N-1$. The dimension of the control parameters n_w depends on the type of base function, e.g. for piecewise constant functions n_w would be n_u, for piecewise linear base functions we would have: $n_w = 2n_u$.

For the state parametrization it uses a multiple shooting method. Similar as in the previous paragraph the time horizon gets split up into M equidistant so–called *multiple shooting intervals* for which the interval boundaries are called *multiple shooting nodes*. On each node k at time t_k the value of the state is described by $s_k = x(t_k)$, $k = 0, \ldots, M$. With this we can define M initial value problems

$$\dot{x}(t) = f_i(t, x(t), u(t), p), \quad t \in [t_k, t_{k+1}]$$
$$x(t_k) = s_k$$

for $k = 0, \ldots, M-1$, of which the numerical solutions are denoted by $x(t; s_k)$. By adding continuity conditions $x(t_{k+1}; s_k) - s_{k+1} = 0$, the original boundary value problem is then replaced by M initial value problems together with continuity conditions. This allows us to discretize the trajectories of the states with s_k for $k = 0, \ldots, M$.

Having now parameterized both $u(t)$ and $x(t)$ by w_j and s_k we can now define a vector

$$y = [w_0, \ldots, w_{N-1}, s_0, s_1, \ldots, s_M, p, t_i]^T$$

which contains discretized values together with the other optimization variables of the continuous optimal control problem. Furthermore, by imposing the constraints
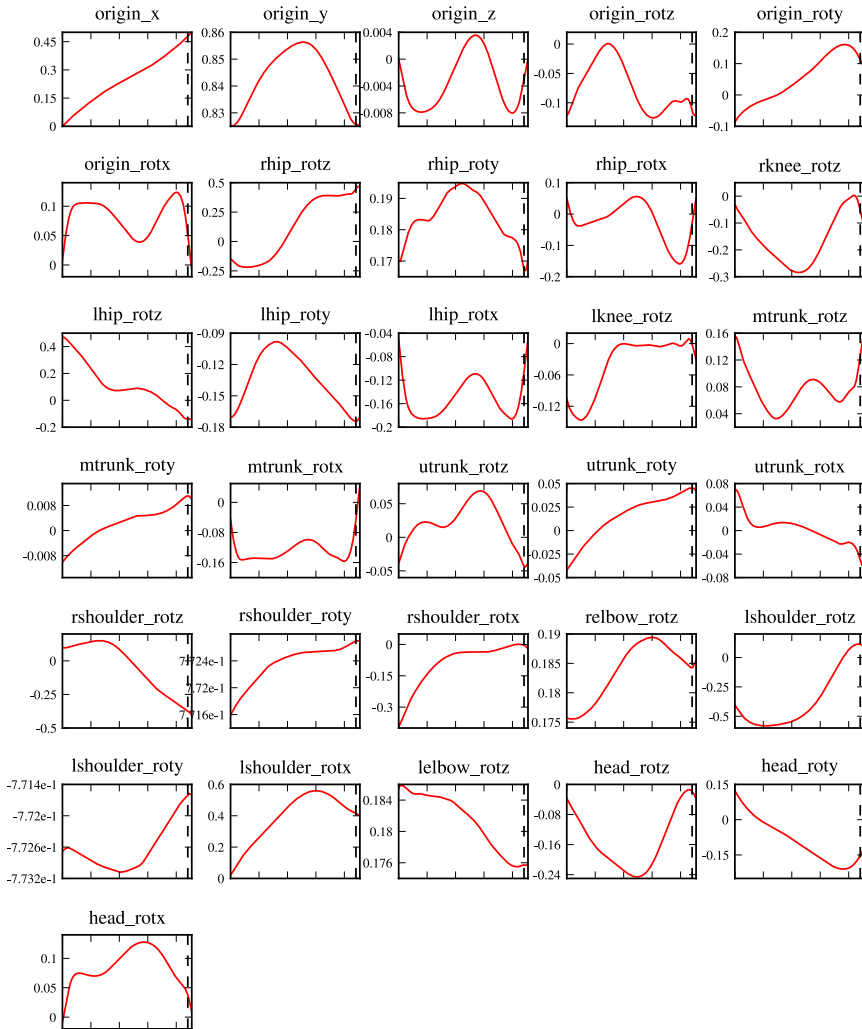
Fig. 2 Trajectories of all states of the optimized gait. The first three trajectories in the uppermost row are the positions of the hip center in m, followed by the hip rotation in radians. The other plots are the joints in radians. The dashed vertical line marks the phase switch between single– and double–support phase.

$(6) - (8)$ only on the multiple shooting nodes, we can formulate a nonliner optimization problem over the variables in the vector y:

$$\min_{y} F(y) \tag{10}$$

subject to:

$$g(y) \geq 0 \tag{11}$$
$$h(y) = 0, \tag{12}$$

which is the discretized version of our original optimal control problem.

This problem could then be solved by using a general purpose sequential quadratic programming (SQP) solver. However, due to the fact that the values w_j and s_k
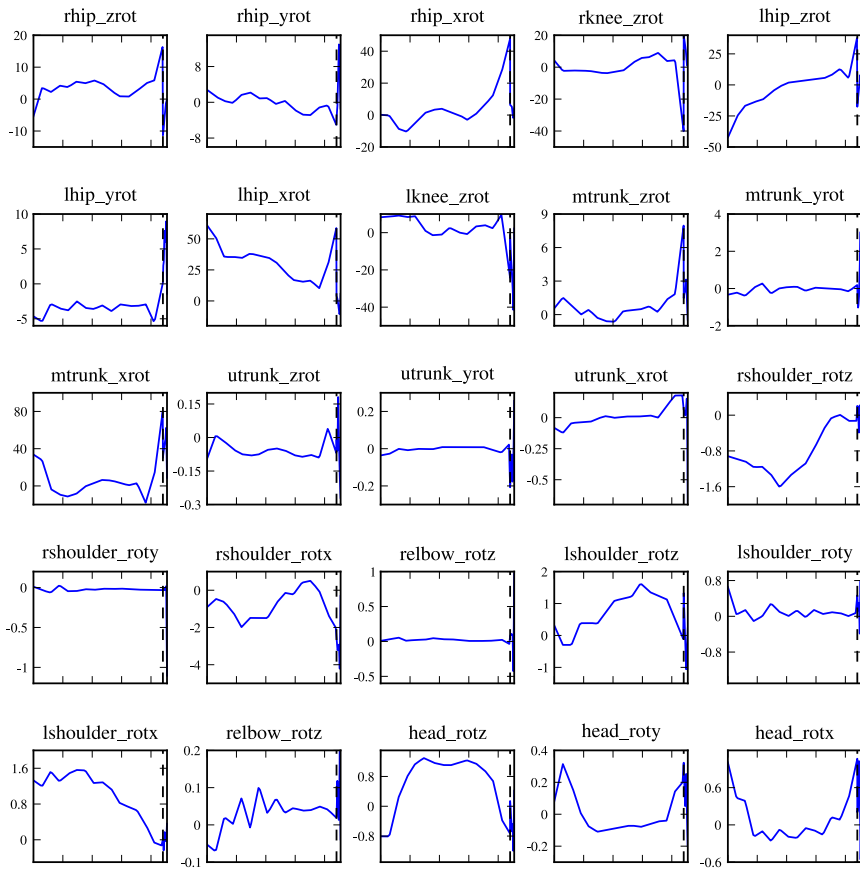
Fig. 3 Torque profiles at the actuated joints in *Nm*. Please note the different scalings of the plots.

only have local influence, we have specially structured matrices in the underlying quadratic problems which are heavily exploited by MUSCOD-II.

In our problem we used $N = M = 20$ and piecewise linear functions as base functions for the control discretization. The discretized nonlinear optimization problem has 2450 variables, 1842 equality and 4951 inequality constraints. The dynamics of our model is integrated with an integrator tolerance of 10^{-5}.

4 Numerical Results and Visualization

A plot of the generated trajectories of the position variables $q(t)$ and torques $u(t)$ are shown in Figure 2 and 3 respectively. A visualization can be found in Figure 4.

There is a natural arm swinging opposite to the leg swing. For the upper body mainly the shoulder joints are actuated, their actuation is however far less compared to the actuation in the joints of the lower body. In the upper body the shoulder abduction and adduction (X–rotations) are the strongest contributors as they cause weight transference from one leg to the other. The double-support phase is only 0.018 seconds. This is about 4% of the the step cycle, which has a duration of 0.44 seconds. Also, the knee of the swing leg bends less compared to the gait of a real human. In a human gait, the Y–position of the ankle is increased when the foot is
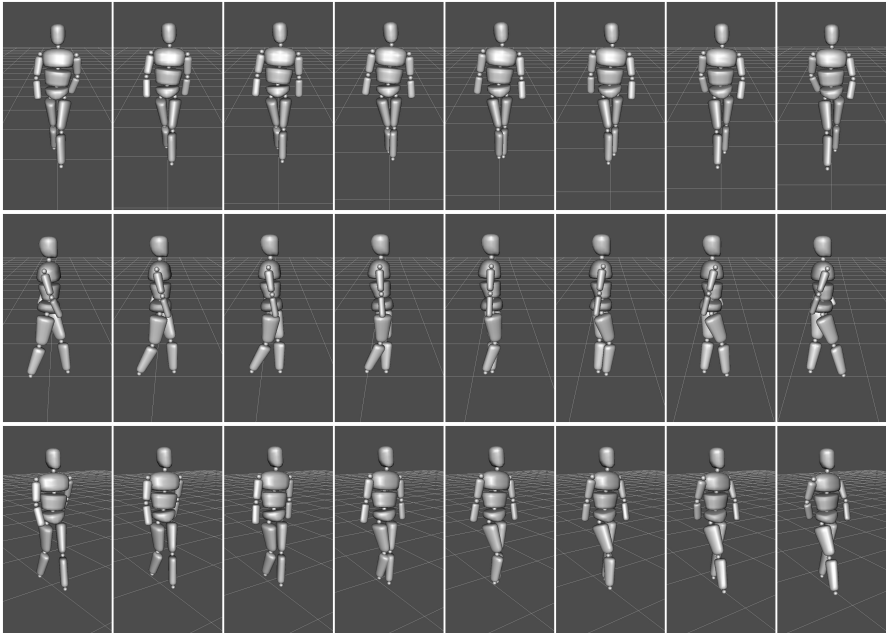
Fig. 4 Visualization of the generated gait cycle from the front, the side, and in perspective view

rolling about the toes, which is compensated by flexion of the knee. However, this elevation of the ankle does not occur with point feet.

The visualization we created can visualize the generated motion using simple stick models and also using sophisticated polygonal models that are manipulated by a skeleton that represents the kinematic structure of our multi–body system. This is achieved by using the animation library CAL3D [3]. Figure 4 shows the generated gait by using a polygonal model from our visualization. Custom models and skeletons can be created by using the open–source 3D content creation suite Blender [1], which allows both modeling of the skeleton and the polygonal mesh. Additionally, our visualization allows the motion to be described by either *ZYX*–Euler angles or Quaternions.

5 Summary of Results and Outlook

We were able to generate a physically valid gait by using optimal control methods as presented in this paper. The generated gait looks natural for our current model topology and the method does not depend on motion capture data. Even though we considered a symmetric gait, this method can also be used for non-symmetric gaits, such as a curved path. Also the model parameters can easily be adjusted to generate pathological gaits (e.g. limited joint movement, carrying a heavy backpack).

One of the biggest challenges currently is the modeling of the foot. So far we have used point feet but for a more sophisticated foot model would allow a more realistic ground contact. We expect this to lead to an automatic bending of the knee at lift-off, which improves the realism of the motion. Also adding a model for muscles would be interesting and would allow formulation of optimization criteria concerning their actuation patterns instead of raw joint torques.

The gait generated by using optimization depends heavily on the constraints and the objective function. Different styles of walking could be obtained by using objective functions other than minimization of energy and head movement. Especially identifying specific objective functions for emotional walking styles would be of interest.

We would also like to apply this method to computer graphics since the presented approach could also be applied to a variety of models and, depending on the objective function, generate different walking styles. The resulting motion would be physically valid and therefore realistic. Moreover, instead of manually creating keyframes between two postures (e.g. sitting and standing), generating an optimal motion could automatically create appealing animations. The method used in this paper can also be applied in robotics to generate a gait that uses the ZMP by adding a constraint to the optimal control problem.

References

1. Blender Foundation: Blender - The Free Open Source 3D Content Creation Suite (2009), `http://www.blender.org`
2. Delp, S.L., Anderson, F.C., Arnold, A.S., Loan, P., Habib, A., John, C.T., Guendelman, E., Thelen, D.G.: Opensim: Open-source software to create and analyze dynamic simulations of movement. IEEE Transactions on Biomedical Engineering 54(11), 1940–1950 (2007)
3. Desmecht, L., Dachary, L., Heidelberger, B.: Cal3D – 3D Character Animation Library (2009), `http://gna.org/projects/cal3d/`
4. Geyer, H., Seyfarth, A., Blickhan, R.: Compliant leg behaviour explains basic dynamics of walking and running. Proceedings of the Royal Society B: Biological Sciences 273(1603), 2861–2867 (2006), `http://rspb.royalsocietypublishing.org/content/273/1603/2861.abstract`, doi:10.1098/rspb.2006.3637
5. Hodgins, J.K., Wooten, W.L., Brogan, D.C., O'Brien, J.F.: Animating human athletics. In: SIGGRAPH 1995: Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques, pp. 71–78. ACM, New York (1995), `http://doi.acm.org/10.1145/218380.218414`
6. Kajita, S., Kanehiro, F., Kaneko, K., Fujiwara, K., Harada, K., Yokoi, K., Hirukawa, H.: Biped walking pattern generation by using preview control of zero-moment point. In: Proceedings of the IEEE International Conference on Robotics and Automation, pp. 1620–1626 (2003)
7. Kim, J.Y., Park, I.W., Oh, J.H.: Experimental realization of dynamic walking of the biped humanoid robot khr-2 using zero moment point feedback and inertial measurement. Advanced Robotics 20, 707–736 (2006), `http://www.ingentaconnect.com/content/vsp/arb/2006/00000020/00000006/art00005`, doi:10.1163/156855306777361622
8. Leineweber, D., Bauer, I., Bock, H., Schlöder, J.: An efficient multiple shooting based reduced SQP strategy for large-scale dynamic process optimization. Part I: Theoretical aspects. Computers and Chemical Engineering 27, 157–166 (2003)
9. de Leva, P.: Adjustments to zatsiorsky-seluyanov's segment inertia parameters. Journal of Biomechanics 29 (9), 1223–1230 (1996)
10. Mombaur, K.: Using optimization to create self-stable human-like running. Robotica 27, 321–330 (2009, published online June 2008)
11. Nakamura, Y., Mori, T., Aki Sato, M., Ishii, S.: Reinforcement learning for a biped robot based on a cpg-actor-critic method. Neural Networks 20(6), 723–735 (2007), `http://www.sciencedirect.com/science/article/B6T08-4N3GFGM-/2/e8817c699dbf751ad76e131d778a598a`, doi:10.1016/j.neunet.2007.01.002
12. Ren, C., Zhao, L., Safonova, A.: Human motion synthesis with optimization-based graphs. In: Computer Graphics Forum (In Proc. of Eurographics 2010, Sweden), vol. 29 (2010)
13. Safonova, A., Hodgins, J.K.: Construction and optimal search of interpolated motion graphs. In: SIGGRAPH 2007: ACM SIGGRAPH 2007 Papers, p. 106. ACM, New York (2007), `http://doi.acm.org/10.1145/1275808.1276510`
14. Schultz, G., Mombaur, K.: Modeling and optimal control of human-like running. To appear in IEEE/ASME Transactions on Mechatronics (2010)

15. Vukobratovic, M.: Zero-moment point – thirty five years of its life. International Journal of Humanoid Robotics 1, 157–173 (2004)
16. Wieber, P.B., Billet, F., Boissieux, L., Pissard-Gibollet, R.: The humans toolbox, a homogeneous framework for motion capture, analysis and simulation. Tech. rep., INRIA Rhone-Alpes (2006)
17. Witkin, A., Kass, M.: Spacetime constraints. In: SIGGRAPH 1988: Proceedings of the 15th Annual Conference on Computer Graphics and Interactive Techniques, pp. 159–168. ACM, New York (1988),
    doi: `http://doi.acm.org/10.1145/54852.378507`
18. Yin, K., Loken, K., van de Panne, M.: Simbicon: Simple biped locomotion control. ACM Trans. Graph. 26(3), Article 105 (2007)

Motion Generation with Geodesic Paths on Learnt Skill Manifolds*

Ioannis Havoutis and Subramanian Ramamoorthy

Abstract. We present a framework for generating motions drawn from parametrized classes of motions and in response to goals chosen arbitrarily from a set. Our framework is based on learning a manifold representation of possible trajectories, from a set of example trajectories that are generated by a (computationally expensive) process of optimization. We show that these examples can be utilized to learn a manifold on which all feasible trajectories corresponding to a skill are the geodesics. This manifold is learned by inferring the local tangent spaces from data. Our main result is that this process allows us to define a flexible and computationally efficient motion generation procedure that comes close to the much more expensive computational optimization procedure in terms of accuracy while taking a small fraction of the time to perform a similar computation.

1 Introduction

Humanoid robots provide a flexible platform for a variety of tasks including rough terrain locomotion and dexterous manipulation. Typically, this flexibility also carries the burden of increased complexity that adversely impacts the practical usability of such systems. For example, if a humanoid robot were tasked with locomotion on an uneven terrain - requiring the ability to continually vary foot placement positions in response to external events, it is hard to define a suitable motion generation strategy for two reasons. Firstly, ensuring that the motion satisfies all requirements ranging

Ioannis Havoutis

Department of Advanced Robotics, Istituto Italiano di Tecnologia, Genova
e-mail: I.Havoutis@iit.it

Subramanian Ramamoorthy

Institute of Perception, Action and Behaviour, School of Informatics,
University of Edinburgh
e-mail: s.ramamoorthy@ed.ac.uk

* An extended version of this paper appears in [4].

from high-level planning goals to intermediate stability constraints and lower level actuator of joint constraints is a hard computational problem. Even if we had the computational resources, it can be hard to actually specify all of these requirements in a well posed analytical formulation. One way to approach such problems is by learning from demonstration trajectories. Here, the problem is to infer the *continuum* of trajectories in the solution space corresponding to a specific skill from a sparse set of demonstrated solutions. Additionally, we must represent this set in a way that allows generation of new motions, directed to previously unseen goals, that are consistent with prior experience.

While there are a number of different state of the art techniques for learning by demonstration algorithms, they all share some weaknesses with respect to this specific goal. Many existing methods focus on reproduction of patterns of movement for an end effector [1], without direct consideration of joint-space motion either to address constraints or exploit additional flexibility, or they focus on tasks where it is acceptable to define independent joint-level trajectories that can be simultaneously used in parallel [5]. While these are good for, say, reproducing human motions, they may not be well suited to the needs of a flexible motion generator in an autonomous system that must be deployed in a continually changing world.

Our approach represents each skill as a manifold that is embedded in the robot's joint space. This manifold represents the set of all possible solutions to a skill and it is inferred from a few example solutions to corresponding optimization problems (or, if available, human demonstrations). When presented with a planning query we can generate a path that is within this set, generated by computing the geodesic path over the manifold.

In this paper, in order to illustrate the behaviour of the algorithm, we utilize example trajectories that are obtained from a computational method which involves numerical optimization. These solutions are computationally expensive and not feasible for online operation. However, they can serve the same role as demonstration data. With this, we have a clear idea of the specific properties of each task being considered, and a measure of algorithm performance against reasonable 'ground truth'.

2   Learning for Motion Synthesis

In the usual formulation, manifold learning is aimed at finding an embedding or 'unrolling' of a nonlinear manifold onto a lower dimensional space while preserving metric properties such as inter-point distances. Popular examples include MDS [3], LLE [7] and ISOMAP [8]. However, much of this work has been focused on summarization, visualization or analysis that explains some aspect of the observed data.

On the other hand, we are interested in preserving properties of trajectories in the data set. So, formally our goal is to learn a model of the tangent space of the low-dimensional nonlinear manifold, conditioned on the adjacency relations of the high dimensional data. Such a learnt manifold model can then be used to compute

geodesic distances, to find projections of points on the manifold and to directly generate geodesic *paths* between points.

2.1 Learning the Manifold

Our nonlinear manifold learning algorithm is based on Locally Smooth Manifold Learning by Dollar et al. [2], which we have adapted with robot motion specific issues in mind. In particular we have replaced the neighbourhood graph creation process with a procedure that considers task space distances as well as ensuring that temporal neighbourhood relations along the demonstrated trajectories are respected, similar to the procedure used in ST-ISOMAP [6].

Given that our D-dimensional data lies on a locally smooth d-dimensional manifold in D-dimensional space, where $d < D$, there exists a continuous bijective mapping $\mathcal{M}$ that converts low dimensional points $y \in \mathbb{R}^d$ from the manifold, to points $x \in \mathbb{R}^D$ of the high dimensional space, $x = \mathcal{M}(y)$. The goal is to learn a mapping from a point on the manifold to its tangent basis $\mathcal{H}(x)$,

$$\mathcal{H} : x \in \mathbb{R}^D \mapsto \left[\frac{\partial}{\partial y_1} \mathcal{M}(y) \cdots \frac{\partial}{\partial y_d} \mathcal{M}(y) \right] \in \mathbb{R}^{D \times d}$$

where each column of $\mathcal{H}(x)$ is a basis vector of the tangent space of the manifold at y, i.e. the partial derivative of $\mathcal{M}$ with respect to y.

We then learn a model of the mapping with parametrization θ, i.e. $\mathcal{H}_\theta$, based on the generalized neighbourhood relations of the data, N, and the centred estimate of the directional derivative between two neighbours, $\Delta^i_{\cdot j}$. The model is trained by minimizing the error function:

$$err(\theta) = \min_{\{\varepsilon^{ij}\}} \sum_{i,j \in N^i} \left\| \mathcal{H}_\theta(\vec{x}^{ij}) \varepsilon^{ij} - \Delta^i_{\cdot j} \right\|^2_2,$$

where ε^{ij} is an unknown alignment factor and N^i is the set of neighbours of x^i.

Solving for the bases and their alignment simultaneously is complex, but if either one is kept constant, solving for the remaining variables becomes a tractable least squares problem. Optimizing the model requires alternating between these two least squares problems, until a local minima has been reached. Typically more than one random restart is performed to avoid local minima [4].

2.2 Optimal Geodesic Paths

By approximating the tangent space of the manifold, we gain access to a variety of geometric operations. Central to our robotics aims is the ability to compute *geodesic paths*; paths that lie on the low dimensional manifold. In this spirit, we now change our notation of points from x to q, to denote poses a robot can achieve in a configuration space.

| (a) Task space | (b) NN graph | (c) Joint space | (d) Tangent space |
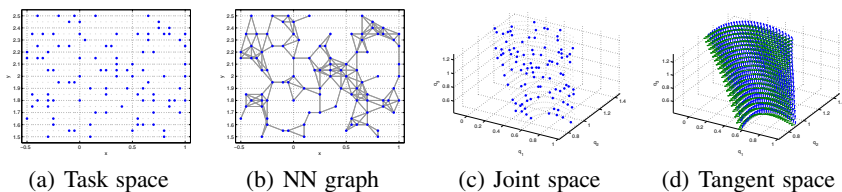
Fig. 1 Learning the optimality manifold of a 3-link arm. (a) The planar task space of the arm and subsampled points (blue) used for leaning. (b) The neighbourhood graph used for learning a manifold. (c) The optimality manifold that we wish to learn. Light gray points are not used for learning but are plotted to give a better estimate of the geometry of the manifold. Note that the manifold is not planar but twist and turns as we move down the q_3 axis. (d) The learnt tangent space model. Blue and green arrows are basis vectors evaluated at points that correspond to the original grid.

Our goal is to find the shortest path between two specified poses q_{start} and $q_{end} \in \mathbb{R}^D$, D being the dimensionality of the configuration space, that respects the geometry of the learnt manifold. In a robotics context, being on the manifold essentially means that the constraints (e.g., optimality w.r.t. a particular task-specific cost) inherent in the training data are satisfied. In practice, we discretize our path into a set of n via points, $\mathbf{q} = q_{start}, \ldots, q_{end}$, with q_{start} and q_{end} being fixed, and we follow a combination of gradient descent steps to minimize the length of the path while not leaving the support of the manifold.

We first initialize a path by linearly interpolating between q_{start} and q_{end}, while following the geometry of the manifold, until the distance between consecutive points is acceptable. With the learnt tangent space we iteratively compute a minimum energy solution that makes q^is "stick" to the manifold and minimizes the length of the path without leaving the support of the manifold. The former is accomplished by following the orthonormal (to the manifold) component of the gradient of

$$err_{\mathscr{M}}(\mathbf{q}) = \min_{\{\varepsilon^{ij}\}} \sum_{i,j \in N^i} \left\| \mathscr{H}_\theta(\bar{q}^{ij})\varepsilon^{ij} - (q^i - q^j) \right\|_2^2,$$

and the latter by following the parallel (to the manifold) component of

$$err_{length}(\mathbf{q}) = \sum_{i=2}^{n} \left\| q^i - q^{i-1} \right\|_2^2,$$

while keeping the endpoints fixed.

The next sections present two examples of our method. The main thrust of our argument here is that the manifold representation provides a concise encoding of all motions corresponding to a skill. This encoding is equivalent to a computationally more expensive optimization process, but requires a fraction of the computational effort. We demonstrate this by first presenting a 3-dim motion problem, where the manifold can be easily visualized and the algorithm intuitively understood. Then, we show a more complex example involving a humanoid robot.

3 Experiments on a Robotic Arm

The planar 3-link arm is a series of three rigid links of unit length that are coupled with hinge joints, producing a redundant system with 3 degrees of freedom that is constrained to move on a 2 dimensional plane (task space).

The skill that we learn in this setting is the set of all solutions to a specific redundancy resolution scheme. Here, we choose the joint space configuration, $\mathbf{q}$, that minimizes the distance to a convenience (robot default or minimum strain) pose, $\mathbf{q}_c$. Formally, $\min \|\mathbf{q} - \mathbf{q_c}\|^2$, *subject to* $f(\mathbf{q}) - \mathbf{x} = 0$, where f is the forward kinematics and $\mathbf{x}$ is the goal endpoint position on the plane. The points trace a smooth nonlinear manifold in joint space (Fig. 1(c)). Note that the manifold is not planar but lies on a convex strip that twists clockwise and tightens as we travel down the q_3 axis. Also, different redundancy resolution strategies would produce different optimality manifolds. In general, this kind of information may not be explicitly known (in the case of human demonstration) or visualizable for more complex problems.

We collect data (joint space points) from a grid in task space and subsample 100 points as our training set (Fig. 1(a)). We compute the neighbourhood graph from the task space distances and learn a model of $\mathcal{H}_\theta$ with 10 RBF's and 100 points, the blue points in Fig. 1(c). We can subsequently evaluate $\mathcal{H}_\theta$ at any point in our joint space. Fig. 1(d) shows the tangent bases evaluated at every point of the previously generated grid. Note that the basis vectors are aligned and vary smoothly, i.e. we obtain a good generalization within the region of support of the data.

3.1 Evaluation

We evaluate the accuracy of the approximation that the learnt manifold provides in two generalization settings. One measures the interpolation ability, where we compare against ground truth data within the region of support of the training data, and the second demonstrates the extrapolation ability, where we compare what our model generates outside the region of support of the training data. We also record the time needed to produce the trajectories. In both cases we compare 50 trajectories with random start and end points that are produced with geodesic paths on the learnt manifold, against what the numerical optimization produces for the same goals. Samples of such paths for both generalization cases are depicted in Fig. 2(a) and (b) (grid points in light gray for comparison).

We compute the *RMSE*, for each trial and for each case, between ground truth and prediction of model, for a total of 10 trials. The averaged errors are depicted in Fig. 2(c). Note that the *RMSE* axis is in log-scale while the difference of the two bars is of 2 orders of magnitude. To be precise, the average *RMSE* for paths generated within the region of support of the data is $1.8935 \times 10^{-4} \pm 3.6013 \times 10^{-5}$ (practically zero), while beyond the support of the data the average *RMSE* is $6.84 \times 10^{-2} \pm 2.19 \times 10^{-2}$. In addition, computing the optimal geodesic paths takes less time on average (Fig. 2(d) in both cases).

(a) Interpolation   (b) Extrapolation   (c) Generalization   (d) Time
                                               errors
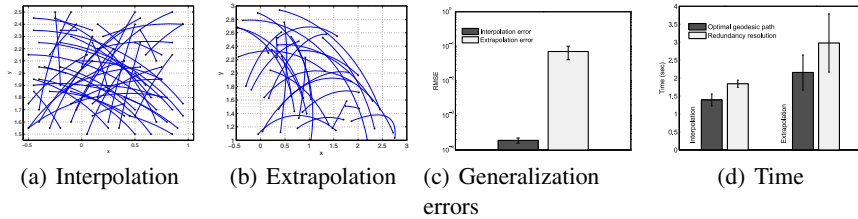
Fig. 2 Results of the 3-link arm experiments. Novel task space trajectories produced with random start and end points where (a) demonstrates generalization within the region of support of the data, while (b) demonstrates generalization beyond the region of support of the training data. (c) RMSE error of generated trajectories against ground truth for the two cases. In the interpolation scenario the error is practically zero (y axis in log-scale). (d) Absolute planning time for the two cases. Note that in the interpolation case the length of the paths is consistently low.

4   Experiments on a Humanoid Robot

To demonstrate the scalability of our approach we also present an example involving a humanoid robot platform. We use the *KHR-1HV* (Fig. 3(a)), that stands approximately 35cm tall and has 19 DoFs. We focus on the task of walking, with the aim of learning the manifold of quasi-static stepping trajectories for random

(a) Robot   (b) Right steps   (c) Left steps

(d) Model   (e) Right neighborhood graph   (f) Left neighborhood graph
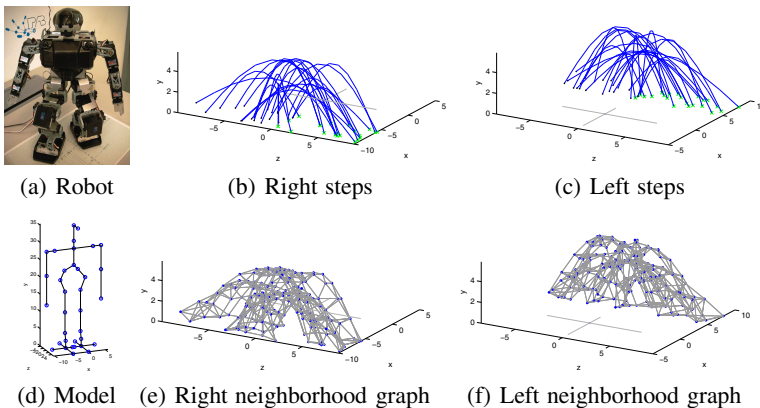
Fig. 3 The *KHR-1HV* humanoid robot used, (d) skeleton model and (a) physical robot. *Task space* representation of the training data through forward kinematics. Random start and end point leg swing trajectories of the left (b) and right (c) legs. (e) and (f) the neighbourhood graphs that result from the task space distances between demonstrated data (units in *cm*). This provides the task-specific distance metric for the high dimensional *joint-space*. Note that depicted here are only feet midpoint positions while the datasets consist of the joint space points that are 19-dimensional.

(a) Generated left steps (b) Generated right steps (c) Error (d) Absolute time
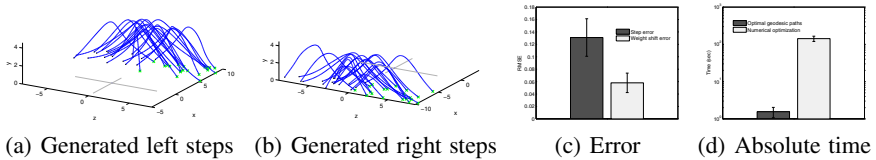
Fig. 4 Experimental results with the humanoid robot. Random start and end point trajectories for left (b) and right (b) leg swings that have been generated from our learnt manifold, via geodesic path optimization (units in *cm*). (c) *RMSE* (*degrees*) of generated data against ground truth. (d) absolute time needed for planning and optimization with our method and the nonlinear optimization method (*y* axis in log-scale) described in the text.

foothold placements, within a reasonably large step interval. We generate data with an unconstrained nonlinear optimization method that uses a hand-crafted cost function. Formally, the optimization problem is of the form $\min_q \mathscr{J}(\mathbf{q})$, *subject to* $f(\mathbf{q}) - \mathbf{x} = 0$, where $\mathscr{J}$ is the cost function, f is the forward kinematics and $\mathbf{x}$ is a goal task space position. The cost function is a mixture of task constraints and stability constraints.

We collected 20 full body joint space trajectories where start and goal points of every step have been randomized within a reasonable reaching distance (Fig. 3(b) and 3(c)). We separated each footstep to a swing phase and a weight shift phase. This way we divided the learning into two components, leg swing manifold and support weight shift manifold, as the measure of optimality is essentially different for each phase. We compute a neighbourhood graph (Fig 3(f) and 3(e)) and learn a manifold for each stepping phase. We set the dimensionality of the manifolds to be 3, being the simplest model that yields a low error.

4.1 Evaluation

The learnt manifolds are able to produce smooth walking trajectories that satisfy the optimization criteria used to produce the training data. Specifically, the average *RMSE* (*degrees*) of the leg swing manifold for the ground truth was as low as 0.12 while the average *RMSE* of the weight shift manifold ranged on average near 0.06 (Fig. 4(c)). This implies that the geometry of the step manifold is more complex and some of its features might be smoothed over by the RBF model. Nonetheless the procedure was able to produce stable walking in the continuum of the reaching space of the robot as depicted in Fig. 4(a) and 4(b) for right and left swings accordingly.

The absolute time needed to generate an optimal geodesic path on the pair of manifolds (swing leg and weight shift) from random start to random end points was approximately 1.5552 ± 0.4785 seconds (in a standard, not particularly fine-tuned, numerical implementation of the algorithm) whereas generating a trajectory with the optimization procedure required approximately *two minutes* on average, an approximately 98% increase in speed. This is a *significant* decrease in absolute planning time, which makes it possible to deploy this algorithm in realistic application
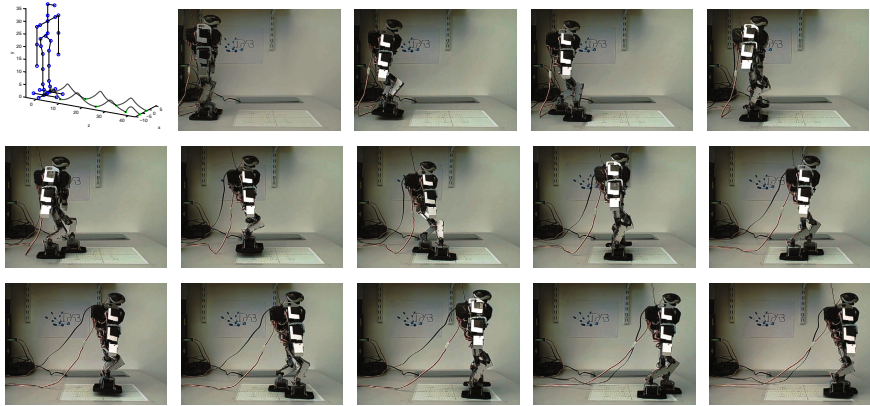
Fig. 5 Random walk generated by geodesic path optimization on the learnt manifolds for randomized task-space footholds and stills of the robot executing the planned motion

scenarios (e.g. RoboCup). A randomized walk sequence entirely generated with our method is depicted in Fig. 5, where foothold positions have been randomly generated and are previously unseen.

5 Conclusions and Future Work

We have demonstrated how a manifold representation can capture the flexibility required of a motion generation scheme operating in a continually changing environment. As used here, we have a computationally efficient procedure that can recover all of the solutions of a more expensive computational optimization procedure while also allowing for learning from data - where all requirements may not be easy to encode in an analytical framework. This work adds to the literature on learning by demonstration by addressing the cases where the task is more complex than simply reproducing specific task space trajectories and involves further kinodynamic requirements in the joint space, etc. We demonstrate this using a couple of robotics examples - a 3-link arm, where the results are easy to visualize, and a humanoid robot, where the stepping task is intuitively understood. Our long term goal is to utilize this procedure as part of a larger system that would be able to learn, plan and execute motions robustly and in real time.

References

[1] Calinon, S., D'halluin, F., Caldwell, D., Billard, A.: Handling of multiple constraints and motion alternatives in a robot programming by demonstration framework. In: IEEE-RAS International Conference on Humanoid Robots, Humanoids (2009)

[2] Dollár, P., Rabaud, V., Belongie, S.: Non-isometric manifold learning: Analysis and an algorithm. In: International Conference on Machine Learning, ICML (2007)

[3] Hastie, T., Tibshirani, R., Friedman, J.H.: The Elements of Statistical Learning. Springer (2001)

[4] Havoutis, I., Ramamoorthy, S.: Geodesic trajectory generation on learnt skill manifolds. In: Proceedings: IEEE International Conference on Robotics and Automation 2010, ICRA 2010 (2010)

[5] Ijspeert, A., Nakanishi, J., Schaal, S.: Trajectory formation for imitation with nonlinear dynamical systems. In: IEEE International Conference on Intelligent Robots and Systems (2001)

[6] Jenkins, O.C., Mataric, M.J.: A spatio-temporal extension to isomap nonlinear dimension reduction. In: International Conference on Machine Learning, ICML (2004)

[7] Roweis, S.T., Saul, L.K.: Nonlinear dimensionality reduction by locally linear embedding. Science 290(5500), 2323–2326 (2000)

[8] Tenenbaum, J.B., de Silva, V., Langford, J.C.: A global geometric framework for nonlinear dimensionality reduction. Science 290(5500), 2319–2323 (2000)

Online CPG-Based Gait Monitoring and Optimal Control of the Ankle Joint for Assisted Walking in Hemiplegic Subjects

Rodolphe Héliot, Katja Mombaur, and Christine Azevedo-Coste

Abstract. The paper introduces an approach to the FES-assisted correction of the drop-foot syndrome in post-stroke hemiplegic patients. The approach is based on a two stage architecture. One stage is dedicated to the online estimation of high-level gait information and the second to the generation of optimal ankle joint trajectories for walking assistance. The general gait information is obtained through the observation of one limb based on a central pattern generator model generating rhythmic trajectories which auto-adapt to real-measurements. This allows us to obtain information about the execution of the walking cycle. Optimal control is used to generate ankle joint dorsi-flexion trajectories during the swing phase of the corresponding deficient leg based on a muscle model and on the information provided by the first stage and some estimated or measured information about the controlled leg. This allows us to minimize a criteria linked to muscle activation, excitation or fatigue while satisfying constraints such as ground clearance, instead of just mimicking a priori chosen foot ankle trajectories which may be suboptimal. The strategy is validated in simulation using experimental data recorded in one healthy subject.

1 Introduction

Hemiplegia is a condition where one side of the body is paretic or paralyzed; it is usually the consequence of a cerebro-vascular accident (CVA). Many survivors to

Rodolphe Héliot
INRIA Rhône Alpes, 655 avenue de l'Europe, 38330 Montbonnot, France
CEA-LETI, Minatec Campus, 17 avenue des Martyrs, 38054 Grenoble, France
e-mail: rodolphe.heliot@cea.fr

Katja Mombaur
IWR, Heidelberg University, Im Neuenheimer Feld 368, 69120 Heidelberg, Germany;
Associate Researcher at LAAS-CNRS, Toulouse, France
e-mail: kmombaur@uni-hd.de

Christine Azevedo-Coste
DEMAR INRIA/LIRMM, 161 rue Ada 34392 Montpellier, France
e-mail: azevedo@lirmm.fr

stroke recover a large amount of function by the natural neurological recovery occurring in the months following stroke. Nevertheless, a persistent, long term disability remains in approximatively 10 to 20% of these patients: the drop foot (DF) syndrome [18]. DF typically involves an inability to dorsiflex the foot during the swing phase of gait. The conventional approach to address this problem is the prescription of an ankle-foot orthosis (AFO). The most commonly used AFO in drop foot is constructed of polypropylene and inserted into a shoe, it provides rigid immobilization and impedes bearing of the foot when walking. The application of surface electrical stimulation (ES) to correct drop foot was initially proposed in 1961; ES is usually applied to the common peroneal nerve inducing an activation of the tibialis anterior (TA) muscle responsible for foot dorsiflexion ; most of the existing systems use a footswitch to synchronize the application of ES with the swing phase [16].

The use of functional electrical stimulation (FES) for the correction of hemiplegic drop foot is well established [7]. The method classically used in most DFS can be described as using a pre-programmed trapezoidal ES envelope so that ES intensity (or pulse width) is linearly ramped up to its maximum value from heel-off. ES intensity (or pulse width) is then kept constant until heel-strike, when it is ramped down to zero. One major issue with existing DFS is that the TA muscle activation is quite different from the one occurring naturally in healthy gait. In normal walking, TA presents two phases of activity: 1) the first phase of TA activity occurs at toe-off, contraction of TA results in foot-lift to provide foot clearance during swing, 2) this phase is followed by a more intense activity phase at loading response, when the TA is contracted to provide with braking action at heel strike [28, 19]. Another lack of usual DFS systems is the absence of adaptability of the stimulation patterns to cope with frequent gait modifications (rhythm, step length, stairs, muscle fatigue...). Discrete events extracted from footswitches or tilt sensors as proposed in commercial devices [25, 7] are not able to provide with sufficient information to continuously modify the stimulation pattern if needed.

Some authors have worked on the dynamic adjustment of the stimulus envelope in order to mimic natural TA contraction. Results have shown an increased dorsiflexion and a reduced delivered stimulation charge, which could diminish fatigue and lower power consumption [19]. However, this was obtained by positioning several sensors and bulky equipment, leading to poor acceptance by patients. Different approaches have been proposed to replace footswitches by more powerful sensors in order to distinguish between different phases of the gait cycle but they involve several sensors [26, 27, 20, 14, 13, 8] or are used for detection event mainly [9, 24]. Thus, designing an adaptive control system that will provide a patient-specific walking pattern with an easy-to-use device remains an open question.

In this paper, we address the drop foot correction problem by optimizing muscle activation in order to achieve a desired motion based on the observation of the movements of the valid leg. The controller only requires a single, small, attitude sensor. The walking cycle is monitored thanks to a model-based observer, based on a nonlinear oscillator that mimics natural Central Pattern Generator (CPG) control (section 2.1). High-level gait information is extracted online, that is used as input (see figure 1) to an optimization routine that derives in real-time adapted stimulation

patterns (section 2.2.2). Using a muscle model described in section 2.2.1, the optimization aims at reducing as much as possible muscle fatigue, while guaranteeing foot clearance over the ground. Preliminary simulation results are presented for gait cycle online monitoring and for optimal control aspects based on real experimental data.

2 Methods

We propose to use a "teleoperation" scheme (see figure 1) for rehabilitation applications, where the healthy parts of the body could be used to control the deficient limbs. For example, considering a hemiplegic patient, one could generate a trajectory for the deficient leg based on the movements of the healthy leg [2]. The deficient leg can then be controlled using FES [22, 10, 4]. Similarly controlling a prosthetic leg in an amputee person could be possible. We have already demonstrated the feasibility of generating cyclic patterns for one leg observing the contralateral leg [11, 3].

In this scheme, a movement observer continuously estimates gait high-level variables, such as gait phase and joint positions. These online estimations can then be used to generate a desired joint trajectory; tracking this trajectory through FES raises a non-trivial technical issue: one needs to invert the musculo-skeletelal model of the patient to derive from the desired joints trajectory the Electrical Stimulation commands. Practically, this model inversion has to be done through optimal control. Indeed, due to the redundancy and non-linearity of musculo-skeletelal models, classic PID controllers cannot be used to compute the commands. With optimal control one can take into account further constraints such as generating joint trajectories that are adapted to deficient limbs, for example taking into account muscle fatigue, stimulation efforts, ... Whatever the optimization criterions and constraints, the optimal control procedure has to run online. Hence there is a need for fast, efficient optimization routines.

This methods section will naturally be divided into two parts: we first describe the online estimation of high-level gait information using a sensor-modulated controller (Fig.1), and we then present the optimal control techniques that are required to compute muscle stimulations.
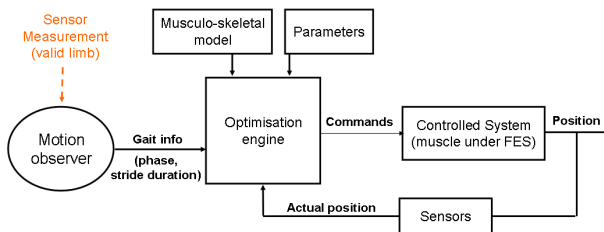
Fig. 1 Control architecture: high-level control variables are generated by the CPG; commands are then derived according to a model through optimal control

2.1   Online Estimation of High-Level Gait Information

A classical way of generating cyclic motion patterns for articulated systems is to synthesize a rhythm generator, mimicking the biological Central Pattern Generator (CPG) principles [23]. CPGs can generate trajectories either based on their intrinsic rhythm, or based on external inputs. As an example, it is possible to synchronize the walking movement of a biped robot with the leg motion of a human wearing a movement sensor [12]. In this type of applications, the CPG, modeled through an oscillator network, generates coordinated joint trajectories, which are then tracked through a low-level controller such as a PID corrector. In the following, we describe a method that allows to estimate online gait-related variables with a CPG that is synchronized with an external sensor. That method will then be used in the teleoperation scheme described above.

2.1.1   Framework

The basic assumption in the proposed approach is that considered motions (walking) reflect a cyclical or periodic activity: this means that all involved signals (sensor outputs, control variables, ...) can be described along a cycle. In order to assign to these variables a kind of relative position in the cycle, the phase φ can be introduced as a coordinate along the limit cycle [21], i.e as a variable which grows uniformly in the direction of the motion and gains 2π during each rotation, thus obeying the equation:

$$\frac{d\varphi}{dt} = \omega_0 \tag{1}$$

where $\omega_0 = 2\pi/T_0$ is the frequency of the oscillations.

It is possible to extend the definition of the phase to the vicinity of the limit cycle, demanding that the phase be constant on each isochrone. In this way, phase can be defined in the neighborhood of the limit cycle.

The goal of the following method is to estimate the phase of an observed signal (sensor input). To this aim, we build an oscillator that synchronizes with the sensory input, and estimates the oscillator phase.

2.1.2   Our Approach

To guarantee that the CPG we design will actually be synchronized on its input, we propose the following two-steps method (Fig. 2). It is based on observer theory [17], that provides mathematical tools to derive convergence and synchronization properties of the system:

1. build a system that models the sensor measurements
2. build an observer of this system, in which are injected the on going sensor measurements
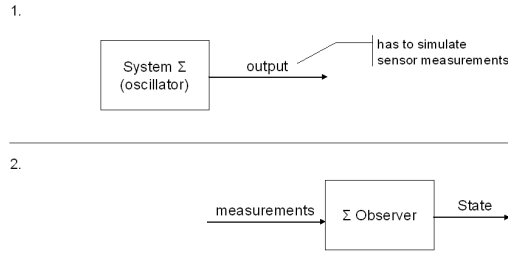
Fig. 2 Schematics of the method

Since we are interested in a cyclic movement (human gait), we chose to use a non-linear oscillator as the model for our sensor measurement. This approach, using an observer of a non-linear oscillator, presents as major advantage: the behavior of the observer can be proved to converge towards the given cyclic input. The synchronization is therefore ensured, and so is the observation of the phase ϕ.

2.1.3   Non-linear Oscillator Model

Let us now consider the modeling under the form of an oscillator of the time evolution of the thigh inclination in standard human gait. From a mechanical point of view, the motion of the human body restricted to a tree-form kinematic structure with variable unilateral ground contacts can be modeled using a Newton-Euler or Lagrangian approach, leading to a nonlinear second order system. From a biomechanical point of view, the standard steady-state human walking corresponds to minimal metabolic energy consumption and is naturally attained after some transient steps. All these facts are reflected at the joint level, therefore in the thigh angle itself and its measurement. It finally appears that searching for an oscillator of second-order type and exhibiting a limit cycle is a natural way of modeling the steady-state behavior of a human leg link measured with an adequate sensor.

As detailed in [12], we have therefore chosen a Van der Pol equation as a model of the system, the damping term of which has been modified to take into account asymmetrical patterns:

$$\ddot{x} - \mu(1 - bx - x^2)\dot{x} + \omega_0^2 x = 0 \tag{2}$$

2.1.4   Observer Design

The modified van der Pol equation (2) can be rewritten as:

$$\Sigma : \begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = \mu(1 - bx_1 - x_1^2)x_2 - \omega_0^2 x_1 \\ y = x_1 \end{cases} \tag{3}$$

Once the three parameters of the oscillator are identified through an optimization procedure that gives the best fitting of the oscillator limit cycle with respect to a set of recorded measurements, a dedicated nonlinear observer can be built under the form:

$$\Sigma' : \begin{cases} \dot{z} = -z + (k_1 - \omega_0^2)y + k_2 y^2 + k_3 y^3 \\ \hat{x}_1 = y \\ \hat{x}_2 = z - k_1 y - k_2 y^2 - k_3 y^3 \end{cases} \tag{4}$$

where the k_i coefficients depend on the oscillator parameters and y is the current sensor output (all technical developments can be found in [12]). By injecting the measurement y as an input to the observer, we thus get estimates $\hat{x}_1$ and $\hat{x}_2$ of x_1 and x_2, respectively.

2.1.5   Phase Estimation through Isochrones

From the estimates $\hat{x}_1$ and $\hat{x}_2$, and since this observer is itself an oscillator, its phase can be computed from its state variables even if the estimated state does not belong exactly to the limit cycle, thanks to isochrones (see section 2.1.1). We thus get a mapping from the phase space to the phase variable : $\varphi = I(\mathbf{x})$, as shown in Fig. 3. This phase variable can later be used as a high-level variable for control purposes (see Fig. 1), as it contains all information relative to speed and position within the gait cycle. An important remark has to be made here: the strong synchronization of the observer with the given input signal is guaranteed: thanks to observer theory, it can be assessed that the behavior of the observer will asymptotically match the behavior of the observed system. In practice, the convergence time of the observer is extremely short with respect to the period of the motion, which means that the filtering that is performed is phase-shift free.
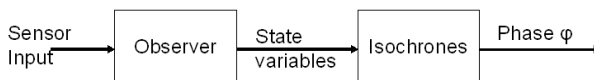
Fig. 3 Computation flow of the phase estimator

2.2   An Optimal Control-Based Approach to Generate Foot Trajectories

Optimal control techniques can serve to generate optimal trajectories for the foot taking into account the dynamics of the system and other constraints to be satisfied during the motion. The goal of the optimization problem can be either to track a prescribed trajectory e.g. to mimic another systems behavior, or - which is in most cases more desirable - optimize a criterion linked to the properties of the system itself. Optimal control problems can be solved off-line to determine the overall

optimal solution in advance, or on-line considering sensory information about the state of the system and re-optimizing in regular time steps. In this section we describe first the dynamical model of the lower leg to be optimized in this study, before we briefly outline the formulation and solution of the optimal control problem.

2.2.1   Dynamical Model of the Lower Leg with the Tibialis Anterior Muscle

In order to address the drop-foot correction problem by an optimal control approach, we need a dynamical model of the lower leg including the dynamics of the force generation in the tibialis anterior (TA) muscle as well as its activation dynamics. We are interested to describe the angular motion of the foot α_F during the swing phase of a walking step under the action of the TA. These equations of motion also depend on the position histories of the ankle x_A, y_A, or - more precisely - the corresponding accelerations $a_{x,A}$, $a_{y,A}$ and the shank orientation α_S and angular velocity ω_S. The foot angle α_F is defined to be zero when the foot is horizontal, and the shank angle α_S is zero when the shank is vertical. The dynamics of the foot angle are formulated as

$$\dot{\alpha}_F = \omega_F \tag{5}$$
$$\dot{\omega}_F = (T_{TA} + T_{grav} + m_F c_F (a_{x,A} \sin \alpha_F - a_{z,A} \cos \alpha_F)) (\Theta_F + m_F c_F^2)^{-1} \tag{6}$$

with the torque generated by gravity

$$T_{grav} = -c_F \cos(\alpha_F) m_F g. \tag{7}$$

Here m_F denotes the mass of the foot and c_F its center of mass location with respect to the ankle. The torque generated by the TA is

$$T_{TA} = F_{TA} \cdot d \tag{8}$$

where $d = 0.037$ is the lever arm of the TA at the ankle and F_{TA} the muscle force which is computed using an appropriate muscle model. The muscle model used in the present version of this study is essentially the model established by van den Bogert and Ackermann described in [1, 6] with the exception that we assume here a constant tendon length.

As in most Hill-type models, the muscle is modeled as a combination of a contractile element, a parallel elastic and a parallel damping element $F_{TA} = F_{CE} + F_{PE} + F_{PD}$.

The force in the contractile element is generally computed as the product of the maximum isometric force $F_{iso,max}$ and three independent factors f_{ad}, f_{tl} and f_{fv} (the activation level, the force-length factor, and the force-velocity factor):

$$F_{CE} = F_{iso,max} f_{ad} f_{tl} f_{fv} \tag{9}$$

The activation level f_{ad} (with $0 \leq f_{ad} \leq 1$) is determined by the activation dynamics of the muscle

$$\dot{f}_{ad} = (\varepsilon - f_{ad})(\frac{\varepsilon}{T_{act}} - \frac{1.0 - \varepsilon}{T_{deact}}) \qquad (10)$$

where ε is the muscle excitation (with $0 \leq \varepsilon \leq 1$) which we take to be the entry or control variable of this model, and $T_{act} = 0.04s$ and $T_{deact} = 0.04s$ are the time constants for activation and deactivation, respectively.

The length of the muscle-tendon complex for the TA is assumed to depend linearly on the joint angle

$$l_{MT} = l_{MT,0} - d_{ankle}\phi \qquad (11)$$

with rest length $l_{MT,0} = 0.381$ and the current relative angle at the foot $\phi := \alpha_S - \alpha_F$ (i.e. ϕ is zero when there is a right angle at the ankle). The tendon length is assumed to be constant $l_T = 0.317$, and the length of the contractile element follows directly from

$$l_{CE} = l_{MT} - l_T; \qquad (12)$$

and therefore the contraction speed of the muscle can be computed as

$$v_{CE} = -d_{ankle}\dot{\phi}. \qquad (13)$$

The force-length factor f_{fl} takes into account the fact that the force generated by muscles depends on the current length of the contractile element l_{CE}

$$f_{fl} = e^{-\left(\frac{l_{CE} - l_{CE,opt}}{Wl_{CE,opt}}\right)^2} \qquad (14)$$

where $l_{CE,opt} = 0.082$ is the fiber length at which optimum force can be generated, and $W = 0.56$ is called the width parameter describing the overlapping of filaments in the sarcomer.

The force-velocity factor f_{fv} models the dependency of the muscle force on the contraction speed of the muscle which can be described by two hyperbolic relationships:

$$f_{fv} = \frac{g_{max}v_{CE} + c}{v_{CE} + c} \quad \text{if } v_{CE} > 0 \text{ (extension)} \qquad (15)$$

$$f_{fv} = \frac{\lambda v_{max} + c}{\lambda v_{max} - v_{CE}/A} \quad \text{else (contraction).} \qquad (16)$$

where $g_{max} = 10/, L_{CE,opt} = 0.82$ is the normalized maximum force during extension and $A = 0.25$. The parameter λ takes into account that the activation level changes the maximum contraction speed

$$\lambda(f_{ad}) = 1 - e^{-3.82 f_{ad}} + f_{ad}e^{(-3.82)}. \qquad (17)$$

The factor c is introduced to produce continuous first derivatives at $v_{CE} = 0$ and is computed as follows:

$$c = \frac{\lambda v_{max} A (g_{max} - 1)}{A + 1}. \qquad (18)$$

The force in the parallel elastic element is computed as

$$F_{PE} = k_1(l_{CE} - l_{slack,PE}) \quad \text{if } l_{CE} \leq l_{slack,PE} \tag{19}$$

$$F_{PE} = k_1(l_{CE} - l_{slack,PE}) + k_{2,PE}(l_{CE} - l_{slack,PE})^2 \quad \text{else.} \tag{20}$$

while the force in the parallel damping element results from

$$F_{DE} = bv_{CE} \tag{21}$$

where b is chosen as $b = 0.001 F_{iso,max}/l_{CE,opt}$.

The above equations of motion require information about the motion of the ankle and the shank as an input, and it is assumed for our model this information is available from measurements. One possibility would be to directly substitute these measurement results into equations (5) and (6), but this is difficult since 2nd order derivatives of the actual measurements are required. We therefore prefer to keep these variables as state variables in the problem, and formulate the corresponding trivial dynamic relationships as additional equations of motions:

$$\dot{x}_A = v_{x,A} \tag{22}$$

$$\dot{v}_{x,A} = a_{x,A} \tag{23}$$

$$\dot{z}_A = v_{z,A} \tag{24}$$

$$\dot{v}_{z,A} = a_{z,A} \tag{25}$$

$$\dot{\alpha}_S = \omega_S \tag{26}$$

Corresponding terms in the objective function will then enforce that the measurements are well approximated (see the next section).

The variables used in this model can now be summarized as vectors of state variables $z^T = (f_{ad}, \alpha_F, \omega_F, x_A, v_{x,A}, z_A, v_{z,A}, \alpha_S)$ as well as a vector of control variables $u^T = (\varepsilon, a_{x,A}, a_{z,A}, \omega_S)$.

2.2.2 Optimal Control Problem Formulation and Solution

The aim of our approach is to determine the best possible motion (according to a criterion to be specified) for the musculo-skeletal model of the lower leg. The idea is that for an artificially stimulated TA muscle of a hemiplegic patient it may not be desirable to just mimic the normal stimulation pattern of a healthy patient with a normal gait or to aim at following the motion of a healthy swing foot. In the next section, we show that in the investigated case it is even impossible to exactly reproduce a healthy motion by only stimulating the TA.

One problem is that gaits of hemiplegic patients are not symmetric and that the motions of the shank and the ankle do not follow normal trajectories such that also for the foot ankle a non-normal trajectory might be optimal or even necessary to avoid foot scuffing. The choice of the right cost function is still subject of discussion, but in this study we evaluate different objective functions related to the energy consumption and fatigue in the stimulated muscle.

The goal of our ongoing research is to solve the optimal control problem online, either optimizing each swing phase prior to its start, based on information of the previous step and the initial conditions, or using predictive control to introduce corrections even during the swing phase based on measurements of the ankle and shank trajectories. It is important to note here that feasibility of this approach is linked to the type of measurements which will be practically accessible. Indeed, the number and volume of embedded sensors will be limited in an everyday use of the system.

As a start, we here investigate the use of optimal control as an off-line problem coping with pre-scribed trajectories for the shank angle and the ankle position, and optimizing the relative motion of the foot. For the time being, we also just concentrate on the swing phase, but the same approach could be used for the full step.

The optimal control problem to be solved has the following general formulation with the state variables x and control variables u listed at the end of the previous section:

$$\min_{T,x,u} \int_0^T \phi(x(t),u(t))dt \tag{27}$$

$$\text{s. t. } \dot{x}(t) = f(t,x(t),u(t)) \tag{28}$$

$$z(0) = x_0, \quad x(T) = x_e \tag{29}$$

$$r(t,x(t),u(t)) \geq 0 \tag{30}$$

The total time T of the trajectory (which here is the time of the swing phase) can be either fixed or free in the computations. The objective function $\phi(x(t),u(t))$ is a weighted sum of different components:

- a minimization of the criterion chosen for the foot motion which can be either

  - related to the energy consumed, such as a minimization of TA activation squared: $\min \int_0^T f_{ad}^2 dt$
  - related to the fatigue of the muscle, such as a minimization of the third power of the activation [1]: $\min \int_0^T f_{ad}^3 dt$
  - criteria related to the muscle excitation ε, e.g. $\min \int_0^T \varepsilon^2 dt$
  - a minimization of the distance to a reference trajectory for a subset of state variables x' (for comparison purposes) $\min \sum_{i=1}^N (x'(t_i) - x'_{ref}(t_i))^2$

- a minimization of the distance to the prescribed trajectories for x_A, z_A and α_S.

Eqn. (28) is a placeholder for the dynamic equations described above in section 2.2.1. Eqn. 29 describes the initial and final conditions for the trajectory, in this case the position variables at the beginning and end of the swing phase which are here fixed to reference values. () denotes inequality constraints to be satisfied by the motion, such as upper and lower bounds on all optimization variables or more complex constraints, in this case e.g. a clearance of the swing foot. If a whole step (including not only single support but also double support) was to be treated by this optimal control approach, problem (27) -(2.2.2) would have to be replaced by the corresponding multi-phase formulation considering different dynamics and constraints and potentially different objective functions in each phase of motion. In the

optimal control framework used, this extension is easily feasible and will be done in a later publication.

For the solution of the optimal control problem we use the powerful optimal control code MUSCOD developed at Heidelberg University [5, 15] which is based on a direct multiple shooting method. There also is a real time model predictive control version of this code which will be used in a later phase of this project.

3   Results

3.1   Phase Estimation

To validate our method, we have analyzed the walking gait of one healthy subject. An ELITE (BTS) motion capture system was used to track passive reflective markers placed on subject anatomical landmarks. Ground efforts were also recorded using a double force plate (AMTI BP600900). Data processing was performed using Smart Analyzer software (BTS). We used shank inclination to compute the gait phase and trigger the optimal control procedure. Based on this input (see Fig. 4, top) and following the method described in section 2.1, we first computed the gait cycle phase (Fig. 4, middle). Based on this phase estimate, a trigger command is issued (Fig. 4, bottom): when a new stride begins (phase equals 0%), step duration is predicted
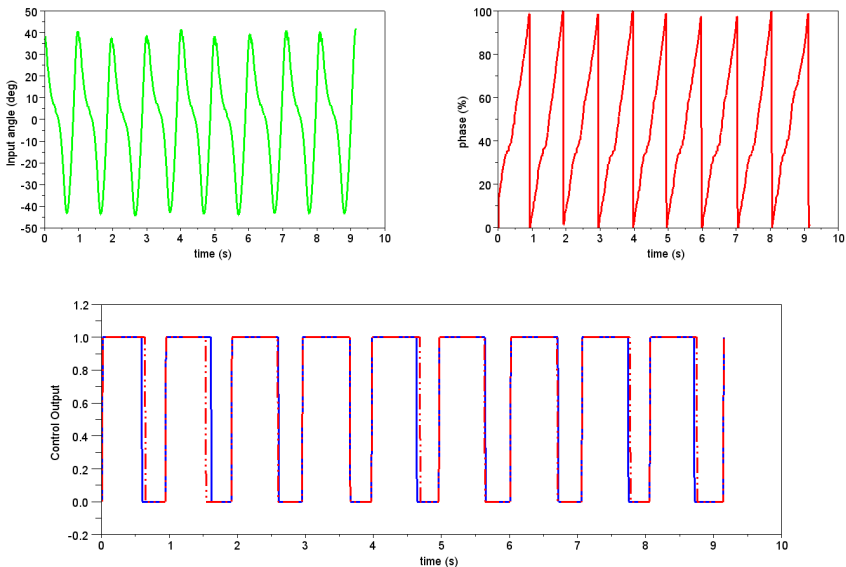
Fig. 4 Top left: Sensor input to phase estimation (shank inclination). Top right: estimated movement phase $(0 - 100\%)$. Bottom: Trigger input featuring step duration prediction (dotted red line), and comparison to ideal, *a posteriori*, estimation (blue line).

based on the previous step duration. This duration is the time needed for the phase variable to go from 0% (corresponding to heel strike, by definition) to the phase value corresponding to toe-off. This value depends on the subject; it was here identified as 63% based on ground reaction force recordings.

3.2   Optimal Control Results

In this section, we present different results of the optimal control problem solution, using different objective functions. In a first step, we wanted to explore if the model of the lower foot as described above, with only the TA activated, is capable to reproduce a measured foot motion. This is done by solving a least squares objective function minimizing the distance between computed and measured solution. As figure 5 shows, even in the optimized case, there is still a considerable difference between computed and measured ankle angle trajectories, i.e. just the TA alone is not capable to produce an exact fit. Apparently the ankle torque generated by gravity alone is not enough to counteract the TA torque as much as needed for the reference trajectory. We therefore have introduced, for the second part of our computations, an antagonistic torque in the model. We have not included a full muscle model for
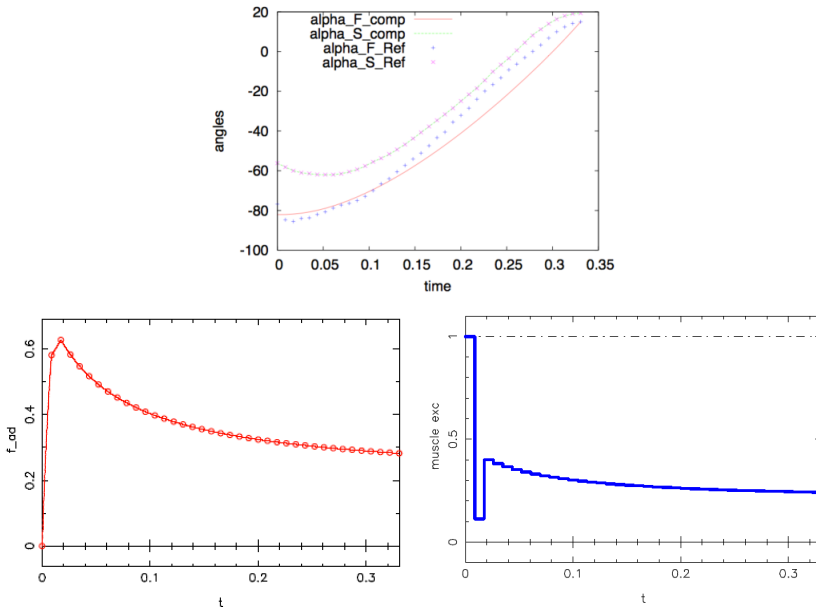
Fig. 5 Optimization results for a best possible approximation of the measured trajectory (in the least squares sense) for the TA alone, without any antagonistic torque: Reference and computed values for foot and shank angles (top), TA activation (bottom left) and TA excitation (bottom right)

the antagonist, but used directly the torque as a control variable, since the only purpose of this test was to see if such an antagonistic torque was enough to achieve an exact tracking of the reference trajectory. And indeed it was, as shown in figure 6. Mimicking healthy ankle joint motion therefore is possible if both agonist and antagonist muscle groups are activated (fig.6). However, in the considered application of using FES to treat the drop foot syndrome, only the TA muscle is electrically stimulated and as we have seen above, this prevents to reproduce healthy individual ankle trajectories (fig.5). Walking training in post-stroke hemiplegic patients is not aimed at obtaining a symmetric gait or reproducing an healthy individual gait. The gait impairment is due both to a motion control deficiency and to physiological deficiencies (weak muscles and spasticity for example). Trying to precisely track a desired trajectory is not a solution, it would fatigue the muscle with in most of the cases poor results.
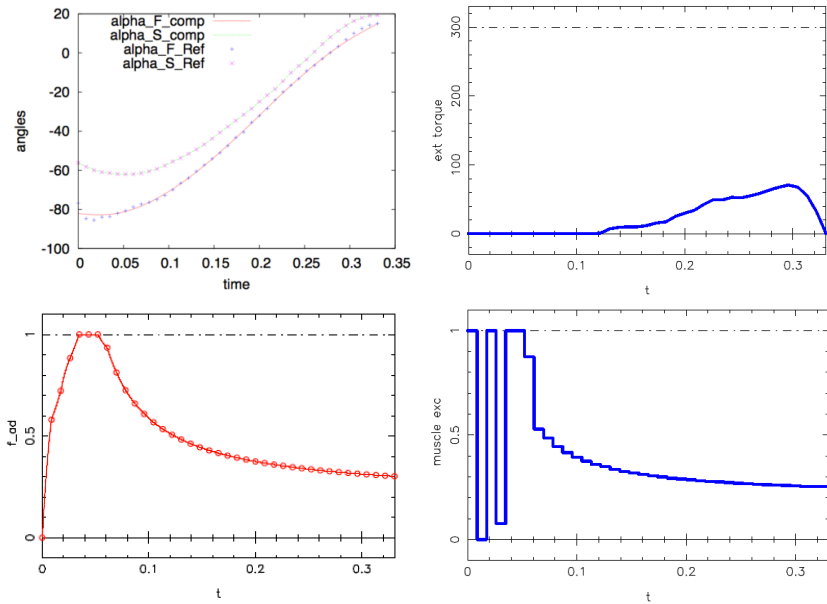
Fig. 6 Optimization results for a best possible approximation of the measured trajectory (in the least squares sense) for the TA combined with an antagonistic torque: Foot and shank angles (top left), antagonistic torque (top right), TA activation (bottom left) and TA excitation (bottom right)

So in the following computations, we were not interested in fitting computations to the reference solution any more, but to determine optimal motions with respect to the criteria listed above describing minimizing muscle effort and muscle fatigue. All following computations are done without the antagonist muscle.

Ankle angle histories and muscle activations for all results discussed are given in figure 7. We have first minimized the muscle activation squared. This is classical and
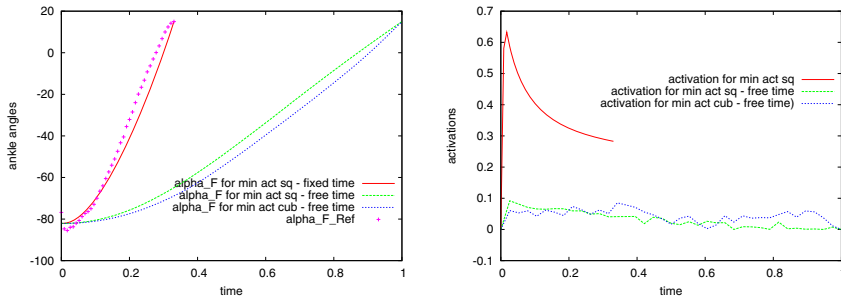
Fig. 7 Optimization results for different objective functions and constrains: minimization of activations squared (related to energy consumption) with time fixes to experimental time as well as free time, and minimization of activations to the third power (related to fatigue) with free time: Foot and shank angles (left) and muscle activation (right)

related to energy consumption minimization. The total swing phase duration in this simulation is fixed to 0.3306sec which corresponds to the measured swing time. The controller is able to generate a motion satisfying the ground clearance constraints as well as start and end point constraints of the measured motion.

We have also compared the results for free swing phase duration from 0 up to 1s (assuming 2s maximum total step time and 50% time spent in swing phase). Time goes to upper bound for all results. Compared to the fixed time solution, muscle activation is clearly diminished for similar amplitude of ankle angle trajectories.

Finally, we present results where we have run the simulations for a cost function related to muscle fatigue, expressed by the third power of muscle activation. One interesting observation is that muscle activation tends to become biphasic.

These illustrative results show that using different criteria it is possible to obtain similar ankle trajectories with very different muscle solicitations. Furthermore the trajectories can be adapted to online information measured with sensors placed on the controlled leg together with general gait information obtained from the motion observer (swing phase duration for instance).

4   Conclusion

We have proposed a control scheme for the FES control of the deficient ankle dorsi-flexion of post-stroke hemiplegic subjects presenting a drop-foot syndrome. The approach is based on two stages: 1) a continuous estimation of the walking gait cycle phase based on the observation of the valid leg behavior and 2) an optimal controller which generates activation of the muscle responsible for ankle flexion. The phase estimation algorithm has been validated in previous work. The main objective of the present paper is to provide the electrical stimulator with adaptive functionalities. The gait phase gives a rhythm information to the controller which, based on information

concerning the deficient leg, optimizes the ankle angle in order to minimize energy consumption while guaranteeing correct step execution during swing phase. Specific attention has been paid at developing low computational cost methods to allow this system to run in real-time. The presented simulation results validate the architecture and show that it is possible to adapt muscle activations during walking in such a way that muscle activations are minimized, thus reducing muscle fatigue. Results show that when trying to optimize a proxy of muscular fatigue, the activation tends to become biphasic like observed in healthy individual subjects.

In a real-life system, the patient will have to be equipped with one sensor on his valid leg in order to compute the gait phase and two sensors on his deficient limb in order to inform the controller about the actual shank inclination and ankle position. Multiple studies have shown that small, wearable inertial sensors can be used to retrieve such information. Once the controller will be available in embedded hardware, we will have to investigate in clinical studies what is the best optimization criterion to be used across multiple patients that both ensures a smooth gait and preservation of muscle in terms of fatigue. Even though, fatigue modeling and counteracting remain an open problem.

References

1. Ackermann, M., van den Bogert, A.J.: Optimality principles for model-based prediction of human gait. Journal of Biomechanics 43(6), 1055–1060 (2010)
2. Azevedo Coste, C., Héliot, R.: Rehabilitation of Functional Posture and Walking: Coordination of healthy and Impaired Limbs. Journal of Automatic Control 15(suppl.), 011–015 (2005), http://hal-lirmm.ccsd.cnrs.fr/lirmm-00135960/en/
3. Azevedo Coste, C., Héliot, R., Pissard-Gibollet, R., Dussaud, P., Andreu, D., Jérôme, F., Laffont, I.: MASEA: Marche Assistée par Stimulation Électrique Adaptative. D'un déclenchement événementiel à un contrôle continu de la stimulation électrique pour la correction du syndrome de pied tombant chez l'hémiplégique. Sciences et Technologie pour le Handicap Numéro Spécial Handicap et Mouvement 22 (2010)
4. Azevedo Coste, C., Popovic, D.B., Dosen, S., Espiau, B., Héliot, R.: Muscle activations optimization and adaptation for Functional Electrical Therapy purposes. In: 13th Czech-French-German Conference on Optimization, p. N/A (2007)
5. Bock, H., Plitt, K.: A multiple shooting algorithm for direct solution of optimal control problems. In: Proceedings 9th IFAC World Congress Budapest, pp. 243–247. Pergamon Press (1984)
6. van den Bogert, A.J.: Tuorial: Musculoskeletal model for simulation of walking. In: Dynamic Walking Conference 2011, Jena (2011)
7. Burridge, J., Taylor, P., Hagan, S., Swain, I.: Experience of clinical use of the odstock dropped foot stimulator. Artif. Organs 21, 254–260 (1997)

8. Chen, Y., Shih, Y., Chen, W., Lin, J., Kuo, T.: Clinical evaluation of the tilt sensors feedback controlled fes for hemiplegia. In: Conf. Proc. IEEE Eng. Med. Biol. Soc., vol. 7, pp. 4737–4740 (2004)

9. Dai, R., Stein, R., Andrews, B., James, K., Wieler, M.: Application of tilt sensors in functional electrical stimulation. IEEE Trans. Rehabil. Eng. 4(2), 63–72 (1996)

10. Dosen, S., Popovic, D.B., Azevedo Coste, C.: Optiwalk. Un nouvel outil pour la conception et la simulation de lois de commande pour le contrôle de la marche de patients atteints de déficits moteurs. Journal Européen des Systèmes Automatisés (JESA) 41(2/2007), 239–259 (2007)

11. Héliot, R., Dosen, S., Azevedo, C., Espiau, B., Popovic, D.: Online adaptation of optimal control of externally controlled walking of a hemiplegic individual. In: 3rd International IEEE EMBS Conference on Neural Engineeering, Kohala Coast, Hawaii (2007)

12. Héliot, R., Espiau, B.: Online generation of cyclic leg trajectories synchronized with sensor measurement. Robotics and Autonomous Systems 56, 410–421 (2008)

13. Jonić, S., Janković, T., Gajić, V., Popović, D.: Three machine learning techniques for automatic determination of rules to control locomotion. IEEE Trans. Biomed. Eng. 46(3), 300–310 (1999)

14. Kostov, A., Hansen, M., Haugland, M., Sinkjaer, T.: Adaptative restriction rules provides functional and safe stimulation pattern for foot drop correction. Artificial Organs 23(5), 443–446 (1999)

15. Leineweber, D., Bauer, I., Bock, H., Schlöder, J.: An efficient multiple shooting based reduced SQP strategy for large-scale dynamic process optimization. Part I: Theoretical aspects. Computers and Chemical Engineering 27, 157–166 (2003)

16. Liberson, W.T., Holmquest, H.J., Scot, D., Dow, M.: Functional electrotherapy: stimulation of the peroneal nerve synchronized with the swing phase of the gait of hemiplegic patients. Arch. Phys. Med. Rehabil. 42, 101–105 (1961)

17. Luenberger, D.: An introduction to observers. IEEE Trans. Automatic Control 16(6), 596–602 (1971)

18. Lyons, G., Sinkjaer, T., Burridge, J., Wilcox, D.: A review of portable fes-based neural orthoses for the correction of drop foot. IEEE Transactions on Neural Systems and Rehabilitation Engineering (2002)

19. Lyons, G., Wilcox, D., Lyons, D., Hilton, D.: Evaluation of a drop foot stimulator fes intensity envelope matched to tibialis anterior muscle activity during walking (2000)

20. Pappas, I., Keller, T., Mangold, S.: A reliable, gyroscope based gait phase detection sensor embedded in a shoe insole. In: Proceedings of IEEE Sensors 2002. First IEEE International Conference on Sensors, vol. 2, pp. 1085–1088 (2002)

21. Pikovsky, A., Rosenblumn, R., Kurths, J.: Synchronization, a universal concept in nonlinear sciences. Cambridge University Press (2001)

22. Popović, D., Sinkjaer, T.: Control of Movement for the Physically Disabled (2003)

23. Righetti, L., Buchli, J., Ijspeert, A.: From dynamic hebbian learning for oscillators to adaptive central pattern generators. In: Proc. Adaptive Motion in Animals and Machines, AMAM 2005 (2005)

24. Shimada, Y., Ando, S., Matsunaga, T., Misawa, A., Aizawa, T., Shirahata, T., Itoi, E.: Clinical application of acceleration sensor to detect the swing phase of stroke gait in functional electrical stimulation. Tohoku J. Exp. Med. 207(3), 242–246 (2005)

25. Weber, D., Stein, R., Chan, K., Loeb, G., Richmond, F., Rolf, R., James, K., Chong, S.: Bionic walkaide for correcting foot drop. IEEE Trans. Neural Syst. Rehabil. Eng. 13(2), 242–246 (2005)

26. Willemsen, A., Bloemhof, F., Boom, H.: Automatic stance-swing phase detection from accelerometer data for peroneal nerve stimulation. IEEE Transactions on Biomedical Engineering 37, 1201–1208 (1990)
27. Williamson, R., Andrews, B., Au, R.: Control of neural prostheses. ii. Event detection using machine learning. In: Proceedings of the RESNA 1996 Annual Conference Exploring New Horizons, vol. 2, pp. 291–293 (1996)
28. Winter, D.: The biomechanics and motor control of human Gait. University of Waterloo Press (1987)

The Combined Role of Motion-Related Cues and Upper Body Posture for the Expression of Emotions during Human Walking

Halim Hicheur, Hideki Kadone, Julie Grèzes, and Alain Berthoz

Abstract. The present study aimed at investigating how emotion affect the kinematic aspect of human walking. The gaits of eight professional actors expressing different types of emotions (neutral, joy, anger, sadness and fear) during walking were recorded and analyzed in the sagittal plane. We show both step-related behavioural changes (in terms of step length, speed,etc.) that are common to different emotions and emotion-specific body configuration changes (mainly at the level of the upper body posture) during emotional gaits. Since the overall speed of walking is another major variant in walking, natural walking at different speeds were recorded in another session for a control.

1 Introduction

Emotional body language (EBL) provides reliable cues to recognize emotions even when viewed from distance and when facial expression is not visible. The few

Halim Hicheur
Laboratoire de Physiologie de la Perception et de l'Action, Collège de France, Paris, France.
UFR Sciences de l'Homme et de la Société, Grenoble, France
e-mail: `halim.hicheur@upmf-grenoble.fr`

Hideki Kadone
Laboratoire de Physiologie de la Perception et de l'Action, Collège de France, Paris, France.
Center for Cybernics Research, University of Tsukuba, Tsukuba, Jpapan
e-mail: `kadone@ccr.tsukuba.ac.jp`

Julie Grèzes
Laboratoire de Physiologie de la Perception et de l'Action, Collège de France, Paris, France.
Lab. of Cognitive Neuroscience, INSERM U960, Ecole Normale Superieure, Paris, France
e-mail: `julie.grezes@ens.fr`

Alain Berthoz
Laboratoire de Physiologie de la Perception et de l'Action, Collège de France / CNRS UMR 7152, Paris, France
e-mail: `alain.berthoz@college-de-france.fr`

currently available studies show that EBL can readily be recognized whether in static postures[18, 5], whole body movements[22, 1] or even simple dynamic point-light displays[6, 4]. Moreover, the results from the few studies that have investigated the neural basis of perceiving EBL underscore the significance of using stimuli consisting of emotional expressions by the entire body as well as those including biological movement[8, 5, 7, 12, 14] rather than using static facial expressions of emotions.

While these studies extended the question of the expression of emotions to the whole body configurations in space, the kinematic properties of the emotional whole body movement received little attention. In the case of arm movements, Pollick et al.[13] showed that humans perceive particularly well different emotions conveyed by different types of arm movements using kinematic cues like the peaks of wrist velocities and acceleration. Atkinson and colleagues[1, 2] generalized these findings to the case of body gestures and provided evidence for distinct contributions of form-related and motion-related cues to the recognition of emotions from whole body movements. Within the domain of locomotor behaviors, Troje[20] proposed a computational approach for analyzing and synthesizing human gait patterns, used for example to specify the kinematic properties of motion-related and configurational cues during gender recognition.

Another approach based on independent components analysis was successfully achieved in the case of emotional gaits[11]. In [15], an ICA method based on the frequency domain was used to find the several basic patterns for each joint angle. Variation of the coefficients of combination of these basic patterns for each emotion was computed and applied for the analysis of emotion affect on these joint angle patterns. They successfully replicated the conventional psychological results from quantitative analysis. While these computational approaches can help to synthesize artificial gaits, the control mechanisms underlying the implementation of emotional gaits in humans remain unclear. Even if locomotion involves a considerable amount of joints (and the associated muscles) for displacing the whole body, it was shown that this complex behavior can be characterized by specific postural strategies both for balance and trajectory control (the "steering behavior", see (Hicheur et al.[9]) for a review). A similar approach could therefore be implemented for the description of the emotional locomotor behavior.

In the present study, kinematic aspect of emotional locomotion will be investigated, based on a physiological viewpoint on the human locomotion control system. Human locomotion controller is composed of involuntary tonic posture control mainly by the brain stem, rhythm control and stretch reflex by the spinal cord, and voluntary precise limb control by the cortical systems[16]. Emotion may affect some or all of these motion control systems. To clarify this point, we extracted features of gait patterns corresponding to these controlled variables (arm movements, step and rhythm related values, and upper body posture including head and trunk), and compared their emotion dependence against speed dependence to examine the implementation of the emotional gaits.

To this purpose, we recorded and analyzed the gaits produced by actors in five emotional states (neutral NE, joy JO, anger AN, sadness SA and fear FE). Second, the comparison between these emotional gaits with neutral gaits performed

at different speeds by naive subjects (normal/natural speed NS, slow speed SS and fast speed FS) allowed discriminating between speed-related changes and emotion-specific changes (mainly at the level of the upper body posture) in the recorded emotional gaits.

2   Methods

2.1   Experiment 1: Recording of the Emotional Gaits

Eight (four males and four females) healthy, well-experienced professional actors were paid for their participation in the experiments. They gave informed consent prior to inclusion in the study. Experiments conformed to the Declaration of Helsinki. The age, height, weight and number of years of practice of the actors were equal to 26.0 ± 1.41 / 27.0 ± 2.71 years (men / women), 1.82 ± 8.12 / 1.63 ± 6.14 meters, 66.8 ± 7.8 / 58.5 ± 5.69 kilograms and 5.6 ± 1.11 / 6.5 ± 2.65 years respectively. The actors had to walk straight ahead for about ten meters in all the experimental conditions. Five conditions (fear FE, sadness SA, anger AN or joy JO and the control, neutral NE) were tested (see Figure 1 for an animated illustration of the recorded emotional gaits) and subjects had to repeat at least five trials for each condition. In the four conditions, subjects had to express an emotion (one of FE, SA, AN and JO).

They were instructed to feel the emotion before starting to walk. It must thus be emphasized that the recorded gaits were performed in a not-specified context except for fear. Indeed, we observed in a preliminary experiment that some actors spontaneously run in the fear condition ("escape" behavior from some imagined fearful entity). To prevent them from running, a scenario of "walking forward in a dark and dangerous room" was specified to actors exceptionally in this condition so that they walk with a rather general feeling of fear without imaging specific entity at some specific location. The actors orally reported that this instruction made the task easier. Thus, the emotions we investigated here are rather specific and a single emotion (as for fear) can be expressed in different ways. Nevertheless, we wanted here to examine how the emotional state (rather than the scenario) intrinsically affects the locomotor behavior, independently of the strategies/scenarios used by the actors for expressing a particular emotion.

2.2   Experiment 2: Recording of Neutral Gaits with Different Walking Speeds

As Experiment 1 revealed that the different emotional gaits were characterized by different walking speeds, the changes observed at the motor patterns level between emotional gaits might be speed-specific rather than emotion-specific. To address this issue, we designed a control experiment where we tested 5 naive subjects who were

asked to walk straight ahead for ten meters, at different speeds (Low, Normal and Fast - LS, NS and FS conditions, respectively). The age, height and weight of the (male) subjects were equal to 27.4 ± 3.0 years, 1.81 ± 0.03 meters, and 73.4 ± 4.3 kilograms, respectively. Subjects performed 5 repetitions for each condition and a total of 75 trials (5 subjects x 3 speeds x 5 repetitions) were recorded. The speed was chosen by the subjects so that they could walk naturally according to the indicated speed condition.

2.3 Motion Capture

Three-dimensional positions of light reflexive markers were recorded using an optoelectronic Vicon V8 motion capture system wired to 24 (16 in Experiment 2) cameras at a 120 Hz sampling frequency. We used the Vicon Plug in Gait model (VICON, Oxford Metrics Limited, Oxford, United Kingdom). Placement of the 3D markers on the body was as follows: i) 4 markers were directly placed on light glasses without lenses. The head markers were here placed differently from the Plug in Gait model where the markers are supposed to be directly placed on the head. The use of glasses allowed an easier control of markers placement across
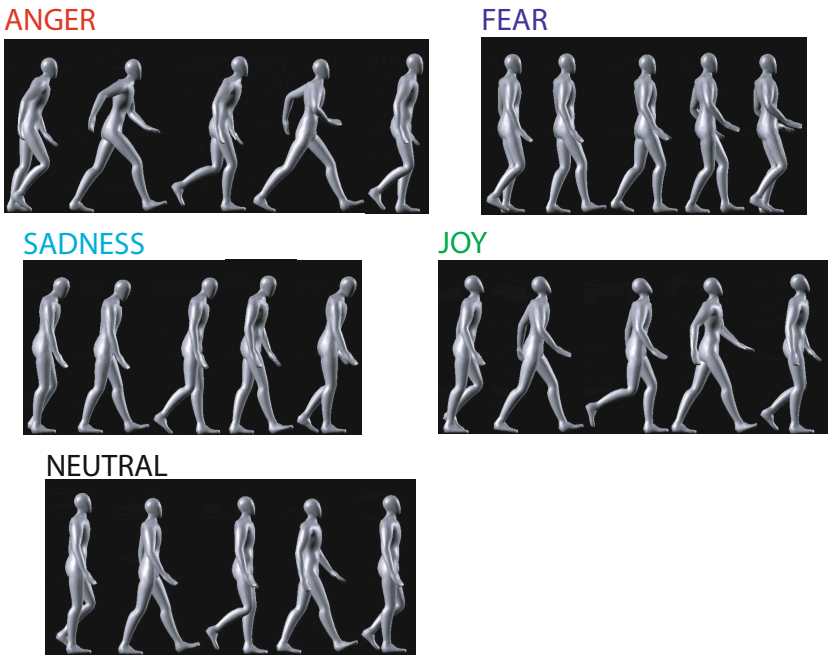
Fig. 1 Animated illustration of an actor (obtained after motion capture) expressing emotion during locomotion, corresponding to anger, fear, sadness, joy, and neutral. The illustration was built by mapping marker positions to the animated character.

subjects, without affecting the quality/accuracy of head markers data: forward and backward markers were aligned with the head roll rotation (naso-occipital) axis for left and right sides ii) left and right shoulder markers were located on left and right acromions, respectively; iii) left and right elbow markers were located on left and right lateral epicondyles (approximating the elbow joint axis) and left and right wrist markers were located on the wrist, on the external face of the lower arm; iv) 4 markers were located on the pelvis: left and right front markers were placed respectively on left and right anterior superior iliac spines while left and right back markers were located on the left and right posterior iliac spines, respectively. v) left and right knee markers were placed on the external side knee joints; vi) left and right ankle markers were located on the lateral malleolus; vii) left and right toe markers were placed at the top of the foot (subjects were allowed to wear shoes), between toes 2 and 3 (1 is for the big toe) viii) left and right heel markers were located at the heel (at the same height as toe markers).

Three dimensional coordinate frame (XYZ) of the motion capture space was defined so that the X-axis is aligned with the walking direction by connecting the start and end points indicated on the floor and the Y-axis is directed leftward perpendicular to the X-axis, and the Z-axis is directed upward along the vertical. This definition gives the laboratory coordinate frame.

2.4 Behavioral Variables

2.4.1 Step Events and Step Parameters

We used heel strike and toe off events for defining steps. These events were detected in the time course of heel and toe Z position profiles. We considered one step as the interval separating two successive heel strikes. In order to investigate how an emotion affects the gait, different step parameters were calculated. The step speed, duration and length were computed. In addition, the stance and swing phases of the step cycles were discriminated using the toe-off events and the percentage of stance duration (relatively to the whole step cycle) was computed. After the calculation of these parameters, every kinematic pattern examined at the level of the step was re-sampled to have the same number of points (n=100) for purpose of comparison between the motor patterns generated by the subjects in the different conditions.

2.4.2 Speed-Related Effects of Emotional and Neutral Gaits

We extracted the walking speed from each step cycle of emotional and neutral gaits. We also extracted the instants (in % of the step cycle) at which subjects shifted from stance to swing as well as the magnitude of angular motion, the instant of maximal peak occurrence and (for the head and trunk absolute angular movements) the mean upper body orientation throughout the step cycle. We assessed the effects of the walking speed on these global and local (joint or segment-related) gait changes in a

2D space for the different joints. This was done for both the "emotions" (FE, SA, NE, JO and FE conditions) and "speed" (LS, NS and FS conditions) groups. The general effect of speed on the temporal organization of walking was computed through linear regression between the walking speed and the instant of stance-to-swing transition.

2.4.3 Joint Angular Motion of the Body Segments

Using the positions provided by the markers mentioned above, we defined a total of 12 body segments (HEAD, TRUNK, and left and right ARM, FOREARM, THIGH, SHANK and FOOT). Joint angular motion of these segments were computed in the sagittal plane [10]. Every segment was projected onto the sagittal plane, which is defined by the XZ plane of the laboratory coordinate frame. Its relative angle to the axes of the laboratory reference frame was computed to obtain the absolute orientation. The angles between the projected segments were computed to obtain joint angles (HEAD-TRUNK, SHOULDER, ELBOW, HIP, KNEE and ANKLE angles). These angles were calculated throughout the movement duration.

2.4.4 Joint Angular Motion Amplitude and Timing

The magnitude of the angular displacement of the body limbs was measured as a root mean square value of joint angles. The temporal structure of the angular profiles was then quantified using a discrete estimation of the timing changes induced by an emotion: the maximal (in absolute values) peaks occurring during the step cycle were computed for each joint (see arrows in Figure 4). For the head movement with respect to the trunk, we observed two oscillations within a step cycle and chose to compute the peak during the swing phase (computing it during the stance period yielded similar results).

2.4.5 Upper Body Orientation in Space

As can be noticed in the animated illustration in Figure 1 , the orientation of the upper body in space is different across the emotions. We quantified this by computing the mean absolute orientation (relative angle to the Z-axis of the laboratory reference frame) of HEAD, TRUNK, ARM, and FOREARM throughout the step cycle.

2.4.6 Discriminative Power as an Evaluation of the Effects of Speed and Emotion on Gait Parameters

In the field of pattern classification, discriminant analysis is used to find a subspace that most separates given classified data vectors. We apply the criteria used in this method to examine the extent of contribution of each feature to separation among emotion (or speed) groups. The discriminative power was calculated as follows:

within the whole group of emotions E (or speeds S for the neutral gaits), we computed the variance of whole data V_{all} including all the groups, and the variance of each emotional group V_{Ei}. The discriminative power DP is computed as:

$$DP = \frac{V_{all}}{\sum_{i=1}^{N} V_{Ei}} \tag{1}$$

with N the number of groups contained in the whole dataset (5 classes of emotions including the neutral or 3 classes of speeds). This criterion evaluates the inverse proportion of intra-class variance against the whole variance. DP is computed for each joint motion profile, both for emotional gaits and gaits performed at different speeds. The greater the DP, the more discriminative the joint segment is. Greater DP means that the joint motion is largely affected by the differences of classes of emotions or speeds.

3 Results

Emotions systematically and significantly affected walking behaviors at the level of the gait pattern changes. This was observable both at the level of global gait parameters (step length, speed and duration as well as stance phase duration) and at the level of the segmental motion of the body segments. In the following section, analysis of variance to compare expected value among emotion groups is conducted by F-test, where the F-values are indicated by $F(a,b)$ with a and b representing the degrees of freedom for the groups and the number of data points respectively.

Our focus is on the emotion specific motion features that are common to the both genders rather than differences between them[20]. Actually, it was difficult to see gender differences in our data.

3.1 Gait Parameters Changes

Emotions significantly modulated the walking speed (see Figure 2, $F(4,284) = 377.62, p < 0.01$). In particular, fear (FE) and sadness (SA) slowed down the walk (by up to 0.7 m/s, compared to the neutral (NE) condition, $F(1,71) = 760.70, p < 0.01$) while anger (AN) accelerated it ($F(1,71) = 219.32, p < 0.01$). The walking speed in FE was also significantly reduced compared to SA (by up to 0.2 m/s, $F(1,71) = 67.04, p < 0.01$). No statistically significant difference was observed when comparing neutral and joy (JO) conditions ($p > 0.05$). These changes in walking speed were associated with step length, step duration and stance phase duration changes ($F(4,284) = 414.04, p < 0.01$, $F(4,284) = 107.00, p < 0.01$, and $F(4,284) = 15.70, p < 0.01$, respectively).

The step length was significantly reduced in FE and SA conditions ($F(1,71) = 796.40, p < 0.01$, it was also significantly greater in SA compared to FE, $F(1,71) = 159.13, p < 0.01$) and was significantly increased in AN ($F(1,71) = 149.61$,
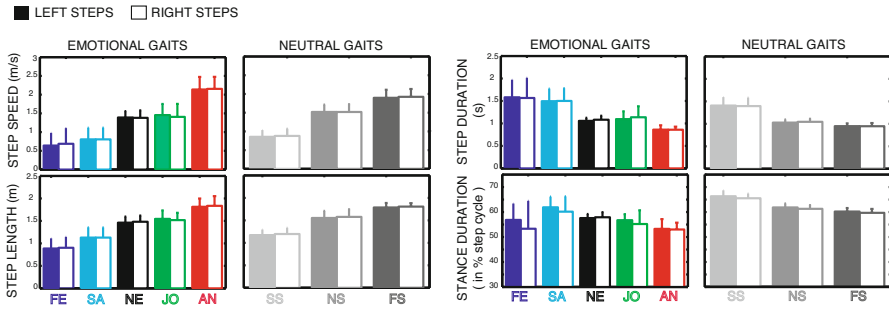
Fig. 2 Gait parameter changes (mean $\pm$ SD) in emotional gaits recorded in actors and in neutral gaits performed at different walking speeds recorded in naive subjects.

$p < 0.01$) when compared to NE. Although the step length was quantitatively greater (by about 3 centimeters) in the joy condition, no statistically significant difference was observed between the joy and neutral condition ($p > 0.05$).

The step duration was significantly increased in FE and SA ($F(1,71) = 203.32$, $p < 0.01$, it was also significantly increased in FE compared to SA, $F(1,71) = 12.16, p < 0.01$) and was significantly reduced in AN ($F(1,71) = 216.47, p < 0.01$)) when compared to NE. The step duration was quantitatively longer in JO compared to NE (by about 0.04 seconds), but this difference was not statistically significant ($p > 0.05$).

The stance phase was of comparable duration between FE and NE ($p > 0.05$) and significantly shorter in FE compared to SA ($F(1,71) = 17.79, p < 0.01$). The stance phase was significantly longer in SA (compared to NE, $F(1,71) = 120.56, p < 0.01$), significantly longer in JO ($F(1,71) = 6.38, p < 0.05$) and significantly shorter in AN ($F(1,71) = 36.04, p < 0.01$). The difference was also statistically significant between AN and JO ($F(1,71) = 7.98, p < 0.01$). It should be noted that the variability of the stance phase duration is greater in the FE and SA conditions (see error bars in Figure 2). Although the stance phase duration is modulated in different ways, these changes are, to a certain extent (if FE is not considered), well fitted by velocity changes (r=0.83, see Figure 3-left).

In the second experiment, in agreement with the instructions to walk at slow (SS), normal/natural (NS) or fast (FS) speeds across trials, we found that the step speed was significantly modulated for the neutral gaits performed ($F(2,98) = 728.31, p < 0.01$). It should be noted that the average walking speed in this group (NS condition) was higher (see Figure 3-right) than the one recorded in the actors' group (NE condition). The speed changes were associated with statistically significant changes at the level of step length, step duration and stance phase duration ($F(2,98) = 192.77, F(2,98) = 147.54$ and $63.72, p < 0.01$, respectively). For each of these parameters, the effect was also statistically significant when comparing SS to NS, SS to FS or NS to FS ($p < 0.01$). The stance phase duration was also a linear function of speed changes ($r = 0.91$, see Figure 3-right).

STANCE TO SWING TRANSITION
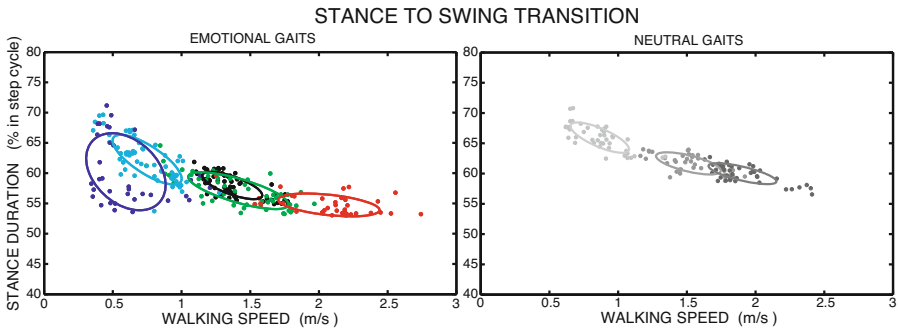
Fig. 3 Relationships of the linear regression fit between walking speed and stance phase duration for emotional gaits (left, including the neutral gait of actors) and neutral gaits performed at various speeds (right). The variance ellipses of the different groups of "emotion" and "speed" are also shown. Note the largest variability of the fear. The linear regression parameters (correlation coefficient (r), root mean square error (rmse) and slope of the regression lines(s)) were r=0.72, rmse=2.77, s=-5.35 in emotional gaits when including "fear" group, r=0.83, rmse=2.03, s=-6.07 when excluding it, and r=0.91, rmse=1.27, s=-5.88 in neutral gaits with various speeds. Note the comparable slopes(s) of the regression lines for emotional and neutral gaits.

Taken together, the results from the "emotions" and from the "speeds" data show that most of the global gait parameters changes occurring in the different emotional gaits can be linearly fitted by speed variations (see Figure 3), independently of the specific emotions. The only exception concerns the particular case of stance phase duration (i.e. for the timing of gait within a particular step cycle) for which changes observed in FE and JO conditions varied independently of the speed. Nevertheless, even for the stance phase parameter, the slope of the regression line relating this parameter to speed variations (see Figure 3) is more or less the same for the "speeds" and the "emotions" gaits ($s = -5.88$ versus s = -6.07, respectively). These local aspects of the effects of the emotion are further examined at the level of the joint angular motion patterns in the next section.

3.2 Joint Angular Motion Patterns

The joint motion patterns are presented in Figure 4 for the "emotions" (left) and "speeds" (right) gaits. The similarity between these patterns was quantified across emotions or across speeds by computing the correlation coefficient between the mean joint angular profile observed in NE (or NS) with those observed in the other conditions (FE,SA, JO and AN, and SS and FS, respectively). While the amplitude of angular motion differs across conditions, the patterns were highly similar ($r > 0.8$) for all joints except for the head-trunk angle in the "emotions" gaits.

Joint Angular Motion Patterns
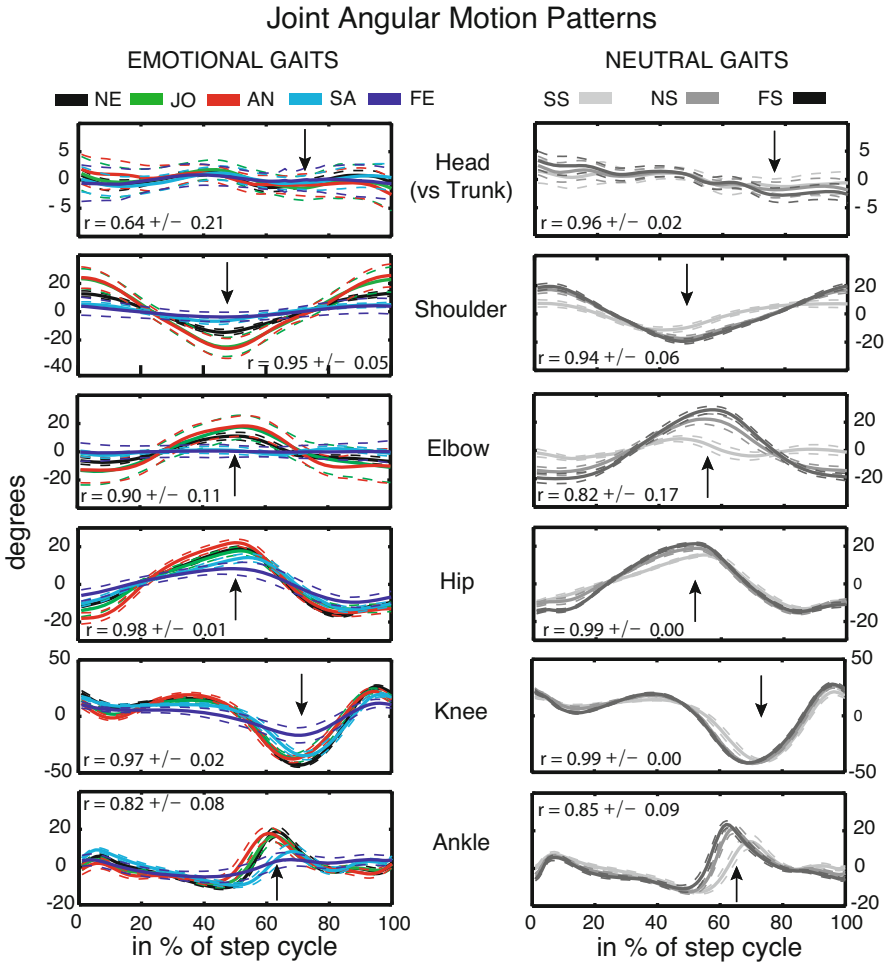
Fig. 4 Joint angular motion profiles in emotional (left) and neutral (right) gaits. The similarity between the mean neutral profile (or the normal speed profile NS for the neutral gaits) and the mean emotional profiles (or the FS and SS profiles) is quantified by the correlation coefficient r (with ± SD across conditions). The arrows indicate the peaks computed for each particular angular profile and used for estimating the timing of joint angular motion (see text for details and Figure 5b for quantitative measures). Note the similarity between the emotional and neutral joint motion patterns. Note also that while the amplitude of joint angular motion (notably for the shoulder and elbow joints in fear and sadness conditions) is considerably reduced for some emotions, the pattern of temporal changes of joint motion is preserved (as quantified by the r coefficient).

The patterns were also comparable between the emotional and the neutral gaits. However, higher variability was observed between emotional gaits, particularly at the head-trunk joint, the shoulder and elbow movements in AN and JO conditions.

In the hypothetical case where the joint angular motion patterns would have been exclusively driven by speed changes, the amplitude and the temporal structure of the "emotional" patterns (Figure 5a and b, left) would have been changed as a function of gait speed, in a way similar to the changes observed in the neutral gaits performed at different speeds (Figure 5a and b, right). At the level of the amplitude of joint angular motion (Figure 5a), this possibility was confirmed for movement of the leg (hip, knee and ankle joints) and the elbow. For these features, the discriminative power (DP) of the speed is high and comparable to the one of the emotional gaits (Figure 5c).

However, this was not true for head and shoulders movements: here a greater variability in the emotional gaits was observed (note the larger width of the variance ellipses of AN and JO groups). While the head motion amplitude remains unchanged by walking speed and ranged between 2 and 4 degrees for the neutral gaits, the head angular motion amplitude was systematically above 4 degrees in AN and JO. The speed DP is particularly poor for the head motion and the emotion DP is therefore greater for AN and JO changes. The shoulder motion amplitude linearly increased with walking speed until reaching 10 degrees for FS: a similar tendency was observed for the "emotions" gaits but here, an abrupt increase of shoulder motion amplitude above 10 degrees was observed for AN and JO conditions. This resulted in a higher DP in "emotions" compared to "speeds" gaits. It should be noted that for the other angles, the DP is comparable between the two groups and is even higher in "speeds" gaits for the hip joint: this is explained in particular by a super imposition of JO and NE variance ellipses.

The timing of the joint motion patterns (Figure 5b) was subject to less variability in the "emotion" gaits and the changes observed across emotions correspond to the changes observed in "speeds". This resulted in comparable speed and emotion DP. The speed DP was even considerably greater for the knee and ankle movements. This was mainly the results of a super imposition of the variance ellipses between SA and FE groups. It should be noted, though, that both speed and emotions have low DPs for the head movement, reflecting, as for the amplitude parameter, that the speed is not as discriminative for this parameter compared with the other joints: this can also be due to a higher variability of head angular motion for both groups.

3.3 *Upper Body Orientation in Space*

The joint angular motion previously examined was completed by "posture" -related measurements (mean head and trunk orientation in sagittal plane). These quantitative measurements confirmed what can intuitively be observed in Figure 1: the emotions greatly affect the angular position in space (the orientation) around which the head and the trunk are rotating (see Figure 6). Compared with the "speeds" gaits for which the head is oriented around 7 degrees (slightly upwards) across speeds (with some variability across subjects in particular for the NS condition), the head is oriented negatively (downwards) for FE, SA and AN conditions. The head is also oriented slightly more upwards in JO group for a speed comparable to the one
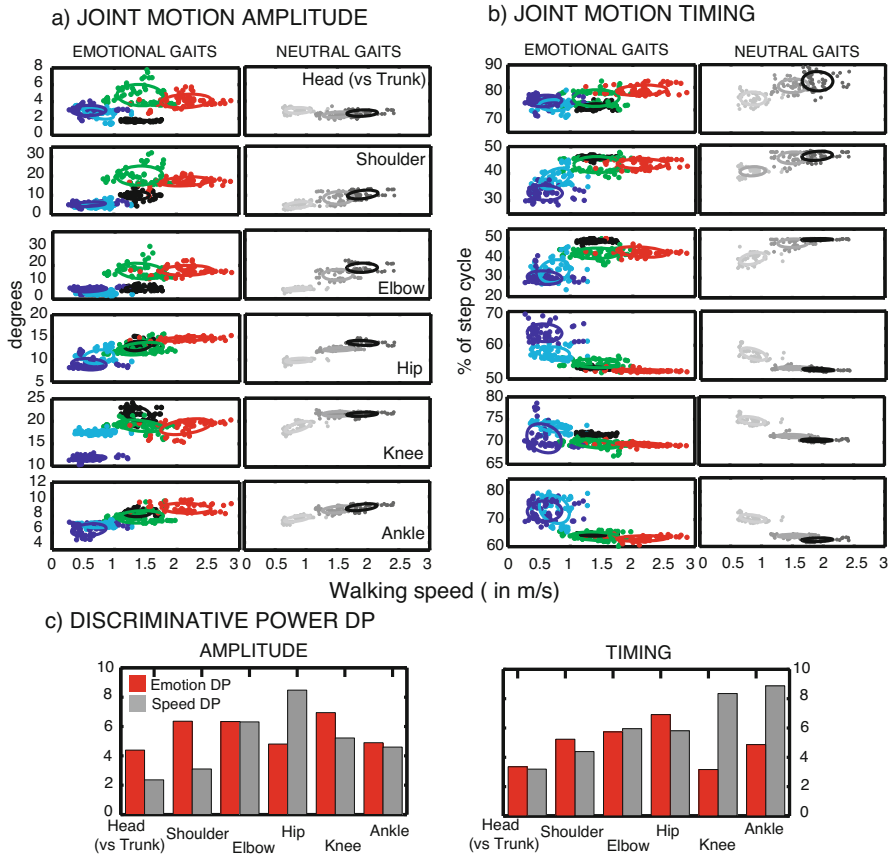
Fig. 5 a) Joint motion amplitude as a function of the recorded walking speed variations in both emotional (left) and neutral (right) gaits. The variance ellipses of each group of emotions or speeds (for neutral gaits) are also plotted. The wider the ellipse is, the more variable the values within a group are. Note that the amplitude evolves in the same direction as walking speed for most joints of neutral and emotional gaits, with higher variability for some emotions (notably the anger and joy at the level of head and arm movements) b) Joint motion timing as quantified using the peaks computed in the joint motion patterns (see arrows in Figure 4). Note that here too, the changes in the timing of joint motion patterns are mostly dictated by walking speed changes, with less variability compared with the joint motion amplitude c) quantification of the speed and emotions' discriminative power DP, respectively on joint motion amplitude (left) and joint motion timing (right). Note the higher emotion discriminative power for head and shoulders' amplitude only.

observed in the NE group. The trunk orientation remains unchanged across "speeds" and the same for AN, JO and SA. However, the trunk was oriented significantly more downwards for FE and AN. The distances between the different variance ellipses were higher for the "emotions" than for the "speed" gaits: this resulted in a
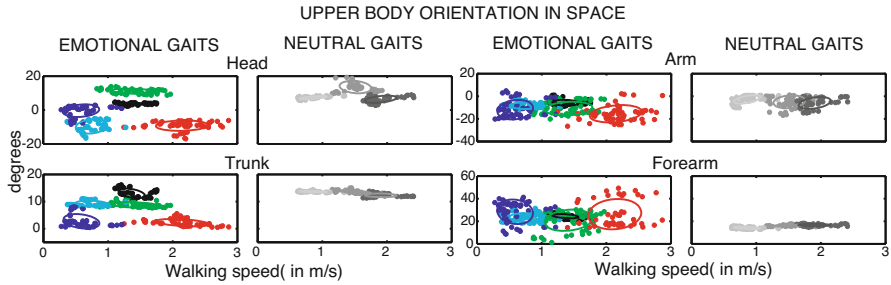
Fig. 6 The mean orientation of the upper body and arms in space as a function of walking speed. Note the clearly identifiable independent effect of the FE, SA and AN emotions on head and trunk orientation, as well as the specific trunk orientation for FE and AN. These parameters are fixed around a single value for the different speeds of the neutral gaits. The orientations of the arms and forearms are more variable for the emotional gaits and the mean forearm orientation is greater in emotional gaits compared to the neutral gaits.
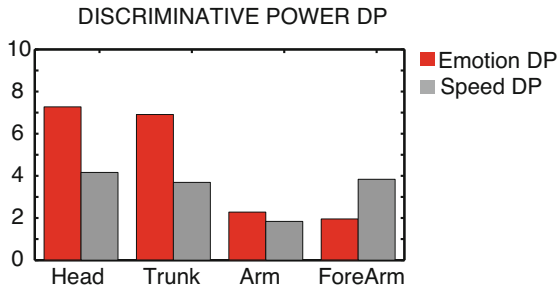
Fig. 7 Emotions and speed discriminative powers for upper body and arms' orientation: note the confirmation of the emotions specificity for head and trunk orientations.

higher emotion DP compared with the speed DP (Figure 7). The orientation of the arms was much more variable across subjects for a same emotion (Figure 6). The arms were oriented around – 10 degrees (slightly backwards) both for emotional and neutral gaits. The variability around this value for "emotion" gaits was related to inter-subjects variability and was not due to a particular emotion's effect. The emotion and speed DPs were low and considerably lower than the head and trunk DPs (Figure 7). While the orientation of the forearms displayed even more variability than the arms' orientation, the mean orientation of the forearms observed for the "emotion" gaits was around 30 degrees. The forearms were thus in a more elevated position (the elbow was in a more flexed position) for the "emotion" gaits but their orientation was more stabilized for the "speed" gaits (around 20 degrees). This higher "stability" resulted in a speed DP greater than the emotion DP. Taken together, the present description of motor changes associated with different emotions revealed that most of the time-varying angular profiles (motion patterns) are explained by speed changes only. Indeed, the neutral gaits performed at different

speeds were associated to similar variations. However, the upper body orientation (the head and the trunk) was found to be emotion-specific. The forearm orientation was also different between "emotion" and "speed" gaits but emotions less systematically affect this parameter (a greater inter-subjects variability was observed).

4 Discussion

This study investigated the motor processes governing the expression of emotions during simple walking task in humans. As mentioned earlier, it should be noted that this study was limited to the analysis of the movements in the sagittal plane. Further investigations are thus needed in order to generalize our findings to other planes. Another limitation of this study relies on the fact that the emotions expressed by actors were specific and do not account for the whole range of behaviors potentially associated with emotion. Still, a consistent effect across subjects for each of the studied emotions on the locomotor behavior was revealed. More specifically, we demonstrated that emotional gaits are associated with body configuration changes that are emotion-specific and speed-related behavioral changes that are common to different emotions. we showed that emotions mainly affect walking in terms of motor patterns (namely the timing and amplitude of the legs' motion) and postural changes (head and trunk orientations in the sagittal plane).

References

1. Atkinson, A.P., Dittrich, W.H., Gemmell, A.J., Young, A.W.: Emotion perception from dynamic and static body expressions in point-light and full-light displays. Perception 33, 717–746 (2004)
2. Atkinson, A.P., Tunstall, M.L., Dittrich, W.H.: Evidence for distinct contributions of form and motion information to the recognition of emotions from body gestures. Cognition 104, 59–72 (2007)
3. Bennett, M.R., Hacker, P.M.: Emotion and cortical-subcortical function: conceptual developments. Progress in Neurobiology 75, 29–52 (2005)
4. Bonda, E., Petrides, M., Ostry, D., Evans, A.: Specific involvement of human parietal systems and the amygdala in the perception of biological motion. J. Neurosci. 16, 3737–3744 (1996)
5. De Gelder, B., Snyder, J., Greve, D., Gerard, G., Hadjikhani, N.: Fear fosters flight: a mechanism for fear contagion when perceiving emotion expressed by a whole body. Proc. Natl. Acad. Sci. U.S.A. 101, 16701–16706 (2004)
6. Dittrich, W.H., Troscianko, T., Lea, S.E., Morgan, D.: Perception of emotion from dynamic point-light displays represented in dance. Perception 25, 727–738 (1996)

7. Grèzes, J., Pichon, S., de Gelder, B.: Perceiving fear in dynamic body expressions. Neuroimage 35, 959–967 (2007)
8. Hadjikhani, N., de Gelder, B.: Seeing fearful body expressions activates the fusiform cortex and amygdala. Current Biology 13, 2201–2205 (2003)
9. Hicheur, H., Glasauer, S., Vieilledent, S., Berthoz, A.: Head direction control during active locomotion in humans. In: Wiener, S.I., Taube, J.S. (eds.) Head Direction Cells and the Neural Mechanisms of Spatial Orientation (Bradford Book ed.), pp. 383–408. MIT Press, Cambridge (2005)
10. Hicheur, H., Terekhov, A.V., Berthoz, A.: Intersegmental coordination during human locomotion: does planar covariation of elevation angles reflect central constraints? J. Neurophysiol. 96, 1406–1419 (2006)
11. Omlor, L., Giese, M.A.: Unsupervised Learning of Spatio-temporal Primitives of Emotional Gait. In: André, E., Dybkjær, L., Minker, W., Neumann, H., Weber, M. (eds.) PIT 2006. LNCS (LNAI), vol. 4021, pp. 188–192. Springer, Heidelberg (2006)
12. Pichon, S., De Gelder, B., Grzes, J.: Emotional modulation of visual and motor areas by dynamic body expressions of anger. Social Neuroscience, 1–14 (2007)
13. Pollick, F.E., Paterson, H.M., Bruderlin, A., Sanford, A.J.: Perceiving affect from arm movement. Cognition 82, B51–B61 (2001)
14. Roether, C.L., Omlor, L., Giese, M.A.: Lateral asymmetry of bodily emotion expression. Current Biology 18, R329–R330 (2008)
15. Roether, C.L., Omlor, L., Christensen, A., Giese, M.A.: Critical features for the perception of emotion from gait. Journal of Vision 9(6), 1–32 (2009)
16. Rossignol, S., Dubuc, R., Gossard, J.P.: Dynamic Sensorimotor Interactions in Locomotion. Physiological Reviews 86, 89–154 (2006)
17. Slawinska, U., Kasicki, S.: Theta-like rhythm in depth EEG activity of hypothalamic areas during spontaneous or electrically induced locomotion in the rat. Brain Res. 678, 117–126 (1995)
18. Sprengelmeyer, R., Rausch, M., Eysel, U.T., Przuntek, H.: Neural structures associated with recognition of facial expressions of basic emotions. Proc. Biol. Sci. 265, 1927–1931 (1998)
19. Takakusaki, K., Saitoh, K., Harada, H., Kashiwayanagi, M.: Role of basal ganglia-brainstem pathways in the control of motor behaviors. Neurosci. Res. 50, 137–151 (2004)
20. Troje, N.F.: Decomposing biological motion: a framework for analysis and synthesis of human gait patterns. J. Vis. 2, 371–387 (2002)
21. Venture, G., Kadone, H., Zhang, T., Hicheur, H., Grèzes, J., Berthoz, A.: Recognizing Emotions Conveyed by Human Gait (under review)
22. Wallbott, H.G., Scherer, K.R.: Cues and channels in emotion recognition. Journal of Personality and Social Psychology 51, 690–699 (1986)

Whole Body Motion Control Framework for Arbitrarily and Simultaneously Assigned Upper-Body Tasks and Walking Motion

Doik Kim, Bum-Jae You, and Sang-Rok Oh

Abstract. A walking motion of a humanoid has been analyzed or developed without considering motions of the remaining parts of the humanoid. In order to perform tasks in the human's living environment, a walking motion and assigned tasks must be considered at the same time. In this paper, a whole body motion generation method, i.e., the motion embedded CoM Jacobian method is introduced. With the method, a balance control and assigned motions are separated and thus, the assigned motions can be generated without considering balance of a humanoid. As experimental examples, whole body motion of a humanoid is assigned by the teleoperation. Arbitrarily assigned upper body motions and independently generated walking motions are combined to generate a balanced whole body motion with the suggested methods.

1 Introduction

Recently, robot's working places are trying to be extended to the human's daily life. Among various robots, a humanoid is one of the most feasible robots that can survive in the human's living environment. In order to live in the human's living environment, a humanoid should overcome many obstacles such as stairs and furnishings, and understand how to manipulate tools and devices such as many kinds of doors and electric home appliances. All these elemental tasks are also combined to conduct more complicated tasks such as cleaning, cooking, errands, and etc.

For these complicated tasks, a humanoid must have at least two functions: mobility and manipulation. The mobility is one of the most important research topics for a humanoid and many excellent results have been reported in recent years[3]. The manipulation is one of traditional research topics in robotics and has been

Doik Kim · Bum-Jae You · Sang-Rok Oh
Interaction and Robotics Research Center, KIST, Seoul, Korea
e-mail: {doikkim,ybj,sroh}@kist.re.kr

successfully realized in many real robots. Nowadays, wheeled mobile robots with two arms are popular type for studying manipulation with mobility. This type of robots can handle objects in the human's living environment with less stability problem than a humanoid robot, and thus they can focus on the manipulation with traveling around a working environment.

In order to increase the usefulness of a humanoid, these two research topics, i.e., mobility and manipulation should be coordinated seamlessly with guaranteeing a balance of a humanoid. To achieve the coordination, this paper introduces a whole body motion generation method which resolves the CoM(Center of Mass) Jacobian of a humanoid with given motions of manipulation and mobility. The introduced methods can use almost all manipulation methods without considering balance and walking situation of a humanoid.

This paper is organized as follows: section 2 gives an overview of the resolution of the motion-embedded CoM(MECoM) Jacobian. Section 3 derives the MECoM Jacobin for several cases. Section 4 describes how to use the MECoM Jacobian and shows several applications. Finally, section 5 concludes the paper.

2   Overall System

In order to balance a humanoid with whole body motion, full dynamics or the CoM of whole body is studied usually. The CoM relation is much simpler than full dynamics and thus it is suitable for real implementation. Additionally, the CoM Jacobian gives a relation between the CoM and the joint motion similar to a normal kinematic relation. Consequently, the CoM Jacobian is one of the most simple and efficient whole body motion relations with balancing information. The CoM Jacobian is proposed by Kagami, et al.[5] and formulated analytically by Sugihara, et al.[9]. The dimension of the whole body motion relation is too complicated to be used in real time or with given task motions. If a humanoid has n-dof in total, then the dimension of the CoM Jacobian is $(3 \times n)$. A usual method to solve the CoM Jacobian is that given motions are augmented to the CoM Jacobian as constraints, and finally the joint motions are solved by using an optimization method. The augmentation of given motions increases the overall dimension of the optimization matrix and thus it needs heavy computation with given motions. In order to overcome the defect of the conventional resolution of the CoM Jacobian, the MECoM(Motion-Embedded CoM) Jacobian is proposed[1, 7].

Basic idea of the MECoM Jacobian is that, in most cases, human motions are occurred without considering balance of its body explicitly, and human motions are assigned locally and independently. For example, if a human carries an object with a certain pose, the aim of arms and legs are different, arms are to maintain pose of the object according to the body motion and legs are to walk toward a destination position. While these motions are assigned to corresponding sections, i.e., arms and legs, balance of the whole body is occurred subconsciously. Thus a human needs not consider the balance control of body consciously. This human behavior is

applied to control a humanoid with the resolution of the MECoM Jacobian, and consequently, the balancing problem is separated from given motions and furthermore, conventional motion planning methods can be applied without any modification to generate balanced motions.

As shown in Fig. 1, input of the MECoM Jacobian can be divided into two categories: 1) CoM-ZMP control for balance and 2) Motions for given tasks. The CoM-ZMP control algorithms are based on the inverted pendulum model. Many research results on the walking patterns are dealt with this balance algorithms[2, 4, 6]. Motion generation methods have been studied for a long time and mostly, the methods can be categories into the joint motion generation and the Cartesian motion generation.

The output of the MECoM Jacobian is the balanced joint motion of a humanoid, i.e., the MECoM Jacobian distributes the two independent inputs into the whole body joint motion which guarantees balance and given motions as much as possible.

Before developing the motion-embedded CoM Jacobian, a humanoid is divided into several parts. A humanoid has four limbs attached at the main body. Hereafter, each limb and the main body will be called a **section**. For a human, a task is done with some sections, usually two arms, and they balance the body with other sections, usually two legs. It means that not all sections are dedicated into one task, but they do their own tasks, for example, balancing or a given task. Thus it is natural that motions of each section are given independently. According to the existence of a given desired motion, the section is classified into **the idle section** and **the busy section**: the idle section has no given motion and the busy section has a given motion. If a section has zero motion, i.e., it is fixed at current position, it can be considered as an idle section or a busy section. In most case, a supporting section which is attached on the ground are considered as a busy section even if no motion is assigned to, since
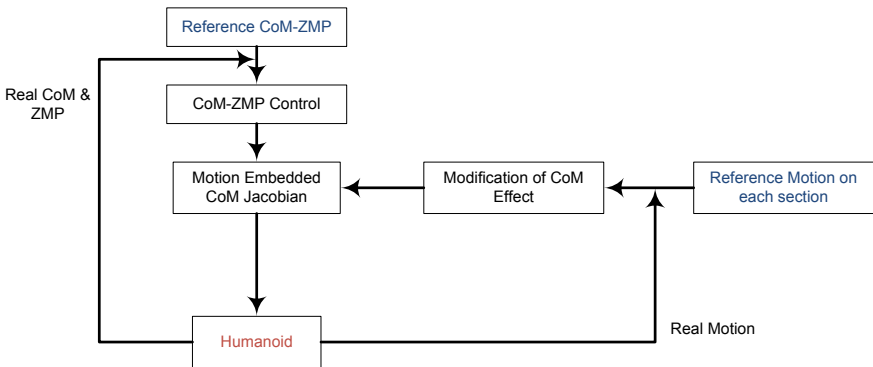
Fig. 1 A basic flow for the resolution of the MECoM Jacobian: balance is maintained by controlling the CoM-ZMP relation, and motions on several sections are given independently. Finally, the balancing control and the given motions are combined by solving the MECoM Jacobian to give balanced motions.

they have a role of balancing the body and thus they should be fixed at the position. Other sections such as arms with zero motion are considered as an idle section if no motion is assigned explicitly.

A given motion of each limb or a main body can be given as a joint motion and/or a Cartesian motion. If a busy section has a motion in the Cartesian space, it will be called **C-busy section** and a section with a motion in the joint space is **J-busy section**.

3   Derivation of Motion-Embedded CoM Jacobian

3.1   CoM Jacobian

This section briefly reviews the CoM Jacobian, and a detail description can be found in [9, 10]. Let us consider a n-DOF humanoid. There are two referential frames to describe a humanoid as shown in Fig. 2. The world coordinate frame is fixed on somewhere in the world and represents the global motion of a humanoid. The body center frame is fixed on a humanoid to describe the local motion. Almost all the properties of a humanoid is described in the body center frame. The leading superscript $^o(\cdot)$ implies that the elements are represented in the body center frame, and without it, a value is based on the world coordinate frame.

The CoM, $\mathbf{p}_G$, of a humanoid is a function of joint angle vector, $\mathbf{q}$, i.e., $\mathbf{p}_G = \mathbf{f}(\mathbf{q})$. Thus, there exists a Jacobian $\mathbf{J}_G$ which can relate $\dot{\mathbf{q}}$ to $\dot{\mathbf{p}}_G$ as:

$$\dot{\mathbf{p}}_G = \mathbf{J}_G \dot{\mathbf{q}} \tag{1}$$

where the $(3 \times n)$ matrix $\mathbf{J}_G$ is defined by

$$\mathbf{J}_G \equiv \frac{\partial \mathbf{p}_G}{\partial \mathbf{q}} \tag{2}$$

Fig. 2 Coordinate System of Mahru: There are two coordinate systems. The world coordinate frame is to represents the inertial coordinate system and the body center frame is attached on the humanoid body and all the local motions represented with leading superscript $^o(\cdot)$ are represented based on the body center frame.
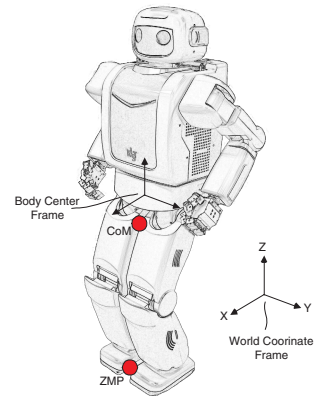
We call $\mathbf{J}_G$ the CoM Jacobian hereafter. $\mathbf{p}_G$ is a quite complex function with multiple arguments. Kagami, et al., proposed the numerical method to calculate it[5], which unfortunately needs a large amount of computation. Sugihara, et al., developed a fast and accurate calculation method of $\mathbf{J}_G$ with the following approach[10].

Firstly, the relative CoM velocity with respect to the body center frame, ${}^o\dot{\mathbf{p}}_G$, can be expressed as:

$$
{}^o\dot{\mathbf{p}}_G = \frac{\sum_{i=0}^{n-1} m_i {}^o\dot{\mathbf{r}}_{G_i}}{\sum_{i=0}^{n-1} m_i} = \frac{\sum_{i=0}^{n-1} m_i {}^o\mathbf{J}_{G_i}\dot{\mathbf{q}}}{\sum_{i=0}^{n-1} m_i}
\tag{3}
$$

where m_i is the mass of link i, ${}^o\mathbf{r}_{G_i}$ is the position of the center of mass of link i with respect to the body center frame, and ${}^o\mathbf{J}_{G_i}$ $(3 \times n)$ is defined by

$$
{}^o\mathbf{J}_{G_i} \equiv \frac{\partial {}^o\mathbf{r}_{G_i}}{\partial \mathbf{q}}
\tag{4}
$$

Therefore, Jacobian ${}^o\mathbf{J}_G$ which relates $\dot{\mathbf{q}}$ to ${}^o\dot{\mathbf{p}}_G$ is

$$
{}^o\mathbf{J}_G = \frac{\sum_{i=0}^{n-1} m_i {}^o\mathbf{J}_{G_i}}{\sum_{i=0}^{n-1} m_i}
\tag{5}
$$

Secondly, suppose link 1 is the **base section**, which is fixed in the world frame (for example, when the right leg is the supporting leg, the right leg is fixed on the ground and becomes the base section), the CoM velocity with respect to the world coordinates frame, $\dot{\mathbf{p}}_G$ is

$$
\begin{aligned}
\dot{\mathbf{p}}_G &= \dot{\mathbf{p}}_o + \omega_o \times \mathbf{R}_o {}^o\mathbf{p}_G + \mathbf{R}_o {}^o\dot{\mathbf{p}}_G \\
&= \mathbf{R}_o \{ {}^o\dot{\mathbf{p}}_G - {}^o\dot{\mathbf{p}}_1 + ({}^o\mathbf{p}_G - {}^o\mathbf{p}_1) \times {}^o\omega_1 \} \\
&= \mathbf{R}_o {}^o\mathbf{J}_G \dot{\mathbf{q}} \\
&\quad + \mathbf{R}_o \{ -{}^o\mathbf{J}_{\mathbf{v}_1} + [({}^o\mathbf{p}_G - {}^o\mathbf{p}_1) \times] {}^o\mathbf{J}_{\omega_1} \} \dot{\mathbf{q}}_1
\end{aligned}
\tag{6}
$$

where $\mathbf{p}_o$ is the position of the body center, ω_o is the angular velocity of the body center, and $\mathbf{R}_o$ is the attitude matrix of the body center with respect to the world frame. ${}^o\mathbf{p}_1$ is the position of the base section, ${}^o\omega_1$ is the angular velocity of the base section, ${}^o\mathbf{J}_{\mathbf{v}_1}$ is the linear velocity part of the base Jacobian and ${}^o\mathbf{J}_{\omega_1}$ is the angular velocity part of the base Jacobian with respect to the body center frame. $[\mathbf{v}\times]$ means outer-product matrix of a vector $\mathbf{v}$ (3×1). $\dot{\mathbf{q}}_i$ is the joint velocity of section i. Note that the base section can be any section that is fixed on the ground, but here, we assigned the base section with the number 1 without loss of generality.

In order to use Eq. (6) in the following section, it is rewritten here as:

$$
\begin{aligned}
\dot{\mathbf{p}}_G &= \sum_{i=1}^{m} \mathbf{R}_o {}^o\mathbf{J}_{G_i} \dot{\mathbf{q}}_i \\
&\quad + \mathbf{R}_o \{ -{}^o\mathbf{J}_{\mathbf{v}_1} + [({}^o\mathbf{p}_G - {}^o\mathbf{p}_1) \times] {}^o\mathbf{J}_{\omega_1} \} \dot{\mathbf{q}}_1
\end{aligned}
\tag{7}
$$

where m is the number of sections.

Now, let us derive the motion-embedded CoM Jacobian from Eq. (7).

3.2 Motion-Embedded CoM Jacobian

3.2.1 Embedment of J-Busy Section

It is easy to embed a joint motion into the CoM Jacobian, since the joint motion can be directly replaced $\dot{\mathbf{q}}_i$ in Eq. (7).

If section j is a J-busy section, Eq. (7) becomes

$$
\dot{\mathbf{p}}_G - \mathbf{R}_o{}^o\mathbf{J}_{G_j}\dot{\mathbf{q}}_j
$$
$$
= \sum_{i=1,i\neq j}^{m} \mathbf{R}_o{}^o\mathbf{J}_{G_i}\dot{\mathbf{q}}_i
$$
$$
+ \mathbf{R}_o\{-{}^o\mathbf{J}_{\mathbf{v}_1} + [({}^o\mathbf{p}_G - {}^o\mathbf{p}_1)\times]{}^o\mathbf{J}_{\omega_1}\}\dot{\mathbf{q}}_1 \tag{8}
$$

The second term of the left hand side compensates the motion of the jth section. Therefore, the other sections shown in the right hand side can generate a joint motion with the compensated CoM motion.

If at least one section, i.e., the base section, is an idle section, then Eq. (8) can compensate motions of the other sections. If all sections are the J-busy section, there is no section to compensate given motions. In this case, an optimization method needs to be applied.

3.2.2 Embedment of C-Busy Section

Let us derive the motion-embedded CoM Jacobian for the C-busy section. Each section of a humanoid is considered as an independent section, i.e., any section can have its own motion independently without considering other sections. In general, the i-th section has the following relation:

$$
{}^o\dot{\mathbf{x}}_i = {}^o\mathbf{J}_i\dot{\mathbf{q}}_i \tag{9}
$$

where ${}^o\dot{\mathbf{x}}_i = [{}^o\dot{\mathbf{p}}_i^T ; {}^o\omega_i^T]^T$ is the end point velocity of the section, ${}^o\dot{\mathbf{p}}_i$ and ${}^o\omega_i$ are the linear and the angular velocity, respectively. $\dot{\mathbf{q}}_i$ is the joint velocity, and ${}^o\mathbf{J}_i$ is the usual Jacobian matrix represented in the body center frame.

In our case, the body center is floating, and thus the end point motion about the world coordinate frame is written as follows:

$$
\dot{\mathbf{x}}_i = \mathbf{X}_i^{-1}\dot{\mathbf{x}}_o + \mathbf{X}_o{}^o\dot{\mathbf{x}}_i \tag{10}
$$

where $\dot{\mathbf{x}}_o = [\dot{\mathbf{p}}_o^T ; \omega_o{}^T]^T$ is the velocity of the body center represented in the world coordinate system, and

$$
\mathbf{X}_i = \begin{bmatrix} \mathrm{i}_3 & [\mathbf{R}_o{}^o\mathbf{r}_i\times] \\ \mathbf{0}_3 & \mathrm{i}_3 \end{bmatrix} \tag{11}
$$

is a (6×6) matrix which relates the body center velocity and the end point velocity of the i-th section. i_3 and $\mathbf{0}_3$ are the (3×3) identity and zero matrix, respectively.

$\mathbf{R}_o$ is the orientation of the body center based on the world coordinate system. $^o\mathbf{r}_i$ is the position vector from the body center to the end of the i-th section based on the body center frame. The transformation matrix $\mathbf{X}_o$ is

$$\mathbf{X}_o = \begin{bmatrix} \mathbf{R}_o & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{R}_o \end{bmatrix}. \tag{12}$$

By combining Eq. (9) and Eq. (10), the end point velocity based on the world coordinate system is

$$\dot{\mathbf{x}}_i = \mathbf{X}_i^{-1}\dot{\mathbf{x}}_o + \mathbf{X}_o{}^o\mathbf{J}_i\dot{\mathbf{q}}_i \tag{13}$$

For simplicity, we will use the relation $\mathbf{J}_i = \mathbf{X}_o{}^o\mathbf{J}_i$, hereafter.

From Eq. (13), we can see that all sections should satisfy the compatibility condition, i.e., *the body center velocity, $\dot{\mathbf{x}}_o$, in Eq. (13) for each section is the same, so that sections are connected without being broken.*, and thus arbitrary two sections, for example, the i-th and j-th section should satisfy the following relation:

$$\mathbf{X}_i(\dot{\mathbf{x}}_i - \mathbf{J}_i\dot{\mathbf{q}}_i) = \mathbf{X}_j(\dot{\mathbf{x}}_j - \mathbf{J}_j\dot{\mathbf{q}}_j). \tag{14}$$

From Eq. (14), the joint velocity of any section can be represented by the joint velocity of any other section. However, all sections will be represented by the base section, since the motion of the body center is represented by the base section, as shown in Eq. (7). The base section can be the supporting leg in the single supporting phase or one of both legs in the double supporting phase if a humanoid is standing. Let us express the base section with the subscript 1, then the joint velocity of any section is expressed as:

$$\dot{\mathbf{q}}_i = \mathbf{J}_i^{-1}\dot{\mathbf{x}}_i - \mathbf{J}_i^{-1}\mathbf{X}_{i1}(\dot{\mathbf{x}}_1 - \mathbf{J}_1\dot{\mathbf{q}}_1), \tag{15}$$

for $i = 2, \cdots, m$, where m is the number of sections. Here,

$$\mathbf{X}_{i1} = \begin{bmatrix} \mathbf{i}_3 & [\mathbf{R}_o(^o\mathbf{r}_i - {}^o\mathbf{r}_1)\times] \\ \mathbf{0}_3 & \mathbf{i}_3 \end{bmatrix}. \tag{16}$$

Note that if a section is a redundant system, any null space optimization scheme can be added in Eq. (15), and $\mathbf{J}_i^{-1}$ becomes a generalized inverse.

By substituting Eq. (15) into Eq. (7), the motion-embedded CoM Jacobian relation becomes

$$\begin{aligned}
\dot{\mathbf{p}}_G = \;& \mathbf{R}_o\{-{}^o\mathbf{J}_{\mathbf{v}_1} + [(^o\mathbf{p}_G - {}^o\mathbf{p}_1)\times]^o\mathbf{J}_{\omega_1}\}\dot{\mathbf{q}}_1 \\
& + \sum_{i=1}^{m} \mathbf{R}_o{}^o\mathbf{J}_{G_i}\mathbf{J}_i^{-1}(\dot{\mathbf{x}}_i - \mathbf{X}_{i1}\dot{\mathbf{x}}_1) \\
& + \sum_{i=1}^{m} \mathbf{R}_o{}^o\mathbf{J}_{G_i}\mathbf{J}_i^{-1}\mathbf{X}_{i1}\mathbf{J}_1\dot{\mathbf{q}}_1
\end{aligned} \tag{17}$$

where $\mathbf{J}_{\mathbf{v}_1} = \mathbf{R}_o{}^o\mathbf{J}_{\mathbf{v}_1}$ and $\mathbf{J}_{\boldsymbol{\omega}_1} = \mathbf{R}_o{}^o\mathbf{J}_{\boldsymbol{\omega}_1}$ are the linear and angular velocity part of the base section Jacobian. Note that if $i = 1$, $\dot{\mathbf{x}}_i - \mathbf{X}_{i1}\dot{\mathbf{x}}_1 = \mathbf{0}$ and $\mathbf{R}_o{}^o\mathbf{J}_{G_i}\mathbf{J}_i^{-1}\mathbf{X}_{i1}\mathbf{J}_1\dot{\mathbf{q}}_1 = \mathbf{R}_o{}^o\mathbf{J}_{G_i}\dot{\mathbf{q}}_1$.

Finally, all desired motions, $\dot{\mathbf{x}}_i$, are embedded in the modified CoM Jacobian. Thus the effect of the CoM movement generated by the desired motion is compensated by base section. Eq. (17) can be rewritten like the usual kinematic Jacobian of the base section:

$$\dot{\mathbf{p}}_{\text{meG}} = \mathbf{J}_{\text{meG}}\dot{\mathbf{q}}_1 \tag{18}$$

where

$$\dot{\mathbf{p}}_{\text{meG}} = \dot{\mathbf{p}}_G - \sum_{i=1}^{m} \mathbf{R}_o{}^o\mathbf{J}_{G_i}\mathbf{J}_i^{-1}(\dot{\mathbf{x}}_i - \mathbf{X}_{i1}\dot{\mathbf{x}}_1), \tag{19}$$

$$\mathbf{J}_{\text{meG}} = \mathbf{R}_o\{-{}^o\mathbf{J}_{\mathbf{v}_1} + [({}^o\mathbf{p}_G - {}^o\mathbf{p}_1)\times]{}^o\mathbf{J}_{\boldsymbol{\omega}_1}\}$$
$$+ \sum_{i=1}^{m} \mathbf{R}_o{}^o\mathbf{J}_{G_i}\mathbf{J}_i^{-1}\mathbf{X}_{i1}\mathbf{J}_1. \tag{20}$$

The modified CoM motion, $\dot{\mathbf{p}}_{\text{meG}}$, consists of two relations: a desired CoM motion(the first term) and the relative effect of motions of each section(the second term). The modified CoM Jacobian, $\mathbf{J}_{\text{meG}}$ also consists of two relations: the effect of the body center(the first term) and the effect of motions of each section(the second term).

The modified CoM Jacobian $\mathbf{J}_{\text{meG}}$ is a $(3 \times n_1)$ matrix where n_1 is the dimension of the base section, which is smaller than that of the original CoM Jacobian. For example, Mahru in Fig. 2 has a 6-dof leg, and thus $n_1 = 6$ if the leg is the base section. After solving Eq. (18), the joint motion of the base section is obtained. The resulting base section motion balances a humanoid robot automatically during the movement of all other sections. With the resulting joint motion of the base section, the joint motion of all other sections are obtained by Eq. (15). The resulting motion follows the desired motion, regardless of the balancing motion of the base section.

4 Application of MECoM Jacobian

4.1 Embedment of a Motion into MECoM Jacobian

As shown in the previous section, there is no difference between manipulation of arms and walking of legs. They are just categorized which type of motion is assigned to a certain section. Currently all sections can have any type of motions, but the base section must have the Cartesian motion, which is usually the zero motion i.e., fixed on the ground. Within the MECoM framework, all motions including walking motion are considered as manipulation or motion with constraints, and the balance is guaranteed by the CoM-ZMP controllers, and finally all these manipulation and balancing results are reflected on the base section.

The balance control focuses on CoM-ZMP controllers, i.e., by changing the CoM-ZMP controller, more stable motion can be obtained without considering given motions. The balance control is tightly related to the CoM-ZMP pattern generation.

According to given motions, the motion embedded CoM Jacobian has two different forms as shown in the previous section: if a given motion is a joint motion, Eq. (8) is used, i.e., the given motion is just replaced. If a given motion is a Cartesian motion, Eq. (15) is used, i.e., the joint motion is obtained from inverse kinematics relation between the given Cartesian motion and the base section.

Consequently, by changing motion generation methods represented in Eq. (15) explicitly or by substituting the results of motion generation methods into Eq. (8) implicitly, we can have more sophisticated motion of a humanoid.

In order to show the replacement of motion generation algorithms, let us consider that a humanoid has arms with more than 6-dof, i.e., arms are redundant. A given desired arm motion is $\dot{\mathbf{x}}_{d_i}$ for the ith section, then we can embed this motion into the MECoM with two methods as follows: Firstly, the given motion is pre-calcuated before embedding into the MECoM Jacobian

$$\dot{\mathbf{q}}_{d_i} = \mathbf{J}_i^\dagger \dot{\mathbf{x}}_{d_i} \tag{21}$$

where $(\cdot)^\dagger$ represents the pseudo-inverse, and the null vector related part is not included in the equation, but it can be easily added in the equation. With this equation, a desired joint motion, $\dot{\mathbf{q}}_{d_i}$ can be obtained and it is embedded into the MECoM Jacobian as a joint motion with Eq. (8). This joint motion embedment is cascaded with the MECoM Jacobian and thus motion generation algorithms are separated perfectly and it can be developed independently as in Eq. (21).

The second method is that the given motion $\dot{\mathbf{x}}_{d_i}$ is embedded explicitly with Eq. (15) as follows:

$$\dot{\mathbf{q}}_i = \mathbf{J}_i^\dagger \dot{\mathbf{x}}_{d_i} - \mathbf{J}_i^\dagger \mathbf{X}_{i1}(\dot{\mathbf{x}}_1 - \mathbf{J}_1 \dot{\mathbf{q}}_1) \tag{22}$$

Note that the null vector related part also can be included in the equation. Eq. (15) in the previous section must be replaced by Eq. (22) and then the final form of the MECoM Jacobian relation is also changed according to the motion generation method as in Eq. (22). This Cartesian motion embedment is an explicit embedment of given motions and the MECoM Jacobian resolution has all the motion generation routines and we don't need additional motion generation routines in this case.

We can develop or use a new motion generation method to perform a certain task without any modification or consideration of the balance of the body as in Eq. (21) and Eq. (22). The following section will show several applications of the resolution of the MECoM Jacobian.

4.2 Applications

As shown in Fig. 2, the MECoM Jacobian method is applied to the humanoid, **Mahru**, developed at the Cognitive Robotics Center, KIST in 2004. Mahru has the

height of about 150cm, and the weight is about 67kg. It has 6-dof for each legs and arms, 1-dof for the waist, 2-dof for the neck, and each hand has 4-dof. In total, it has 35-dof.

4.2.1   Tele-Operation

A humanoid has many degrees-of-freedom, and thus it is hard to control its whole body in real time. To overcome these difficulties, we developed a tele-operation system. In order for the portability, we used a motion capture suit, and human motions are captures from it. The captured motions are interpreted into the humanoid motions, and it is transferred to the humanoid.

As indicated in the section 3.2, we can assign independent motions to each section of the humanoid. For a human-like motion, the upper part of the humanoid is controlled by the joint motion. For stability, the lower part of the humanoid is controlled by the Cartesian motion. From foot-prints of the lower part, the CoM-ZMP pattern is calculated. Consequently, we can assign whole body motion and reference CoM-ZMP for the balance to the humanoid in real time, as shown in Fig. 3. A detail explanation on this application can be found in [8].

Fig. 3 Teleoperation with mixed given motion: The upper body of the humanoid has a joint motion comes from the motion capture suit and the lower body of the humanoid has a Cartesian motion which represents the foot print. Two different input motions are combined with the MECoM Jacobian. The lower body of the humanoid has delayed by about two steps because of the detection procedure of the human walking motion.

4.2.2   Door Opening

In order to interact with an object in the environment, we added a force control algorithm to the upper body motion generation routine. A compliant motion is generated from the force sensor attached on the wrist, and this motion is combined with the captured motion. The final joint motions are embedded into the MECoM Jacobian routine with Eq. (8). The combination of the compliant motion and the captured motion is pre-calculated before embedding. Fig. 4 shows that the humanoid, Mahru, opens a door with given tele-operated motions and force controlled motion.

Fig. 4 Teleoperation with force control: In order to open a door, we embedded a compliance control algorithm into the upper body motion. By doing this, the embedded motion can follow the captured motion and can interact with an object in the environment simultaneously.

5   Concluding Remarks

In this paper, the resolution of the motion embedded CoM Jacobian is introduced as a whole-body motion generation method. In order for a humanoid to survive in the human's daily life, a whole body motion generation method will be critical and the MECoM Jacobian method is suggested as one of promising methods. With the MECoM Jacobian method, a walking motion is also a type of manipulation and it can be handled as a usual motion generation method. The supporting leg or the base section is the only part that is dedicated to the balance of the whole body and it is affected by the balance control which is separated from given motions. Most conventional motion generation methods can be embedded seamlessly as a joint motion or a Cartesian motion into the MECoM Jacobian method. By dividing a complicated whole-body motion generation problem into several independent problems such as a balance control problem and a motion generation problem, it is possible to perform complicated tasks as shown in section 4.

References

1. Choi, Y., Kim, D., Oh, Y., You, B.J.: Posture/walking control for humanoid robot based on kinematic resolution of com jacobian with embedded motion. IEEE Trans. on Robotics 23(6), 1285–1293 (2007)
2. Choi, Y., You, B.J., Oh, S.R.: On the stability of indirect ZMP controller for biped robot systems. In: International Conference on Intelligent Robots and Systems, pp. 1966–1971 (2004)
3. Hirai, K., Hirose, M., Haikawa, Y., Takenaka, T.: The development of honda humanoid robot. In: Proceedings of the IEEE International Conference on Robotics and Automation, vol. 2, pp. 1321–1326 (1998)
4. Hong, S., Oh, Y., Kim, D., You, B.J.: A walking pattern generation method with feedback and feedforward control for humanoid robots. In: IROS 2009: Proceedings of the 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 1078–1083. IEEE Press, Piscataway (2009)

5. Kagami, S., Kanehiro, F., Tamiya, Y., Inaba, M., Inoue, H.: Autobalancer: An online dynamic balance compensation scheme for humanoid robots. In: 4th International Workshop on Algorithmic Foundation on Robotics, WAFR 2000 (2000)
6. Kajita, S., Kanehiro, F., Kaneko, K., Fujiwara, K., Harada, K., Yokoi, K., Hirukawa, H.: Biped walking pattern generation by using preview control of zero-moment point. In: Proceedings of the IEEE International Conference on Robotics and Automation, ICRA 2003, vol. 2, pp. 1620–1626 (2003)
7. Kim, D., Choi, Y., Kim, C.: Motion-embedded cog jacobian for a real-time humanoid motion generation. In: 2nd International Conference on Informatics in Control, Automation and Robotics, ICINCO 2005, pp. 55–61 (2005)
8. Kim, S.K., Hong, S.M., Kim, D.: A walking motion imitation framework of a humanoid robot by human walking recognition from imu motion data. In: International Conference on Humanoid Robots, Humanoid 2009, pp. 343–348 (2009)
9. Sugihara, T., Nakamura, Y.: Whole-body cooperative balancing of humanoid robot using cog jacobian. In: International Conference on Intelligent Robots and Systems, pp. 2575–2580. EPFL, Lausanne (2002)
10. Sugihara, T., Nakamura, Y., Inoue, H.: Realtime humanoid motion generation through ZMP manipulation based on inverted pendulum control. In: IEEE International Conference on Robotics and Automation, Washington, DC, pp. 1404–1409 (2002)

Structure Preserving Optimal Control of Three-Dimensional Compass Gait

Sigrid Leyendecker, David Pekarek, and Jerrold E. Marsden

Abstract. The benefits of structure preserving algorithms for the numerical time-integration of mechanical systems, also called mechanical integrators, are widely accepted in forward dynamic simulations. However, in the field of motion planning and optimal control via direct methods, so far, these benefits have been less used. The dynamic optimisation method DMOC, does exploit the structure preserving properties of a variational integrator within an optimal control problem. This work considers the optimal control of a bipedal compass gait by modeling the double stance configuration as a transfer of contact constraints between the feet and the ground and develops a structure preserving simulation method for this context.

1 Introduction

When planning or predicting motion of multibody systems, one can pursue quite different strategies. One possibility is to rely purely on kinematic considerations. One can capture motion with a camera or simply prescribe certain desired poses for the motion. This information can be used as input for inverse kinematics, where a trajectory, meeting the prescribed conditions is reconstructed. However, thereby no dynamics is taken into account. If one is interested in the forces that cause dynamics, then one is faced with a control problem. In this paper we consider the problem of determining an optimal control that produces a walking gait for a three-dimensional

Sigrid Leyendecker
University of Erlangen-Nuremberg, Konrad-Zuse-Straße 3/5, 91052 Erlangen, Germany
e-mail: `sigrid.leyendecker@ltd.uni-erlangen.de`

David Pekarek
Northwestern University, 2145 Sheridan Road, Evanston, IL 60208
e-mail: `d-pekarek@northwestern.edu`

Jerrold E. Marsden
California Institute of Technology, 1200 E. California Boulevard, Pasadena, CA 91125

compass biped model. This control task has been previously addressed with various biped models in the literature, for instance in [1, 2]. However, this work is unique in the use of the DMOCC dynamic optimisation method [3] which exploits the geometric structure [4] and variational dynamics of the biped model [5] and yields a structure preserving simulation. DMOCC is a constrained version of the previously developed method called discrete mechanics and optimal control DMOC, see [6, 7, 8]. DMOC and DMOCC can be classified as direct methods that find local solutions of nonlinear optimal control problems, i.e. the state and control variables are discretised directly in order to transform the optimal control problem into a finite dimensional nonlinear constrained optimisation problem that can be solved by standard nonlinear optimisation techniques such as sequential quadratic programming (see e.g [9, 10]). This is in contrast to other methods like, e.g. shooting [11, 12, 13, 14], multiple shooting [15, 16], or collocation methods [17, 18], relying on a direct integration of the associated ordinary differential equations parametrised by states and controls or the controls only (see also [19] and [20] for an overview).

Part of the constraints retricting the optimisation problem are the discrete equations of motion. In general, the standard derivation of forward dynamics integration methods starts with a continuous equation of motion and replace the continuous quantities, in particular the derivatives with respect to time, by discrete approximations. In contrast to that, the variational theory of discrete mechanics provides a theoretical framework that parallels continuous variational dynamics. Discrete analogues to the Euler-Lagrange equations, Noethers theorem, and the Legendre transform are derived from a discrete Lagrangian by performing similar steps as in the continuous theory. The resulting time stepping schemes are structure preserving, i.e. they are symplectic-momentum conserving and exhibit good energy behaviour, meaning that no artificial dissipation is present and the energy error stays bounded over longterm simulations. To be more specific, the term structure preserving means that the approximate solution, i.e. the discrete trajectory, inherits certain characteristic properties of the continuous motion. For example, the evolution of the system's momentum maps (e.g. angular momentum is a momentum map for the biped) exactly represents externally applied forces, in particular they are conserved along the approximate motion of unforced systems. In addition to momentum maps, the symplectic structure underlying dynamics is respected, and as a consequence, geometric integrators (sometimes also termed mechanical integrators) yield good energy behaviour. See e.g. [21] for a short introduction to the basic idea of structure preserving integration while [22, 23] are detailed and elaborate works on geometric mechanics in the continuous time and discrete time setting. Furthermore, the introduction of [24] gives a short overview on works on symplectic integrators in various different contexts. Besides improving the fidelity of the approximate solution compared to standard methods, the preservation of these quantities stabilises the numerical integration. On the one hand, this is important for longterm forward dynamics simulations since it guarantees that energy is neither gained nor dissipated numerically. On the other hand, the inheritance of the continuous dynamics' structural

properties increases the robustness of the approximate solution, i.e. also with a relatively coarse time grid, meaningful results can be obtained in optimal control simulations (requiring comparatively low computation times) which can then be used as an initial guess for simulations with higher resolution. Based on the choice of the discrete Lagrangian in the variational integrator, the accuracy of the resulting approximate trajectory can be influenced, see [8] for an analysis of DMOC.

In contrast to many other works taking a rotation-based approach in minimal coordinates to multibody systems, (see e.g. [25, 26]), here, the multibody system is described in terms of redundant coordinates subject to holonomic constraints. On the one hand, difficulties and in particular singularities associated with rotational parameters are circumvented in this way. On the other hand, the formulation of complex three-dimensional multibody systems is easily possible in a straightforward and intuitive way. The resulting equations of motion assume the form of index three DAEs for which structure preserving integration methods are well developed, see [27, 28]. Disadvantages like the large dimension and possible ill-conditioning of the resulting discrete nonlinear system due to the presence of the Lagrange multipliers can be remedied by using the discrete null space method that eliminates the constraint forces. Details on the discrete null space method in the context of forward dynamic integration can be found in [29, 30, 31]. An extension to the optimal control of multibody systems can be found in [3] on which this work here is relying heavily. However, no contact between bodies is considered there. In the context of the walker, the change between the stance and the swing leg during the double stance configuration imposes additional challenges, wherefore in this paper, the variational formulation of [32] is developed further to describe this transfer of contacts. While the variational theory for nonsmooth systems is just mentioned briefly in this paper, details can be found in [33].

Section 2 introduces the biped model and gives details on the constrained multibody formulation. The continuous optimal control problem for the walker is formulated in Section 3, while the corresponding problem in discrete time is described in Section 4. Finally, computational results are demonstrated in Section 5.

2 Compass Gait Walker Model

In this work, a relatively simple model is used to illustrate the performance of the developed structure preserving numerical simulation method. The three-dimensional compass biped is modelled as a spherical kinematic pair in which the rigid legs are combined at the hip by a spherical joint, see Figure 1. A point mass in the hip represents the weight of the upper part of the body.

The contact between a foot and the ground is modelled as a perfectly plastic impact, constraining the foot to stay fixed on the ground during the other leg's swing phase. The contact is transferred instantaneously when the second foot hits the ground and the first one is released. During a swing phase, the walker has six
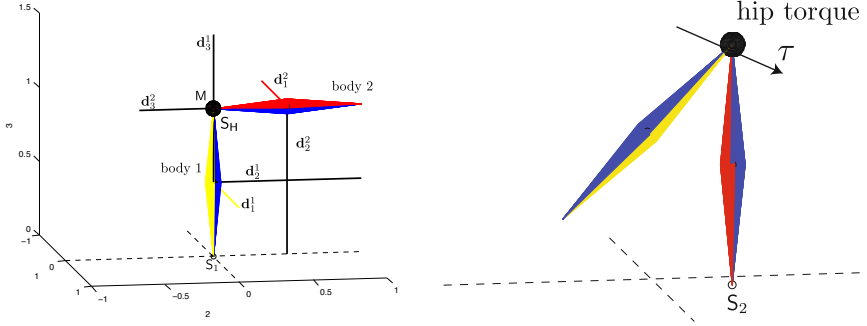
Fig. 1 Compass biped model with directors (left) and with actuating torque in the hip joint (right)

degrees of freedom. However, only a three-dimensional torque acts in the hip joint yielding an underactuated system.

2.1 Multibody Configuration

A constrained formulation is used for the dynamics of the complete multibody system as well as for a single rigid body (see e.g. [34, 35, 30]). The $n = 27$-dimensional time-dependent configuration variable of the walker $\mathbf{q}(t) \in Q = \mathbb{R}^{27}$ in the time interval $[t_0, t_N] \subset \mathbb{R}$ consists of the configurations of the two rigid bodies $\mathbf{q}^1, \mathbf{q}^2 \in \mathbb{R}^{12}$ and the placement $\mathbf{q}^M \in \mathbb{R}^3$ of the point mass M^M in the hip. It reads

$$\mathbf{q}(t) = \begin{bmatrix} \mathbf{q}^1(t) \\ \mathbf{q}^2(t) \\ \mathbf{q}^M(t) \end{bmatrix} \quad \text{with} \quad \mathbf{q}^\alpha(t) = \begin{bmatrix} \varphi^\alpha(t) \\ \mathbf{d}_1^\alpha(t) \\ \mathbf{d}_2^\alpha(t) \\ \mathbf{d}_3^\alpha(t) \end{bmatrix} \quad \alpha = 1, 2$$

where $\varphi^\alpha \in \mathbb{R}^3$ denotes the placement of the centre of mass and the directors $\mathbf{d}_I^\alpha \in \mathbb{R}^3, I = 1, 2, 3$ represent the orientation of the α-th body. Each director triad is constrained to stay orthonormal during the motion, see Figure 2.

The α-th body's Euler tensor with respect to the centre of mass can be related to the inertia tensor $\mathbf{J}^\alpha$ via $\mathbf{E}^\alpha = \frac{1}{2}(\mathrm{tr}\mathbf{J}^\alpha)\mathbf{I} - \mathbf{J}^\alpha$, where $\mathbf{I}$ denotes the 3×3 identity matrix. The principal values of the Euler tensor E_i^α together with the body's total mass M_φ^α are ingredients in the α-th rigid body's mass matrix. The constant symmetric positive definite mass matrix of the multibody system reads

$$\mathbf{M} = \mathrm{diag}(M_\varphi^1 \mathbf{I} \quad E_1^1 \mathbf{I} \quad E_2^1 \mathbf{I} \quad E_3^1 \mathbf{I} \quad M_\varphi^2 \mathbf{I} \quad E_1^2 \mathbf{I} \quad E_2^2 \mathbf{I} \quad E_3^2 \mathbf{I} \quad M^M \mathbf{I}).$$

Fig. 2 Configuration of a rigid body with respect to an orthonormal frame $\{\mathbf{e}_I\}$ fixed in space
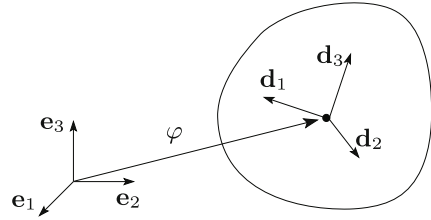
2.2 Constraints

The redundant configuration variable is subject to two types of constraints. On the one hand, internal constraints ensure that all bodies are rigid. On the other hand, external constraints combine the bodies by joints into the walker model. Rigidity of the two bodies gives rise to orthonormality constraints for the two director triads, thus there are $m_{int} = 12$ internal constraints.

$$\mathbf{g}_{int}(\mathbf{q}) = \begin{bmatrix} \mathbf{g}_{int}^1(\mathbf{q}) \\ \mathbf{g}_{int}^2(\mathbf{q}) \end{bmatrix} \quad \text{with} \quad \mathbf{g}_{int}^\alpha(\mathbf{q}^\alpha) = \frac{1}{2}\left((\mathbf{d}_j^\alpha)^T \cdot \mathbf{d}_k^\alpha - \delta_{jk}\right) = 0 \quad \begin{array}{l} j,k = 1,2,3 \\ \alpha = 1,2 \end{array}$$

During the second leg's swing phase, the first foot is fixed on the ground in x_{S_1} by a spherical joint S_1, see Figure 1. The corresponding constraint is $\mathbf{g}_{S_1}(\mathbf{q}) = \mathbf{0}$. Furthermore, the spherical joint S_H connects the two legs in the hip via $\mathbf{g}_{S_H}(\mathbf{q}) = \mathbf{0}$ and the point mass is held in place by the condition $\mathbf{g}_M(\mathbf{q}) = \mathbf{0}$, thus the total number of external constraints is $m_{ext} = 9$.

It is assumed that a perfectly plastic impact with no sliding takes place [36], when the second foot hits the ground (i.e. it hits the contact surface in the unknown place x_{S_2}), the foot is fully immobilised by the constraint $\mathbf{g}_{S_2}(\mathbf{q}) = \mathbf{0}$. Depending on the actual phase of the gait, the relevant constraints are collected in the $m = m_{int} + m_{ext} = 21$-dimensional constraint function vector $\mathbf{g}_1$ or $\mathbf{g}_2$ given by

$$\mathbf{g}_1(\mathbf{q}) = \begin{bmatrix} \mathbf{g}_{int}(\mathbf{q}) \\ \mathbf{g}_{S_1}(\mathbf{q}) \\ \mathbf{g}_{S_H}(\mathbf{q}) \\ \mathbf{g}_M(\mathbf{q}) \end{bmatrix} \quad \text{or} \quad \mathbf{g}_2(\mathbf{q}) = \begin{bmatrix} \mathbf{g}_{int}(\mathbf{q}) \\ \mathbf{g}_{S_2}(\mathbf{q}) \\ \mathbf{g}_{S_H}(\mathbf{q}) \\ \mathbf{g}_M(\mathbf{q}) \end{bmatrix} \quad \text{with} \quad \begin{array}{l} \mathbf{g}_{S_1}(\mathbf{q}) = \varphi^1 + \rho_{S_1}^1 - \mathbf{x}_{S_1} \\ \mathbf{g}_{S_2}(\mathbf{q}) = \varphi^2 + \rho_{S_2}^2 - \mathbf{x}_{S_2} \\ \mathbf{g}_{S_H}(\mathbf{q}) = \varphi^1 + \rho_{S_H}^1 - \varphi^2 - \rho_{S_H}^2 \\ \mathbf{g}_M(\mathbf{q}) = \mathbf{q}^M - \rho_{S_H}^1 - \varphi^1 \end{array}$$

where the vectors $\rho_J^\alpha = (\rho_J^\alpha)_I \mathbf{d}_I^\alpha$ point from the center of mass of the α-th body to the specific joint $J \in \{S_1, S_2, S_H\}$. Altogether, the walker model has $n - m = 6$ degrees of freedom.

Furthermore, path constraints $h(\mathbf{q}) \geq 0$ depending on the geometry of the legs ensure that the walker does not penetrate itself during the gait. For the actual double coned legs with radius r, the path constraints read

$$h(\mathbf{q}) = \left\| \varphi^2 - \varphi^1 \right\| - 2r \geq 0$$
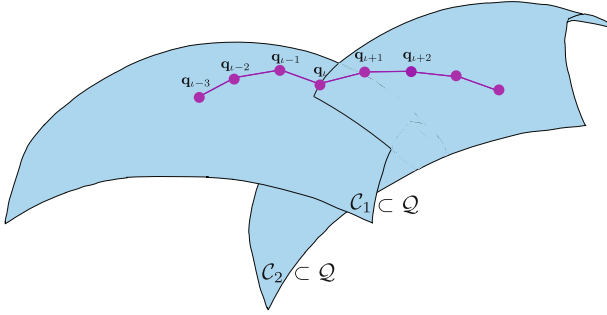
Fig. 3 When the second foot hits the ground, the first foot is released, thus the system instantaneously leaves the constraint manifold C_1 and enters the constraint manifold C_2

2.3   Transfer of Contact

It is important to note that the placement of the second foot on the ground is not known a priori. The (scalar) contact condition for the second foot reads $g_c(\mathbf{q}) = (\varphi^2 + \rho_{S_2}^2) \cdot \mathbf{e}_3 = 0$. In the instant the contact takes place, the point of contact $\mathbf{x}_{S_2}$ is determined which then defines the constraint function $\mathbf{g}_{S_2}$. The corresponding constraint manifolds are defined as $C_1 = \{\mathbf{q} \in \mathbb{R}^{27} | \mathbf{g}_1(\mathbf{q}) = \mathbf{0}\}$ and $C_2 = \{\mathbf{q} \in \mathbb{R}^{27} | \mathbf{g}_2(\mathbf{q}) = \mathbf{0}\}$, respectively. The transfer of contact is illustrated for the discrete trajectory in Figure 3.

In general, contact conditions are unilateral constraints. Thus, in the presented case, when modeling the transfer of contact as the concurrent release of the bilateral constraint $\mathbf{g}_1$ and the establishing of the new bilateral constraint $\mathbf{g}_2$, one has to verify for the resulting motion that the constraint forces point into the ground, thus they prevent the foot from penetrating the ground and do never prevent the lifting of the foot. Furthermore, the velocity of the previous point of contact (the just released previous stance foot) must have a positive component normal to the contact surface.

2.4   Nullspace Matrix and Nodal Reparametrisation

In DMOCC, the system of discrete equations of motion (being subject to the kinematic constraints described in Section 2.2) serves as constraints for the optimisation. To reduce the system's dimension to the minimal possible number, the discrete null space method is used, see [31, 3]. For each swing phase, the null space matrix and nodal reparametrisation used later in Section 4.1 are given here.

The $n \times (n-m)$ null space matrices $\mathbf{P}_1(\mathbf{q}) : \mathbb{R}^{n-m} \to T_{\mathbf{q}}C_1$ and $\mathbf{P}_2(\mathbf{q}) : \mathbb{R}^{n-m} \to T_{\mathbf{q}}C_2$ mapping to the tangent space of the constraint manifold in the specific gait phase read

$$\mathbf{P}_1(\mathbf{q}) = \begin{bmatrix} \widehat{\rho_{S_1}^1} & \mathbf{0} \\ -\widehat{\mathbf{d}_1^1} & \mathbf{0} \\ -\widehat{\mathbf{d}_2^1} & \mathbf{0} \\ -\widehat{\mathbf{d}_3^1} & \mathbf{0} \\ \widehat{\rho_{S_1}^1 - \rho_{SH}^1} & \widehat{\rho_{SH}^2} \\ \mathbf{0} & -\widehat{\mathbf{d}_1^2} \\ \mathbf{0} & -\widehat{\mathbf{d}_2^2} \\ \mathbf{0} & -\widehat{\mathbf{d}_3^2} \\ \widehat{\rho_{S_1}^1 - \rho_H^1} & \mathbf{0} \end{bmatrix} \quad \text{and} \quad \mathbf{P}_2(\mathbf{q}) = \begin{bmatrix} \widehat{\rho_{SH}^1} & \widehat{\rho_{S_2}^2 - \rho_{SH}^2} \\ -\widehat{\mathbf{d}_1^1} & \mathbf{0} \\ -\widehat{\mathbf{d}_2^1} & \mathbf{0} \\ -\widehat{\mathbf{d}_3^1} & \mathbf{0} \\ \mathbf{0} & \widehat{\rho_{S_2}^2} \\ \mathbf{0} & -\widehat{\mathbf{d}_1^2} \\ \mathbf{0} & -\widehat{\mathbf{d}_2^2} \\ \mathbf{0} & -\widehat{\mathbf{d}_3^2} \\ \mathbf{0} & \widehat{\rho_{S_2}^2 - \rho_{SH}^2} \end{bmatrix}$$

The matrix $\mathbf{Q}(\mathbf{q}) : \mathbb{R}^n \to \eta(T_{\mathbf{q}}^*C)$ projects onto the embedding of the cotangent space $T_{\mathbf{q}}^*C$ in $\mathbb{R}^n$ and is canonical for regular Lagrangians. It is given by

$$\mathbf{Q}(\mathbf{q}) = \mathbf{I}_{n \times n} - \mathbf{G}^T \cdot \left[\mathbf{G} \cdot \mathbf{M}^{-1} \cdot \mathbf{G}^T \right]^{-1} \cdot \mathbf{G} \cdot \mathbf{M}^{-1}$$

where $\mathbf{G}$ denotes the Jacobian of the constraints. An equidistant time grid $\{t_0, t_0 + \Delta t, \ldots, t_0 + N\Delta t = t_N\}$ is defined using the constant time step $\Delta t \in \mathbb{R}$ and the discrete approximation to the configuration at a time node reads $\mathbf{q}_n \approx \mathbf{q}(t_n)$. The discrete generalised coordinates $\mathbf{u}_{n+1} = \begin{bmatrix} \theta_{n+1}^1 \\ \theta_{n+1}^2 \end{bmatrix}$ consist of the incremental rotation vectors $\theta_{n+1}^1, \theta_{n+1}^2 \in \mathbb{R}^3$ for the two bodies. With the corresponding rotation matrices obtained via the exponential map $\exp : so(3) \to SO(3)$, the nodal reparametrisations $\mathbf{q}_{n+1} = \mathbf{F}_1(\mathbf{u}_{n+1}, \mathbf{q}_n) \in C_1$ and $\mathbf{q}_{n+1} = \mathbf{F}_2(\mathbf{u}_{n+1}, \mathbf{q}_n) \in C_2$ in the specific gait phases read

$$\begin{aligned} (\mathbf{d}_I^\alpha)_{n+1} &= \exp\left(\widehat{\theta_{n+1}^\alpha} \right) \cdot (\mathbf{d}_I^\alpha)_n \qquad I = 1,2,3 \quad \alpha = 1,2 \\ \varphi_{n+1}^\alpha &= x_{S_\alpha} - (\rho_{S_\alpha}^\alpha)_{n+1} \\ \varphi_{n+1}^\beta &= \varphi_{n+1}^\alpha + (\rho_H^\alpha)_{n+1} - (\rho_H^\beta)_{n+1} \\ \mathbf{q}_{n+1}^M &= \varphi_{n+1}^1 + (\rho_H^1)_{n+1} \end{aligned}$$

During stance phase of first foot, the active constraint is $\mathbf{g}_1$ and $(\alpha, \beta) = (1,2)$ in the last three equations. The second foot's stance phase is characterised by $\mathbf{g}_2$ and $(\alpha, \beta) = (2,1)$ holds.

2.5 Actuation

Although the system has six degrees of freedom, only a three-dimensional torque $\tau \in \mathbb{R}^3$ acts in the hip joint, thus the system is underactuated. The generalised forces τ are mapped to the redundant control force $\mathbf{f} \in \mathbb{R}^n$ (since the optimal control problem in Section 3 is formulated in terms of the n-dimensional redundant configuration and control force) via the input transformation matrix

$$\mathbf{B}(\mathbf{q}) = \frac{1}{2} \left[\mathbf{0} \;\; -\widehat{\mathbf{d}}_1^1 \;\; -\widehat{\mathbf{d}}_2^1 \;\; -\widehat{\mathbf{d}}_3^1 \;\; \mathbf{0} \;\; \widehat{\mathbf{d}}_1^2 \;\; \widehat{\mathbf{d}}_2^2 \;\; \widehat{\mathbf{d}}_3^2 \;\; \mathbf{0} \right]$$

To ensure regularity of the constrained optimisation problem in DMOCC, the discrete generalised forces τ_n are defined in the interval $[t_n, t_{n+1}]$, then their effect is transformed to the nodes via $\tau_{n-1}^+ = \frac{\Delta t}{2} \tau_{n-1}$ and $\tau_n^+ = \frac{\Delta t}{2} \tau_n$ and finally the redundant forces at t_n are given by

$$\mathbf{f}_{n-1}^+ = \mathbf{B}^T(\mathbf{q}_n) \cdot \tau_{n-1}^+ \qquad \text{and} \qquad \mathbf{f}_n^- = \mathbf{B}^T(\mathbf{q}_n) \cdot \tau_n^-$$

3 Optimal Control of the Walker

3.1 Objective Functional

The objective functional $J(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{f}) = \int_{t_0}^{t_N} C(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{f}) \, dt$ is to be minimised with respect to the state trajectory $(\mathbf{q}(t), \dot{\mathbf{q}}(t))$ and the control trajectory $\mathbf{f}(t)$. Motivated by the specific cost of transport used e.g. in [37], we consider the control effort per step length sl, i.e. $C(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{f}) = \frac{\|\mathbf{f}\|}{sl(\mathbf{q})}$ as a cost function. Although the walker is in principle free to move in any direction, the step length is measured as the projection of the distance between the feet in the double stance configuration onto a predefined walking direction, see Figure 4.

3.2 Periodic Boundary Conditions

Let $\mathbf{q}_t$ denote the double stance configuration (which is unknown and thus to be determined by the optimisation). It is assumed that the swing phases of the left and right leg of the walker are identical mirror images of each other, see Figure 4. Therefore, only half a gait cycle is optimised while the final state is requested to be a mirror image of the initial state that is translated by the steplength into the walking direction. For the compass biped, periodicity of the gait leads to 12 independent conditions, which can e.g. be formulated in terms of the alignment of certain directors (and conjugate momenta) at the unknown initial configuration with the mirror image of directors (and corresponding momenta) at the unknown final configuration. Thereby, the

$(\mathbf{q}_0, \mathbf{p}_0)$ $(\mathbf{q}_l, \mathbf{p}_l)$ $(\mathbf{q}_N, \mathbf{p}_N)$
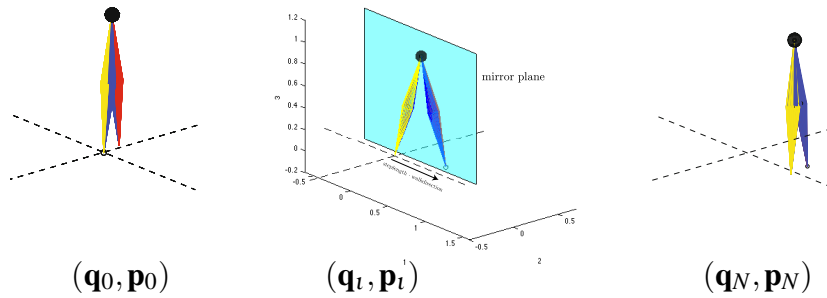
Fig. 4 Initial (left) and final (right) configuration of half a gait cycle and mirror plane in the double stance configuration (middle)

prescribed walk direction and the direction of gravity span the mirror plane which goes through the midpoint (measured in the plane's normal direction) between the contact points of the feet with the ground. Another three conditions relate the unknown torques in the first and last time interval to each other as mirror images. These relations are described by the function $\mathbf{r}(\mathbf{q}_0, \mathbf{p}_0, \mathbf{q}_l, \mathbf{q}_N, \mathbf{p}_N, \tau_0, \tau_{N-1}) = \mathbf{0}$ involving the initial and final configuration, conjugate momenta and torques, respectively and the impact configuration. Let $mirr : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ denote the mirror function, then one possibility is to require alignment of

· the first leg's directors at t_0, i.e. $(\mathbf{d}_I^1)_0$, with the mirror image of the second leg's directors at t_N, i.e. $mirr((\mathbf{d}_I^2)_N)$ for $I = 1, 2, 3$ and
· the conjugate momenta $(\mathbf{p}_I^1)_0$ with $mirr((\mathbf{p}_I^2)_N)$ for $I = 1, 2, 3$ and
· the directors $(\mathbf{d}_I^2)_0$ with $mirr((\mathbf{d}_I^1)_N)$ for $I = 1, 2, 3$ and
· the conjugate momenta $(\mathbf{p}_I^2)_0$ with $mirr((\mathbf{p}_I^1)_N)$ for $I = 1, 2, 3$ and
· the torque τ_0 with $mirr(\tau_{N-1})$.

Using the discrete Legendre transformation (4), these conditions can be transformed into relations between $\mathbf{q}_0$ and $\mathbf{q}_{N-1}$ and between $\mathbf{q}_1$ and $\mathbf{q}_N$, respectively, which simplifies their implementation in the framework of the discrete optimisation problem.

3.3 Variational Principles

Before and after the impact, the Lagrange-d'Alembert principle for constrained forced dynamics is used to derive the equations of motion from a variational principle. The Lagrangian $L : TQ \rightarrow \mathbb{R}$ with $L(\mathbf{q}, \dot{\mathbf{q}}) = \frac{1}{2}\dot{\mathbf{q}} \cdot \mathbf{M} \cdot \dot{\mathbf{q}} - V(\mathbf{q})$ represents the difference of kinetic and potential energy $V(\mathbf{q})$. In the case of the walker, the potential energy represents the influence of gravity. The Lagrange multipliers $\lambda_1, \lambda_2 \in \mathbb{R}^m$ correspond to the constraints active during the specific gait phases.

In the presence of a perfectly plastic impact, the dynamics takes place in a non-smooth setting involving modifications to the path space such that one takes variations over curves with isolated points of diminished smoothness or continuity. In [33], a nonautonomous variational approach to nonsmooth dynamical problems is elaborated. While mentioning this here only very briefly, we refer to [33] for details. Essentially, we consider paths $c(s) = (c_t(s), c_q(s))$ in an extended configuration space $[t_0, t_N] \times Q$ parametrised in $s \in [0,1]$. Now, the time reads $t = c_t(s)$ in $C^\infty([0,1], [t_0, t_N])$ with strictly positive derivative and the configuration is given by the continuous function $\mathbf{q}(t) = c_q(c_t^{-1}(t))$ being (piecewise) twice continuously differentiable (away from the impact) and having a singularity at the impact configuration $\mathbf{q}(t_t) = c_q(s_t)$ due to which the action integral has to be split. With the variation $\delta c(s_t) = (\delta c_t(s_t), \delta c_q(s_t)) \in T_{\mathbf{q}_t}([t_0, t_N], C_2)$, it reads

$$
\begin{aligned}
&\delta\left(\int_{t_0}^{t_t^-} L(\mathbf{q}, \dot{\mathbf{q}}) - \mathbf{g}_1(\mathbf{q}) \cdot \lambda_1 \, dt\right) + \int_{t_0}^{t_t^-} \mathbf{f} \cdot \delta\mathbf{q} \, dt + \mathbf{f}_t^c \cdot \delta c_t(s_t) + \mathbf{f}_q^c \cdot \delta c_q(s_t) + \\
&\delta\left(\int_{t_t^+}^{t_N} L(\mathbf{q}, \dot{\mathbf{q}}) - \mathbf{g}_2(\mathbf{q}) \cdot \lambda_2 \, dt\right) + \int_{t_t^+}^{t_N} \mathbf{f} \cdot \delta\mathbf{q} \, dt = 0
\end{aligned}
\tag{1}
$$

Note that describing the transfer of contact, and in particular the perfectly plastic impact with no sliding, requires the joining of two variational principles at t_t by incorporating the virtual work of the contact force $\mathbf{f}^c = (\mathbf{f}_t^c, \mathbf{f}_q^c) \in T_{\mathbf{q}_t}^*([t_0, t_N], C_2)$. In particular, it will be seen in the transition equations (2) below, that the configuration component $\mathbf{f}_q^c$ imposes a jump in the system's momentum normal to the contact surface while $\mathbf{f}_t^c$ induces jump in the energy $E = \frac{\partial L}{\partial \dot{\mathbf{q}}} \cdot \dot{\mathbf{q}} - L$.

3.4   Optimal Control Problem

Deriving the differential-algebraic equations of motion from the variational formulation (1), results in the following optimal control problem in the time continuous setting.

$$
\min_{\mathbf{q}, \dot{\mathbf{q}}, \mathbf{f}} J(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{f})
$$

subject to
    constrained equations of motion in $[t_0, t_t[$

$$
\begin{aligned}
\frac{\partial L(\mathbf{q}, \dot{\mathbf{q}})}{\partial \mathbf{q}} - \frac{d}{dt} \frac{\partial L(\mathbf{q}, \dot{\mathbf{q}})}{\partial \dot{\mathbf{q}}} - \mathbf{G}_1^T(\mathbf{q}) \cdot \lambda_1 + \mathbf{f} &= \mathbf{0} \\
\mathbf{g}_1(\mathbf{q}) &= \mathbf{0}
\end{aligned}
$$

    constrained equations of motion in $]t_t, t_N]$

$$
\begin{aligned}
\frac{\partial L(\mathbf{q}, \dot{\mathbf{q}})}{\partial \mathbf{q}} - \frac{d}{dt} \frac{\partial L(\mathbf{q}, \dot{\mathbf{q}})}{\partial \dot{\mathbf{q}}} - \mathbf{G}_2^T(\mathbf{q}) \cdot \lambda_2 + f &= \mathbf{0} \\
\mathbf{g}_2(\mathbf{q}) &= \mathbf{0}
\end{aligned}
$$

transition equations from $(\mathbf{q}(t_t^-), \dot{\mathbf{q}}(t_t^-)) \in TC_1$ to $(\mathbf{q}(t_t^+), \dot{\mathbf{q}}(t_t^+)) \in TC_2$

$$\frac{\partial L(\mathbf{q}, \dot{\mathbf{q}})}{\partial \dot{\mathbf{q}}}\Big|_{t_t^-}^{t_t^+} - \mathbf{f}_q^c = 0, \qquad E\Big|_{t_t^-}^{t_t^+} - \mathbf{f}_t^c = 0 \qquad (2)$$

periodic boundary conditions

$$\mathbf{r}(\mathbf{q}_0, \mathbf{p}_0, \mathbf{q}_t, \mathbf{q}_N, \mathbf{p}_N, \tau_0, \tau_{N-1}) = \mathbf{0}$$

path constraints
$$h(\mathbf{q}) \geq 0$$

Here, $\mathbf{G}_1(\mathbf{q}) = D\mathbf{g}_1(\mathbf{q})$ and $\mathbf{G}_2(\mathbf{q}) = D\mathbf{g}_2(\mathbf{q})$ denote the Jacobians of the constraint functions. The transition equations (2) describe the change in momentum and energy due to the perfectly plastic impact that immobilises the foot.

4   Constrained Discrete Dynamics and Optimal Control of the Walker

4.1   Discrete Variational Principles and Equations of Motion

Without loss of generality, it is assumed that the time of contact establishing between the second foot and the ground coincides with a time node t_t. This is possible since periodic boundary conditions are imposed on the initial and final states, see Section 3.2. Furthermore, it spares the necessity to consider variations with respect to time in the discrete setting. Details on the discrete variational theory for nonsmooth systems with unknown collision time can be found in [33]. The key ingredient of variational integrators is the discrete Lagrangian L_d, which approximates the action of the continuous Lagrangian in one time interval. In this work, a midpoint approximation is used, i.e. $L_d(\mathbf{q}_n, \mathbf{q}_{n+1}) = L(\frac{\mathbf{q}_n + \mathbf{q}_{n+1}}{2}, \frac{\mathbf{q}_{n+1} - \mathbf{q}_n}{\Delta t})$. In analogy to the continuous variational principles (1), the joining of two discrete constrained Lagrange-d'Alembert principle at t_t reads

$$\delta \left(\sum_{n=0}^{t-1} L_d(\mathbf{q}_n, \mathbf{q}_{n+1}) - \frac{\Delta t}{2} \mathbf{g}_1(\mathbf{q}_n) \cdot \lambda_{1,n} - \frac{\Delta t}{2} \mathbf{g}_1(\mathbf{q}_{n+1}) \cdot \lambda_{1,n+1} \right) + \sum_{n=0}^{t-1} (\mathbf{f}_n^- \cdot \delta \mathbf{q}_n +$$

$$\mathbf{f}_n^+ \cdot \delta \mathbf{q}_{n+1} + \mathbf{f}_q^c \cdot \delta \mathbf{q}_t + \delta \left(\sum_{n=t}^{N-1} L_d(\mathbf{q}_n, \mathbf{q}_{n+1}) - \frac{\Delta t}{2} \mathbf{g}_2(\mathbf{q}_n) \cdot \lambda_{2,n} - \frac{\Delta t}{2} \mathbf{g}_2(\mathbf{q}_{n+1}) \cdot \lambda_{2,n+1} \right)$$

$$+ \sum_{n=t}^{N-1} (\mathbf{f}_n^- \cdot \delta \mathbf{q}_n + \mathbf{f}_n^+ \cdot \delta \mathbf{q}_{n+1}) = 0$$

$$(3)$$

with the variations $\delta\mathbf{q}_0,\ldots,\mathbf{q}_{t-1},\mathbf{q}_{t+1},\ldots,\delta\mathbf{q}_N \in TQ$, the constrained variation at the impact $\delta\mathbf{q}_t \in T_{\mathbf{q}_t}C_2$, the impact force $\mathbf{f}_q^c \in T^*_{\mathbf{q}_t}C_2$ and variations of the Lagrange multipliers $\delta\lambda_{1,0},\ldots,\delta\lambda_{1,t},\delta\lambda_{2,t},\ldots,\delta\lambda_{2,N} \in \mathbb{R}^m$. Due to their derivation via a discrete variational principle, the discrete equations of motion resulting from (3), called discrete Euler-Lagrange equations, inherit the structure preserving properties from the real continuous dynamics. A discrete symplectic form as well as momentum maps arising from symmetries according to Noether's theorem are conserved exactly along the discrete trajectory, see [22].

The discrete equations of motion resulting from (3) involve configurations variables, forces and Lagrange multipliers. However, in the context of boundary conditions as well as for post-processing and interpretation of the discrete equations of motion as a balance of discrete momentum, the knowledge of the conjugate momenta is useful. At each time node, there exist two expressions for the conjugate momenta, taking into account the past or the following time interval, respectively. The constrained forced discrete Legendre transformation are given by

$$\begin{aligned}
\mathbf{p}_n^- &= -D_1 L_d(\mathbf{q}_n, \mathbf{q}_{n+1}) + \tfrac{\Delta t}{2}\mathbf{G}^T(\mathbf{q}_n)\cdot\lambda_n - \mathbf{f}_n^- \\
\mathbf{p}_n^+ &= D_2 L_d(\mathbf{q}_{n-1}, \mathbf{q}_n) - \tfrac{\Delta t}{2}\mathbf{G}^T(\mathbf{q}_n)\cdot\lambda_n + \mathbf{f}_{n-1}^+
\end{aligned} \tag{4}$$

The projected discrete Legendre transforms, which still yield an n-dimensional conjugate momentum, read

$$^Q\mathbf{p}_n^- = \mathbf{Q}(\mathbf{q}_n)\cdot\left[-D_1 L_d(\mathbf{q}_n, \mathbf{q}_{n+1}) - \mathbf{f}_n^-\right], \quad ^Q\mathbf{p}_n^+ = \mathbf{Q}(\mathbf{q}_n)\cdot\left[D_2 L_d(\mathbf{q}_{n-1}, \mathbf{q}_n) + \mathbf{f}_{n-1}^+\right]$$

Finally, the reduced discrete Legendre transforms result in $(n-m)$-dimensional conjugate momentum and are defined as

$$^P\mathbf{p}_n^- = \mathbf{P}^T(\mathbf{q}_n)\cdot\left[-D_1 L_d(\mathbf{q}_n, \mathbf{q}_{n+1}) - \mathbf{f}_n^-\right], \quad ^P\mathbf{p}_n^+ = \mathbf{P}^T(\mathbf{q}_n)\cdot\left[D_2 L_d(\mathbf{q}_{n-1}, \mathbf{q}_n) + \mathbf{f}_{n-1}^+\right]$$

First Swing Phase

For $n = 1,\ldots,t-1$ the discrete variational principle (3) yields the following system which is to be solved for $\mathbf{q}_2,\ldots,\mathbf{q}_t,\lambda_{1,1},\ldots,\lambda_{1,t-1}$.

$$\begin{aligned}
D_2 L_d(\mathbf{q}_{n-1}, \mathbf{q}_n) + D_1 L_d(\mathbf{q}_n, \mathbf{q}_{n+1}) - \Delta t\mathbf{G}_1^T(\mathbf{q}_n)\cdot\lambda_{1,n} + \mathbf{f}_{n-1}^+ + \mathbf{f}_n^- &= \mathbf{0} \\
\mathbf{g}_1(\mathbf{q}_{n+1}) &= \mathbf{0}
\end{aligned} \tag{5}$$

Note that the first equation can be interpreted as a balance of discrete momentum $\mathbf{p}_n^+ = \mathbf{p}_n^-$. Equivalently, in reduced form using the null space matrix $\mathbf{P}_1$ and the discrete reparametrisation $\mathbf{q}_{n+1} = \mathbf{F}_1(\mathbf{u}_{n+1}, \mathbf{q}_n) \in C_1$, the balance of projected momentum $^P\mathbf{p}_n^+ = {}^P\mathbf{p}_n^-$ reading

$$\mathbf{P}_1^T(\mathbf{q}_n)\cdot\left[D_2 L_d(\mathbf{q}_{n-1}, \mathbf{q}_n) + D_1 L_d(\mathbf{q}_n, \mathbf{q}_{n+1}) + \mathbf{f}_{n-1}^+ + \mathbf{f}_n^-\right] = \mathbf{0} \tag{6}$$

is to be solved for $\mathbf{u}_2,\ldots,\mathbf{u}_t$. In contrast to the $(n+m)$-dimensional system (5), (6) is only $(n-m)$-dimensional.

Second Swing Phase

The discrete variational principle (3) yields for $n = \iota + 1, ..., N - 1$ the system

$$D_2 L_d(\mathbf{q}_{n-1}, \mathbf{q}_n) + D_1 L_d(\mathbf{q}_n, \mathbf{q}_{n+1}) - \Delta t \mathbf{G}_2^T(\mathbf{q}_n) \cdot \lambda_{2,n} + \mathbf{f}_{n-1}^+ + \mathbf{f}_n^- = \mathbf{0}$$
$$\mathbf{g}_2(\mathbf{q}_{n+1}) = \mathbf{0}$$

which is o be solved for $\mathbf{q}_{\iota+2}, ..., \mathbf{q}_N, \lambda_{2,\iota+1}, ..., \lambda_{2,N-1}$. Equivalently using the null space matrix $\mathbf{P}_2$ and the discrete reparametrisation $\mathbf{q}_{n+1} = \mathbf{F}_2(\mathbf{u}_{n+1}, \mathbf{q}_n) \in C_2$ the reduced system

$$\mathbf{P}_2^T(\mathbf{q}_n) \cdot \left[D_2 L_d(\mathbf{q}_{n-1}, \mathbf{q}_n) + D_1 L_d(\mathbf{q}_n, \mathbf{q}_{n+1}) + \mathbf{f}_{n-1}^+ + \mathbf{f}_n^- \right] = \mathbf{0} \qquad (7)$$

is to be solved for $\mathbf{u}_{\iota+2}, ..., \mathbf{u}_{N-1}$.

Transfer of Contact

As explained earlier, without loss of generality, it can be assumed that the impact of the second foot on the ground takes place at t_ι, thus $g_c(\mathbf{q}_\iota) = 0$. Then automatically $\mathbf{g}_2(\mathbf{q}_\iota) = \mathbf{0}$ follows, since the point of contact defines $\mathbf{g}_2$. Note that a contact force $\mathbf{f}_q^c \in T_{\mathbf{q}_\iota}^* C_2$ which immobolises the second foot in it's point of contact is normal to C_2, thus it is given by $\mathbf{f}_q^c = \mathbf{G}_2^T(\mathbf{q}_\iota) \cdot \lambda_c$. Substituting this in the discrete form $\mathbf{p}_\iota^+ - \mathbf{p}_\iota^- + \mathbf{f}_q^c = \mathbf{0}$ of the transition equations (2) yields

$$D_2 L_d(\mathbf{q}_{\iota-1}, \mathbf{q}_\iota) + D_1 L_d(\mathbf{q}_\iota, \mathbf{q}_{\iota+1}) - \tfrac{\Delta t}{2} \mathbf{G}_1^T(\mathbf{q}_\iota) \cdot \lambda_{1,\iota} - \mathbf{G}_2^T(\mathbf{q}_\iota) \cdot (\tfrac{\Delta t}{2} \lambda_{2,\iota} + \lambda_c) +$$
$$\mathbf{f}_{\iota-1}^+ + \mathbf{f}_\iota^- = \mathbf{0}$$
$$\mathbf{g}_2(\mathbf{q}_{\iota+1}) = \mathbf{0}$$
$$(8)$$

This is an underdetermined system. To solve for $\mathbf{q}_{\iota+1}, \lambda_{1,\iota}, \lambda_{2,\iota}, \lambda_c$, one possibility is to augment (8) by the constraints on momentum level $\mathbf{G}_1(\mathbf{q}) \cdot \mathbf{M}^{-1} \cdot \mathbf{p}_\iota^+ = \mathbf{0}$ and $\mathbf{G}_2(\mathbf{q}) \cdot \mathbf{M}^{-1} \cdot \mathbf{p}_\iota^- = \mathbf{0}$. However, since only constraints on the configuration variables are imposed elsewhere, this would be somewhat inconsequent. Therefore, the fact that $^{Q_1}\mathbf{p}_n^+ = \mathbf{p}_n^+$ holds is used and the transition equations read $^{Q_1}\mathbf{p}_\iota^+ - \mathbf{p}_\iota^- + \mathbf{f}_q^c = \mathbf{0}$. Next, projection with the second discrete null space matrix and insertion of the discrete reparametrisation $\mathbf{q}_{\iota+1} = \mathbf{F}_2(\mathbf{u}_{\iota+1}, \mathbf{q}_\iota) \in C_2$ yields

$$\mathbf{P}_2^T(\mathbf{q}_\iota) \cdot \left[\mathbf{Q}_1(\mathbf{q}_\iota) \cdot \left(D_2 L_d(\mathbf{q}_{\iota-1}, \mathbf{q}_\iota) + \mathbf{f}_{\iota-1}^+ \right) + D_1 L_d(\mathbf{q}_\iota, \mathbf{q}_{\iota+1}) + \mathbf{f}_\iota^- \right] = \mathbf{0} \quad (9)$$

to be solved for $\mathbf{u}_{\iota+1}$.

4.2 Discrete Constrained Optimisation Problem

As for the discrete Lagrangian in Section 4.1, the integral of the continuous cost function in one time interval is approximated by C_d. Furthermore, indicating the dependence on the discrete generalised coordinates $\mathbf{u}_d = \{\mathbf{u}_n\}_{n=1}^{N}$ and forces $\tau_d = \{\tau_n\}_{n=0}^{N-1}$ directly, the discrete objective function can be expressed as

$$J_d(\mathbf{u}_d, \tau_d) = \sum_{n=0}^{N-1} C_d(\mathbf{u}_n, \mathbf{u}_{n+1}, \tau_n)$$

The constrained optimisation problem reads

$$\min_{\mathbf{u}_d, \tau_d} J_d(\mathbf{u}_d, \tau_d)$$

subject to
  reduced forced discrete equations of motion (6) for $n = 1, ..., \iota - 1$

$$\mathbf{P}_1^T(\mathbf{q}_n) \cdot \left[D_2 L_d(\mathbf{q}_{n-1}, \mathbf{q}_n) + D_1 L_d(\mathbf{q}_n, \mathbf{q}_{n+1}) + \mathbf{f}_{n-1}^+ + \mathbf{f}_n^- \right] = \mathbf{0}$$

reduced forced discrete equations of motion (7) for $n = \iota + 1, ..., N - 1$

$$\mathbf{P}_2^T(\mathbf{q}_n) \cdot \left[D_2 L_d(\mathbf{q}_{n-1}, \mathbf{q}_n) + D_1 L_d(\mathbf{q}_n, \mathbf{q}_{n+1}) + \mathbf{f}_{n-1}^+ + \mathbf{f}_n^- \right] = \mathbf{0}$$

transition equations (9)

$$\mathbf{g}_c(\mathbf{q}_\iota) = \mathbf{0}$$
$$\mathbf{P}_2^T(\mathbf{q}_\iota) \cdot \left[\mathbf{Q}_1(\mathbf{q}_\iota) \cdot \left(D_2 L_d(\mathbf{q}_{\iota-1}, \mathbf{q}_\iota) + \mathbf{f}_{\iota-1}^+ \right) + D_1 L_d(\mathbf{q}_\iota, \mathbf{q}_{\iota+1}) + \mathbf{f}_\iota^- \right] = \mathbf{0}$$

periodic boundary conditions

$$\mathbf{r}(\mathbf{q}_0, \mathbf{p}_0, \mathbf{q}_\iota, \mathbf{q}_N, \mathbf{p}_N, \tau_0, \tau_{N-1}) = \mathbf{0}$$

path constraints for $n = 1, ..., N$

$$h(\mathbf{q}_n) \geq 0$$

5 Results

In the walker model, the mass of the rigid legs is $M_\varphi^1 = M_\varphi^2 = 5$ while that of the point mass is $M^M = 10$. The legs are double cones of radius $r = 0.05$ and cone length $l = 0.5$. Gravity points with an acceleration of $g = -9.81$ into the negative $\mathbf{e}_3$-direction. The simulation of the half step takes place in the time interval $[0, 0.7]$ and
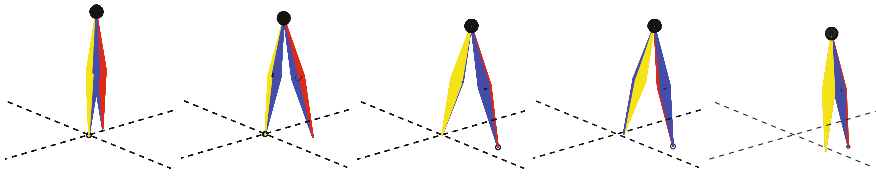
Fig. 5 Snapshots of the compass biped gait

$N = 13$, i.e. 14 time nodes are used and the double stance configuration is assumed to be approximately in the middle of the interval, thus $\iota = 6$. Note that the periodic boundary conditions allow t_ι to be anywhere in the time interval. The restricted optimisation problem described in Section 4.2 is solved in Matlab using `fmincon` choosing an active-set algorithm and supplying user defined analytic gradients of the objective function and the constraints, respectively.

The initial guess is quite rough and does not fulfill the discrete dynamics. At t_0, the biped stands in the $(\mathbf{e}_2, \mathbf{e}_3)$-plane, with the stance leg (yellow) rotated by the angle $\frac{\pi}{18}$ around the negative $\mathbf{e}_1$-axis and the swing leg (red) rotated by the same angle around the positive $\mathbf{e}_1$-axis. Then,

$$
\begin{aligned}
(\theta_1^1)_n = (\theta_1^2)_n &= \tfrac{\pi}{18(N-1)} && n = 2,\ldots,13 \\
(\theta_2^1)_n = \tfrac{0.12}{\frac{N}{2}-1}, \qquad (\theta_3^1)_n = (\theta_3^2)_n &= -\tfrac{0.16}{\frac{N}{2}-1}, \qquad (\theta_2^2)_n = \tfrac{0.25}{\frac{N}{2}} && n = 1,\ldots,6 \\
(\theta_2^1)_n = -\tfrac{0.12}{\frac{N}{2}-1}, \qquad (\theta_3^1)_n = (\theta_3^2)_n &= \tfrac{0.16}{\frac{N}{2}-1}, \qquad (\theta_2^2)_n = -\tfrac{0.25}{\frac{N}{2}-2} && n = 7,\ldots,13
\end{aligned}
$$

and all discrete generalised forces are set to zero.

The gait resulting from the discrete objective function

$$
J_d(\mathbf{u}_d, \tau_d) = \frac{\Delta t}{sl(\mathbf{u}_d)} \sum_{n=0}^{N-1} \|\tau_n\|
$$

approximating $J(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{f})$ introduced in Section 3.1 is computed. See Figure 5 for snapshots of the motion. Figure 6 shows the evolution of the feet trajectory coordinates (left) and projection of the feet and hip trajectories to the $(\mathbf{e}_1, \mathbf{e}_2)$-plane (right) during three steps. Only a half step has been simulated, however, the fulfilment of periodic boundary condition on configuration as well as on momentum level ensures the smooth transition between the steps. The vertical dotted lines in the left plot indicate the double stance configurations and the yellow and red circles in the right plot mark the placement $\mathbf{x}_{S_1}$ and $\mathbf{x}_{S_2}$ of the stancefoot during the specific gait phases, respectively. The path constraints preventing the self-penetration of the walker are never active in the presented solution. Due to the presence of gravity and the fixing of one foot on the ground, the only symmetry of the Lagrangian of the walker is rotation around the gravity axis through the foot position. Figure 7 shows that during the specific gait phases, angular momentum with respect to the attachment point is conserved exactly. This illustrates the structure preservation guaranteed by
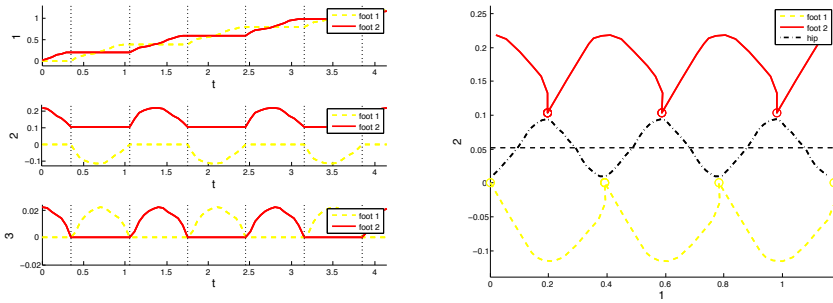
Fig. 6 Evolution of the feet trajectory coordinates (left) and projection of the feet and hip trajectories to the $(\mathbf{e}_1, \mathbf{e}_2)$-plane (right).
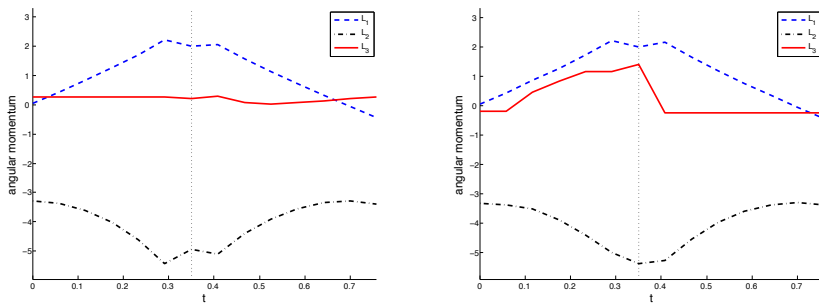
Fig. 7 Angular momentum with respect to $\mathbf{x}_{S_1}$ (left) and with respect to $\mathbf{x}_{S_2}$ (right)

the discrete Euler-Lagrange equations. From the discrete configuration and force trajectories, the steplength for this gait is determined to be $sl = 0.1960$ and the value of the objective function is $J_d = 11.4020$. Due to the midpoint evaluation of the discrete Lagrangian in (3), the discrete trajectory presented in this work is second order accurate. However, a numerical convergence study goes beyond the scope of this work and will be presented elsewhere.

6   Conclusion

A structure preserving method for the numerical simulation of the optimal control of a bipedal walker's compass gait has been developed and illustrated with an example. In the discrete formulation of the optimal control problem in Section 4.2, structure preservation is guaranteed by the derivation of the discrete equations of motion, and in particular the discrete transfer of contact equations, via a discrete variational principle.

References

1. Chevallereau, C., Aoustin, Y.: Optimal reference trajectories for walking and running of a biped robot. Robotica 19, 557–569 (2001)
2. Roussel, L., Canudas-de-Wit, C., Goswami, A.: Generation of energy optimal complete gait cycles for biped robots. In: Proc. IEEE Conf. on Robotics and Automation (1998)
3. Leyendecker, S., Ober-Blöbaum, S., Marsden, J.E., Ortiz, M.: Discrete Mechanics and Optimal Control for Constrained systems. Optimal Control Applications and Methods 31, 505–528 (2010)
4. Bullo, F., Zefran, M.: On modeling and locomotion of hybrid mechanical systems with impacts. In: Proceedings of the 37th IEEE Conference on Decision & Control, Tampa, Florida, USA, pp. 2633–2638 (1998)
5. Pekarek, D., Marsden, J.E.: Variational collision integrators and optimal control. In: Proc. of the 18th International Symposium on Mathematical Theory of Networks and Systems (2008)
6. Junge, O., Marsden, J.E., Ober-Blöbaum, S.: Discrete mechanics and optimal control. In: Proceedings of the 16th IFAC World Congress (2005)
7. Ober-Blöbaum, S.: Discrete mechanics and optimal control. University of Paderborn (2008)
8. Ober-Blöbaum, S., Junge, O., Marsden, J.E.: Discrete mechanics and optimal control: an analysis. ESAIM: Control Optimisation and Calculus of Variations 17, 322–352 (2011)
9. Gill, P.E., Murray, W., Saunders, M.A.: SNOPT: An SQP algorithm for large-scale constrained optimization. Numerical Analysis Report, Department of Mathematics. University of California, San Diego (1997)
10. Schittkowski, K.: Nonlinear programming codes. Lecture Notes in Economics and Mathematical Systems, vol. 183, pp. viii+242. Springer (1980)
11. Stoer, J., Bulirsch, R.: Introduction to numerical analysis, vol. 12, pp. xvi+744. Springer (2002)
12. Kraft, D.: On converting optimal control problems into nonlinear programming problems. Computational Mathematical Programming F15, 261–280 (1985)
13. Hicks, G.A., Ray, W.H.: Approximation methods for optimal control systems. Can. J. Chem. Engng. 49, 522–528 (1971)
14. Mombaur, K.D., Longman, R.W., Bock, H.G., Schlöder, J.P.: Open-loop stable running. Robotica 23, 21–33 (2005)
15. Deuflhard, P., Bornemann, F.: A modified Newton method for the solution of ill-conditioned systems of nonlinear equations with application to multiple shooting. Numer. Math. 22, 289–315 (1974)
16. Bock, H.G., Plitt, K.J.: A multiple shooting algorithm for direct solution of optimal control problems. In: Proc. 9th IFAC World Congress (1984)
17. von Stryk, O.: Numerical solution of optimal control problems by direct collocation. Optimal Control – Calculus of Variations, Optimal Control Theory and Numerical Methods, Internat. Ser. Numer. Math. 111, 129–143 (1993)
18. Biegler, L.T.: Solution of dynamic optimization problems by successive quadratic programming and orthogonal collocation. Comput. Chem. Engng. 8, 243–248 (1984)
19. Betts, J.: Survey of numerical methods for trajectory optimization. Journal of Guidance, Control, and Dynamics 2, 193–207 (1998)

20. Binder, T., Blank, L., Bock, H.G., Bulirsch, R., Dahmen, W., Diehl, M., Kronseder, T., Marquardt, W., Schlöder, J.P., von Stryk, O.: Introduction to model based optimization of chemical processes on moving horizons. Online Optimization of Large Scale Systems: State of the Art 2, 295–340 (2001)
21. Stern, A., Desbrun, M.: Discrete geometric mechanics for variational time integrators. In: Proc. of the SIGGRAPH 2006 ACM SIGGRAPH 2006 Courses, pp. 75–80 (2006)
22. Marsden, J.E., West, M.: Discrete mechanics and variational integrators. Acta Numerica 10, 357–514 (2001)
23. Hairer, E., Lubich, C., Wanner, G.: Geometric numerical integration. Springer, Berlin (2006)
24. Leyendecker, S., Ober-Blöbaum, S.: A variational approach to multirate integration for constrained systems. In: Fisette, P., Samin, J.C. (eds.) Multibody Dynamics – Computational Methods and Applications. Springer (2012) (to appear)
25. Schiehlen, W.: Multibody systems handbook. Springer, Berlin (1990)
26. Géradin, M., Cardona, A.: Flexible Multibody Dynamics. John Wiley & Sons, Berlin (2001)
27. Gonzalez, O.: Mechanical systems subject to holonomic constraints: differential-algebraic formulations and conservative integration. Physica D 132, 165–174 (1999)
28. Wendlandt, J., Marsden, J.E.: Mechanical integrators derived from a discrete variational principle. Physica D 106, 223–246 (1997)
29. Betsch, P.: The discrete null space method for the energy consistent integration of constrained mechanical systems. Part I: Holonomic constraints. Comput. Methods Appl. Mech. Engrg. 194, 5159–5190 (2005)
30. Betsch, P., Leyendecker, S.: The discrete null space method for the energy consistent integration of constrained mechanical systems. Part II: Multibody dynamics. Int. J. Numer. Meth. Engng. 67, 499–552 (2006)
31. Leyendecker, S., Marsden, J.E., Ortiz, M.: Variational integrators for constrained dynamical systems. ZAMM 88, 677–707 (2008)
32. Fetecau, R.C., Marsden, J.E., Ortiz, M., West, M.: Nonsmooth Lagrangian Mechanics and Variational Collision Integrators. Siam J. Applied Dynamical Systems 2, 381–416 (2003)
33. Pekarek, D.: Variational Methods for Control and Design of Bipedal Robot Models. California Institute of Technology (2010)
34. Antman, S.S.: Nonlinear Problems in Elasticity. Springer, Berlin (1995)
35. Betsch, P., Steinmann, P.: Constrained integration of rigid body dynamics. Comput. Methods Appl. Mech. Engrg. 191, 467–488 (2001)
36. Hurmuzlu, Y.: Dynamics of bipedal gait. Part I – objective functions and the contact event of a planar five-link biped. Part II – stability analysis of a planar five-link biped. ASME Journal of Applied Mechanics 60, 331–344 (1993)
37. Duindam, V.: Port-based modeling and control for efficient bipedal walking robots. University of Twente (2006)

Quasi-straightened Knee Walking for the Humanoid Robot

Zhibin Li, Bram Vanderborght, Nikos G. Tsagarakis, and Darwin G. Caldwell

Abstract. Most humanoid robots do not walk in a very human-like manner due to their style of bent knee walking. Typically for decoupling the motion in the sagittal and the coronal planes, the acceleration term in the zero moment point (ZMP) equation is set to zero, resulting in a constant height of the center of mass (COM). This constraint creates the bent knee profile that is fairly typical for walking robots, which particularly requires high torque transmission from motors. Hence, it is interesting to investigate an improved trajectory generator that produces a more straight knee walking which is more energy efficient and natural compared to those performed by the bent knee walking. This issue is addressed by adding a virtual spring-damper to the cart-table model. This strategy combines the preview control for generating the desired horizontal motions of the COM, and the virtual model for generating the vertical COM motion. The feasibility is evaluated by a mathematical investigation of the sensitivity of ZMP errors in MATLAB simulation of a multi-body humanoid model. The walking pattern is applied to the simulated humanoid iCub using the dynamic simulator OpenHRP3. The simulated iCub successfully performs walking gaits. Simulation results are presented and compared to the biomechanical study from human gaits. Both the knee joint torque and energy consumption of all joints required by the proposed strategy are reduced compared to that of the conventional cart-table scheme.

1 Introduction

To date, many successful humanoids such as Asimo [12] and HRP-2 [6] demonstrate their outstanding capability of performing a variety of stable walking tasks.

Zhibin Li · Nikos G.Tsagarakis · Darwin G. Caldwell
Istituto Italiano di Tecnologia, via Morego 30, 16163 Genova, Italy
e-mail: {zhibin.li,nikos.tsagarakis,darwin.caldwell}@iit.it

Bram Vanderborght
Vrije Universiteit Brussel, Brussels, Belgium
e-mail: bram.vanderborght@vub.ac.be

Nevertheless, most of them walk with bent knees that gives an unnatural looking. Moreover, knee motors usually have the highest torque and power [8]. In contrast, in human walking the knee is almost completely stretched [1] and performs mostly negative work [14]. Indeed, in many passive walkers, the knee joint is not actuated and only a knee-locking mechanism is used [7].

WABIAN-2 achieves a more human-like walking than many other robots because it can stretch its knees and avoid singularities by using extra degree of freedoms (DOFs) from the waist joint [9]. The waist joint provides two complementary DOFs for solving the inverse kinematics so it permits a flexible design of knee joint trajectories. Its pattern generator uses predetermined knee joint trajectories consist of two sine motions in order to realize straight knee walking. The knee singularity is avoided since the knee trajectory is predefined in the joint space and requires no inverse kinematics. An essential benefit of this motion is the lower torque requirement and reduced energy consumption [10]. Nandha et al. found a hip trajectory satisfying the zero moment point (ZMP) by the method in [3], and solved the inverse kinematics by defining an initial foot trajectory. The knee stretch motion is redesigned by the cubic spline interpolation to prevent the singularity [2]. But the foot motion needs to be recalculated to find the inverse kinematics solution for the new knee trajectory. Both methods have a common groundwork of planning the knee trajectory in joint space.

Our study presents an alternative approach using a pattern generation method in the Cartesian space without predefining or redesigning knee joint trajectories in the joint space. We investigate a pattern generation method which creates a more straightened (but not fully straightened) knee walking profile by combining the well recognized cart table model with virtual spring-damper models [11].

The paper is organized as follows. Section 2 mathematically investigates the feasibility of integrating the z motion with the cart-table model and presents the modeling of the virtual spring damper. Section 3 provides the gait generation results from a multi-body humanoid model in MATLAB as well as the successful walking gaits from the dynamic simulator OpenHRP3. We conclude the study in Section 4.

2   Mathematical Modeling

The spatial COM motion is decoupled into the horizontal plane and the vertical axis respectively. The control architecture consists of two stages of trajectory generation. In the first stage, the preview control [4] generates the horizontal motion and the virtual spring damper produces the vertical motion. In the second stage, the preview control modifies the horizontal motion to compensate for the errors caused by the vertical motion as well as the simplified modeling. In this paper, we focus on the generation of the vertical motion in order to achieve the quasi-straightened knee walking.

2.1 Feasibility

Prior to applying the virtual model, we mathematically examine the feasibility of combining the cart-table model and the virtual model by computing the sensitivity of the ZMP error linearized around the nominal COM height with zero acceleration. The cart-table model assumes that the cart stays on a level table, while the virtual spring is meant to create vertical displacement. Introducing the virtual spring-damper theoretically violates the assumption of the cart-table model. However, the following analysis shows that the introduced error can be minimized and minor if the vertical acceleration is relatively small compared to the gravity constant.

The general ZMP equations of a multi-body rigid system considering the angular momentum effect are as follows.

$$x_{zmp} = \frac{m(\ddot{z}+g)x - m\ddot{x}z - \dot{L}_y}{m(\ddot{z}+g)} \tag{1a}$$

$$y_{zmp} = \frac{m(\ddot{z}+g)y - m\ddot{y}z + \dot{L}_x}{m(\ddot{z}+g)} \tag{1b}$$

In the proposed method, the upper body of the robot is kept in an upright posture and only legs alternate during walking. Hence we assume a minor inertia effect since the momentum created by two legs counteracts each other to some extent so the rate of the angular momentum L_y and L_x are neglected in this study. Regarding z, $\ddot{z}$ as two variables, we obtain the ZMP equation which comprises z and $\ddot{z}$.

A general ZMP equation neglecting the rate of angular momentum is

$$x_{zmp} = x - \frac{\ddot{x}z}{\ddot{z}+g}. \tag{2}$$

The simplified ZMP equation of cart-table model used by the preview controller is

$$x'_{zmp} = x - \frac{\ddot{x}z_c}{g}, \tag{3}$$

where z_c is the constant COM height.

We examine the ZMP error e_x in the x axis and the same rule holds for e_y. Therefore in the following content e_x is investigated and hereafter. The ZMP error e_x introduced by the vertical motion is

$$\begin{aligned} e_x &= x_{zmp} - x'_{zmp} \\ &= \ddot{x}\frac{z_c\ddot{z} + g(z_c - z)}{g(\ddot{z}+g)}. \end{aligned} \tag{4}$$

Partial differential equations of ZMP error e_x are

$$\frac{\partial(e_x)}{\partial(z)} = -\frac{\ddot{x}}{\ddot{z}+g}, \tag{5}$$

and

$$\frac{\partial(e_x)}{\partial(\ddot{z})} = \ddot{x}(\frac{z_c}{g(\ddot{z}+g)} - \frac{(z_c\ddot{z}+gz_c-gz)}{g(\ddot{z}+g)^2}). \tag{6}$$

Linearizing the partial derivatives around $z = z_c$ and $\ddot{z} = 0m/s^2$, the errors of the ZMP calculations caused by z and $\ddot{z}$ respectively are

$$\begin{aligned}\frac{\partial(e_x)}{\partial(z)}\Delta z &= -\frac{\ddot{x}}{\ddot{z}+g}\Delta z \\ &= -\frac{\ddot{x}}{g}\Delta z,\end{aligned} \tag{7}$$

and

$$\begin{aligned}\frac{\partial(e_x)}{\partial(\ddot{z})}\Delta\ddot{z} &= \frac{\ddot{x}z}{g^2}\Delta\ddot{z} \\ &= \frac{\ddot{x}z_c}{g^2}\Delta\ddot{z}.\end{aligned} \tag{8}$$

The ratio of errors caused by $\Delta\ddot{z}$ and Δz is

$$\begin{aligned}\left|\frac{\partial(e_x)}{\partial(\ddot{z})}\Delta\ddot{z}\right|/\left|\frac{\partial(e_x)}{\partial(z)}\Delta z\right| &= \left|-z/(\ddot{z}+g)\right|\left|\Delta\ddot{z}/\Delta z\right| \\ &= \frac{\left|\Delta\ddot{z}/g\right|}{\left|\Delta z/z_c\right|}.\end{aligned} \tag{9}$$

For an average human height, assume the value of parameters are $z_c \approx 0.95m$, $|\Delta z| \leq 0.02m$, and $|\Delta\ddot{z}| \leq 2m/s^2$. Substituting these values into (9), we gain the insight that the height variation $\Delta z/z_c$ is relatively small compared to the acceleration variation $\Delta\ddot{z}/g$. So, e_x introduced by the vertical COM motion is mainly determined by the magnitude of acceleration variation $\Delta\ddot{z}$. Therefore, the error e_x can be reduced within a reasonable bound by minimizing $\Delta\ddot{z}/g$. The same conclusion holds for the error e_y of the ZMP y_{zmp} in the y axis. A generalized investigation of the contribution
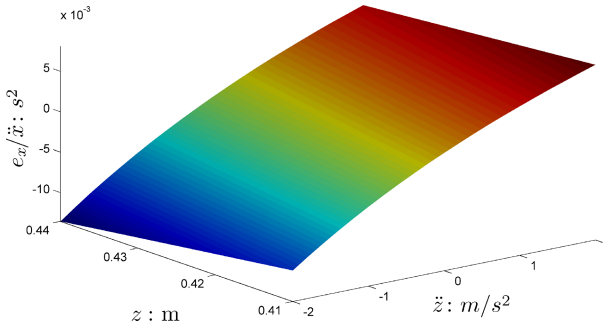
Fig. 1 The ZMP error e_x normalized by the horizontal acceleration $\ddot{x}$

of errors for a different humanoid robot can be done using the same equations by substituting different sets of parameters.

In order to evaluate the error caused by the vertical motion, we configure a set of parameters $0.41m \leq z \leq 0.44m$, $-2m/s^2 \leq \ddot{z} \leq 2m/s^2$, and $\ddot{x} = 1m/s^2$ for the iCub robot. The parameter scan computes numerically the error e_x according to (4) given a unit of horizontal acceleration $\ddot{x} = 1m/s^2$. In Fig. 1, it can be seen that the difference of the slope along the z and $\ddot{z}$ axes indicating the difference level of parametric perturbations from the parameter variation of z and $\ddot{z}$ respectively.

2.2 Virtual Spring-Damper Model

On the basis of the cart-table model, the virtual spring-damper model relaxes the constraint of the constant COM height. This will permit greater stretching of knee joints which will reduce the knee torque and provide a more natural motion.

In Fig. 2, the virtual springs connect the COM and the ankle joints. During walking, the virtual springs are compressed thus generating virtual forces. Since the preview control solves the horizontal motion, only the vertical force component of the spring is employed to determine the vertical dynamics. In the z axis, a virtual damper is added at the tip of each spring to prevent the vertical oscillations. l_0 is the original rest length of the spring; l_{s_l}, l_{s_r} are the spring length for left and right leg respectively; l_{upper} is the length of the thigh; l_{lower} is the length of the shin; d_{hip} is the horizontal distance from the hip joint to the pelvis center; $d_{hip_{COM}}$ is the initial distance between the pelvis center and the COM; x,y,z are the position of the COM in the world coordinate; x_{fl},y_{fl},z_{fl} and x_{fr},y_{fr},z_{fr} are the position of the left and right foot in the world coordinate; $K = k/m$ is the mass-less stiffness of the virtual spring; $C = c/m$ is the mass-less viscous coefficient of the virtual damper; g is the gravity constant $9.81m/s^2$. We define the mass-less coefficient K and C which are the standard stiffness and viscous coefficient normalized by the mass, thus system dynamics is preserved regardless of a specific mass of the robot. Tuning K and C is intuitive according to their physical meanings.
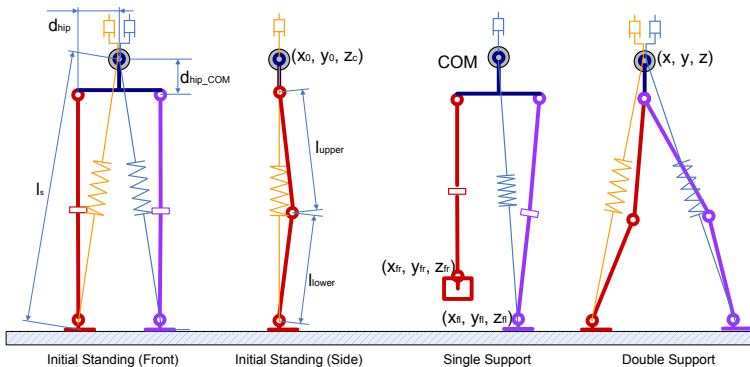

Fig. 2 Virtual spring-damper model

Define x, y, z and x_f, y_f, z_f are the position of the COM and the ankle of the support leg respectively. The acceleration exerted by the spring can be derived according to Hooke's law.

The rest length of the the virtual spring is

$$l_0 = \sqrt{\left(l_{upper} + l_{lower} + d_{hipCOM}\right)^2 + d_{hip}^2}. \tag{10}$$

The length of the the virtual spring of the stance leg is

$$l_s = \sqrt{(x - x_f)^2 + (y - y_f)^2 + (z - z_f)^2}. \tag{11}$$

The force produced by the virtual spring is

$$f = k(l_0 - l_s). \tag{12}$$

The acceleration caused by the force of the virtual spring is

$$\begin{aligned} a &= \frac{k}{m}(l_0 - l_s) \\ &= K(l_0 - l_s). \end{aligned} \tag{13}$$

The vertical component of the acceleration is

$$\ddot{z} = K(l_0 - l_s)\frac{z - z_f}{l_s}. \tag{14}$$

Setting the origin of the world coordinate at the height of the ankle joint, we have $z_f = 0$, yields

$$\begin{aligned} \ddot{z} &= K(l_0 - l_s)\frac{z}{l_s} \\ &= K(\frac{l_0}{l_s} - 1)z. \end{aligned} \tag{15}$$

The acceleration contributed by the virtual spring force is thus obtained as in (15). By adding the gravity constant and the acceleration produced by the virtual damper, the vertical component of the overall acceleration can be easily computed as

$$\ddot{z} = K(\frac{l_0}{l_s} - 1)z - g - C\dot{z}. \tag{16}$$

2.3 Mathematical Formulation

We have studied smooth transition strategies to minimize the magnitude of the acceleration deviation $\Delta\ddot{z}$ in order to compensate for the overall ZMP error in the second stage of preview control. The strategies include three different means of altering

the spring damper parameters according to the walking phases. The walking phases such as standing, single support, and double support are defined by the foot ground contact.

1. Transition from initial standing posture to single support phase: each leg in the standing phase uses half of the stiffness of that of the stance leg in the single support phase.
2. Transitions from single support phase to double support phase: set the initial spring length in the event of touch-down as its temporary rest length l_0', ensuring the vertical acceleration exerted by touch-down leg increases from zero.
3. Transitions from double support phase to single support phase: restore the original spring rest length l_0 of the stance leg.

The simulated robot at the initial standing phase has the mass-less stiffness of $0.5K$ in each leg. So the overall stiffness of two legs is K. By doing so when the robot starts the first single support phase, its new support leg also has the same stiffness K as the overall stiffness of two legs in the standing posture, therefore, the acceleration term $\ddot{z}$ doesn't vary significantly when the gait starts. When the robot enters the double support phase, the touch-down leg is not fully straightened for avoiding the knee singularity. So at the very beginning of touch-down, the virtual spring length l_s is already shorter than its rest length l_0. Consequently, it could generate a non-zero initial force and result in an offset force input which produces large acceleration $\ddot{z}$. To avoid this, the virtual spring length l_s in the event of touch-down is set as the temporary rest length, denoted as l_0', to ensure that the $\ddot{z}$ exerted by touch-down leg increases from zero. This realizes a smooth transition of the vertical acceleration. When the coming single support phase starts, the original rest length of the spring is restored for the support leg. The usage of the virtual damper primarily filters the force spikes caused by this stiffness variation. With these smooth transition strategies, the variation of Δz and $\Delta \ddot{z}$ can be treated as small parametric disturbances which can be compensated by the second loop of the preview controller.

In the standing position, the initial COM position is denoted as (x_0, y_0, z_c). The spring force produced by each leg is computed as in (15). Using superposition, we obtain the equation of the equilibrium point where the force of two virtual springs counterbalances the gravity.

$$(\frac{2l_0}{\sqrt{d_{hip}^2 + z_c^2}} - 2)z_c = \frac{g}{K}. \tag{17}$$

Rewrite (17), yields

$$z^4 + \frac{g}{K}z^3 + (\frac{g^2}{4K^2} + d_{hip}^2 - l_0^2)z^2 + \frac{gd_{hip}^2}{K}z + \frac{g^2 d_{hip}^2}{4K^2} = 0. \tag{18}$$

Solving (18) gives the value of z_c as the constant COM height in the state space equation of the preview controller. Given the initial condition $z(0) = z_c, \dot{z}(0) = 0, \ddot{z}(0) = 0$, the COM state $(z(i), \dot{z}(i), \ddot{z}(i))$ can be computed by numerical integrations according to the dynamic equations.

In the single support phase, the dynamic equation is

$$\ddot{z} = K(l_0/l_s - 1)z - g - C\dot{z}, \tag{19}$$

The virtual spring length of the stance leg are

$$l_{s_l} = \sqrt{(x - x_{fl})^2 + (y - y_{fl})^2 + z^2} \tag{20a}$$

$$l_{s_r} = \sqrt{(x - x_{fr})^2 + (y - y_{fr})^2 + z^2} \tag{20b}$$

for the left and right support leg respectively.

In the double support phase, the dynamic equation is

$$\ddot{z} = K(l_0/l_{s_{old}} + l'_0/l_{s_{new}} - 2)z - g - 2C\dot{z}, \tag{21}$$

where l'_0 is the temporary rest spring length of the latest touch-down leg $l_{s_{new}}$.

3   Simulation

The joint trajectories of the stable walking are generated by the gait pattern generator in MATLAB. In the first control stage, the simplified model presented in the previous section is used to generate the spatial trajectory of the COM and a multibody model including the mass and inertia is used to compute the explicit ZMP as in (1). In the second control stage, the error of the desired ZMP and the explicit ZMP is used by the preview controller to generate a modification of the horizontal motion to minimize the ZMP error. The final output of the COM trajectory and the foot trajectory are used to solve the inverse kinematics to obtain the joint trajectories as the reference inputs for joint tracking controllers. Fig. 3 shows the entire control architecture of the trajectory generator.

In Fig. 4, the red lines are the results from gait pattern generation without smooth transition strategies, while the blue lines are those with the strategies applied. Fig. 4(a) shows that without the smooth transition strategies the acceleration $\ddot{z}$ is large, resulting in large e_x which causes the real ZMP to drift away from the one formulated by the cart-table model as shown in Fig. 4(b). Thus, it is more difficult for the preview controller to compensate for the ZMP errors in the second control stage. In Fig. 4(b), the smooth transition strategies minimize e_x within $7mm$, so the cart-table model provides a good representation of the system dynamics even with a certain range of the vertical motion.

The dynamic simulation of the iCub [13] robot is performed in OpenHRP3 [5]. Fig. 5 shows the real iCub robot and its rigid body model in OpenHRP3. The snapshots of the bent knee and the straightened knee walking are shown in Fig. 6(a) and Fig. 6(b) respectively. The difference in the vertical motion are highlighted by the straight/arc lines of the COM in Fig. 6. The walking manner has more natural
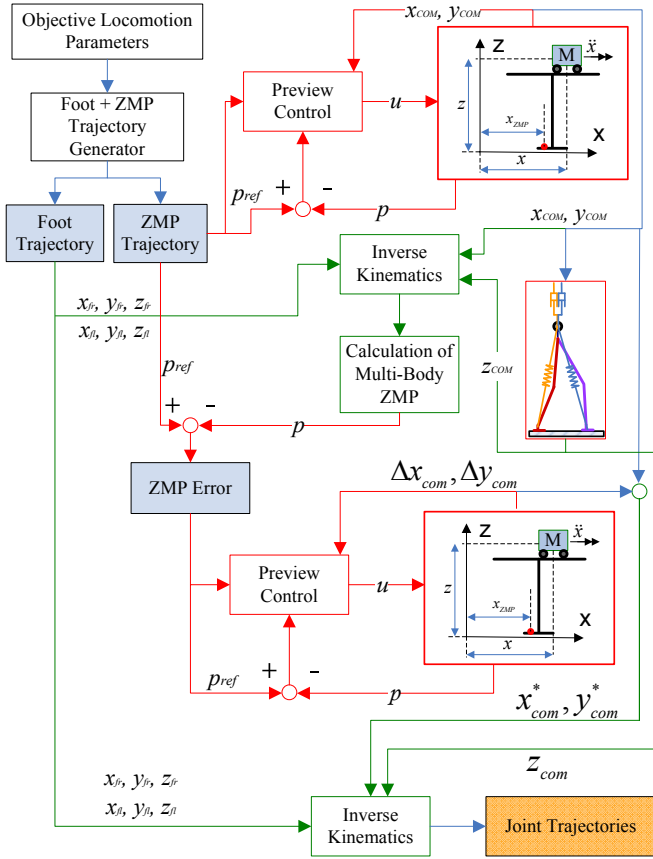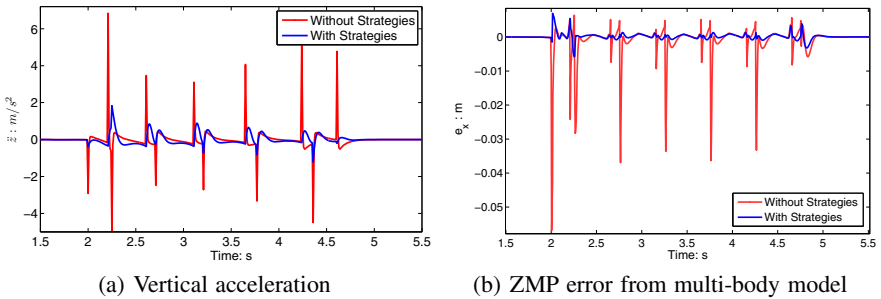
Fig. 3 Overall control architecture

(a) Vertical acceleration

(b) ZMP error from multi-body model

Fig. 4 Vertical acceleration and the resultant ZMP error in gait generation

(a) iCub robot   (b) iCub model
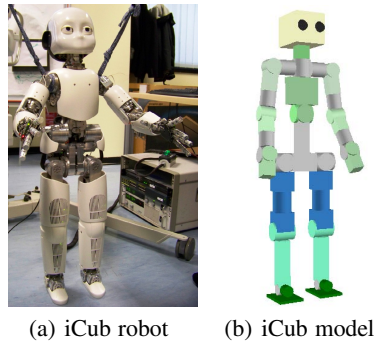
Fig. 5 iCub robot and its model in OpenHRP3 simulator

(a) Bent knee walking

(b) Straightened knee walking
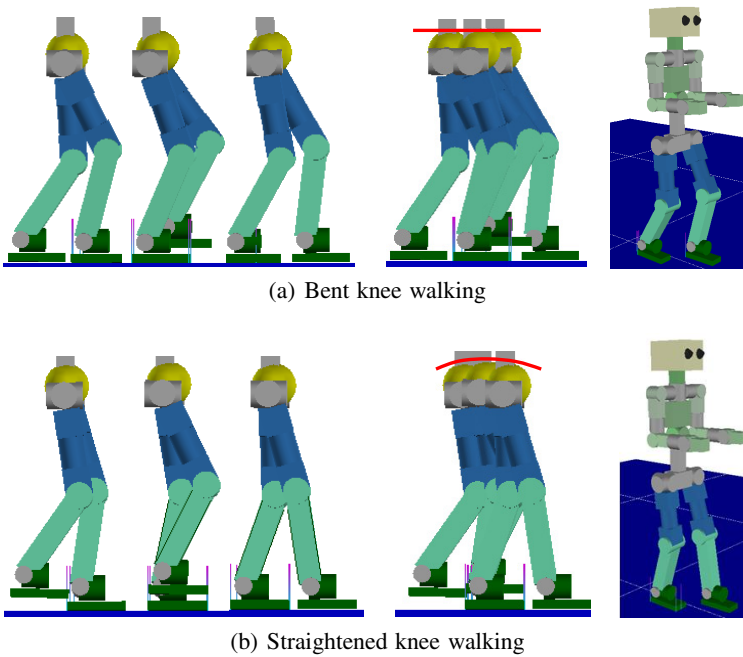
Fig. 6 iCub robot and walking simulation in OpenHRP3 simulator

looking because the robot stretches out its shin to place a new foothold while the stance leg is more straightened rather than a common highly bent profile.

Fig. 7(a) compares the knee joint angles from the simulation of the cart-table model and the proposed scheme. Fig. 7(b) shows the two knee torque profiles from the dynamic simulator OpenHRP3 in the single support phase. It can be seen that the motor torque is reduced in the straightened knee walking. The difference of the

(a) Knee trajectory

(b) Knee torque

(c) Knee angular velocity
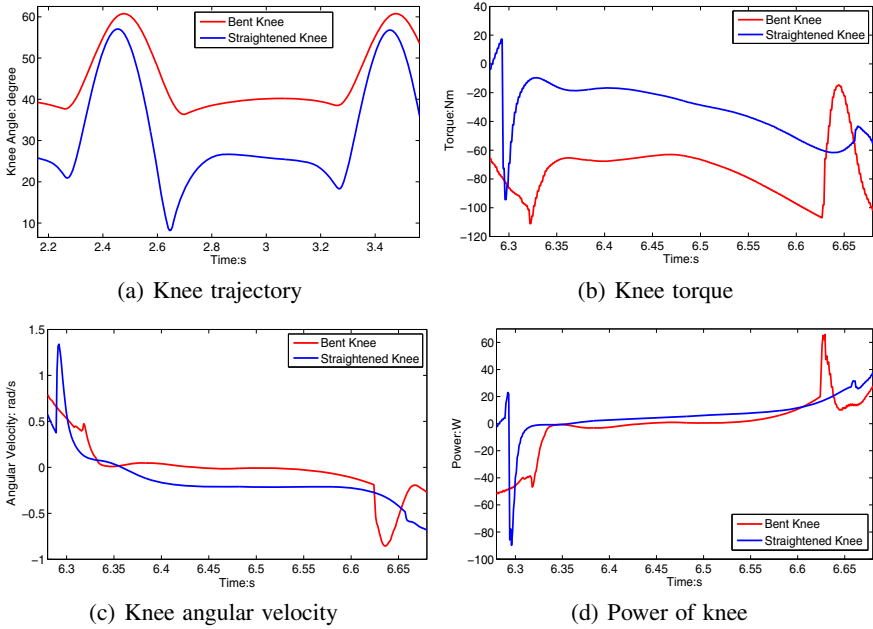
(d) Power of knee

Fig. 7 Simulation results
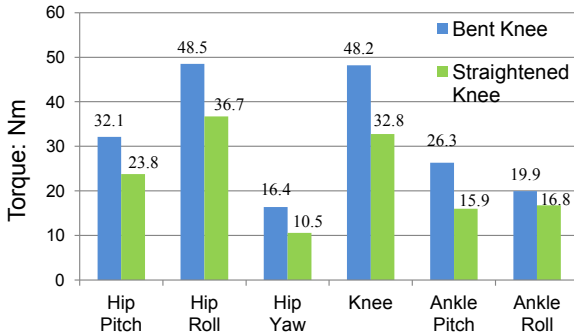
Fig. 8 RMS torque of all joints

angular velocity is shown in Fig. 7(c). Hence, the power of knee can be computed as shown in Fig. 7(d). In the conventional bent walking, knee actuator consumes 4.31J of mechanical energy during a single support phase, while 3.87J is required in a more straightened knee walking, which saves 10.2% of the mechanical energy.

Note that the heat dissipation of electric motors is measured in terms of the current square. Since the motor current is proportional to the motor torque, the root mean square (RMS) torque can be used as an index to evaluate the heat dissipation.

The RMS torque of all the joints are computed based on the original torque data obtained in the OpenHRP3 simulator, shown in Fig. 8. The proposed method significantly reduces the heat dissipation at all joints. The The total RMS torque of all joints decreases from 191.4*Nm* (bent knee) to 136.5*Nm* (straightened knee), saving 28.7% energy from unnecessary heat waste.

(a) Knee profile of human   (b) Knee profile of robot

(c) Human GRF   (d) Simulated GRF
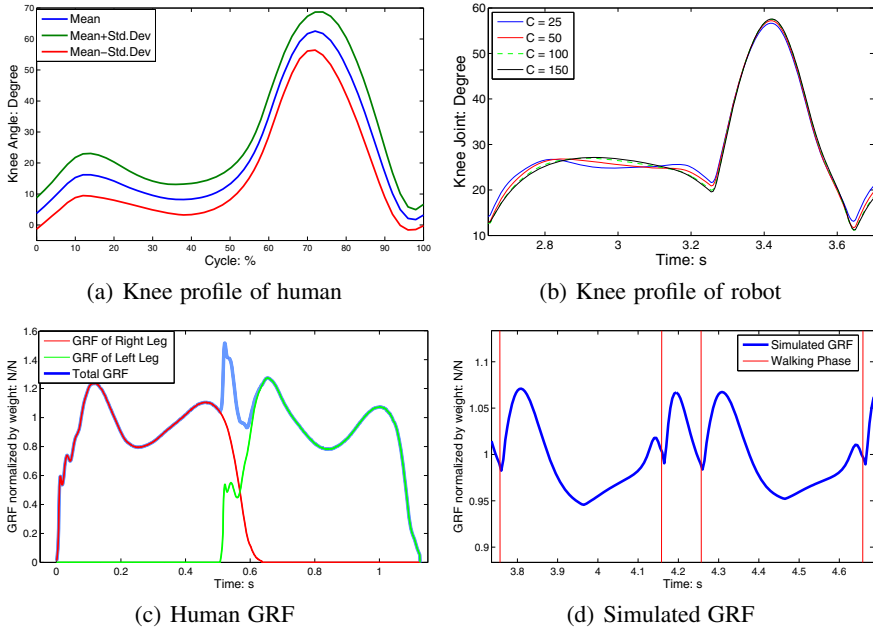
Fig. 9 Comparison of vertical ground reaction force

The work in [14] shows statistically that the maximum knee angle during the stance phase is approximately $23°$ for humans as shown in Fig. 9(a), while the simulated robot with straightened knee is around $26°$ as shown in Fig. 9(b). When the robot places a new touch-down leg, the knee joint angle increases due to the compression of the virtual spring, then decreases because of the decompression of the virtual spring. This results in a convex pattern of the knee joint during the stance phase. Fig. 9(a) reveals a similar convex curve of knee joint in human gait.

Moreover, the compression and decompression of the virtual spring consequently create a double force peaks. This phenomenon is also observed in the study of biomechanical research of human gait [14]. In Fig. 9(c) and Fig. 9(d), the normalized GRF of the human gait and the simulated robot are depicted. In Fig. 9(d), the red vertical lines indicate the switching between the single and double support phases. There are several similar features which reflect the similarities, despite that the GRF of the simulated robot has a smaller magnitude than that of humans.

1. Both the human and the simulated robot show two force peaks larger than the body weight during the single support phase.
2. The GRF has a force peak in the middle of the double support phase.
3. The first force peak is larger than the second one in single support phase.

Certainly, the GRF feature of human comes from different nature than that of the simulated robot. However, the results shown in this study might suggest the possibility of reproducing the similar dynamic features for the robot if a proper modeling is exploited. For example, the double force peak during single support phase originates from the bouncing behavior of the virtual spring and the superposition of two spring force delivers a maximum force magnitude during the mid double support phase. The viscous force from the virtual damper partially dissipates the kinetic energy therefore the second force peak has smaller magnitude than the first one during the single support phase.

4   Conclusion

The proposed method combines the preview control and the virtual spring-damper model for generating walking patterns with more straightened knees which is more similar to humans. The dynamic simulation in OpenHRP3 confirms the effectiveness of proposed control scheme. The investigation of knee joint torque and power shows the feature of energy efficiency. The proposed method saves 10.2% of the mechanical energy of the knee joint and 28.7% of energy from unnecessary heat dissipation for all joint actuators. Therefore it could potentially contribute a longer operation time for stand-alone application.

In this study, we claim a more natural walking manner in terms of more straightened knees during walking. Interestingly, the knee joint profile and the GRF data show the similarities to some extent between the robot and human. Nevertheless, other features such as toe-off and heel-strike are still missing in the proposed method. It could be the research of interest to further study a novel control scheme that generates a more human-like foot motion.

References

1. Alexander, R.M.: Exploring Biomechanics: Animals in Motion. Scientific American Library (1992)
2. Handharu, N., Yoon, J., Kim, G.: Gait pattern generation with knee stretch motion for biped robot using toe and heel joints. In: International Conference on Humanoid Robots, Daejeon, Korea, pp. 265–270 (2008)
3. Huang, Q., Yokoi, K., Kajita, S., Kaneko, K., Arai, H., Koyachi, N., Tanie, K.: Planning walking patterns for a biped robot. IEEE Transactions on Robotics and Automation 17(3), 280–289 (2001)

4. Kajita, S., Kanehiro, F., Kaneko, K., Fujiwara, K., Harada, K., Yokoi, K., Hirukawa, H.: Biped walking pattern generation by using preview control of zero-moment point. IEEE International Conference on Robotics and Automation 2, 1620–1626 (2003)

5. Kanehiro, F., Hirukawa, H., Kajita, S.: OpenHRP: Open architecture humanoid robotics platform. The International Journal of Robotics Research 23(2), 155–165 (2004)

6. Matsui, T., Hirukawa, H., Ishikawa, Y., Yamasaki, N., Kagami, S., Kanehiro, F., Saito, H., Inamura, T.: Distributed real-time processing for humanoid robots. In: IEEE International Conference on Embedded and Real-Time Computing Systems and Applications, pp. 205–210 (2005)

7. McGeer, T.: Powered flight, child's play, silly wheels and walking machines. In: IEEE International Conference on Robotics and Automation, Scottsdale, USA, pp. 1592–1597 (1989)

8. Ogura, Y., Aikawa, H., Lim, H.-O., Takanishi, A.: Development of a Human-like Walking Robot Having Two 7-DOF Legs and a 2-DOF Waist. In: IEEE International Conference on Robotics and Automation, pp. 134–139 (2004)

9. Ogura, Y., Lim, H.-O., Takanishi, A.: Stretch walking pattern generation for a biped humanoid robot. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, vol. 1, pp. 352–357 (2003)

10. Ogura, Y., Kataoka, T., Aikawa, H., Shimomura, K., Lim, H.-O., Takanishi, A.: Evaluation of various walking patterns of biped humanoid robot. In: IEEE International Conference on Robotics and Automation, pp. 603–608 (2005)

11. Pratt, J., Chew, C.M., Torres, A., Dilworth, P., Pratt, G.: Virtual model control: An intuitive approach for bipedal locomotion. The International Journal of Robotics Research 20, 129–143 (2001)

12. Takenaka, T., Matsumoto, T., Yoshiike, T.: Real time motion generation and control for biped robot-1st report: Walking gait pattern generation-. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 1084–1091 (2009)

13. Tsagarakis, N., Metta, G., Sandini, G., Vernon, D., Beira, R., Becchi, F., Righetti, L., Santos-Victor, J., Ijspeert, A., Carrozza, M., et al.: iCub: the design and realization of an open humanoid platform for cognitive and neuroscience research. Advanced Robotics 21(10), 1151–1175 (2007)

14. Winter, D.A.: Biomechanics and motor control of human movement. John Wiley & Sons, Inc. (2009)

Modeling and Control of Dynamically Walking Bipedal Robots

Tobias Luksch and Karsten Berns

Abstract. Today's bipedal robots still cannot compete with humans regarding efficiency, velocity, and robustness of locomotion. Thus, this paper suggests a control concept for dynamic walking based on insights into human motion control. Key features include exploitation of passive dynamics, hierarchical control, and reflexes, while not requiring a full dynamical model. Walking stability is achieved by a set of postural reflexes based on the motion of the extrapolated center of mass. It shows that only a small number of joints must be simultaneously actively actuated during the different phases of walking. Besides the control concept, the anthropomorphic biped model and its properties like compliant actuation are presented as they prove to be essential for the walking performance. Specifically, the approach requires non self-locking and torque-controllable joints with parallel elasticity and low friction, similar to the human muscle-tendon system. The approach is validated for 3D dynamic walking within a physical simulation framework. Results show an efficient, fluent, and fast gait that can cope with considerable disturbances. The resulting joint trajectories show significant resemblance to human walking data.

1 Introduction

Despite several decades of research, locomotion of bipedal robots is still far from achieving the graceful motions and the dexterity observed in human walking. Most of today's bipeds are controlled by analytical approaches based on multibody dynamics, pre-calculated joint trajectories, and Zero-Moment Point considerations to ensure stability [25]. These efforts have been yielding impressive results concerning two-legged locomotion or other movement skills [21, 12, 11]. However, these approaches show several drawbacks like strong model dependency, high energetic and

Tobias Luksch · Karsten Berns
Robotics Research Lab, University of Kaiserslautern, Germany
e-mail: {luksch,berns}@cs.uni-kl.de

computational costs, or vulnerability to unknown disturbances. In contrast, human locomotion is elegant, highly robust, fast, and energy efficient. These considerations gave rise to the two main hypothesis examined in this article: on the one hand, it is postulated that a control system based on insights into human motion control can yield human-like walking capabilities in two-legged robots. On the other hand, it is argued that certain properties of the human morphology are necessary to deploy and to fully exploit such a control system. To this end, a control methodology is derived based on key features of human walking control and is applied to the locomotion of a biped model featuring some characteristics of the human body.

Naturally, there already has been and still is research on the transfer of control and morphology aspects from biology to walking machines. For one thing, the design of most two-legged robots is – at least kinematically – oriented towards human morphology. Regarding the utilization of this morphology as well as its control, examples include the exploitation of inherent dynamics of the mechanical system and elasticities [5, 18, 23], or neural and reflex based approaches [27, 17, 6]. Hence the presented approach aims at differing from previous work regarding the extend of including biological analysis and the resulting applicable control aspects, the manner in which these aspects are transformed into a robot control system, and the complexity of the considered robotic target platform.

The remainder of this paper is structured as follows: Sec. 2 shortly introduces the proposed control concept and its biological motivation as well as its application to dynamic walking control. The biped model is described in Sec. 3, giving some details on its kinematics and actuation system. Sec. 4 presents experimental validation by analyzing the robot's dynamic walking gait. The paper is concluded in Sec. 5.

2   Bio-Inspired Control Approach

A review of biomechanical and neuroscientific research as well as clinical gait analysis reveals several control aspects that can be transferred to technical systems:

- The robot's design should be based on functional morphology. By exploiting the passive system dynamics[1] and self-stabilizing properties of the muscle-tendon system and low-level muscle reflexes, control effort and energy consumption are significantly reduced [26, 7, 4].
- The hierarchical structure of the CNS (central nervous system) facilitates to cope with the high complexity and redundancy of body and control [10, 2].
- Natural locomotion control emerges as a combination of feed-forward and feed-back control. Suitable control units can be derived from gait analysis [20].
- Besides passive control, human walking is stabilized by reflex action, ranging from local reflexes to supraspinal postural control [28, 9].

---

[1] Here, passive system dynamics are understood as the inherent dynamical motions introduced by the mechanical system, e. g. by inertia of segment masses.

- Based on motor synergies extracted from EMG (electromyography, i.e. muscle activity) recordings of normal walking, the gait cycle can be divided into several distinct phases featuring bilateral synchronization [10].
- Reflexes action is modulated depending on the current task or gait phase, respectively, i.e. the same stimulus can lead to different reactions [28].

2.1 Structure of Control Concept

These insights are incorporated into the following control concept [13, 14]. Since passive dynamics should be exploited, no whole body joint trajectories are used. Rather, as in biological systems, the motion commands emerge from a hierarchical network of control units. Being based on behavioral robot control ideas, these units already have a semantic interpretation and thus are located above the level of individual neurons.
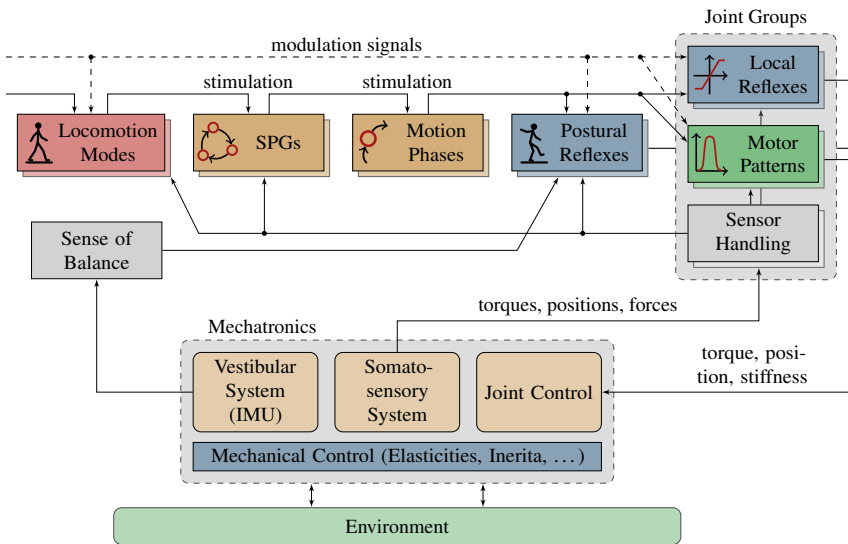
Fig. 1 The hierarchical organization of control units and the interaction of the control system, the robot's mechatronics and the environment

Fig. 1 illustrates the hierarchical layout defined by the flow of stimulation, inhibition, and modulation between six classes of control units. *Locomotion modes* are located at the highest level and represent the form of locomotion like walking or standing. They stimulate *spinal pattern generators* (SPG), state machine-like units triggered by kinetic or kinematic events instead of using oscillators with fixed timing.

Motion phases provide for synchronized stimulation of feed-forward control commands and activate the appropriate feedback units. Feed-forward control is

issued by *motor patterns* in the form of local torque impulses directed at only one or a few adjacent joints. These torque patterns shape the natural motion of the robot created by the passive system dynamics to create the desired motion. All torque impulses are all defined by the same parametrized sigmoid function to simplify the design of the control system.

Feedback is implemented by local and postural reflexes: *local reflexes* only affect spatially related joints based on data of adjacent sensors and introduce a tight sensor-actor coupling. *Postural reflexes* require whole body sensor information and can use simplified dynamic models to calculate their reaction. Modulation signals like the desired walking velocity can influence the output of control units.

The bottom of Fig. 1 represents the mechatronics of the robot and its interaction with the environment. The joint control detailed in Sec. 3 allows to set the desired torque, position, or stiffness of each joint. Further, the robot is equiped with with various sensor systems that take the function of parts of the human somatosensory and proprioceptive senses.

2.2   *Controlling Dynamic Walking*

The suggested control method is applied to the task of bipedal dynamic walking and stable standing. To do so, certain system premises must be met regarding both the robot's morphology and the architectural framework of the control system. The characteristics of the morphology and the actuation system will be discussed in Sec. 3. Concerning the control framework, behavior-based architectures have been established as a preferred control approach for robotic systems acting in situations and environments that are not known a priori. The behavior-based control architecture iB2C (**i**ntegrated **B**ehavior-**B**ased **C**ontrol)[2] is well suited for implementing the aspired hierarchical layout, the control unit classes, or stimulation and inhibition mechanisms. Further details on iB2C and the implementation of the control system introduced above can be found in [19] and [13].

Having selected a target locomotion mode, functional control units need to be selected and designed to achieve the desired motions. Findings from human gait analysis and neuroscience are consulted to identify suitable units of the six classes. In the following, the resulting control network for dynamic walking is presented.

At the highest level, the locomotion mode for periodic walking stimulates the walking SPG as soon as the walking initiation process [15] has finished. This state-machine-like SPG is responsible for cyclic walking and the respective walking phases. It is derived from findings based on statistical analysis of EMG recordings of human walking [10, 24] and manages five motion phases, namely weight acceptance, leg propulsion, trunk stabilization, leg swing, and heel strike. Fig. 2 depicts the progression of these phases and the corresponding kinetic and kinematic events that trigger the state transitions. Bilateral synchronization as it is observed in human gait is

---

[2] iB2C as well as the corresponding robot control framework MCA2 can be downloaded at `http://rrlib.cs.uni-kl.de`.
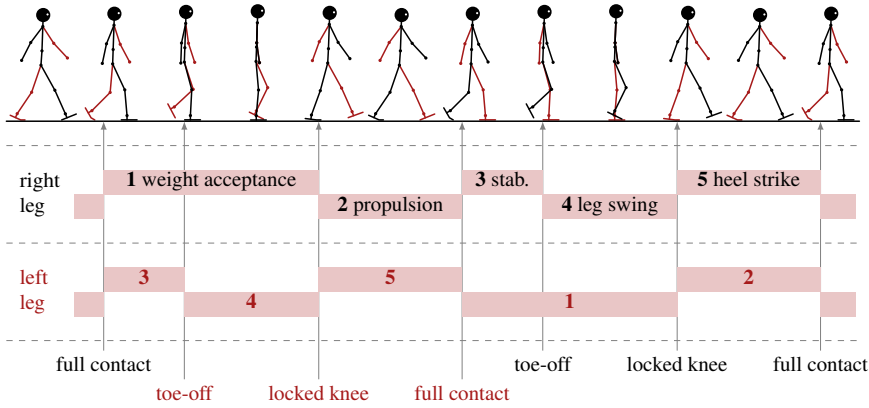
Fig. 2 Phases of walking illustrated for the left and right side. The sensor events responsible for triggering phase transitions are shown at the bottom.

achieved by simultaneously switching phases of both body sides. As a benefit, this also reduces the amount of sensor events necessary to trigger walking phases and thus increases the robustness of the state machine.

Based on results from biomechanical gait studies, motor patterns are identified for each of the phases to shape the natural motion of the robot based on passive dynamics towards a walking gait by applying selective torque commands (see [13] for more details). This process consults kinematic and kinetic analysis as well as EMG data. While the latter cannot easily be used to obtain quantitative statements, muscle activities and synergies still provide valuable clues concerning possible control goals of the CNS during individual phases of locomotion. Furthermore, this data indicates which DOFs the nervous system has evolved to consider as relevant during each phase of motion, thus helping to cope with the "DOF problem" as stated by Bernstein [1]. Concerning feedback control, local reflexes similar to those working in human walking are inserted, e. g. the cutaneous reflex.

Controlling the stability during walking is based on the combined efforts of local reflexes as well as several postural reflexes. The latter independently control the robot's forward velocity, its lateral stability, and the pose of the upper body. This is done by adjusting the torques of the ankle, hip, or spine joints, or, most importantly, by adapting the foot placement. Similar to the supraspinal postural reflexes in human balance control, these control units cannot purely rely on local sensor signals but must make use of simplified forward models.

There is evidence from gait analysis that human subjects control postural stability based on the estimated movement of the center of mass (CoM). Hof et al. suggest a control strategy for foot placement relying on the extrapolated center of mass [8]. To adopt a similar approach for the functioning of postural reflexes, an estimation of the current position of the CoM becomes necessary. Fig. 3 illustrates the simplified pendulum model used for this estimation. For both the frontal and lateral direction the distance d of the horizontal component of the CoM from each foot contact point
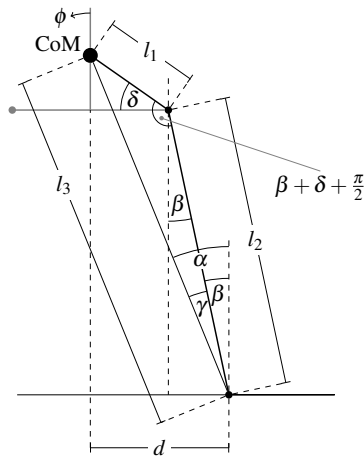
Fig. 3 Pendulum model for estimating the ground projection d of the center of mass relative to the stance foot position

is calculated. Following Hof's suggestions, the extrapolated center of mass (XCOM) can be derived from these values by including the velocity $\dot{d}$ of the respective COM projection normalized by the eigen frequency ω_0 of the assumed inverted pendulum model:

$$xcom = d + \frac{\dot{d}}{\omega_0} = d + \frac{\dot{d}}{\sqrt{\frac{g}{l_3}}}$$

The calculated values are used as indication for postural adjustments. More specifically, the trajectories of the COM and XCOM positions during normal, undisturbed walking are approximated by simple functions. The derivations of the actual values from these trajectories serve as excitatory signals for the postural reflexes.

3 Biped Model

The suggested control concept requires certain characteristics of the underlying mechatronics system. For instance, beneficial effects of the passive system dynamics cannot be exploited if the robot's joints are self-locking or have overly high friction. Similarly, self-stabilizing properties of elastic elements can only be used if such elastic elements are present. A gait then emerges from the combination of local torque commands issued by the control units and the interaction of the "intelligent mechanics" with the environment.

The choice of kinematic degrees of freedom is based on human gait analysis identifying the most relevant joints for walking. On the right side of Fig. 4 the rotational axes of the model are illustrated. Instead of an additional toe joint in the foot, a
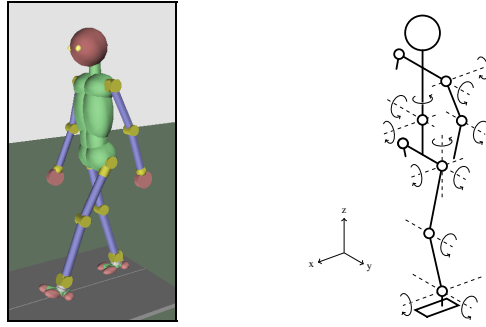
Fig. 4 Left: Simulated biped as visualized by the simulation framework. Right: Joint layout of the biped model. The hip and spine joints are modeled by three, the shoulder and ankle joints as two subsequent revolute joints.

curved geometry of the frontal part of the foot is selected to allow for the rolling action during the second half of the stance phase. The spine is reduced to three DoFs for simplicity. Similar, the complexity of shoulder and arm kinematic is decreased as the upper extremities are of minor importance during walking. The overall system amounts to 21 DoFs, the robot's height to 1.8 m. Weight distribution is based on average human data, with the total weight adding up to 76 kg.

Similar to the kinematics, the actuation system must possess certain properties to make a human-like motion control possible. Hill's mechanical muscle model features serial and parallel elastics elements, dampers, and a contracting unit for each muscle. A human joint is actuated by at least two of those muscles in an antagonistic setup. This allows for both torque control and position control with a large range of stiffness. For this work, direct joint actuation is assumed, neglecting the antagonistic setup. This has the advantage of reducing complexity in design and control. A drawback could arise as no biarticular structures can be implemented mechanically.

It is assumed that each joint can swing freely with low friction if no control commands are given. A direct torque demand τ_{target} is possible. Beside this, position control towards α_{target} with variable compliance simulates the series elastic elements of the muscle. A parallel elastic element with a fixed spring constant and equilibrium point concludes the joint actuation characteristics.

A block diagram of the resulting joint model is shown in Fig. 5. The stimulation values s_τ and s_α are used for a weighted fusion of the torque command and the torque resulting from the position control loop:

$$\tau_{\text{control}} = \frac{s_\tau(\tau_{\text{target}}s_\tau) + s_\alpha(\tau_{\text{pos}}s_\alpha)}{s_\tau + s_\alpha}$$

Thus, stimulation s_α can be interpreted as compliance of the joint. The characteristic of the parallel spring is modeled to be quadratic with a fixed equilibrium point.
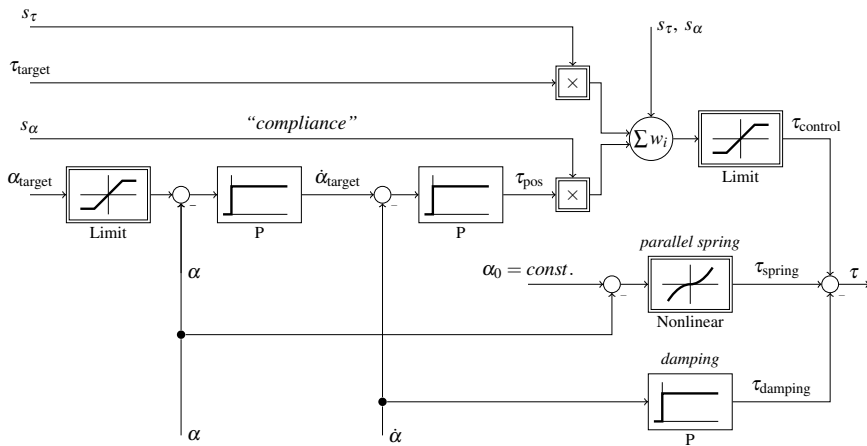
Fig. 5 Block diagram of the joint model calculating the resulting torque based on the given torque and position commands as well as the corresponding stimulation values

As for the joint actuation, some requirements have to be met regarding the sensor configuration, too. Corresponding to the capabilities of sensor organs in human muscles, each joint outputs the current angular position and the acting torque. Cutaneous and load receptor information is required by certain reflexes in human motion control. Consequently, the model includes force sensors in each of the robot's feet providing data on ground contact and load distribution. Finally, postural reflexes depend on an estimation of the upper trunk's pose and movements. For this purpose, an inertial measurement unit (IMU) is installed in the upper body.

4 Experimental Validation

For experimental validation, the suggested biped model is implemented within a full-featured dynamics simulation framework. The left side of Fig. 4 shows the visualization of the biped model, with the shapes also serving as collision geometries. First steps towards developing a robot prototype and a suitable actuation system have been made [16, 3] but will not be discussed in the scope of this article.

It shows that the presented control concept is indeed capable of achieving three-dimensional dynamic walking of an anthropomorphic bipedal robot, including the transition from actively balanced standing to walking [15]. The naturally looking gait emerges from the combined control outputs of the phase-dependent stimulated reflexes and motor patterns, passive dynamics, and the interaction of the robot with its environment. Less than half of the 21 DOFs need to be actively controlled simultaneously during each phase of walking. The remaining joints act passively at different degrees of compliance. Even more, for each phase the control of only 1–4 DOFs by motor patterns is sufficient to direct the passive dynamics towards a walking gait.
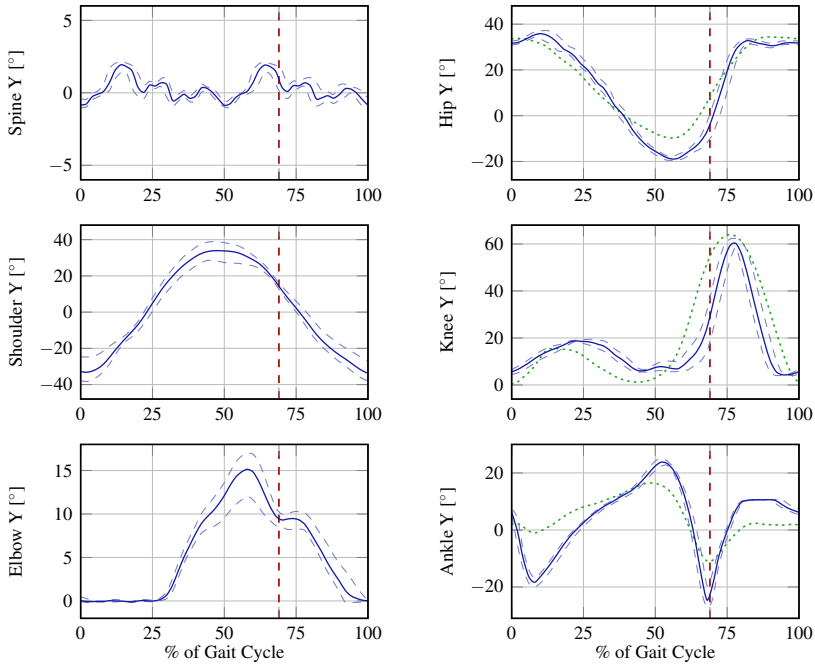
Fig. 6 Joint angles in the sagittal over the course of one gait cycle. The solid lines represent the mean values over 30 steps of normal walking on even ground. The dashed lines mark the minimum and maximum values. The vertical dashed line indicates the transition from stance to swing. For the hip, knee, and ankle joint, human data is given as dotted line for comparison.

These facts comply with the "principle of minimal intervention" as formulated for human movement control [22].

Fig. 6 shows average angle trajectories of the joints rotating in the sagittal plane from one heel strike to the next. The vertical dashed line marks the transition from stance to swing. It can be seen that the spine joint is kept relatively stable. The sine-like trajectory of the shoulder joint results from the passive dynamics combined with short torque impulses as the arm is mainly swinging freely. Except for low elasticities, the elbow acts purely passively. As in human walking, the knee flexion during the swing phase emerges also mainly due to passive dynamics.

These kinematic as well as the kinetic results show strong resemblance to human walking (dotted lines on the right side of Fig. 6). Joint angle trajectories possess the same characteristics and even similar amplitude despite the fact that no joint angle control based on human data has been applied. Further similarities can be found in the trajectories of the absolute position of body segments, joint torques, and power values. This resemblance can be explained by the likewise exploitation of passive system dynamics at comparable segment masses and lengths, and by the

assumption that indeed essential features of human motion control have been iden-
tified and transferred.

Other then in purely passive dynamic walking machines, the presented approach
allows for active variation of the resulting gait by using modulation signals (dashed
line in Fig. 1). This mechanism is validated for controlling the walking velocity. For
the modeled gait, the velocity can be adjusted between 1.1 m/s and 1.4 m/s, with the
corresponding emerging step length ranging from 0.65 m to 0.95 m.

The efficiency of the resulting gait is analyzed by an estimation of the mechan-
ical joint work and power output based on the torque and rotational displacement
of the joints. The total mechanical power consumption during walking amounts to
approximately 300 W. By normalizing this value to the robot's weight and the walk-
ing velocity, the cost of transport can be approximated. Comparing this result to a
similar estimation of joint angle controlled machines yields an efficiency improve-
ment by the factor five [5]. Still, energy consumption considerably exceeds that of
humans or actuated passive walkers. However it should be noted that the control
system has not been optimized towards low energy consumption at all. Certainly,
simulation can only provide a rough guess on the necessary power when applying
the control concept to a real robot, but nevertheless a considerable energetic benefit
is to be expected. The observed peak values of joint angles, motor torques, or power
can easily be achieved by today's hardware.

Already during normal walking, local and postural reflexes help to stabilize the
robot against self-induced disturbances and against the impacts occurring at each
foot contact. To further evaluate the robustness of the control system, the robot is
confronted with different ground geometries and arbitrary external forces acting on
body segments. Importantly, it should be noted that all experiments are performed
with identical model and control parameters.

To illustrate postural control on one example, Fig. 7 presents the effects of terrain
inclined orthogonally to the walking direction. It compares lateral foot placement
and the activity of the *Lateral Balance Ankle* postural reflex on level terrain and
ground sloped downwards to the right by 1.5°.

The most obvious distinction can be observed in the action of the *Lateral Foot
Placement* postural reflex. The inclination pushes the robot to the right and rotates
the whole machine around the x-axis with every step it takes. The reflex reacts to
this disturbances by placing the feet further to the right than in normal walking. As
illustrated by the foot point diagrams, the left foot shown as green circle is placed al-
most in front of the last stance foot, whereas the right foot is positioned up to 60 cm
to the right. As a result, the robot is walking sidewards down the slope, whereas on
level ground, the minor foot placement corrections do not deflect the robot distinctly
from its path. In addition, the *Lateral Balance Ankle* reflex also exhibits increased
activity. It shows that the robot cannot maintain stability during walking on side-
wards slopes of more than 2°. Mainly, the robot is expecting level ground at heel
strike but instead the foot must be lowered even further. Since the hip is kept rigid
during this phase, the robot tilts to the side.
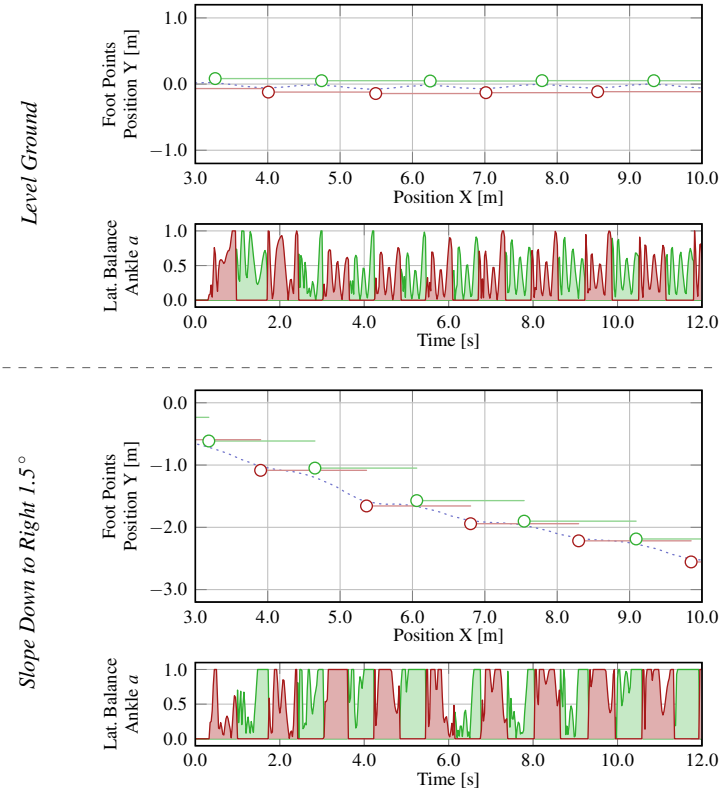
Fig. 7 Comparison of lateral foot placement and the activity of the *Lateral Balance Ankle* postural reflex on level terrain and ground sloped to the right

Further experiments on robustness focus on ground sloped in walking direction, external pushes, or walking over steps and on irregular terrain. The control manages downhill slopes of up to 5° and uphill slopes of 2° before walking becomes instable. Main difficulties lie in decreased lateral stability during downhill walking caused by increasingly long steps and insufficient ground clearance during uphill walking. While the robot can cope with higher pushes of short duration, the limit for constant external forces in walking direction arises to 15 N, lateral forces should remain below 8 N. The height of steps the robot can walk up is restricted to about 3 cm, mainly due to the small ground clearance of the feet during the swing phase. Stepping downwards remains stable to up to 5 cm. During walking on irregular terrain modeled by random, smoothed height fields, differences in height should not exceed 3 cm.

5   Summary and Outlook

This paper presented a control methodology for two-legged locomotion and the corresponding biped model, both based on findings from studies on human motion control. Located above the neural level, the control system is structured as a hierarchical network of local feed-forward and feedback units. Neither a complete dynamic model nor pre-calculated joint trajectories are used. Sensor event-based spinal pattern generators coordinate the stimulation and synchronization of control units and the compliance setting of the passive joints. By mostly applying local torque commands to only a subset of the robot's DoFs instead of using joint angle control, passive dynamics can be exploited. Walking robustness is enhanced by several postural reflexes based on simple CoM considerations.

Tested on a dynamically simulated anthropomorphic biped with 21 DoFs, the control system can achieve stable, 3D dynamic walking of variable velocity. It is able to cope with the high complexity and the compliance of the modeled biped. The emerging, naturally looking walking gait shows significant similarities to human walking. Simultaneous actuation of only a subset of joints is sufficient to direct the natural motion of the robot as introduced by passive dynamics towards a walking gait. At a high walking velocity of up to 5 km/h, the gait is more energy efficient than in most joint angle controlled robots. The control system shows considerable inherent robustness against unknown and unexpected disturbances.

Future work will include modulation signals for e. g. changing the walking direction, additional postural reflexes to increase robustness, or further locomotion modes like running. To better cope with the parameter multitude, machine learning and optimization techniques will be applied.

References

1. Bernstein, N.: The co-ordination and regulation of movements. Pergamon (1967)
2. Bizzi, E., Cheung, V., d'Avella, A., Saltiel, P., Tresch, M.: Combining modules for movement. Brain Research Reviews 57 (2007)
3. Blank, S., Wahl, T., Luksch, T., Berns, K.: Biologically inspired compliant control of a monopod designed for highly dynamic applications. In: Proc. of IEEE Int. Conf. on Intelligent Robots and Systems (2009)
4. Blickhan, R., Seyfarth, A., Geyer, H., Grimmer, S., Wagner, H., Günther, M.: Intelligence by mechanics. Philosophical Transactions of the Royal Society of London, Series A 365 (2007)
5. Collins, S., Ruina, A., Tedrake, R., Wisse, M.: Efficient bipedal robots based on passive-dynamic walkers. Science 307 (2005)
6. Endo, G., Nakanishi, J., Morimoto, J., Cheng, G.: Experimental studies of a neural oscillator for biped locomotion with QRIO. In: Proc. of IEEE Int. Conf. on Robotics and Automation (2005)
7. Fischer, M., Blickhan, R.: The tri-segmented limbs of therian mammals: kinematics, dynamics, and self-stabilization - a review. Journal of Experimental Zoology Part A: Comparative Experimental Biology 305A(11) (2006)

8. Hof, A.: The 'extrapolated center of mass' concept suggests a simple control of balance in walking. Human Movement Science 27(1) (2008)
9. Horak, F.B.: Postural orientation and equilibrium: what do we need to know about neural control of balance to prevent falls? Age and Ageing 35-S2 (2006)
10. Ivanenko, Y.P., Poppele, R.E., Lacquaniti, F.: Motor control programs and walking. The Neuroscientist 12(4) (2006)
11. Kaneko, K., Harada, K., Kanehiro, F., Miyamori, G., Akachi, K.: Humanoid robot HRP-3. In: Proc. of IEEE Int. Conf. on Intelligent Robots and Systems (2008)
12. Kim, J., Park, I., Oh, J.: Walking control algorithm of biped humanoid robot on uneven and inclined floor. Journal of Intelligent and Robotic Systems 48(4) (2007)
13. Luksch, T.: Human-like Control of Dynamically Walking Bipedal Robots. RRLab Dissertations. Verlag Dr. Hut (2010)
14. Luksch, T., Berns, K.: Controlling dynamic motions of biped robots with reflexes and motor patterns. In: Proc. of Int. Symposium on Adaptive Motion of Animals and Machines (2008)
15. Luksch, T., Berns, K.: Initiating normal walking of a dynamic biped with a biologically motivated control. In: Proc. of Int. Conf. on Climbing and Walking Robots (2008)
16. Luksch, T., Berns, K., Mombaur, K., Schultz, G.: Using optimization techniques for the design and control of fast bipeds. In: Proc. of Int. Conf. on Climbing and Walking Robots (2007)
17. Manoonpong, P., Geng, T., Porr, B., Wörgötter, F.: The RunBot architecture for adaptive, fast, dynamic walking. In: Proc. of IEEE Symposium on Circuits and Systems (2007)
18. Pratt, J.: Exploiting inherent robustness and natural dynamics in the control of bipedal walking robots. PhD thesis. MIT Press, Cambridge (2000)
19. Proetzsch, M., Luksch, T., Berns, K.: Development of complex robotic systems using the behavior-based control architecture iB2C. Robotics and Autonomous Systems 58(1) (2010)
20. Rossignol, S., Dubuc, R., Gossard, J.P.: Dynamic sensorimotor interactions in locomotion. Physiological Reviews 86 (2006)
21. Takenaka, T., Matsumoto, T., Yoshiike, T., Shirokura, S.: Real time motion generation and control for biped robot. In: Proc. of IEEE Int. Conf. on Intelligent Robots and Systems (2009)
22. Todorov, E.: Optimality principles in sensorimotor control. Nature Neuroscience 7(9) (2004)
23. Vanderborght, B., Verrelst, B., Ham, R.V., Damme, M.V., Beyl, P., Lefeber, D.: Development of a compliance controller to reduce energy consumption for bipedal robots. Autonomous Robots 24(4) (2008)
24. Vaughan, C., Davis, B., O'Connor, J.: Dynamics of human gait. Human Kinetics Publishers, Champaign (1992)
25. Vukobratovic, M., Borovac, B.: Zero-Moment Point - thirty five years of its life. International Journal of Humanoid Robotics 1 (2004)
26. Witte, H., Hoffmann, H., Hackert, R., Schilling, C., Fischer, M., Preuschoft, H.: Biomimetic robotics should be based on functional morphology. Journal of Anatomy 204(5) (2004)
27. Zaier, R., Kanda, S.: Adaptive locomotion controller and reflex system for humanoid robots. In: Proc. of IEEE Int. Conf. on Intelligent Robots and Systems (2008)
28. Zehr, E., Stein, R.B., Komiyama, T.: Function of sural nerve reflexes during human walking. The Journal of Physiology 507(1) (1998)

In Humanoid Robots, as in Humans, Bipedal Standing Should Come before Bipedal Walking: Implementing the Functional Reach Test

Vishwanathan Mohan, Jacopo Zenzeri, Giorgio Metta, and Pietro Morasso

Abstract. This chapter describes a computational architecture for coordinating the degrees of freedom of the humanoid robot iCub during bipedal standing, with particular reference to the Whole Body Reaching and the Functional Reach Test.

1 Introduction

In humans the ability to stand up on two legs is a necessary prerequisite for bipedal walking. Moreover, there is ample neurophysiological evidence that standing and walking are rather independent control mechanisms. Therefore, we suggest that also humanoid robots should be trained first to master the unstable standing posture and then learn to walk.

We shall address such issue in relation with the humanoid robot iCub [1], which has the size of a three years old child (height is 105 cm and weight is 14.2 kg) and has 53 degrees of freedom (DoF): 7 DoFs for each arm, 9 for each hand, 6 for the head, 3 for the trunk and spine and 6 for each leg. iCub is still unable to stand or walk, but only to crawl, as baby toddlers of the same age. Therefore, the goal of this paper is to carry out a preliminary study of the computational processes that may allow iCub to achieve the sensorimotor competence that is necessary for bipedal standing. The study builds upon what has already been achieved in the bimanual coordination of iCub's movements [2], using a biomimetic, force-field based computational model. The model has been evaluated and validated both in a simulated environment and

Vishwanathan Mohan · Jacopo Zenzeri
RBCS Dept, Istituto Italiano di Tecnologia, Genoa, Italy
e-mail: {vishwanathan.mohan,jacopo.zenzeri}@iit.it

Giorgio Metta · Pietro Morasso
RBCS Dept, Istituto Italiano di Tecnologia, Genoa, Italy and DIBRIS,
University of Genoa, Italy
e-mail: giorgio.metta@iit.it, pietro.morasso@unige.it

in real movements. On the contrary, the present study is limited to the simulation stage for 'developmental constraints', because the sensorimotor system of iCub has not matured enough to achieve the features that are necessary for standing (postural control system) and walking (bipedal locomotion system).

As a matter of fact, the postural control system must face two main problems: P1) stabilize the inverted pendulum that characterizes the bipedal standing posture either during quiet standing or when compensating the postural perturbations induced by movements of the upper part of the body; P2) coordinate the redundant set of DoFs of the lower and upper parts of the body in whole body gestures. The focus of this paper is on P2.

The easiest way to solve P1 would be to use a 'stiffness strategy', in particular at the ankle joint. However, this is not what humans do, because the ankle stiffness is dominated by the elasticity of the Achilles tendon and the corresponding stiffness is consistently smaller than the toppling torque due to gravity [3, 4]. Different studies have shown that the ankle torque which is missing from the intrinsic properties of the soft ankle tendons is likely to be supplemented by an active control process [5, 6]: this process can be characterized as an intermittent control mechanism that generates frequent, ballistic bias impulses by soleus and gastrocnemius. It was also shown [7] that this discrete-time feedback controller is much more robust and thus more plausible that an alternative continuous time control mechanism [8] if one considers the large transmission delays in the feedback loop and the intrinsic instability of the bodily inverted pendulum. There is a functional merit to the low-stiffness solution of the postural stabilization mechanism because a rather compliant ankle joint avoids high impact forces with the ground and more easily adapts to uneven surfaces. In other words, a low-stiffness postural controller is more robust and is capable of guaranteeing stability in a much wider range of situations than a stiff controller.

P2 implies a quite different computational problem because functional movements during bipedal standing recruit, in principle, all the DoFs of the global kinematic chain, with a high degree of redundancy, whereas P1 can be considered a 1 DoF or 2 DoFs 'ankle strategy'. As already mentioned, we plan to address this problem by using a biomimetic, force-field based computational model, which takes inspiration from the Passive Motion Paradigm (PMP [9]), extended to include terminal attractor properties [10]. We already used this approach for the coordination of bimanual movements of the humanoid robot iCub [2] and for modeling whole body reaching (WBR) movements in humans [11]. Here we investigate the feasibility of applying this model to the coordination of WBR movements in iCub, with particular emphasis on a specific form of WBR, the Functional Reach Test (FRT), which has been invented as a dynamic clinical measure of balance [12]. FRT measures the distance between the length of the arm and a maximal forward reach in the standing position, while maintaining a fixed base of support. FRT has been tested for both validity and reliability and is used in patients with diagnoses as different as stroke, Parkinson, vestibular hypofunction, multiple sclerosis and hip fractures. FRT has also been associated with an increased risk of fall and frailty in elderly people who are unable to reach more than 15 cm. This study fits in the general framework of

whole body motion of humanoid, highly redundant robots. A number of approaches have been developed. In many cases the underlying inverse kinematic problem is addresses either by formulating it as an optimization problem [13] or by a weighted least norm solution [14]. More specifically, in the case of ASIMO robot redundancy is resolved by selecting one particular out of the infinite number of solutions based on additional criteria like distance to the joint limits [15] or by learning whole body inverse kinematics with a recurrent neural network [16]. Another approach [17, 18] is based on an operational space formulation, which provides dynamic models at the task level and structures for decoupled task and posture control. In most cases, the critical element of the computational architecture is the inversion of a matrix of large dimension, which makes the computational process heavy and scarcely robust. Our proposal, based on the PMP, avoids any matrix inversion and is based on a network which only includes well-posed transformations.
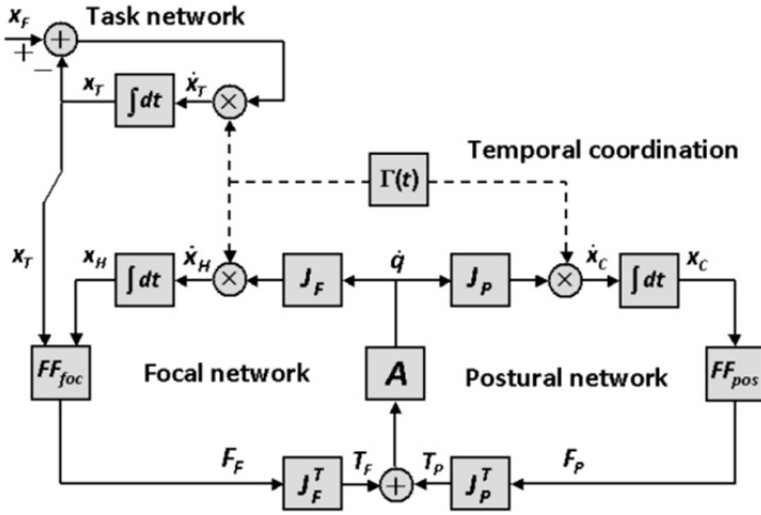
2 FRT Network for iCub

The network architecture which has been developed for allowing iCub to face the Functional Reaching Test is an extension of the architecture developed for modeling whole body reaching movements in humans [11]. The architecture is composed of four parts: 1) Task sub-network, 2) Focal sub-network, 3) Postural sub-network, 4) Temporal coordination unit (see Fig. 1).

The three networks are stable dynamical systems with terminal-attractor characteristics, which is provided by the temporal coordination unit. This unit generates a time-varying gain which is transmitted to the three sub-networks and allows them to reach final equilibrium at the same time:

$$\begin{cases} \Gamma(t) = \frac{\dot{\xi}}{(1-\xi)} \\ \xi(t) = 6\left(\frac{t-t_0}{\tau}\right)^5 - 15\left(\frac{t-t_0}{\tau}\right)^4 + 10\left(\frac{t-t_0}{\tau}\right)^3 \end{cases} \tag{1}$$

Here t_0 is the initiation time and τ is the duration of the coordinated forward reaching movement; $\xi(t)$ is a minimum jerk time base generator, but any smooth function with similar temporal features would yield the same results in terms of temporal coordination of the three sub-networks.

The task sub-network generates a moving target $\mathbf{x}_T$ in 3D which attracts both hands of iCub (represented by the time-varying vector $\mathbf{x}_H$) with a suitable force field, generated by the focal sub-network. $\mathbf{x}_T$ evolves from the initial position of the hands, which are supposed to be jointed, to a final position $\mathbf{x}_F$. This is related to the most common form of FRT, i.e. the bimanual test. We might also implement in the same framework a unimanual paradigm in which one hand is attracted by a forward moving target and the other is either fixed or is used as a further balancing tool. In WBR experiments $\mathbf{x}_F$ may be situated beyond arm's length but should be placed inside the reachable workspace, defined as the set of points that can be reached by keeping the projection on the ground of the center of mass (CoM) within

Fig. 1 FRT network for iCub. $\mathbf{x}_F$, $\mathbf{x}_T$, $\mathbf{x}_H$, $\mathbf{x}_C$, $\mathbf{q}$: vectors of the final and current target position, position of the CoM, and the joint configuration, respectively; $\mathbf{J}_F$, $\mathbf{J}_P$: Jacobian matrices of the focal and postural sub-networks, respectively; $\mathbf{FF}_{foc}$, $\mathbf{FF}_{pos}$: force field generators of the focal and postural sub-networks, respectively; $\mathbf{A}$: admittance matrix; Γ: temporal coordination function.

the support base of the standing robot. Obviously, the support base is a function of the position of the feet and in FRT they are supposed to be parallel and symmetric with respect to the body. In FRT experiments we also positioned $\mathbf{x}_F$ just outside the reachable workspace, in the anterior-posterior direction.

The focal sub-network generates an attractive force field of elastic type which is applied to both hands, implementing the focal part of the task whose goal is to allow the hand to reach the target: $\mathbf{F}_F = \mathbf{K}_F(\mathbf{x}_T - \mathbf{x}_H)$. $\mathbf{K}_F$ is a 3x3 matrix and for simplicity we assumed that it is diagonal. Moreover, in the logic of FRT the force field should be directed in anterior-posterior direction and thus only one component of the matrix is non-zero. The field $\mathbf{F}_F$ is mapped from the extrinsic space to the joint space ($\mathbf{T}_F$) by the following transformation, where $\mathbf{J}_F$ is the Jacobian matrix of the overal kinematic chain (from feet to hands): $\mathbf{T}_F = \mathbf{J}_F^T \mathbf{F}_F$. The admittance matrix $\mathbf{A}$ transforms this torque field into a movement vector $\dot{\mathbf{q}}$ of the kinematic chain, which is mapped to the extrinsic space by the same Jacobian matrix, generating the trajectory of the hand $\mathbf{x}_H$ and thus closing the loop. This dynamical mechanism allows the hand to reach $\mathbf{x}_F$ at the same time in which $\mathbf{x}_T$ reaches $\mathbf{x}_F$, but there is no guarantee that the CoM remains withing the support base in the process. Thus, if only driven by this mechanism, iCub would reach the target but fall forward immediately after. The postural sub-network is intended to prevent such unfortunate event.

The postural sub-network modifies the torque field $\mathbf{T}_F$, generated by the focal sub-network, by adding a component $\mathbf{T}_P$ that takes into account the position of the CoM on the support base ($\mathbf{T}_{TOT} = \mathbf{T}_F + \mathbf{T}_P$). In the vein of the so called 'hip strategy', which characterizes human postural movements, the postural force field is applied to the hip joint and pulls it backward as function of the distance of the CoM from the forward limit of the support base. The activation of the field is meant to induce the following effects: 1) a smaller forward shift of the CoM; 2) a backward shift of the hip; 3) a forward tilt of the trunk associated with the lowering of the CoM. It is worth remarking that this complex control pattern is not explicitly represented but is implicitly coded by the dynamics of the network. The motion of the CoM $\mathbf{x}_C$ is derived from the motion of the whole kinematic chain using a different Jacobian matrix $\mathbf{J}_F$ that only takes into account the ankle and knee joints. The force field applied to the hip was implemented by a non-linear function that diverges to very high values when $\mathbf{x}_C$ approaches the forward limit of the support base $\mathbf{x}_{MAX}$.

In summary, the integrated dynamics of the interacting sub-networks is characterized by the following equations, which achieve a balance between the forward pull applied to the hand and the backward pull applied to the hip:

$$\begin{cases} \dot{\mathbf{x}}_T = \Gamma(t)\,(\mathbf{x}_F - \mathbf{x}_T) \\ \dot{\mathbf{x}}_H = \Gamma(t)\mathbf{J}_F\mathbf{A}\mathbf{T}_{TOT} \\ \dot{\mathbf{x}}_C = \Gamma(t)\mathbf{J}_P\mathbf{A}\mathbf{T}_{TOT} \\ \mathbf{T}_{TOT} = \mathbf{T}_F + \mathbf{T}_P \\ \mathbf{T}_F = \mathbf{J}_F^T\mathbf{K}_F\,(\mathbf{x}_F - \mathbf{x}_T) \\ \mathbf{T}_P = \mathbf{J}_P^T\mathbf{K}_P\frac{\mathbf{x}_C}{\mathbf{x}_{MAX} - \mathbf{x}_C} \end{cases} \qquad (2)$$

3   Simulation Experiments with FRT Network

The computational architecture described in the previous section was tested by using the iCub simulator. The robometric parameters of iCub (length, mass) are summarized here: leg (0.213 m, 0.95 kg); thigh (0.224 m, 1.5 kg); trunk (0.127 m, 4 kg); humerus (0.152 m, 1.15 kg); forearm + hand (0.137 m, 0.5 kg). The head weight is 2 kg.

As suggested by the FRT protocol, the initial posture of the test is characterized the following set of joint angles (ordered from the ankle joint to the elbow joint): $85°$, $92°$, $85°$, $330°$, $0°$. These are absolute values, referred to a horizontal line. With this posture the initial position of the hand reaches a distance of 29.05 cm beyond the vertical line and the CoM is shifted forward 3.87 cm with respect to the ankle joint. The final position of the target was set 5 cm beyond the maximum reachable forward distance and the limit for the CoM displacement ($\mathbf{x}_{MAX}$) was set equal to 13 cm, considering that the length of iCub's foot is 15 cm.

The basic parameters of the FRT control network are 1) the gain of the focal field $\mathbf{KF}$, 2) the gain of the local field $\mathbf{KP}$, 3) and the admittance matrix of the whole kinematic chain $\mathbf{A}$. The latter is a 5x5 matrix but we assumed for simplicity that it is diagonal and thus we only have to choose 5 parameters. The results of
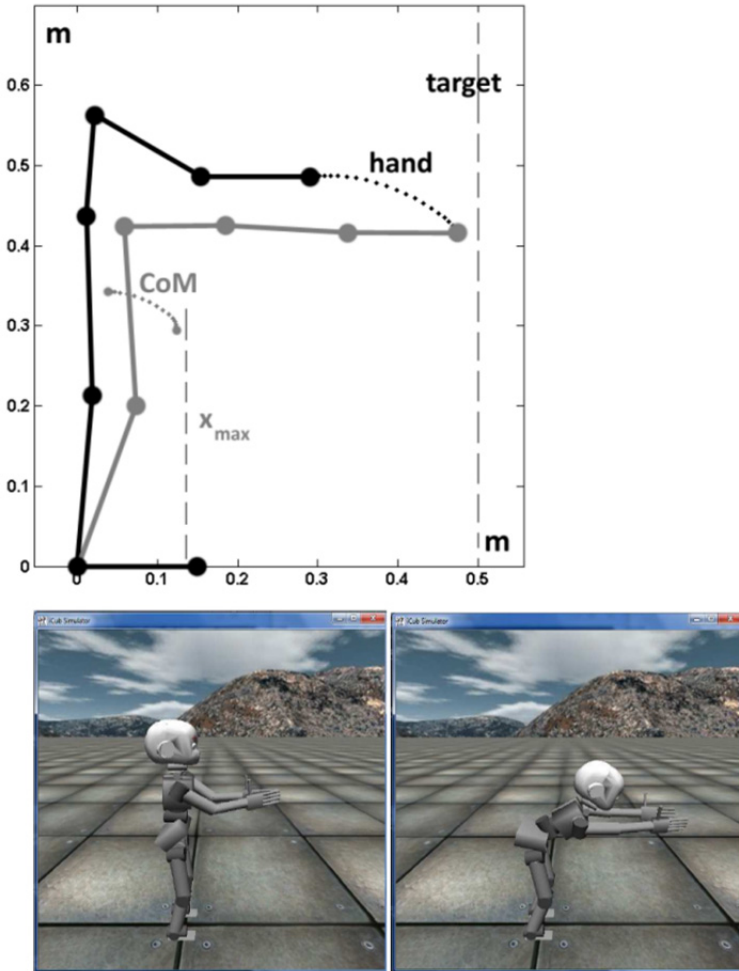
Fig. 2 Initial and final poses of iCub in the Functional Reaching Test

the simulations make us confident that the choice of these parameters is not critical. The simulations reported in the following were obtained with the following list of values: **KF**=700N/m; **KP**=2N; **A1**(ankle)=0.02rad/Nms; **A2**(knee)=0.01rad/Nms; **A3**(hip)=0.3rad/Nms; **A4**(shoulder)=0.1rad/Nms; **A5**(elbow)=0.07rad/Nms. In particular, as discussed in [11], the choice of the **A** parameters allows iCub to choose among equivalent solutions of the planned movement, as a consequence of the redundancy of the kinematic chain.

Figure 2 shows the initial and final posture of the FRT, which allowed iCub to reach forward at a distance of 47.39cm, with an increase of 18.35cm with respect to the initial posture. Incidentally, this value is greater than the threshold of 15cm which is considered clinically relevant in relation with the risk of falling.
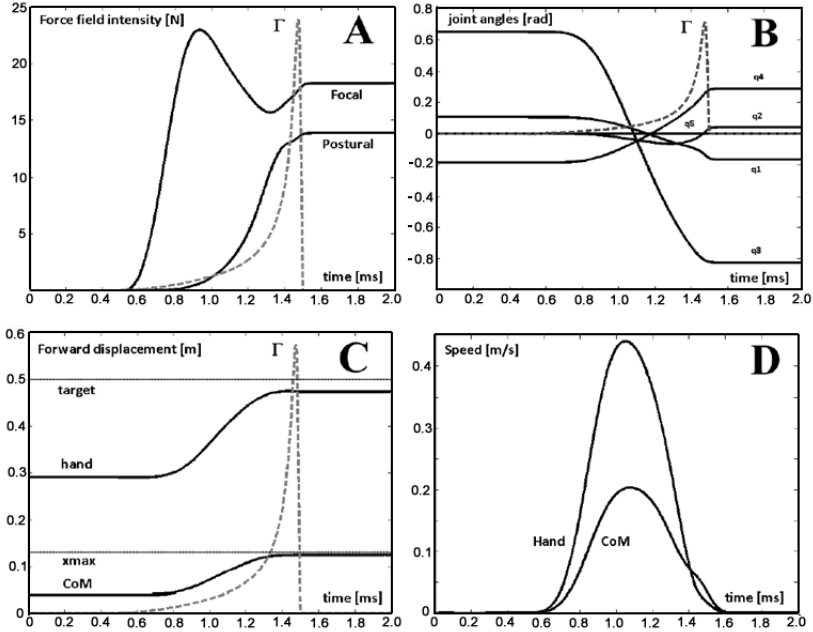
Fig. 3 *Panel A*: time course of the forces generated by the focal and the postural sub-networks, respectively. *Panel B*: Time course of the joint rotation angles, after subtracting the mean value: q1(ankle)=1.38 rad; q2(knee)=1.60 rad; q3(hip)=0.83 rad; q4(shoulder)=5.95 rad; q5(elbow)=0 rad. The angular values are absolute, referred to the horizontal line. *Panel C*: forward shift of the hand (Functional Reach), related to the forward position of the target (5 cm beyond the workspace) and forward shift of the CoM, related to the maximum stable position on the support base. *Panel D*: velocity profiles of the hand and the CoM. Panels A, B, C also display the time course of the Γ function.

Figure 3 shows the evolution of the different relevant variables. Panel A displays the intensities of the focal and postural force fields, respectively. Panel B shows the joint rotations patterns from the initial to the final posture, reached at the time of termination of the Γ function. Please note that some angles evolve monotonously from initial to termination time whereas other do not. In particular, the elbow joint angle remains equal to 0 throughout the whole movement for two reasons: 1) it was set to 0 initially in agreement with the FRT protocol and 2) it remained 0 because both force fields were directed horizontally (the focal field forward and the postural field backward, respectively). Panel C plots the forward displacements of the hand

and the CoM, respectively. The curves evolve monotonously, as should do, to the final shift values that must be compared with the final position of the target and the maximum admitted forward shift of the CoM, respectively. It turns out that the hand stops 5 cm before the target, because the latter is outside the workspace of the robot; the CoM stops just 6 mm before the fixed limit. Finally, panel D displays the speed profiles of the hand and the CoM respectively: they appear to be bell-shaped and synchronized, in agreement with the basic findings of the research in WBR [19, 20, 21]. Panels A, B, C also display the time course of the Γ function, emphasizing its role in the ordered coordination and synchronization of so many different variables.

4 Discussion

The proposed coordination model is not a controller of the standing posture but a mechanism of synergy formation, which operates on a 'mass-less body schema', allowing the redundant DoFs to be coordinated in a principled way during manipulation tasks that may affect the stability of the standing posture. As in human beings, we believe that also in humanoid robots it is convenient to separate coordination and control processes. The former one takes into account task requirements and affordances; the latter takes into account body dynamics and actuator constraints. Of course, the two processes are linked but we believe that the 'divide and conquer strategy' is the best one in the sense that provides solutions that can be scaled up naturally, achieving at the same time robustness and flexibility.

References

1. Metta, G., Sandini, G., Vernon, D., Natale, L., Nori, F.: The iCub humanoid robot: An open platform for research in embodied cognition. In: PerMIS: Performance Metrics for Intelligent Systems Workshop, Washington DC, USA, August 19-21 (2008)
2. Mohan, V., Morasso, P., Metta, G., Sandini, G.: A biomimetic, force-field based computational model for motion planning and bimanual coordination in humanoid robots. Auton. Robots 27, 291–307 (2009)
3. Loram, I.D., Lakie, M.: Direct measurement of human ankle stiffness during quiet standing: the intrinsic mechanical stiffness is insufficient for stability. J. Physiol. 545, 1041–1053 (2002)
4. Casadio, M., Morasso, P., Sanguineti, V.: Direct measurement of ankle stiffness during quiet standing: implications for control modelling and clinical application. Gait & Posture 21, 410–424 (2005)

5. Loram, I.D., Maganaris, C.N., Lakie, M.: Human postural sway results from frequent, ballistic bias impulses by soleus and gastrocnemius. J. Physiol. 564(1), 295–311 (2005)
6. Bottaro, A., Yasutake, Y., et al.: Bounded stability of the quiet standing posture: an intermittent control model. Hum. Mov. Sci. 27, 473–495 (2008)
7. Asai, Y., Tasaka, Y., et al.: Postural stabilization of quiet standing by means of PD feedback: and intermittent, switching control strategy is more robust than the linear, continuous-time strategy and better captures the spectral properties of human sway. PLoS ONE 4, e6169 (2009)
8. Maurer, C., Peterka, R.: A new interpretation of spontaneous sway measures based on a simple model of human postural control. J. Neurophysiol. 93, 189–200 (2005)
9. Mussa Ivaldi, F.A., Morasso, P., Zaccaria, R.: Kinematic Networks. a distributed model for representing and regularizing motor redundancy. Biol. Cybern. 60, 1–16 (1988)
10. Zak, M.: Terminal attractors for addressable memory in neural networks. Phys. Lett. A 133, 218–222 (1988)
11. Morasso, P., Casadio, M., Mohan, V., Zenzeri, J.: A neural mechanism of synergy formation for whole body reaching. Biol. Cybern. 102, 291–307 (2009)
12. Duncan, P.W., Weiner, D.K., Chandler, J., Studenski, S.: Functional reach: a new clinical measure of balance. J. Gerontol. 45, M192–M197 (1990)
13. Hollerbach, J.M., Suh, K.C.: Redundancy resolution of manipulators through torque optimization. IEEE J. Robotics and Automation 3(4), 308–316 (1987)
14. Chan, T.F., Dubey, R.V.: A weighted least-norm solution based scheme for avoiding joint limits for redundant joint manipulators. IEEE J. Robotics and Automation 11(2), 286–292 (1995)
15. Gienger, M., Janssen, H., Goerick, C.: Task-oriented whole body motion for humanoid robots. In: Proceedings of the IEEE-RAS/RSJ International Conference on Humanoid Robots (2005)
16. Rolf, M., Steil, J.J., Gienger, M.: Efficient exploration and learning of whole body kinematics. In: IEEE 8th Intl Conf. on Development and Learning (ICDL 2009), Shanghai, CH (2009)
17. Khatib, O., Sentis, L., Park, J., Warren, J.: Whole-body dynamic behavior and control of human-like robots. Intl J. of Humanoid Robotics 1, 29–43 (2004)
18. Khatib, O., Demircan, E., De Sapio, V., Sentis, L., Besier, T., Delp, S.: Robotics-based synthesis of human motion. Journal of Physiology - Paris 103, 211–219 (2009)
19. Stapley, P.J., Cheron, G., Grishin, A.: Does the coordination between posture and movement during human whole-body reaching ensure center of mass stabilization? Exp. Brain Res. 129, 134–146 (1999)
20. Pozzo, T., Stapley, P.J., Papaxanthis, C.: Coordination between equilibrium and hand trajectories during whole body pointing movements. Exp. Brain Res. 144, 343–350 (2002)
21. Kaminski, T.R.: The coupling between upper and lower extremity synergies during whole body reaching. Gait & Posture 26, 256–262 (2007)

A New Optimization Criterion Introducing the Muscle Stretch Velocity in the Muscular Redundancy Problem: A First Step into the Modeling of Spastic Muscle

F. Moissenet, D. Pradon, N. Lampire, R. Dumas, and L. Chèze

Abstract. Over the past few decades, musculo-skeletal modeling has been proposed as an in silico alternative to the invasive in vivo measurement of internal forces (e.g., musculo-tendon and joint reaction forces). However, even if great efforts have been made to improve the models, they remain partially validated and not adapted to pathologic subjects with orthopeadic and/or neurologic disorders. Indeed, even if a geometric scaling can be done using medical imaging techniques, the personalization of motor control specificities remains problematic. Consequently, when optimization techniques are used to solve the muscular redundancy problem, the selected criteria, that should reflect the motor control strategies, are not adapted to the gait disorders, such as muscle spasticity. The goal of this study was to introduce the muscle stretch velocity in the objective function, since muscle spasticity is linked to this parameter. We show that the maximization of the squared muscle stretch velocity provide more physiologic results during the stance phase and could be a way to introduce a spasticity criterion.

F. Moissenet
Laboratoire d'Analyse du Mouvement et de la Posture, CNRFR - Rehazenter,
1 rue André Vésale, L-2674 Luxembourg, Luxembourg
e-mail: `florent.moissenet@mailoo.org`

D. Pradon
Laboratoire d'Analyse de la Marche, Hopital Raymond Poincaré,
104 bd Raymond Poincaré, F-92380 Garches, France
e-mail: `didier.pradon@rpc.aphp.fr`

N. Lampire
Laboratoire d'Analyse Quantifiée de la Marche, ADAPT Loiret,
658D rue des Bourgoins, F-45200 Amilly, France
e-mail: `nicolas.lampire@gmail.com`

R. Dumas · L. Chèze
Laboratoire de Biomécanique et Mécanique des Chocs, UMR T9406,
Université de Lyon, Université Lyon 1 / IFSTTAR, Villeurbanne, France
e-mail: {`raphael.dumas,laurence.cheze`}`@univ-lyon1.fr`

1   Introduction

With over 10 million people affected worldwide and over 125.000 new cases per year in France, strokes are nowadays one of the most common cause of disability among adults. Indeed, 75% of survivors have permanent sequelae and often gait disorders such as muscle spasticity [29]. Spasticity is one of the consequences of the upper motor neuron syndrome associated to a full or partial lesion of the descending pathways (i.e., corticospinal tracts). It was defined by Lance [20] as "a motor disorder characterized by a velocity-dependant increase in tonic stretch reflexes (i.e., muscle tone) with exaggerated tendon jerks, resulting from hyperexcitability of the stretch reflex". This disorder can be reduced using treatments such as botulinum toxin but unfortunately, medical management remains a real difficulty and has few functional impacts [31]. To overcome this difficulty, it may be relevant to understand the gait organization of the patient and to identify the mechanisms that lead to it. Several factors may influence this organization such as energy consumption, environment (e.g., constraints, obstacles), stability and security, pain and history (e.g., skills, fatigue) [35].

A common way to explore the gait organization is to use musculo-skeletal modeling [2, 5, 6, 13, 15, 28, 30, 33, 34, 38]. However, even if great efforts have been made to improve the models, they only remain partially validated [25, 26] and not adapted to pathologic subjects with orthopeadic and/or neurologic disorders [17]. Indeed, even if a geometric scaling can be done using medical imaging techniques [1, 3, 14, 19, 32, 36], the personalization of motor control specificities remains problematic. Consequently, when optimization techniques are used to solve the muscular redundancy problem, the selected criteria, that should reflect the motor control strategies, are not adapted to the patient specificities, such as muscle spasticity.

The modeling of muscle spasticity can be reached by defining a new criterion in the optimization. By maximizing the muscle activation over a muscle stretch velocity threshold, it should be possible to model the specific motor control of a spastic muscle. Among the optimization techniques proposed in the literature to solve the muscular redundancy problem, one is based on a framework consisting of a forward dynamics assisted data tracking [16, 24, 27, 39] (i.e., the traditional approach). This method extends an initial optimization cost function (e.g., the minimization of the sum of the squared muscular activations [2, 37]) by introducing the weighted minimization of the difference between the sum of the individual muscular moments and the net joint torques computed from the inverse dynamics procedure. However, the fulfillment of this torque tracking belongs to the selected optimization weight and this can affect the kinematic parameters (e.g., segments and muscles position, velocity and acceleration) and lead to unphysiologic muscle stretch velocities. Consequently, the traditional approach can not be directly used to model muscle spasticity.

This study aims to extend this traditional approach by introducing the muscle stretch velocity parameter in the optimization process in order to keep a physiologic muscle stretch velocity pattern. This can be done by extending the cost function with the maximization of muscle stretch velocity (i.e., the new approach). The results in

terms of muscle fiber lengths and velocities are then compared with those obtained using the traditional approach and the reference data (i.e., the muscle fiber lengths and muscle stretch velocities computed from inverse kinematics, and the EMG signals).

2 Material and Method

Data of five gait cycles of a right hemiplegic patient (58yrs, 160cm, 69kg) are used in this study. Clinically, the patient presents a stiff knee gait (i.e., insufficient flexion of the knee during the swing phase) associated with spasticity and overactivity during the swing phase of the right Rectus Femoris (RF). Kinematics of the subject was collected using optoelectronic cameras (Motion Analysis, Santa Rosa, USA), sampled at 100Hz, and a typical set of cutaneous markers based on the Helen Hayes protocol [7]. Ground-reaction forces and moments were measured using forceplates (AMTI, Watertow, USA) sampled at 1000Hz. Finally, muscular activities were assessed using a 16 EMG channels system (Motion Lab Systems, Baton Rouge, USA).

Based on these data, a forward dynamics assisted data tracking [16, 24, 27, 39] was run using the software package SIMM (Musculographics, Santa Rosa, CA) to perform the gait simulation. This software package integrates a 3D lower limbs musculo-skeletal model, based on the works of Delp et al. [10], made of 13 segments, 14 degrees of freedom and 86 musculo-tendon units.

The forward dynamics assisted data tracking, implemented in SIMM, corresponds to a minimization problem with only inequality constraints through the scalar cost function J_1 (i.e., the traditional approach):

$$\min_{\mathbf{a}} J_1 = \frac{1}{2} \left(\mathbf{a}^T \mathbf{W}_1 \mathbf{a} + k_1 \mathbf{c}_{eq}^{\ T} \mathbf{c}_{eq} \right)$$
$$\text{subject to} : \mathbf{0} \le \mathbf{a} \le \mathbf{1} \tag{1}$$
$$\text{with} : \begin{cases} \mathbf{c}_{eq} = \mathbf{L}\mathbf{f} - \tau = \mathbf{0} \\ \mathbf{f} = h(\mathbf{a}, \mathbf{l}, \mathbf{v}) \end{cases}$$

where $\mathbf{a}$ and $\mathbf{f}$ are respectively the $j \times 1$ muscle activations vector and the $j \times 1$ musculo-tendon forces vector. τ is the $i \times 1$ net joint moments vector of the i degrees of freedom, computed by inverse dynamics. $\mathbf{L}$ is the $i \times j$ moment arms matrix of the j musculo-tendon units computed by using the musculo-skeletal model. $\mathbf{W}_1$ is a $j \times j$ diagonal weight matrix and k_1 is a scalar weight factor. $\mathbf{1}$ and $\mathbf{0}$ are, respectively, the identity matrix and the zero vector of appropriate dimensions. In order to perform the optimization in the activations space, SIMM integrates a Hill-type model [18, 40] that links the musculo-tendon forces $\mathbf{f}$ to the activations $\mathbf{a}$ through the function h. This function introduces the $j \times 1$ muscle fiber lengths vector $\mathbf{l}$ and the $j \times 1$ muscle stretch velocities vector $\mathbf{v}$. These parameters, $\mathbf{l}$ and $\mathbf{v}$, are updated during the convergence process. The integration scheme, at each time step, calculates joint

torques and joint kinematics using the activation and kinematic states of the previous time step [12].

In order to introduce the muscle stretch velocity in the optimization process, the traditional approach (Eq. 1) is extended as follow through the scalar cost function J_2 (i.e., the new approach):

$$\min_{\mathbf{a}} J_2 = \frac{1}{2} \left(\mathbf{a}^T \mathbf{W}_1 \mathbf{a} + k_1 \mathbf{c}_{eq}^T \mathbf{c}_{eq} + k_2 \begin{bmatrix} 1 \\ \tilde{\mathbf{v}} \end{bmatrix}^T \begin{bmatrix} 1 \\ \tilde{\mathbf{v}} \end{bmatrix} \right)$$

$$\text{subject to}: \mathbf{0} \leq \mathbf{a} \leq \mathbf{1}$$

$$\text{with}: \begin{cases} \mathbf{c}_{eq} = \mathbf{L}\mathbf{f} - \tau = \mathbf{0} \\ \mathbf{f} = h(\mathbf{a}, \mathbf{l}, \mathbf{v}) \end{cases} \tag{2}$$

where $\tilde{\mathbf{v}}$ is a $k \times 1$ vector composed of a selection of muscle stretch velocities that we want to introduce in the optimization process, and k_2 is a scalar weight factor.

The new approach is applied on the right Rectus Femoris stretch velocity and its potential is estimated by comparing $\hat{\mathbf{l}}$ and $\hat{\mathbf{v}}$, respectively the muscle fiber lengths and muscle stretch velocities obtained after optimization, with those obtained using the traditional approach and the reference data (i.e., computed from inverse kinematics). Predicted musculo-tendon forces are also studied and compared with EMG signals in order to evaluate the impact of the approach on this output. EMG signals were processed following the recommendations of De Luca [9, 8] and the amplitude of EMG signal envelops was adjusted to the amplitude of the predicted musculo-tendon forces to compare the pattern of these data. Weight matrices and factors were arbitrary set to $\mathbf{W}_1 = \mathbf{W}_2 = \mathbf{1}$, $k_1 = 10^2$ and $k_2 = 10^5$.

Table 1 Root mean square error (i.e., RMSE) and correlation (i.e., R2 coefficient) between both the fiber lengths and stretch velocities, obtained when using the traditional approach and the new approach, and the reference data

Muscle	Approach	Coefficient	Fiber length		Stretch velocity	
			Stance	Swing	Stance	Swing
RF	Traditional	RMSE (m, m/s)	0.0057	0.0052	0.0416	0.0302
		R2	0.9189	0.5436	0.6460	0.7173
	New	RMSE (m, m/s)	0.0081	0.0108	0.0087	0.0615
		R2	0.9909	0.2266	0.9431	0.1311
ST	Traditional	RMSE (m, m/s)	0.0042	0.0040	0.0102	0.0100
		R2	0.9976	0.9877	0.8171	0.8582
	New	RMSE (m, m/s)	0.0044	0.0039	0.0030	0.0112
		R2	0.9999	0.9827	0.9809	0.8929
VL	Traditional	RMSE (m, m/s)	0.0016	0.0017	0.0026	0.0022
		R2	0.9968	0.9883	0.9606	0.9757
	New	RMSE (m, m/s)	0.0016	0.0035	0.0025	0.0224
		R2	0.9965	0.4560	0.9651	0.3843
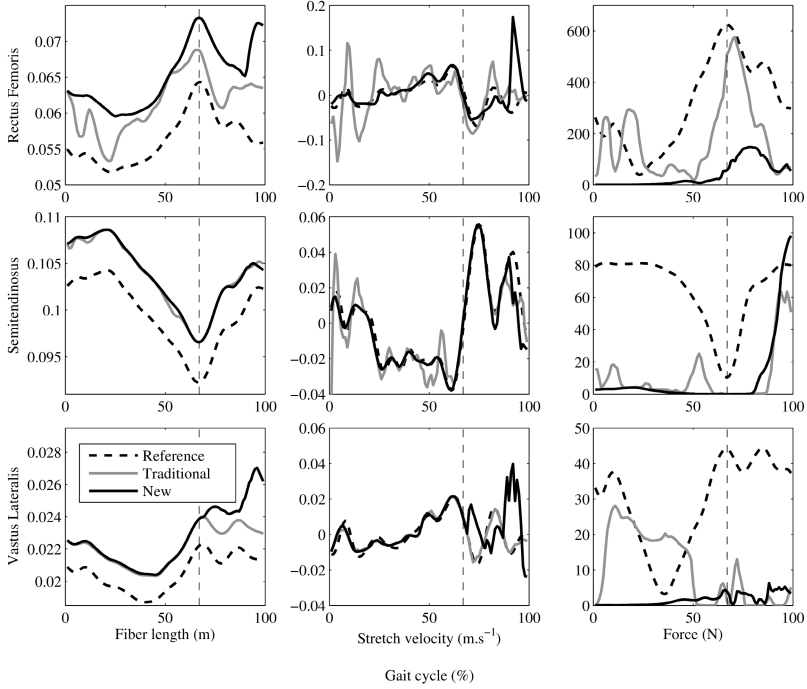
Fig. 1 Fiber length, stretch velocity and force of Rectus Femoris, Semitendinosus and Vastus Lateralis obtained using the traditional approach and the new approach, compared with reference data

3 Results

The mean fiber length, stretch velocity and force (Fig. 1) of two biarticular muscles, Rectus Femoris (RF) and Semitendinosus (ST), and one monoarticular muscle, Vastus Lateralis (VL), over the different recorded gait cycles, are presented here. Table 1 gives the root mean square error (i.e., RMSE) and the correlation (i.e., R2 coefficient) between both the fiber lengths and stretch velocities, obtained when using the traditional approach or the new approach, and the reference data.

Regarding fiber lengths, the new approach gives better results (Tab. 1: RF-R2) or similar results (Tab. 1: ST-R2, VL-R2) in terms of pattern than the traditional approach during stance phase. However, during swing phase, the new approach can bring to a high increase of muscle fiber length (Fig. 1: Rectus Femoris and Vastus Lateralis). For both the traditional and the new approaches, a positive offset appears for all muscle fiber lengths (Fig. 1). Regarding the stretch velocities, the

new approach gives better results (Tab. 1: RF-R2, ST-R2) or similar (Tab. 1: VL-R2) in terms of pattern than the traditional approach during stance phase. However, during the swing phase, the new approach gives similar results (Tab. 1: ST-R2) or worse results (Tab. 1: RF-R2, VL-R2) than the traditional approach. Finally, regarding musculo-tendon forces, the traditional approach reproduces well the EMG peaks during the initial stance and at toe off for Rectus Femoris (Fig. 1). For semitendinosus, only the EMG peak at the end of the swing phase is reproduced (Fig. 1). Finally, Vastus Medialis force presents a different pattern than the EMG signal with a midstance activity (Fig. 1). On the whole, the new approach modifies the amplitude of the forces (e.g., Rectus Femoris and Vastus Medialis forces are decreased, Semitendinosus force is increased), simplifies the force pattern, that includes less peaks and greatly reduces muscle activity during stance phase.

4 Discussion

Musculo-skeletal modeling has been proposed as an in silico alternative to the invasive in vivo measurement of internal forces (e.g., musculo-tendon and joint reaction forces) [2, 5, 6, 13, 15, 28, 30, 33, 34, 38]. However, even if great efforts have been made to improve the models, they only remain partially validated (i.e., qualitative comparison of the activation patterns with EMG signals, quantitative comparison of the joint reaction forces with data of instrumented prosthesis [4, 14, 23, 34]) and not adapted to pathologic subjects with orthopeadic and/or neurologic disorders. In order to define patient-specific models, it is necessary to adapt geometric parameters [21], kinematic parameters [11, 15, 22] and kinetic parameters but also, when optimization techniques are used to solve the muscular redundancy problem, the selected criteria, that aim to reflect the motor control strategies.

In this study, a new optimization criterion has been proposed by extending a traditional forward dynamics assisted data tracking approach [16, 24, 27, 39]. It has been chosen to introduce the muscle stretch velocity parameter in the cost function in order to avoid unphysiologic results in terms of muscular kinematics (i.e., muscle fiber length and stretch velocity). This new approach has been compared to the traditional one and the results obtained after the inverse kinematics procedure. The simulation has been performed under the software package SIMM, but could be generalized to any other software package using the same type of optimization procedure (i.e., forward dynamics assisted data tracking approach).

Regarding the results, the traditional approach fails to produce physiologic muscle fiber length and stretch velocities. Indeed, these parameters present high and fast unphysiologic variations during both stance and swing phases. Moreover, maximum muscle stretch velocity peaks of the Rectus Femoris (and so potentially spastic contractions) are obtained during the initial stance phase. However, clinically, spastic contractions of the Rectus Femoris often appear during the pre-swing and

initial swing phase, limiting the second knee flexion peak. These troubles mainly appear for biarticular muscles (i.e., RF and ST in this study), since monoarticular muscles (i.e., VL in this study) seem not to be affected. Accordingly, the traditional approach seems not well adapted to the study of spastic muscles. The new approach produces more physiologic muscle fiber length and stretch velocity patterns for both monoarticular and biarticular muscles during stance phase (e.g., the maximum muscle stretch velocity peak of the Rectus Femoris is obtained during pre-swing). However, some offsets appear during the computation of muscle fiber lengths compared with the reference data. These offsets, also present when using the traditional approach, affect muscle-tendon length, muscle moment-arm and therefore musculo-tendon force. They can be linked to the computation (i.e., integration) of the kinematics at each time step of the convergence process, based on the optimized activation and kinematic states of the previous time step. Since the equality constraints fulfillment belongs to the selected optimization weight, the kinematic parameters (e.g., segments and muscles position, velocity and acceleration) can be affected. Moreover, the new approach brings high unphysiologic variations of muscle fiber lengths and stretch velocities for all muscles during the swing phase. Again, this can be linked to the introduction of the weighted minimization of the equality constraints error, since the joint torques are closed to zero during the swing phase.

Several limitations need to be addressed regarding the present study. First, the reference data, in terms of muscular kinematics (i.e., muscle fiber length and stretch velocity), are directly computed from inverse kinematics. To do this, SIMM computes the muscle fiber length by finding the static equilibrium force in the musculo-tendon unit. This can be done using the force-length relationships of the Hill model [12]. However, since the muscle activation is basically considered at 0 when only computing inverse kinematics, only the passive component of the musculo-tendon unit is taken into account. Thus, the reference data are purely kinematic data and can not reflect the consequences of the muscle active component activity. This can be problematic when investigating pathologic muscles, such as spastic muscles, where muscle activations can have a high influence on the muscular kinematics. Second, the optimization criterion proposed here, defined as the maximization of the muscle stretch velocity, does not have a physiologic sense. This criterion, and its weight factors, were defined arbitrary for this study. It would be necessary to investigate if the motor control rules integrate or not such a mechanism.

To conclude, this study shows that the use of a traditional forward dynamics assisted data tracking approach can bring unphysiologic results in terms of muscular kinematics (i.e., muscle fiber length and stretch velocity). In order to plan the modeling of muscle spasticity, the new approach proposed here can be used to get more physiologic results during the stance phase, especially in terms of muscle stretch velocity.

References

1. Albracht, K., Arampatzis, A., Baltzopoulos, V.: Assessment of muscle volume and physiological cross-sectional area of the human triceps surae muscle in vivo. Journal of Biomechanics 41(10), 2211–2218 (2008)
2. Anderson, F., Pandy, M.: Static and dynamic optimization solutions for gait are practically equivalent. Journal of Biomechanics 34, 153–161 (2001)
3. Blemker, S., Asakawa, D., Gold, G., Delp, S.: Image-based musculoskeletal modeling: applications, advances, and future opportunities. Journal of Magnetic Resonance Imaging 25(2), 441–451 (2007)
4. Brand, R., Pedersen, D., Davy, D., Kotzar, G., Heiple, K., Goldberg, V.: Comparison of hip force calculations and measurements in the same patient. The Journal of Arthroplasty 9(1), 45–51 (1994)
5. Cleather, D., Bull, A.: An optimization-based simultaneous approach to the determination of muscular, ligamentous, and joint contact forces provides insight into musculoligamentous interaction. Annals of Biomedical Engineering 39(7), 1925–1934 (2011)
6. Crowninshield, R., Brand, R.: A physiologically based criterion of muscle force prediction in locomotion. Journal of Biomechanics 14, 793–801 (1981)
7. Davis, R.I., Ounpuu, S., Tyburski, D., Gage, J.: A gait data collection and reduction technique. Human Movement Sciences 10, 575–587 (1991)
8. De Luca, C., Gilmore, L., Kuznetsov, M., Roy, S.: Filtering the surface emg signal: Movement artifact and baseline noise contamination. Journal of Biomechanics 43(8), 1573–1579 (2010)
9. De Luca, S.: The use of surface electromyography in biomechanics. Journal of Applied Biomechanics 13, 135–163 (1997)
10. Delp, S., Loan, J., Hoy, M., Zajac, F., Topp, E., Rosen, J.: An interactive graphics-based model of the lower extremity to study orthopaedic surgical procedures. IEEE Transactions on Biomedical Engineering 37, 757–767 (1990)
11. Dumas, R., Moissenet, F., Gasparutto, X., Cheze, L.: Influence of the joint models on the lower limb musculo-tendon and contact forces during gait. Proceedings of the Institution of Mechanical Engineers, Part H, Journal of Engineering in Medicine 226(2), 146–160 (2012)
12. Erdemir, A., McLean, S., Herzog, W., van den Bogert, A.: Model-based estimation of muscle forces exerted during movements. Clinical Biomechanics 22, 131–154 (2007)
13. Fraysse, F., Dumas, R., Cheze, L., Wang, X.: Comparison of global and joint-to-joint methods for estimating the hip joint load and the muscle forces during walking. Journal of Biomechanics 42, 2357–2362 (2009)
14. Fregly, B., Besier, T., Lloyd, D., Delp, S., Banks, S., Pandy, M., D'Lima, D.: Grand challenge competition to predict in vivo knee loads. Journal of Orthopaedic Research 30, 503–513 (2012)
15. Glitsch, U., Baumann, W.: The three-dimensional determination of internal loads in the lower extremity. Journal of Biomechanics 30(11/12), 1123–1131 (1997)
16. Happee, R.: Inverse dynamic optimization including muscular dynamics, a new simulation method applied to goal directed movements. Journal of Biomechanics 27, 953–960 (1994)
17. Hicks, J., Schwartz, M., Delp, S.: Modeling and simulation of walking. In: Diagnosis and Management of Movement Abnormalities in Cerebral Palsy. Cambridge Press (2009)
18. Hill, A.: The heat of shortening and the dynamic constraints of muscle. Proceedings of the Royal Society 126, 136–195 (1938)

19. Jolivet, E., Laredo, J., Skalli, W.: Patient-specific finite element model of the hip including soft tissues. In: 16th Congress of the European Society of Biomechanics (2008)
20. Lance, J.: The control of muscle tone, reflexes, and movement: Robert wartenberg lecture. Neurology 30(12), 1303–1313 (1980)
21. Lenaerts, G., De Groote, F., Demeulenaere, B., Mulier, M., Van der Perre, G., Spaepen, A., Jonkers, I.: Subject-specific hip geometry affects predicted hip joint contact forces during gait. Journal of Biomechanics 41(6), 1243–1252 (2008)
22. Li, G., Kawamura, K., Barrance, P., Chao, E., Kaufman, K.: Prediction of muscle recruitment and its effect on joint reaction forces during knee exercises. Annual Review of Biomedical Engineering 26, 725–733 (1998)
23. Lu, T., O'Connor, J., Taylor, S., Walker, P.: Validation of a lower limb model with in vivo femoral forces telemetered from two subjects. Journal of Biomechanics 31, 63–69 (1998)
24. McLean, S., Su, A., van den Bogert, A.: Development and validation of a 3-d model to predict knee joint loading during dynamic movement. Journal of Biomechanical Engineering 125, 864–874 (2003)
25. Modenese, L., Phillips, A., Bull, A.: An open source lower limb model: Hip joint validation. Journal of Biomechanics 44, 2185–2193 (2011)
26. Moissenet, F., Chèze, L., Dumas, R.: Anatomical kinematic constraints: consequences on musculo-tendon forces and joint reactions. Multibody System Dynamics 28, 125–141 (2012)
27. Neptune, R.: Optimization algorithm performance in determining optimal controls in human movement analyses. Journal of Biomechanical Engineering 121, 249–252 (1999)
28. Pandy, M., Andriacchi, T.: Muscle and joint function in human locomotion. Annual Review of Biomedical Engineering 12, 401–433 (2010)
29. Pellerin, C., Mauget, Y., Bouju, A., Rouanet, F., Petitjean, M., Dabadie, P.: Accident vasculaire cérébral. Médecine d'Urgence, 107–117 (2003)
30. Pierrynowski, M., Morrison, J.: A physiological model for the evaluation of muscular forces in human locomotion: Theorical aspects. Mathematical Biosciences 75, 69–101 (1985)
31. Rousseaux, M., Compère, S., Launay, M., Kozlowski, O.: Variability and predictability of functional efficacy of botulinum toxin injection in leg spastic muscles. Journal of Neurological Sciences 232, 51–57 (2005)
32. Scheys, L., Loeckx, D., Spaepen, A., Suetens, P., Jonkers, I.: Atlas-based non-rigid image registration to automatically define line-of-action muscle models: a validation study. Journal of Biomechanics 42(5), 565–572 (2009)
33. Seireg, A., Arvikar, R.: The prediction of muscular load sharing and joint forces in the lower extremities during walking. Journal of Biomechanics 8(2), 89–102 (1975)
34. Stansfield, B., Nicol, A., Paul, J., Kelly, I., Graichen, F., Bergmann, G.: Direct comparison of calculated hip joint contact forces with those measured using instrumented implants. an evaluation of a three-dimensional mathematical model of the lower limb. Journal of Biomechanics 36(7), 929–936 (2003)
35. States, R., Pappas, E., Salem, Y.: Overground physical therapy gait training for chronic stroke patients with mobility deficits. Cochrane Database Systematic Reviews CD006075(3) (2009)
36. Taddei, F., Martelli, S., Valente, G., Leardini, A., Benedetti, M., Manfrini, M., Viceconti, M.: Femoral loads during gait in a patient with massive skeletal reconstruction. Clinical Biomechanics 27, 273–280 (2012)

37. Thelen, D., Anderson, F., Delp, S.: Generating dynamic simulations of movement using computed muscle control. Journal of Biomechanics 36, 321–328 (2003)
38. Wehner, T., Claes, L., Simon, U.: Internal loads in the human tibia during gait. Clinical Biomechanics 24(3), 299–302 (2009)
39. Yamaguchi, G., Zajac, F.: Restoring unassisted natural gait to paraplegics via functional neuromuscular stimulation: a computer simulation study. IEEE Transactions on Biomedical Engineering 37(9), 886–902 (1990)
40. Zajac, F.: Muscle and tendon: properties, models, scaling, and application to biomechanics and motor control. Critical Reviews in Biomedical Engineering 17(4), 359–411 (1989)

Forward and Inverse Optimal Control of Bipedal Running

Katja Mombaur, Anne-Hélène Olivier, and Armel Crétual

Abstract. This paper discusses forward and inverse optimal control problems for bipedal human-like running, with a focus on inverse optimal control. The (forward) optimal control problem looks for the optimal solution for a problem formulation, i.e. given objective function and given dynamic constraints. The inverse optimal control problem is more challenging and consists in determining the objective function and potentially unknown parts in the dynamic model that best reproduce a solution that is known from measurements. Periodic running motions are modeled as hybrid dynamic models with multiple phases and discontinuities, based on a three-dimensional multibody system model with 25 degrees of freedom. We investigate a recorded running motion on a treadmill at 10 km/h running speed and identify the best possible objective function based on some hypotheses for potential contributions to this objective function. For this, we apply a previously developed inverse optimal control technique which uses a combination of a direct multiple shooting method and a derivative-free optimization technique, and we demonstrate here that it also works for problems of the given complexity.

Katja Mombaur
Interdisciplinary Center for Scientific Computing (IWR), Heidelberg University, INF 368, 69120 Heidelberg, Germany, Associate Researcher at LAAS-CNRS, Toulouse
e-mail: kmombaur@uni-hd.de, kmombaur@laas.fr

Anne-Hélène Olivier
INRIA Rennes, Campus universitaire de Beaulieu, 35042 Rennes, France
e-mail: anne-helene.olivier@inria.fr

Armel Crétual
M2S lab, Université de Rennes 2, Campus Ker Lann,
Avenue Robert Schumann 35170 Bruz, France
e-mail: armel.cretual@univ-rennes2.fr

1   Introduction

It is a common assumption that motions of humans and animals are optimal due to evolution, learning and training [1]. As a logical consequence, from a mathematical perspective dynamic motion tasks such as walking and running can be formulated as optimal control problems. In this paper we discuss two ways in which optimal control can be used to explore human locomotion: (a) in the classical forward optimal control sense to generate natural motions without any help by motion capture data; and (b) in the inverse optimal control sense, to identify objectives of human motion from experiments (compare fig. 1). We show in particular the application of these optimal control and inverse optimal control approaches to three-dimensional human running motions.
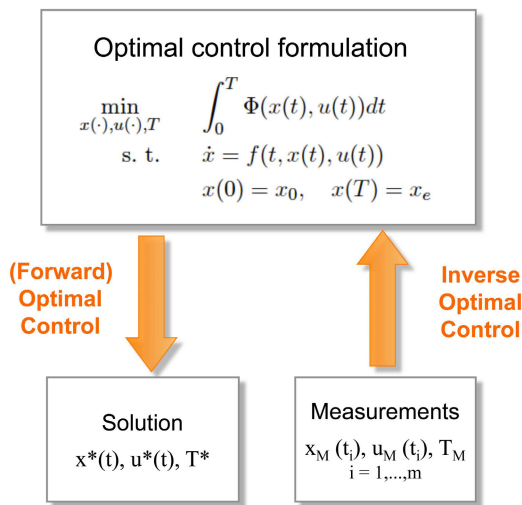
Fig. 1 Two different types of problems: The classical (forward) optimal control problem consists in finding the optimal solution for a given objective function and given dynamic constraints. The inverse optimal control problem consists in determining the objective function (and potentially unknown parts in the dynamic model) that best reproduce a solution that is (partly) known from measurements.

Human and human-like running motions are investigated in various fields of research such as humanoid robotics, computer graphics, biomechanics, sports science or orthopedics. In robotics, achieving real running motions on a humanoid robot system is a benchmark that various companies and research labs have tried to achieve [2, 3]. In computer graphics, the goal is to generate realistic human-like running motions on an avatar (e.g. [4]). For both tasks, forward optimal control can be efficiently applied to generate feasible and realistic running motions. We have done this in previous research, aiming e.g. at the generation of natural looking running motions [5], or of the most stable running in the open-loop sense of an anthropomorphic system [6].

When looking at biological running systems, the inverse question becomes more interesting. In biomechanics, orthopedics or sports sciences the goal is to fully understand mechanics and control of normal and pathological gaits in humans, and to use this knowledge to improve motions after an accident or illness in the orthopedic case, or to increase performance to the highest possible level in sports. In this case the question for the optimization criterion underlying each of these situations becomes interesting. Recently, Miller et al. [7] have investigated different objective functions for running motions based on a 2D model with muscles and found that a minimization of muscle activations squared comes closer to a measured solution than a minimization of the cost of transport. Ackermann and Bogert [8] studied the same objective function for walking. However, none of these authors pursued the study further to look more precisely for the objective function or combination of objective functions underlying the measurements which would be solving an inverse optimal control problem.

In this paper, we investigate running motions on a treadmill at a slow pace (10 km/h). We show how a previously developed method for inverse optimal control [9] can be applied to identify objective functions underlying this type of motions.

The paper is organized as follows: Section 2 describes the multibody system model of three-dimensional human running used in this study. Section 3 outlines the formulation and solution of (forward) optimal control problems for the generation of optimal running motions. Section 4 discusses the formulation and solution of the inverse optimal control problem and gives results for a particular running example. Section 5 gives some conclusions and outlines future research directions.

2   Modeling Human Running Motions

To describe the human body we use a multibody system model in three dimensions with 12 bodies and 25 degrees of freedom (DOF) in flight - 6 global DOF associated with the positon and orientation of the pelvis and 19 internal DOF related to internal joint angles. The system is equipped with torque actuators for each of the 19 internal DOF describing the action of the human muscles. For geometry and inertia parameters, we use the anthropometric data given by de Leva [11]. The model describes human-like forefoot running, i.e. there is no flat foot ground contact but only point-like contact with the ball of foot. The higher the running speed is, the more this assumption is true. At a speed of 10 Km/h, many runners intuitively roll over the foot from heel to toe, instead of using a ball contact. Since modeling such a complex contact with a variable foot shape is still an issue of discussion in biomechanics [10], we have chosen to use the simplification of a pointlike contact for the time being.

Detailed muscle dynamics are not included in the model yet, but we plan to do so at a later stage of this research. For more details on the current model, see the 3D model in [5]. With respect to the model described in this paper, we have removed the spring-damper elements for the present study.

Running motions consist of a sequence of alternating flight phases and single-leg contact phases. The running motion considered in this study is assumed to be periodic and symmetric, i.e. right and left steps are assumed to be identical. Each phase of the motion (flight phase and single-leg contact phase) is described by its own set of ordinary differential or differential-algebraic equations. Between phases, there may be discontinuities in the velocities, e.g. at touchdown of the foot on the floor which are assumed to be fully inelastic. The running model therefore takes the form of a periodic hybrid dynamical system. The above assumptions allow us to reduce the model of the periodic running motion to the model of a single step with a subsequent leg shift and periodicity constraints (see figure 2). The total time of the step T as well as the individual phase times are free variables of the model.

We describe the motion during flight phase by a set of ordinary differential equations of the following form:

$$M(q,p)\ddot{q} + N(q,\dot{q},p)\dot{q} = F(q,\dot{q},p,\mathscr{M}), \tag{1}$$

with mass matrix M and the vector N combining all nonlinear effects. F is the vector of all external forces (such as gravity, muscle torques $\mathscr{M}$, drag etc.) We have used the automatic model generator of HuMAnS by Wieber [12] to generate the terms M and N of the 3D running model (see [5] for details).
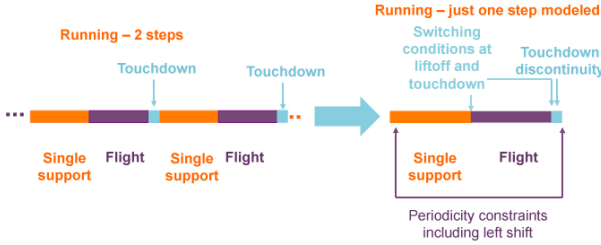
Fig. 2 The model of an (infinite) sequence of periodic running steps can be reduced to the model of one step with leg shift and periodicity constraints

During single leg contact phase, we keep the same number of coordinates even though the number of DOF is reduced by the fact that one ball of foot is fixed to the ground by a constraint of the form $g(q) = 0$. This results in redundant coordinates and a system of differential algebraic equations (DAE) for the equations of motion:

$$\dot{q} = v \tag{2}$$

$$\dot{v} = a \tag{3}$$

$$\begin{pmatrix} M & G^T \\ G & 0 \end{pmatrix} \begin{pmatrix} a \\ \lambda \end{pmatrix} = \begin{pmatrix} -N+F \\ \gamma \end{pmatrix} \tag{4}$$

$$g_{pos} = g(q(t),p) = 0 \tag{5}$$

$$g_{vel} = G(q(t),p) \cdot \dot{q}(t) = 0. \tag{6}$$

with acceleration $a = \ddot{q}$ and Lagrange multipliers λ. G denotes the Jacobian of the position constraints $G = (\partial g/\partial q)$, and γ the corresponding Hessian $\gamma = ((\partial G/\partial q)\dot{q})\dot{q}$. Eqns. (5) and (6) describe the invariant manifolds on position and velocity level that the solution must satsify. In the optimization, we take into account the unilateral nature of the ground contact constraint (i.e. the ground can not pull but only push against the foot) by formulating an inequality constraint on the Lagrange multiplier associated with the normal contact force. Phase changes between flight phase and contact phase are not explicitly time dependent but state dependent:

$$s(q(\tau_s), v(\tau_s), p) = 0. \tag{7}$$

Touch-down occurs when the foot gets down to the height of the ground, and lift-off takes place when the vertical contact force (represented by the negative of the respective Lagrange multiplier in eqn. (4)) becomes zero.

The discontinuities of the velocities at touchdown (resulting from the fact that the velocity of the foot contact point is instantly set to zero at inelastic contact and that this shock wave propagates through the whole body) can be computed as:

$$\begin{pmatrix} M & G^T \\ G & 0 \end{pmatrix} \begin{pmatrix} v_+ \\ \Lambda \end{pmatrix} = \begin{pmatrix} Mv_- \\ 0 \end{pmatrix} \tag{8}$$

where v_- and v_+ are the velocities immediately before and after impact, respectively. Matrices M and N are the same as in eqn. (4) above.

Periodicity constraints are imposed in the model on all velocity variables v and a reduced set of position variables q_{red} eliminating the coordinate describing the forward running direction of the robot, after formulating the leg shift (compare fig. 2).

3   Generation of Natural Human Running by Means of Forward Optimal Control

In humanoid robotics or computer graphics one often faces the task to generate a natural motion for a system with given geometry and inertia characteristics. To achieve this task without any information at all from motion capture or any other human measurements, optimal control techniques can be efficiently used. Optimization has a clear advantage over approaches based on pure simulation. Simulation always requires to fix important quantities in advance: if it is performed on a forward dynamics model the input forces and torques have to be pre-specified to obtain the resulting motion, and if performed on an inverse dynamics model the position and velocity histories have to be fixed to be able to calculate the required driving torques and forces. However, typically none of the quantities is exactly known a priori. Optimization-based simulation allows to leave forces, torques and the motion free and to determine them all simultaneously according to some desired optimization criterion.

In order to generate optimal running motions, the following optimal control problem must be solved:

$$\min_{x(\cdot),u(\cdot),\tau} \sum_{j=1}^{n_{ph}} \left(\int_{\tau_{j-1}}^{\tau_j} \phi_j(x(t),u(t))\,dt + \Phi_j(\tau_j,x(\tau_j)) \right) \tag{9}$$

$$\text{s. t.} \quad \dot{x}(t) = f_j(t,x(t),u(t)) \quad \text{for} \quad t \in [\tau_{j-1},\tau_j],$$
$$j = 1,...,n_{ph}, \; \tau_0 = 0, \tau_{n_{ph}} = T \tag{10}$$
$$x(\tau_j^+) = x(\tau_j^-) + J(\tau_j^-) \quad \text{for} \quad j = 1,...,n_{ph} \tag{11}$$
$$g_j(t,x(t),u(t)) \geq 0 \quad \text{for} \quad t \in [\tau_{j-1},\tau_j] \tag{12}$$
$$r_{eq}(x(0),..,x(T)) = 0 \tag{13}$$
$$r_{ineq}(x(0),..,x(T)) \geq 0 \tag{14}$$

Eqn. (9) describes the objecive function to be minimized, where the first part $\int \phi_j dt$ denotes integral objective functions of Lagrange type, and the second part Φ_j Mayer type objective functions depending only on the end values. $x(t)$ denotes the vector of state variables, summarizing position and velocity variables of the runner, and $u(t)$ is the vector of control variables of the system, in this case the muscle torques $\mathcal{M}_i$. τ is the vector of phase switching times with total step time $T = \tau_{n_{ph}}$.

Eqn. (9) describes the objective function in a general form and is further discussed below. Eqns. (10) and (11) are placeholders for the hybrid dynamic model of the running motion discussed in the previous section. In addition, there are continuous inequality constraints of form (12), including lower and upper bounds on all variables, but also more complex relations between several variables, and coupled and decoupled pointwise equality (13) and inequality constraints (14), such as start and end point constraints, phase switching conditions or periodicity constraints.

For the solution of these multi-phase optimal control problems we use the powerful optimal control code MUSCOD which has been developed at IWR Heidelberg [13, 14]. This code can be applied to mechanical DAEs of the above form, as
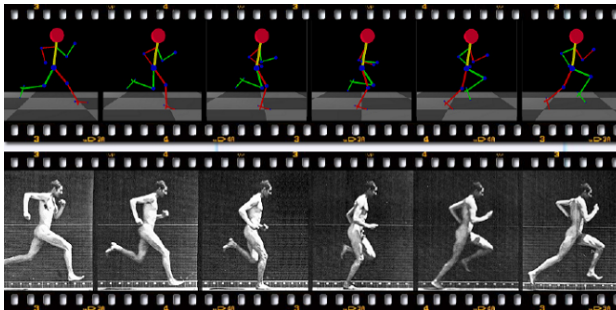
Fig. 3 Qualitative comparison of optimized running motion and a photographed sequence by Muybridge in the late 19th century[19]

described in [15]. MUSCOD uses a direct method (also called first-discretize-then-optimize approach) for control discretization, and the multiple shooting technique for state parameterization. The result of these two discretization steps for which identical grids are chosen is a nonlinear programming problem (NLP) of large dimension which is solved by a specially tailored sequential quadratic programming (SQP) method.

Our experience has shown that objective functions $\phi_j(t, x(t), u(t), p)$ minimizing a sum of weighted torques squared, and possible additionally minimizing the variation of torques, lead to very natural running motions [5], as a qualitative comparison with human motion shows (see figure 3). Since the focus of this paper is the inverse optimal control problem for running motions, we will not discuss this forward optimal control problem further, and refer to [5] for details instead.

4 Identification of Human Objectives during Locomotion by Means of Inverse Optimal Control

This investigation goes beyond the qualitative reasoning of the previous section, since for a thorough biomechanical analysis one is interested in the true objective function applied by the human. The specific optimization criterion for locomotion in different situations is generally unknown, and it can be expected that in most cases a combination of multiple criteria is used. The optimal behavior during a particular locomotion task can be observed and measured by motion capture, force plate or EMG measurements etc. The inverse optimal control problem consists in determining, from a solution that is (partly) known from measurements, the optimization criterion that has produced this solution.

4.1 Formulation and Solution of Inverse Optimal Control Problem

To solve the inverse optimal control problem, we make the assumption that a set of reasonable independent base functions $\Psi_i(t)$ for the objective function can be established. The relative contribution of these base functions $\Psi_i(t)$ expressed by a weight factor α_i remains to be determined by the algorithm. The inverse optimal control problem can be formulated as:

$$\min_{\alpha} \ \sum_{j=1}^{m} ||z^*(t_j; \alpha) - z_M(t_j)||^2 \tag{15}$$
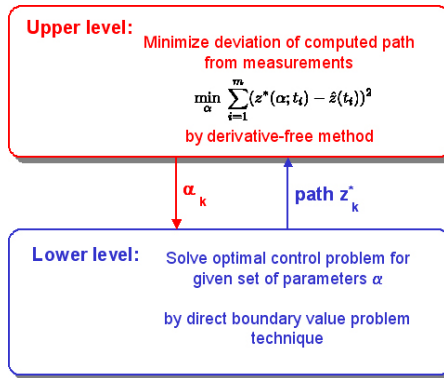
where $z^*(t; \alpha)$ is determined by the solution of

$$\min_{x,u,T} \ \int_0^T \left[\sum_{i=1}^{n} \alpha_i \Psi_i(x(t), u(t)) \right] dt \tag{16}$$

s. t. constraint eqns. (10) - (14)

The vector z represents the observation vector of states and possibly control variables with $z = h(x, u)$, h being the observation function. z can be either a subset of the state and control variables or other quantities that can be directly or indirectly measured and determined from the computed solution. For a more detailed discussion, see below.

Inverse optimal control problems are difficult since they require the solution of an optimal control problem and a parameter identification problem at the same time. We have developed a bilevel optimization technique (compare fig. 4) capable to solve this type of problems based on two powerful existing techniques. The upper level in this method is responsible for the iterations over the unknown weight factors α such that the fit between measurements and optimal control problem solution is improved. In each upper level function evaluation a forward optimal control problem has to be solved for the current set of α_i. We apply a derivative-free optimization technique (i.e. it only requires function evaluations and no gradient information) which is always favorable if function evaluations are expensive and noisy and derivative information can therefore not be generated reliably. We use the derivative-free optimization code BOBYQA (Bound Optimization BY Quadratic Approximation) by Powell [16] which is also capable to handle bounds on the weight parameters.

Upper level: Minimize deviation of computed path from measurements

$$\min_{\alpha} \sum_{i=1}^{m} (z^*(\alpha; t_i) - \hat{z}(t_i))^2$$

by derivative-free method

α_k path z^*_k

Lower level: Solve optimal control problem for given set of parameters α

by direct boundary value problem technique

Fig. 4 Solution of inverse optimal control problem as bilevel optimization problem

In the lower level, the task is to efficiently solve the forward optimal control problem which arrises in each iteration of the upper level. For this we apply the direct boundary value problem approach MUSCOD that was already described in the previous section.

This bilevel inverse optimal control method has already been applied to identify the cost functions that produce locomotion paths in point to point motions in free space [9] as well as in interaction scenarios. The purpose of this paper is to show that it can equally be applied to solve the multi-phase problems based on complex multibody system models that we are facing in the present study.

4.2   Objective Functions of Human Running Identified from Measurements

An important prerequisite for any inverse optimal control computation is the establishment of a basis of potential objective functions which then form the base functions $\Psi_i(x(t), u(t))$ in equation (16). These functions are context specific and should reflect the current expert guesses on the problem under investigation. It is obvious that their choice is crucial, since any inverse optimal control problem solution can only be as good as choice of base functions permits. We recall that the goal of the inverse optimal control task is to find an objective function that gives a meaningful explanation in terms of characteristic physical properties of the system, such as energy, velocities, muscle activations etc. It is therefore not helpful to perform inverse optimal control using a mathematical basis of functions such as Fourier series since the resulting weights of the Fourier terms would not give the desired insights into the system's physical behavior. But for a choice of base functions formulated in terms of physical quantities it must be guaranteed that they are independent - in the sense that there should not be different (combinations of) base objective function producing the same solution (a simple example for this would be a minimization of time or a minimization of distance for a motion at constant speed, which would have exactly the same solution). In such a redundant situation it would obviously be impossible for inverse optimal control to make a selection between any linear combination of these two functions.

For the investigation of running motions, we have to distinguish two different situations (and we claim that there is a policy change between these two):

1. Running at full speed (sprinting) reaching the physical limits:
   In this case, we formulate the following hypotheses for potential contributions to the objective function

   - maximization of running speed
   - maximization of frequency /minimization of step time
   - minimization of duty factor (relative contact time, see [17])
   - ...

2. Running at controlled slow to medium speed (jogging) at which runners typically are capable to set their speed to the desired value and then adjust their running style. For this mode, we formulate the following hypotheses:

   - minimization of joint torques (squared)
   - minimization of joint accelerations (squared)
   - minimization of head motions, compare [18] (e.g. velocities squared)
   - maximization of stride length
   - ...

In this paper, we focus on the second case, i.e. running at controlled low to medium jogging speed, and we investigate a particular experiment with running speed $10 km/h = 2.78 m/s$. To this end, we conducted an experiment for which one male

participant volunteered. He gave written and informed consent before his inclusion and the study conformed to the Declaration of Helsinki. We asked him to run on a treadmill. After a training period to be familiar with the treadmill, we imposed the running speed to the constant value of $10km/h$ to match the case of a low medium jogging speed. 3D kinematic data were recorded with twelve high resolution Vicon MX cameras (Oxford Metrics®) at a sampling rate of 120Hz. Thirty-four reflective markers were attached to the participants'skin on standardized landmarks with respect to the anthropometric table of de Leva (see figure 5).

The average speed of $10km/h$ is imposed as a constraint in the optimal control problem. We do not investigate the objective functions underlying running at maximum speed, since this might require detailed muscle models in order to more precisely describe the existing physical limits.
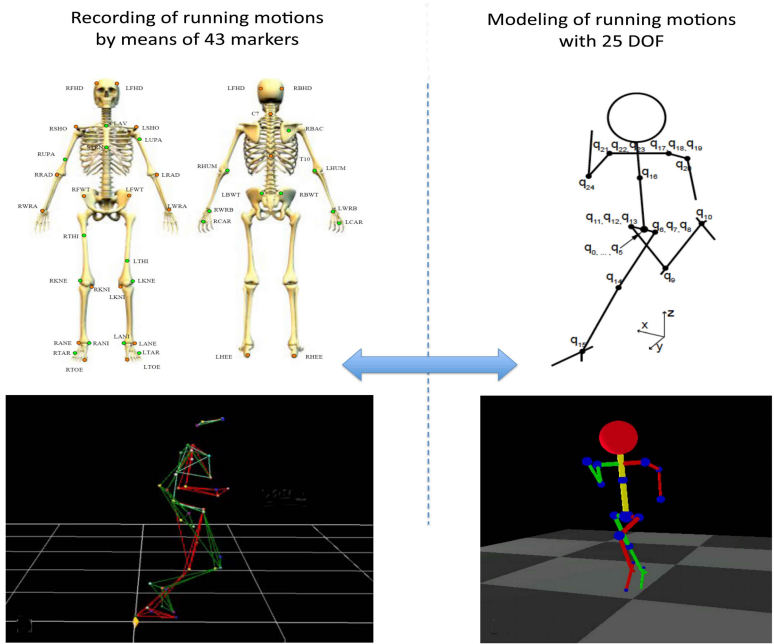
Fig. 5 Recorded marker positions are used to reconstruct movements of joint centers and other characteristic points: the deviation between corresponding points in the model and the experiments is minimized by inverse optimal control

Another issue to be solved before starting the computations is the choice of quantities z that are used to determine the match between computational model with 25 DOF and measurements of the 43 marker positions see figure 5 in the least squares sense in eqn. (15). It is still an ongoing discussion in biomechanics and also in humanoid robotics on which level a fit between human measurements on the one side

and the mathematical model or the humanoid robot on the other side should best be achieved. There are different possibilities, e.g.:

- matching joint angles and optionally also position and orientation of the central body (need to be derived from measured marker positions, but are directly available as state variables in the model);
- matching marker positions (are directly available from measurements, but have to be computed as functions of the state variables in the model);
- matching positions of characteristic points such as joint centers or dominant bones (typically require computations on the measurement side based on measured markers as as well on the model side based on state variables).

Also combinations of the choices above are of course possible and might be reasonable. We have for the time being chosen the last option and fit measurements and model at the following points in the 3D runner: right and left hip points, knees, ankles, balls of the feet, shoulders, elbows and wrists as well as head center, xyphoid, suprasternale and root point in the pelvis. Using 3D position information for these 18 points, this leads to a matching vector z of dimension 54 in eqn. (15).

Based on the reasoning above, we have formulated the following parameterized objective function which combines base functions for the minimization of all 19 joint torques squared, a minimization of the head motion (which has to be expressed in terms of the angular velocities of the trunk since in our model the head is rigidly connected to the trunk) and a maximization of stride length (i.e. a minimization of the negative stride length):
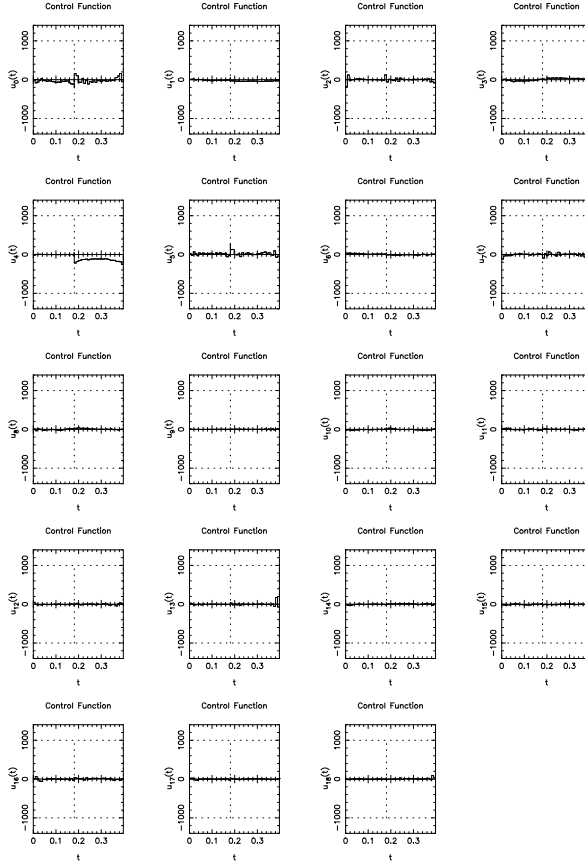
$$\min \left\{ \int_0^T (\sum_{i=1}^{19} \alpha_j \mathscr{M}_i^2) dt + \int_0^T (\beta_1 \omega_{x,tr}^2 + \beta_2 \omega_{y,tr}^2) dt - \gamma_1 \Delta y_{pelvis} \right\} \qquad (17)$$

We have not added the terms related to the accelerations of the limbs since we feel that there may be some redundancy with the torque criterion. We assume here that the weight factors respect body symmetries, i.e. are identical for corresponding terms in the left and right body half. This means that the 19 weight factors for the torque terms can be reduced to 10 unknown parameters. Overall, we then have 13 weight factors to be determined by inverse optimal control.

Our computations have resulted in the following best set of weight factors for the objective function proposed above:

Table 1 Weight factors of objective function contributions determined by inverse optimal control

α_{hip}	α_{knee}	α_{ankle}	α_{trunk}	$\alpha_{shoulder}$	α_{elbow}	β	γ
0.56	3.7	4.0	9.1	9.1	5.8	435	337
10.7				3.0		300	
0.0				0.0			

Fig. 6 Torques of the computational solution that best reproduces the measurements

Figures 6 and 7 show the resulting 19 torque and 25 position histories of the solution with the above objective function. Velocity variables are not shown for reasons of space. This solution represents the best possible match between the measurements and the computations with the chosen model and the selected basis of objective function. The fit is quite good but there are still some differences which can only be further reduced if some additional adjustment in the model and the problem formulation are made:

- The model of the three-dimensional runner (section 2) needs to be refined and several DOF have to be added since it can be shown that even with a a free choice of torque inputs (solving a pure fitting problem and leaving the inverse optimal control question apart) no perfect fit can be achieved. In particular the upper body and the head need additional DOFs. Kinematic and dynamic data used for this
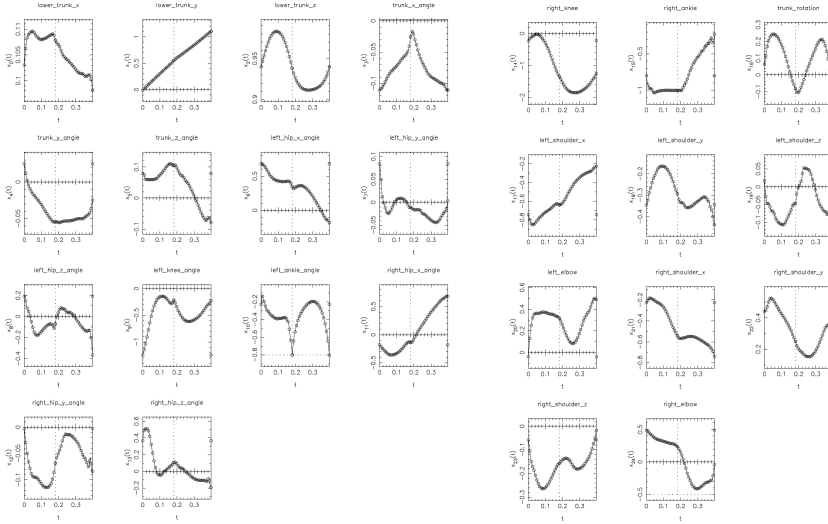
Fig. 7 25 position variables of the solution of inverse optimal control

study was taken from anthropometric tables and scaled to the subject's size and weight. However, for a more detailed fit, the data should ideally be further personalized by taking into account the subject's individual proportions. Also the foot contact modeling with a point contact of the ball which is more realistic for very fast running might be revised for this slower running speed.

- Additional terms might be required in the objective function (17). These can be new functions expressed in variables of the current model such as positions, velocities, accelerations, torques, etc. Other examples are terms related to muscle activity or fatigue or to the effect of passive tissue. These terms would require an additional enhancement of the runner model by adequate muscle models.

5   Conclusion and Perspectives

In this paper, we have discussed the use of optimal control for the investigation of three-dimensional human running motions. We have summarized previous results for the generation of natural running motions by forward optimal control. The focus of this paper was on inverse optimal control, and we have shown first results for a running motion on a treadmill at 10 km/h. The best possible combination of weight factors for an objective function of terms related to torque minimization, head motion minimization and stride maximization was determined. The inverse optimal control computations have been preformed using a previously developed inverse optimal control technique which uses a combination of a direct multiple shooting

method and a derivative-free optimization technique. We have shown here for the first time, that it can also be applied to multi-phase problems with discontinuities and complex multi-body system models.

While these first results are already quite promising, there still remains some work to do until we have achived a reliable explanation of the true objectives underlying human running. As discussed above, we are currently working on some refinements of the model in terms of additional DOF, personalized parameters, and possibly also the addition of muscle models. In addition, other base terms will be added in the objective functions based on biomechanical literature. Later, we will also validate the objective function by performing inverse optimal control computations at different speeds (experiments for 12, 14, 16, and 18 km/h have already been performed but not yet been analyzed) and by extending the study from one to several subjects. With this, it would be possible to answer e.g. if objective functions are individual or can be generalized, and if and at which speeds there is the policy change between running at controlled speed and running at high speed discussed in section 4.

References

[1] McNeill Alexander, R.: Optima for Animals. Princeton University Press (1996)
[2] Toyota partner robot website:
    `http://www.toyota.co.jp/en/special/robot/`, running video on youtube: `http://www.youtube.com/watch?v=yBUcfFbndVY`
[3] Honda Asimo Website: `http://asimo.honda.com/`
[4] Hodgins, J.K.: Simulation of human running. In: Proceedings of IEEE International Conference on Robotics and Automation, vol. 2, pp. 1320–1325 (1994)
[5] Schultz, G., Mombaur, K.: Modeling and Optimal Control of Human-like Running. IEEE/ASME Transactions on Mechatronics 15(5), 783–792 (2010)
[6] Mombaur, K.: Using optimization to create self-stable human-like running. Robotica 27, 321–330 (2009) (published online June 2008)
[7] Miller, R.H., Umberger, B.R., Caldwell, G.E.: Optimality criteria for human running investigated by forward dynamics simulations. Biomechanics in Sports 29(11 suppl. 2), 813–816 (2011)
[8] Ackermann, M., van den Bogert, A.J.: Optimality principles for model-based prediction of human gait. Journal of Biomechanics 43(6), 1055–1060 (2010)
[9] Mombaur, K., Truong, A., Laumond, J.P.: From human to humanoid locomotion - an inverse optimal control approach. Autonomous Robots 28(3), 369–383 (2010)

[10] Ren, L., Howard, D., Ren, L., Nester, C., Tian, L.: A generic analytical foot rollover model for predicting translational ankle kinematics in gait simulation studies. Journal of Biomechanics 43(2), 194–202 (2010)

[11] de Leva, P.: Adjustments to Zatsiorsky-Seluyanov's segment inertia parameters. Journal of Biomechanics 29(9), 1223–1230 (1996)

[12] Wieber, P.B.: Humans toolbox (July 2007),
`http://www.inrialpes.fr/bipop/software/humans/`

[13] Leineweber, D.B., Bauer, I., Bock, H.G., Schlöder, J.P.: An efficient multiple shooting based reduced SQP strategy for large-scale dynamic process optimization. Comput. Chem. Engng. 27 (2003)

[14] Bock, H.G., Plitt, K.J.: A multiple shooting algorithm for direct solution of optimal control problems. In: Proceedings of the 9th IFAC World Congress Budapest, pp. 243–247 (1984)

[15] Mombaur, K.D., Bock, H.G., Schlöder, J.P., Longman, R.W.: Open-loop stable solution of periodic optimal control problems. ZAMM (Journal of Applied Mathematics and Mechanics) 85(7) (2005)

[16] Powell, M.J.D.: The BOBYQA algorithm for bound constrained optimization without derivatives, Report No. DAMTP 2009/NA06, Centre for Mathematical Sciences, University of Cambridge, UK

[17] Minetti, A.E., McNeill Alexander, R.: A Theory of Metabolic Costs for Bipedal Gaits. J. Theor. Biol. 186(4), 467–476 (1997); Process optimization - part I: theoretical aspects. Comput. Chem. Engng. 27, 157–166 (2003)

[18] Pozzo, T., Berthoz, A., Lefort, L.: Head stabilization during various locomotor tasks in humans: I. Normal subjects. Exp. Brain Res. 82, 97–106 (1990)

[19] Muybridge, E.: The Human Figure in Motion. Dover Press (1957)

Synthesizing Human-Like Walking in Constrained Environments

Jia Pan, Liangjun Zhang, and Dinesh Manocha

Abstract. We present a new algorithm to generate plausible walking motion for high-DOF human-like articulated figures in constrained environments with multiple obstacles. Our approach combines hierarchical model decomposition with sample-based planning to efficiently compute a collision-free path in tight spaces. Furthermore, we use path perturbation and replanning techniques to satisfy the kinematic and dynamic constraints on the motion. In order to generate realistic human-like motion, we present a new motion blending algorithm that refines the path computed by the planner with motion capture data to compute a smooth and plausible trajectory. We demonstrate the results of generating motion corresponding to placing or lifting object, walking and bending for a 34-DOF articulated model.

1 Introduction

How to generate or synthesize natural human walking is an important topic for many communities, such as humanoid robotics, computer animation, virtual prototyping, human factor and biomechanics. This is a challenging problem due to both combinatorial and behavioral complexities of human body and there are no known accurate and efficient algorithms to simulate its motion. Even the simplest human-like models that represent the skeleton as an articulated figure need at least 30-40 joints to model different motions such as navigation, sitting, walking, running, object

Jia Pan · Dinesh Manocha
Department of Computer Science, UNC Chapel Hill, Brooks Computer Science Building, 201 South Columbia Street, Chapel Hill, NC 27599-3175, USA
e-mail: {panj,dm}@cs.unc.edu

Liangjun Zhang
Department of Computer Science, Stanford University, 318 Campus Drive, Clark Center S245, Stanford, CA, 94305-5447
e-mail: zhanglj@cs.stanford.edu

manipulation, etc. The high dimensionality of the configuration space of the articulated model makes it difficult to efficiently compute the motion. In addition to collision-free and kinematic constraints, we also need to ensure that the resulting trajectory satisfies the posture and dynamic constraints and looks realistic.

There exists extensive literature relevant to simulating human-like motion in robotics, biomechanics, animation and related areas. However, they mainly focus on human walking in open environments. In the other side, many applications need a natural human walking in constrained environments, i.e. environments with many obstacles that people need to deviate from common behavior so as to avoid collisions.

Main Results: We present an original hybrid approach that combines motion planning algorithms for high-DOF articulated figures with motion capture data to generate collision-free motion that satisfies both kinematic and dynamic constraints. Our approach performs whole-body planning by coordinating the motion of different parts of the body and later refines the trajectory with mocap data.

The rest of the paper is organized as follows. We first give a brief overview of our approach in Section 2. Then the motion blending algorithm is described in Section 3. Finally, we highlight the performance of our approach in Section 4.

2   System Overview

In this section, we present an overview of our approach on generating natural human motion in constrained environments. The overall pipeline of our algorithm is given in Fig 1(a). We do not make any assumptions about the environment or the obstacles in the scene. We assume that the human-like model is represented by an articulated model in Fig 1(b) with serial and parallel joints and there is no limit on the number of DOFs.
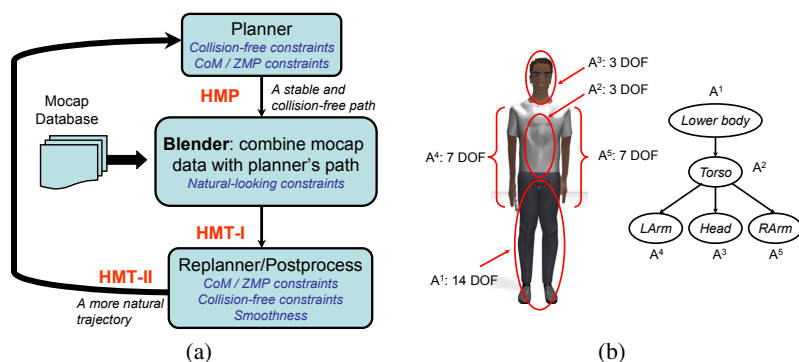
Fig. 1 (a) An overview of our hybrid approach, which can combine the motion computed by planner and the motion from mocap databases to generate a collision-free, dynamic and natural human motion. (b) Our 5-component decomposition scheme for a 34-DOF human-like model. We compute a trajectory for each component in an incremental manner.

Our approach first uses a sample-based high-DOF planner to compute a collision-free path, and it also takes into account the foot placement constraint and static/dynamic balancing constraints. In order to deal with a large DOFs, we use a hierarchical decomposition scheme and present an efficient decomposition planner [2].

In practice, generating natural-looking motion using planning algorithms is considered non-trivial, due to the following reasons: firstly, the randomness of the motion planner can cause jerky and unnatural motion, especially when parts of the robot are in open space; secondly, computing a collision-free trajectory corresponds to searching in a very high-dimensional space, which can be quite challenging in computation; finally, the constrained environments may have tight spaces or narrow passages and this makes it hard for even sample-based planners to search for a valid trajectory. In order to address these issues, we present a novel motion blending algorithm, which refines the motion computed by the planner with motion capture data. We also ensure that the resulting motion is collision-free and satisfies all the other constraints.

For more details about the system, please refer to our recent paper [1].

3 Motion Blender

The output of the high-DOF planner is a collision-free path, we call it the *Human Motion Path* (HMP). The decomposition planner only considers collision-avoidance constraints and uses random sampling. This approach can lead to a jerky motion along the trajectory. As a result, we augment or modify HMP by using mocap data, if available.

We process the postures in HMP in a per-component manner. First we analyze the postures in HMP based on three criteria which take into account the local environment (i.e. nearby obstacles) around the posture and the quality of the computed path. The three heuristic criteria are *space clearance* (CLR), *posture similarity* (PS), and *torque variation* (TV).

Space clearance evaluates whether the human model with a given configuration $\mathbf{q}$ is in the constrained environment or not. To compute this, we generate samples in the neighborhood of $\mathbf{q}$ uniformly. We check whether each sample collides with any obstacles in environments and use $\text{CLR}(\mathbf{q}) = \frac{\#non-collision\ samples}{\#all\ samples}$ as a metric to estimate the space clearance, while $\#(\cdot)$ is the counting function. In other words, we compute the possibility that a small variation of $\mathbf{q}$ will produce an in-collision configuration, which implicitly describes the local distribution of obstacles near human model's current position. If $\text{CLR}(\mathbf{q})$ is larger than a given threshold (e.g. 50%), we estimate that the human model is in the open free space. Otherwise, we estimate that this posture is in the constrained space.

Posture similarity evaluates whether the current configuration $\mathbf{q}$ of human model is similar enough to ascertain $\mathbf{q}_d$ in the motion capture database. We use the horizontal translation- and vertical rotation-invariant metric to measure the distance between

two configurations. If PS($\mathbf{q}$) is larger than the threshold, then the human model is regarded to be in its natural configuration.

Torque variation gives a rough evaluation of motion's validness according to dynamics. For each configuration $\mathbf{q}$ on HMP, we can estimate its velocity $\dot{\mathbf{q}}$ and acceleration $\ddot{\mathbf{q}}$. Then we can compute the torque τ for human body by inverse dynamics. In a natural-looking motion, the torque of each joint tends to change gradually and this boils down to minimizing the integral $\int_0^T |\dot{\tau}|^2 dt$. Using calculus of variation, this means $\ddot{\tau} = 0$, i.e., τ should changes linearly between $\tau(\mathbf{q}_s)$ and $\tau(\mathbf{q}_g)$, where $\mathbf{q}_s$ and $\mathbf{q}_g$ are the first and last configuration of HMP. When $\tau(t)$ deviates from the linear formulation, it may result in a more unnatural motion.

Based on these criteria, we process different components of each posture with different strategies: 1) For components that lie in a constrained space, we primarily rely on the samples in HMP, even though the computed path may not be natural-looking. In these cases, the planner computes a collision-free and statically stable path, and any large changes to that path may result in collisions. As a result those postures of HMP are used in the final path, we only allow small perturbations during the refinement. 2) For components of the human-model that lie in open space and appear to be natural, we tend to retain those postures. Otherwise, we compute a configuration based on the mocap database. The output of this phase is a trajectory that combines HMP with the mocap data and we refer to it as *Human Motion Trajectory I* (HMT-I). HMT-I may not be collision-free or even smooth, but it contains some important information from the mocap data that can bias the result of the motion planner towards a natural-looking trajectory. We finally perform a decomposition-based replanning with HMT-I as the guidance path, and compute a collision-free trajectory called *Human Motion Trajectory II* (HMT-II).

4   Results

We designed three challenging environments with many obstacles and tested our approach to generate collision-free motion by specifying the initial and final configurations of the human model. All the benchmarks consist of multiple obstacles and it would be difficult to edit mocap data directly for such settings. Rather we generated the initial path (HMP) using our planner and used some postures from the CMU mocap database to make the motion appear to be more natural.
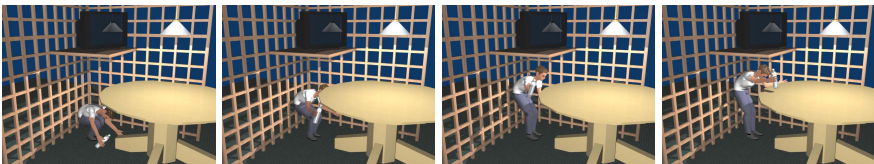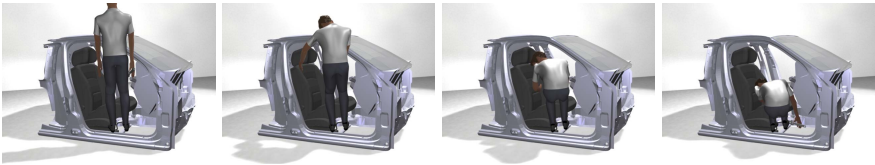

Fig. 2 Object Retrieval: the human model stands up and places the object on the table
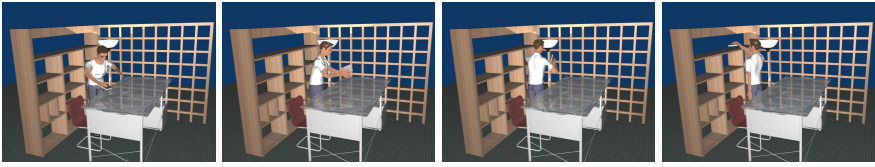
Fig. 3 Object Placement: The human model picks the book from the table and puts it on the bookshelf

Fig. 4 Walking & Bending: The human model walks towards the car, avoiding some obstacles. It bends and stretches to put the tool inside the car

Fig. 5 More bending: The human bends and stretches to put the tools under the chair. This environment is far more constrained than the one of Walking & Bending (Fig 4).
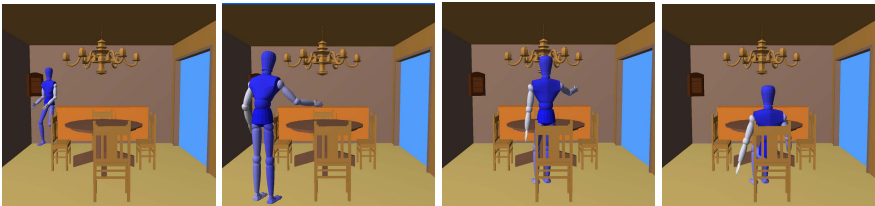
Fig. 6 Walking in the dinning room: The human walks within a dinning room toward a chair and sits down

In the first benchmark (Object Retrieval, Fig 2), human begins from a knee-bending posture, tries to pick up an object and then puts it on the table. There are three main obstacles: grate behind, a ceiling above and the table.

The second benchmark is the manipulation task (Object Placement, Fig 3). In this case, the human holds an object (e.g. a book) and rotates backward to put it on a shelf. The grate, lamp, and bookshelf result in a tight and constrained environment.

The third benchmark has two main motion components: walking and bending (Fig 4). The human first walks along a passage with an obstacle around its head and then puts a tool into the car. The task of putting the object inside the car is

challenging due to limited space. Fig 5 show a more challenging variation of this benchmark with more human bending.

We also test our algorithm to generate a walking motion in constrained living room environment as shown in Figure 6. Our framework can successfully synthesize natural human motion automatically in these environments in about 3-5 minutes.

Conclusion

We have presented an algorithm that combines a high-DOF motion planning algorithm with mocap data to generate plausible human motion and satisfy geometric, kinematic and dynamic constraints. We use a hierarchical decomposition of a high-DOF articulated model and use that decomposition for constrained coordination and to satisfy different constraints. We also present automated techniques to search a mocap database for plausible motion clips and generate a smooth, blended motion. We highlight the performance on generating motion for different tasks including object placement, object retrieval and walking & bending.

References

1. Pan, J., Zhang, L., Lin, M.C., Manocha, D.: A hybrid approach for synthesizing human motion in constrained environments. In: International Conference on Computer Animation and Social Agents, CASA (2010) (to appear)
2. Zhang, L., Pan, J., Manocha, D.: Motion planning of human-like robots using constrained coordination. In: International Conference on Humanoid Robots, Humanoids (2009)

Locomotion Synthesis for Digital Actors

Julien Pettré

Abstract. Motion capture technologies are commonly used in the field of computer animation for interactive applications. They enable synthesizing highly realistic motions for human figures, but they suffer a lack of flexibility. Editing is required to answer the needs of interactivity, or to match the motion with some new geometrical and environmental constraints. During the last two decades, the computer animation research community expended a great deal of effort to use prerecorded sets of motion capture to synthesize animations with unknown (a priori) constraints. This paper provides a short overview on these recent motion capture edition techniques. We also describe more into details a method for synthesizing locomotion with continuous control over velocity parameters. We expect these previous works to be of interest for robotocists who attempt to control humanoid robots motion by imitation techniques, and who first need to be able to synthesize with control input motions.

1   Introduction

Motion synthesis and control for humanoid 2-legged robots is a challenging problem. Generating walking motions is crucial but difficult because dynamic balance must be ensured to prevent falling. A number of dedicated techniques have been proposed: for example, walking motions are computed so that the ZMP always exists in the expected supporting foot. This asks for the walking motion to be computed in advance, in real-time, and to be stabilized with respect to the robot's sensors output. Some simplified dynamic models (e.g., ZMP of the cart-table model for HRP-2 [9]) can be used to efficiently achieve such computations.

More recently, imitation-based motion synthesis appeared for humanoid robots [29]. The general principle of such methods is to consider a kinematic motion

---

Julien Pettré
INRIA-Rennes, Campus de Beaulieu 35042 Rennes, France
e-mail: `julien.pettre@irisa.fr`

(typically, motion captured) and to successively apply a kinematic and a dynamic filter to fit the motion to the specific kinematic nature of the robot (degrees of freedom, limbs size, self-collision removal) and then, to ensure that the motion is dynamically balanced and performable with respect to mechanical constraints.

Given the potential of motion imitation techniques in the field of Robotics to provide humanoids with ability to perform natural human motions, it is made important to overview kinematic motion synthesis techniques that were developed in the field of Computer Graphics to animate virtual humans. This paper provides an overview of motion capture edition techniques that were proposed in the past two decades and which rapidly became highly popular in the entertainment industry because of the trade-off they provide between computation performances and intrinsic realism of results.

The next Section provides an overview of existing technologies as well as method to generate new motions from motion capture. Following that, we detail a method to synthesize locomotion with continuous velocity control.

2   Motion Capture Edition Techniques

2.1   *Motion Capture*

In the field of computer animation, the general goal of motion capture is to record the movements performed by a real actor and to translate them onto a digital actor. Usually, the position and orientation of main human limbs are recorded. Then, the movements are transposed onto a kinematics chain which models the mobility of the digital actor. Movement transposition can be directly achieved when the real and the digital actors have similar morphology and size. When the real and the digital actors have different morphologies, a motion retargetting stage is required, which can be achieved online [12], [8] or offline during a preprocessing stage [6].

Several motion capture technologies and systems are available. First, real actors can wear exoskeletons: the exoskeleton follows the real actor's movements and orientation of can be directly measured. Such equipment allows tracking the articular orientations only, whereas the global position of the actor remains unknown. Second, the orientation of limbs can also be captured using inertial sensors. Such systems suffer positional drifts in time, but can be used in various situations (e.g., outdoor). Third, actors can also wear magnetic sensors: they are localized when moving in a magnetic field emitted by the system. However, the captured volume is limited and the system is highly sensitive to external perturbation as any metallic objects would interfere with the system. Fourth, optical systems are probably the most used in the entertainment industry because of their high accuracy. The motion capture volume is directly defined by the number of cameras composing the system. Human movements are visually captured using image sensors and reconstructed in 3D by triangulating some tracked points. Again, several types of optical systems can
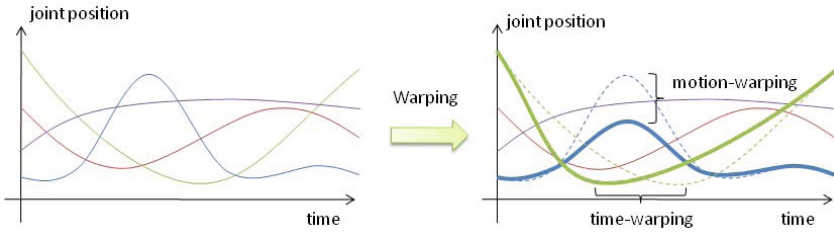
Fig. 1 Motion warping techniques work from a unique motion signal. On the left image, an example of motion made of few degrees of freedom with one motion signal each is represented. Users define constraint in space and time to edit motion signals. Motion signals are smoothly deformed to satisfy constraints as shown on the right image. An example of a time warping and of a space warping over one signal each is represented.

be distinguished, but markers, active or passive, are generally used to facilitate the accurate tracking of specific points of the human body.

The motion capture process results into a fixed recorded motion sequence which can be replayed at will. By nature, fixed sequences can be used in fixed contexts (environment, scenario) and are not suitable for interactive applications where motion has to fit new physical constraints or events caused by the interactive intervention of users. This is the strongest limitation of motion capture. To overcome this limitation many *motion edition* techniques were proposed to preserve the intrinsic realism of motion while satisfying new geometrical or physical constraints.

2.2   *Space and Time Warping*

Space and time warping were early proposed as techniques to edit motion capture sequences [32]. They work from a single motion capture source as represented in Figure 1. Space warping allow to locally deform articular trajectories in order to satisfy some user-defined constraints. A user constraint is for example specific posture the body has to reach at a given time. Motion signals are then deformed over a time window - centered on time the posture has to be reached - so that the defined constraints are satisfied whilst motion signals remain continuous and smooth. Time-warping techniques do not edit the captured body postures but the speed at which they are replayed in time. Such a method can for example locally accelerate or decelerate an initial motion. Timing can also be globally changed. Motion and time warping techniques can be simultaneously combined and can be expressed as edition functions applied to a motion signals $\theta_i(t)$ (where i refer to each degree of freedom):

$$\theta_i'(t) = f(\theta_i, t'), \; with \; t' = g(t) \tag{1}$$

where f is a space-warping function and g a time-warping function. These functions can be individually defined for each articular trajectory, however, commonly a
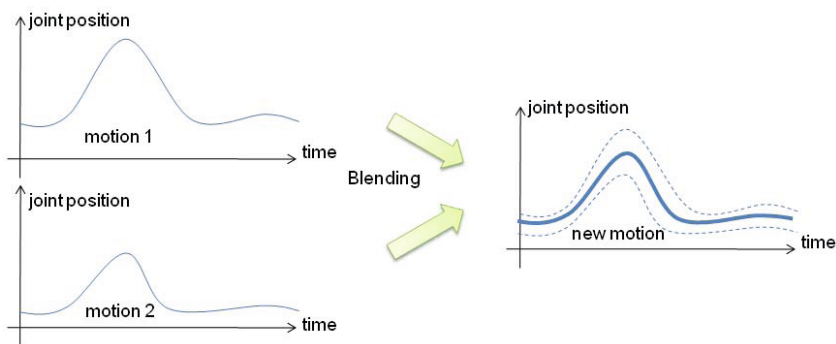
Fig. 2 Example of a motion blending from 2 motion sources made of one motion signal only. Motion signals of the two sources have identical structure which is a general requirement in order to create believable motions.

unique time-warping function is defined and uniformly applied to all motion signals. Several similar methods or extensions were proposed in the literature [3, 24, 5, 17].

2.3 Motion Blending

Motion blending techniques work from 2 motion sources or more. The key-idea of these technique is to interpolate the recorded angular trajectories between the two sources in order to generate a new motion sequence as represented in Figure 2. Interpolating motion signal has no physical meaning, it is then made important to interpolate together motions with similar content. This avoids generating unbelievable sequences. This requirement generally needs for annotated motion data and carefully organized motion database. Such process is clearly defined by Rose and colleagues in [23]: motions of the same type of action are called verb, whereas variations of a same action correspond to adverbs. Motions annotated with the same verb but with different adverbs can be interpolated together to generate new motion with controlled style. The motion blending technique can thus be expressed as the interpolation of n motion sources $\theta_i(t)$:

$$\theta'_i(t) = \sum_{i=1}^{n} w_i \theta_i(t) \tag{2}$$

where w_i are the interpolation weights and θ' the interpolated motion. Even when motion sources have similar structure, they may have different timings and durations. The previously described time-warping techniques are then employed to adapt timings and improve the matching of motion structure [10]. Other methods for organizing a motion database were proposed, such as by using finite state machines or hierarchical structures [17, 15]. The annotation of a motion database can be tedious, automatic annotation techniques were proposed based on machine learning
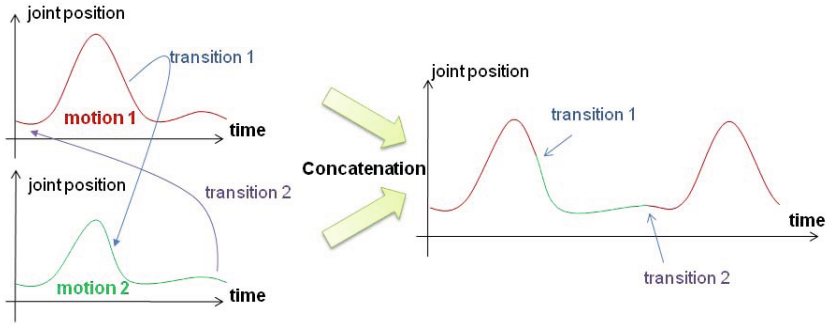
Fig. 3 Example of a motion concatenation using 2 sources made of one motion signal only. Motion signals are split and reordered where transition can be smoothly achieved, i.e., when position value and variation are similar (transition 1 and 2). Concatenation enables creating endless motion by repeating portions of the motion sources.

or principal component analysis [2, 4, 19]. In order to precisely control some motion parameters such as locomotion velocities, extracting recorded motion features is first required. Several methods were proposed such as in [20, 27, 13, 21, 14].

2.4 Motion Concatenation

Motion concatenation techniques are probably the most used in practice for interactive animation of human figures. They work from a unique or several motion sequences. They identifies all the possible transitions in motions, regardless of whether they belong the same motion or not. A transition is detected between motion i at time t_1 motion j at time t_2 when $\theta_i(t_1) \approx \theta_j(t_2)$ (as previously mentioned, we may have $i = j$). All the difficulty of these techniques resides in the identification of similarities in various motions (i.e., definition of the "$\approx$" comparison operator). Then, a concatenation between motions i and j can be achieved by transiting from motion i at time t_1 to motion j at time t_2. This principle is illustrated in Figure 3 with 2 motions with two possible transitions: we show endless motion can be synthesized from these two short sequences.

Graphs efficiently capture feasible motion transitions and concatenations: transitions are the graph nodes, whereas motion clips between two transitions are its edges. Any path in the resulting graph is smooth motion made of several clips. For this reason, concatenation appeared in the literature as Motion Graphs [11]. Note that inversely, it is possible to consider edges being transitions whereas nodes are portions of motion. Both conventions were used in other papers on motion concatenation [1, 16]. Some key-elements compose motion concatenation techniques and differentiate the proposed approaches, such as: the method for searching for transitions (i.e., identifying *similar* postures and motions), the method for transiting from clip to clip, the method for pruning to create sparse graphs, and the method

for searching graphs to create user-controlled motions. Previously described motion blending techniques can for example be used to perform transitions.

The key-ideas brought by these early works on motion concatenation lead to many developments. The automatic graphs construction process was refined in [7, 34, 33] to better match the needs of interactive control. Sung and colleagues combined motion graphs with probabilistic motion planning techniques to animate groups of human figures [30]: rough motions are obtained by assembling motion clips, which is further edited to satisfy precise constraints in space and time. The building of motion controllers and near optimal path search is achieved in [31, 18] to reach user-defined states in the context of interactive control. Hybrid approaches combining motion graphs and motion blending techniques enabled continuous motion control [26, 25].

3   Locomotion Synthesis with Continuous Control

Previous sections emphasize the importance of motion blending techniques to enable continuous control over motion parameters. We now describe a method entirely based on a motion blending technique to interactively synthesize locomotion with continuous control over the tangential and angular velocities. This method fas first introduced in [22] and detailed in [21]. Velocity control satisfies the needs of many applications such as path following problems. This method is composed of 3 key-elements. First, a *motion capture database* is made of carefully preprocessed locomotion sequences. Second, given an input command, a *selection process* chooses 3 of these sequences to be blend. Finally, *motion blending* weights are determined by solving a simple linear system and motion with desired velocities is synthesized. Each of these components is described in the following sections.

3.1   *Motion Database*

Our motion capture database contains various locomotion sequences. Each sequence describes one complete locomotion cycle. All sequences start and end at a same identifiable event, such as for example the right foot heel strike event. Such a convention ensures that all the captures contained into the database have the same structure. The database does not only contain motion captures but also provide their duration T_i, their mean tangential velocity $\tilde{v}_i$ as well as their mean angular velocity $\tilde{\omega}_i$ at which the locomotion cycle m_i is performed. To extract these features, we apply fitting techniques: the global motion of the root is considered and linear fitting is apply over the trajectory. In the case of turning motions, an arc of circle is fitted. Mean velocities are deduced from the fitted shapes with respect to the motion durations. To summarize, motion database D is a set of n motions described as follows:

$$D = (m_i, T_i, \tilde{v}_i, \tilde{\omega}_i), i : 1 \to n \tag{3}$$
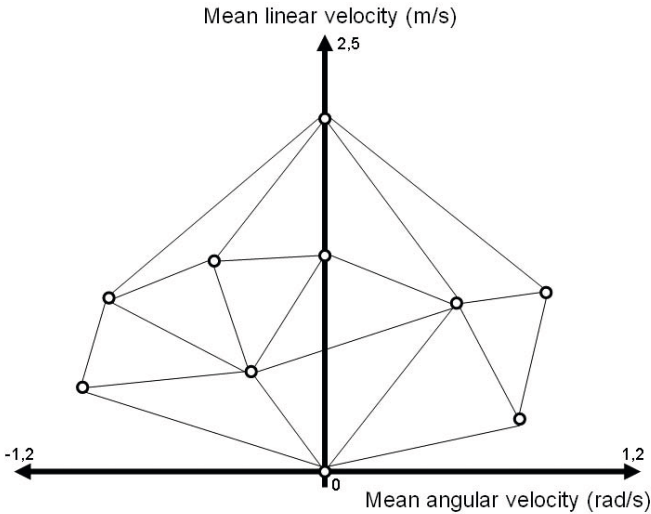
Fig. 4 Schematic representation of the motion capture database by projecting its content into the velocity space. Horizontal axis represent angular velocities, vertical axis represents tangential velocities. Each point represents one motion, its coordinates are the measured mean tangential and angular velocities. Delaunay triangulation over this set of points is represented as well: such a structuring allow efficient nearest neighbour searches. Thus, for any input velocity, we immediately deduce the motions having the closest features.

It is made important to carefully choose the motions to be added to the database. At least two criteria should be considered. First, motion database should enough cover the velocity space (see Figure 4: in order to enable synthesis of slow or fast walking motions with various angular velocities, it is important to dispose of corresponding similar examples. Second, a trade-off has to be found between motion density and sparsity: a too dense database will provoke frequent changes of the motion blending formula (see the section below) and will decrease the resulting motion smoothness. A too sparse database will impose blending motions with highly different features and may violate the need for similarity between blended motions. The displayed example of database in Figure 4 is an example of correctly prepared database. It demonstrates that few motions are required in practice.

The building of the database is made at a preprocessing stage. The following steps are done at runtime.

3.2 Motion Capture Selection and Weighting

We now consider an input command (v_d, ω_d), the user-desired locomotion velocities. The goal of this step is to search the database for examples the mean velocities of which are close to (v_d, ω_d). When projecting the input command the velocity space as shown in Figure 5, we simply retain the three motions that correspond to
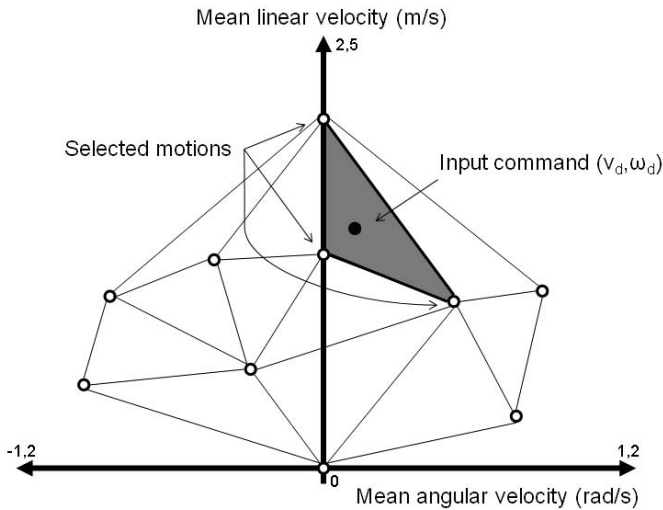
Fig. 5 Selection of 3 motion capture sources is made by searching motion clips the mean velocities of which are close to the input command

the vertices of the triangle the input command belong. The resulting motions are denoted $m_1(T_1,\text{~}v_1,\text{~}\omega_1)$, $m_2(T_2,\text{~}v_2,\text{~}\omega_2)$ and $m_3(T_3,\text{~}v_3,\text{~}\omega_3)$.

The more the input command is close to the features of one of the selected motion, the more this motion should be influent in the following blending process. We can put this principle into practice by computing the blending weights w_1, w_2 and w_3, respectively corresponding to motions m_1, m_2 and m_3, by solving the following simple linear system:

$$\begin{cases} w_1 + w_2 + w_3 = 1 \\ w_1\text{~}v_1 + w_2\text{~}v_2 + w_1\text{~}v_3 = v_d \\ w_1\text{~}\omega_1 + w_2\text{~}\omega_2 + w_1\text{~}\omega_3 = v_d \end{cases} \qquad (4)$$

3.3 Motion Blending

In the previous section we described how to select three motion sources from the database and how to weight them. We also previously described the principle of motion blending techniques in Section 2.3. We now synthesize a locomotion cycle m_d with desired mean locomotion velocities (v_d, ω_d) by simply interpolating motion sources with respective weights:

$$m_d = w_1 m_1 + w_2 m_2 + w_3 m_3 \qquad (5)$$

There are however several subtleties to be considered during this process. First is to choose the interpolation space. Motions are made of angular trajectories. They each

define the relative orientation of the character limbs. Interpolation is made easier when limbs orientations are expressed using quaternions [28]. Such a representation is classically used in the field of computer animation. Second is about motion timing. Interpolating the three sources using their initial timing is not possible because motions have various duration. It is made crucial to normalize the motion time dimensions before interpolating them. The duration T_d of the interpolated motion is deduced as follows:

$$T_d = w_1 T_1 + w_2 T_2 + w_3 T_3 \qquad (6)$$

Finally note that the motion selection step and the blending formula are recomputed each time the input command changes. Nevertheless, it is not required to compute the whole locomotion with desired velocities. It is possible to only extract postures one after the other. It is then made important to maintain frame information in terms of normalized time to achieve coherent extraction.

When the desired walking velocities are covered by our example database (i.e., they belong the convex hull represented in Figure 4), there always exists solution weights. If desired velocities are continuously changed, the combination and weighting of examples smoothly change accordingly. We thus enable continuously controlling walking velocities.

3.4   Example

This method has been applied to locomotion planning problems. In our example, a collision free path is computed into an environment made of obstacles using a probabilistic roadmap approach. This path is transformed into a trajectory by computing a velocity profile for following the path. We proposed a dedicated technique that enables bounding velocities and accelerations. Bounds are chosen with respect to
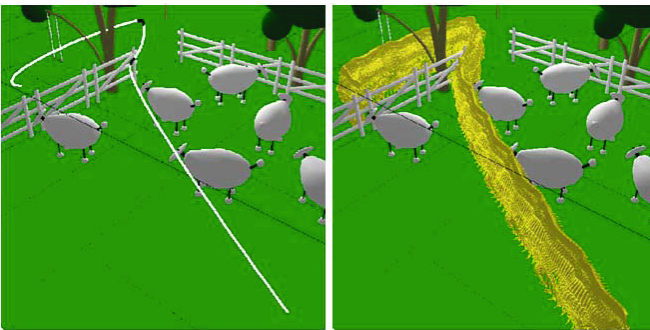
Fig. 6 Left: a path is planned in an environment made of obstacles using a probabilistic roadmap technique. Right: The path is transformed into a trajectory by computing a velocity profile. Resulting velocities are used as inputs of our animation method to synthesize a locomotion strictly following the path.

the content of the motion database. Resulting velocities are finally used as inputs of our motion blending technique. The resulting motion is illustrated in Figure 6. Details are provided in [22].

4 Conclusion

Motion capture provide realistic motions, technologies are each year easier to use and systems more affordable. Perspectives of using motion captures to drive the motion of humanoid robots recently appeared in the literature. This make more important for roboticists to understand how to overpass limitations of motion capture-based animation of human figures. This paper proposed a short overview 20 years of research over this problem. We described in few pages the three main methods to edit motion capture sequences and adapt them to some given environments or to match the needs of given scenarios: motion warping, motion blending, and motion concatenation techniques. Our description is certainly too short to enable re-implementing those techniques, we however hope it can be used as an introduction for the reader to the domain of motion capture-based computer animation.

References

1. Arikan, O., Forsyth, D.A.: Interactive motion generation from examples. In: SIGGRAPH 2002: Proceedings of the 29th Annual Conference on Computer Graphics and Interactive Techniques, pp. 483–490. ACM, New York (2002)
2. Arikan, O., Forsyth, D.A., O'Brien, J.F.: Motion synthesis from annotations. ACM Trans. Graph. 22(3), 402–408 (2003)
3. Bruderlin, A., Williams, L.: Motion signal processing. In: SIGGRAPH 1995: Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques, pp. 97–104. ACM, New York (1995)
4. Glardon, P., Boulic, R., Thalmann, D.: Pca-based walking engine using motion capture data. In: Computer Graphics International Conference, pp. 292–298 (2004)
5. Gleicher, M.: Motion editing with spacetime constraints. In: I3D 1997: Proceedings of the 1997 Symposium on Interactive 3D Graphics, pp. 139–148. ACM, New York (1997)
6. Gleicher, M.: Retargetting motion to new characters. In: SIGGRAPH 1998: Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques, pp. 33–42. ACM, New York (1998)
7. Gleicher, M., Shin, H.J., Kovar, L., Jepsen, A.: Snap-together motion: assembling run-time animations. In: SIGGRAPH 2003: ACM SIGGRAPH 2003 Papers, pp. 702–702. ACM, New York (2003)
8. Hecker, C., Raabe, B., Enslow, R.W., DeWeese, J., Maynard, J., van Prooijen, K.: Real-time motion retargeting to highly varied user-created morphologies. In: SIGGRAPH 2008: ACM SIGGRAPH 2008 Papers, pp. 1–11. ACM, New York (2008)
9. Kajita, S., Kanehiro, F., Kaneko, K., Fujiwara, K., Yokoi, K., Hirukawa, H.: A realtime pattern generator for biped walking. In: ICRA, pp. 31–37 (2002)

10. Kovar, L., Gleicher, M.: Flexible automatic motion blending with registration curves. In: Proceedings of ACM SIGGRAPH Symposium on Computer Animation, SCA 2003 (2003)
11. Kovar, L., Gleicher, M., Pighin, F.: Motion graphs. In: SIGGRAPH 2002: Proceedings of the 29th Annual Conference on Computer Graphics and Interactive Techniques, pp. 473–482. ACM, New York (2002)
12. Kulpa, R., Multon, F., Arnaldi, B.: Morphology-independent representation of motions for interactive human-like animation. Computer Graphics Forum, Eurographics 2005 Special Issue 24(3), 343–352 (2005)
13. Kwon, T., Shin, S.Y.: Motion modeling for on-line locomotion synthesis. In: SCA 2005: Proceedings of the 2005 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, pp. 29–38. ACM, New York (2005)
14. Kwon, T., Shin, S.Y.: A steering model for on-line locomotion synthesis. Comput. Animat. Virtual Worlds 18(4-5), 463–472 (2007)
15. Lau, M., Kuffner, J.J.: Behavior planning for character animation. In: SCA 2005: Proceedings of the 2005 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, pp. 271–280. ACM, New York (2005)
16. Lee, J., Chai, J., Reitsma, P.S.A., Hodgins, J.K., Pollard, N.S.: Interactive control of avatars animated with human motion data. In: SIGGRAPH 2002: Proceedings of the 29th Annual Conference on Computer Graphics and Interactive Techniques, pp. 491–500. ACM, New York (2002)
17. Lee, J., Shin, S.Y.: A hierarchical approach to interactive motion editing for human-like figures. In: SIGGRAPH 1999: Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques, pp. 39–48. ACM Press/Addison-Wesley Publishing Co., New York (1999)
18. Lo, W.-Y., Zwicker, M.: Real-time planning for parameterized human motion. In: SCA 2008: Proceedings of the 2008 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, pp. 29–38. Eurographics Association, Aire-la-Ville (2008)
19. Olivier, A.-H., Kulpa, R., Pettré, J., Crétual, A.: A Velocity-Curvature Space Approach for Walking Motions Analysis. In: Egges, A., Geraerts, R., Overmars, M. (eds.) MIG 2009. LNCS, vol. 5884, pp. 104–115. Springer, Heidelberg (2009)
20. Park, S.I., Shin, H.J., Shin, S.Y.: On-line locomotion generation based on motion blending. In: SCA 2002: Proceedings of the 2002 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, pp. 105–111. ACM, New York (2002)
21. Pettré, J., Laumond, J.-P.: A motion capture based control-space approach for walking mannequins. Computer Animation and Virtual Worlds 17(2), 109–126 (2006)
22. Pettré, J., Laumond, J.-P., Siméon, T.: A 2-stages locomotion planner for digital actors. In: SCA 2003: Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, pp. 258–264. Eurographics Association, Aire-la-Ville (2003)
23. Rose, C., Cohen, M.F., Bodenheimer, B.: Verbs and adverbs: multidimensional motion interpolation. IEEE Computer Grahics and Applications 18(5), 32–40 (1998)
24. Rose, C., Guenter, B., Bodenheimer, B., Cohen, M.F.: Efficient generation of motion transitions using spacetime constraints. In: SIGGRAPH 1996: Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques, pp. 147–154. ACM, New York (1996)
25. Safonova, A., Hodgins, J.K.: Construction and optimal search of interpolated motion graphs. In: SIGGRAPH 2007: ACM SIGGRAPH 2007 Papers, p. 106. ACM, New York (2007)

26. Shin, H.J., Oh, H.S.: Fat graphs: constructing an interactive character with continuous controls. In: SCA 2006: Proceedings of the 2006 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, pp. 291–298. Eurographics Association, Aire-la-Ville (2006)

27. Shiratori, T., Nakazawa, A., Ikeuchi, K.: Rhythmic motion analysis using motion capture and musical information. In: Proceedings of IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems, MFI 2003, July 30-August 1, pp. 89–94 (2003)

28. Shoemake, K.: Animating rotation with quaternion curves. In: SIGGRAPH 1985: Proceedings of the 12th Annual Conference on Computer Graphics and Interactive Techniques, pp. 245–254. ACM, New York (1985)

29. Suleiman, W., Yoshida, E., Kanehiro, F., Laumond, J.-P., Monin, A.: On human motion imitation by humanoid robot. In: ICRA, pp. 2697–2704 (2008)

30. Sung, M., Kovar, L., Gleicher, M.: Fast and accurate goal-directed motion synthesis for crowds. In: SCA 2005: Proceedings of the 2005 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, pp. 291–300. ACM, New York (2005)

31. Treuille, A., Lee, Y., Popović, Z.: Near-optimal character animation with continuous control. In: SIGGRAPH 2007: ACM SIGGRAPH 2007 Papers, p. 7. ACM, New York (2007)

32. Witkin, A., Popovic, Z.: Motion warping. In: SIGGRAPH 1995: Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques, pp. 105–108. ACM, New York (1995)

33. Zhao, L., Normoyle, A., Khanna, S., Safonova, A.: Automatic construction of a minimum size motion graph. In: SCA 2009: Proceedings of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, pp. 27–35. ACM, New York (2009)

34. Zhao, L., Safonova, A.: Achieving good connectivity in motion graphs. In: SCA 2008: Proceedings of the 2008 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, pp. 127–136. Eurographics Association, Aire-la-Ville (2008)

Whole-Body Motion Synthesis with LQP-Based Controller – Application to iCub

Joseph Salini, Sébastien Barthélemy, Philippe Bidaud, and Vincent Padois

Abstract. This paper deals with the dynamic control of humanoid robots interacting with their environment, and more specifically the behavioral synthesis for dynamic tasks. The particular problem that is considered here is the sequencing of elementary activities subjected to physical constraints, both internal as torque limits and external as contacts, within the framework of posture/tasks coordination. For that we propose to convert the set of tasks into weighted quadratic functions and to minimize their cost with a Linear Quadratic Program. The combination of elementary tasks leads to complex actions, and the continuous evolution of the weights ensures smooth transitions over time, as it is shown in the results.

1 Introduction

This paper focuses on the design of a generic and efficient framework for dynamic whole-body motions. Interesting issues appear, like the postural balance control when the robot realizes numerous objectives, the management of a task hierarchy or the transition control over time. The issue addressed in this paper is to find how to build scenarii of complex dynamic activities for humanoid robots, including physical interaction with the environment and ensuring a match between the control and the dynamics of the robot.

Generally, the whole-body dynamic control is solved using linear algebra methods as pseudo-inverses and projectors which give analytic solutions. In computer graphics, these tools give rise to uses for generation of physical simulations [2], and in robotics, some authors as in [11, 10] have performed the control of humanoid

Joseph Salini · Sébastien Barthélemy · Philippe Bidaud · Vincent Padois
Institut des Systèmes Intelligents et de Robotique
Université Pierre et Marie Curie / CNRS UMR 7222 Pyramide T55
4, Place Jussieu 75252 Paris Cedex 05 - France
e-mail: {salini,barthelemy,bidaud,padois}@isir.upmc.fr

robots with these methods. But the integration of physical constraints in the equations is not easy, especially inequality constraints.

To overcome these difficulties and provide a more natural way to take into account the constraints, some researchers propose the use of optimization programs, and especially Linear Quadratic Programs (LQP) which optimize a quadratic cost function subject to linear constraints. The work presented in [4] shows how to control humanoids and find a robust balance behavior with these particular programs. Furthermore, [1, 13] realize the animation of humanoids with a LQP.

Concerning the task hierarchy, numerous works have been done by the past and the problem is still complicated. [11, 12, 9] propose general frameworks for managing several tasks on redundant systems and return analytic solutions. Using LQP, [1] propose to use a weighted-sum objective as cost function which makes a trade-off between the different objectives, whereas [7] describes the hierarchy as a cascade of LQP with no interference between lower level tasks and upper ones. However, continuity in the transitions is not guaranteed during evolution of the hierarchy.

Here, the authors use a quadratic function to regroup the elementary tasks related to the functional elements of the robot which control a single degree of freedom (a joint) or several degrees of freedom simultaneously (a frame). The whole-body activity is tuned by weights related to the tasks which evolve in time, hence it is shown that relatively sophisticated scenarii can be designed while ensuring torque continuity and consequently a good control.

First, this paper exposes the control of humanoid robots with a LQP and explains the construction of the constraints and the cost function. Second, it addresses the
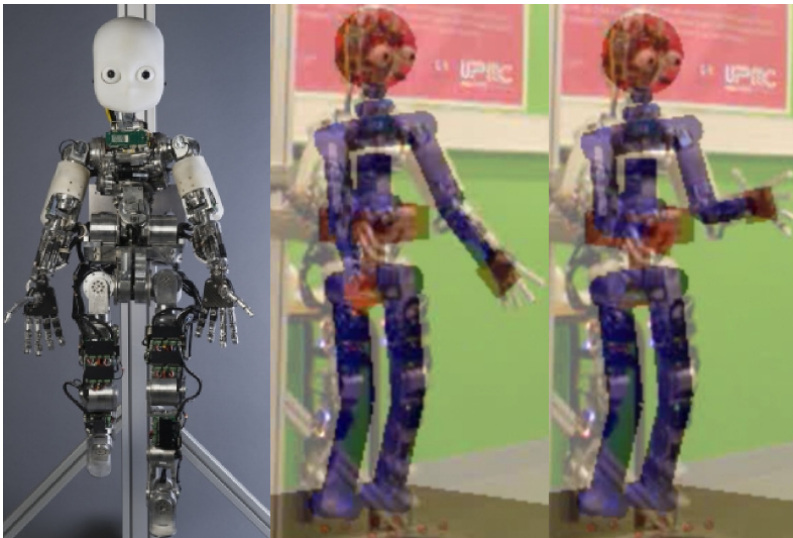
Fig. 1 Left: the real robot iCub. Middle and Right: Superimposition of the real and virtual iCub, performing the same simulation.

issue of the task hierarchy, their advantages and drawbacks. Third, it shows the application in simulation of the method described in this paper to the humanoid robot iCub [8] shown in Fig. 1.

2 Definition of Constraints

A humanoid robot is a mechanical system composed of several rigid bodies, connected to each other in a tree structure by joints. Each active joint has a limited motion range and is equipped with an actuator that can generate a bounded torque. This kind of robot is basically an under-actuated system, with a trunk, a head, lower and upper limbs, and it interacts with its environment with physical contacts. On these systems, one goal is often to make them perform several basic tasks in a coordinated manner. In this paper, the control of the robot is done by a LQP-based dynamic controller which needs one set of linear constraints and one quadratic cost function to model the problem, and applies the solution for whole-body motion.

The set of constraints used in the LQP generally represents the physical limits of the robot, both internal and external. Some constraints can be added or removed to change the perception of the world by the controller, but they must be carefully selected in particular to deal with variations in the contact conditions.

2.1 Equations of Motion

The robot dynamic behavior follows the Euler-Lagrange equations of motion:

$$\mathsf{M}\ddot{\mathbf{q}} + \mathsf{N}\dot{\mathbf{q}} = \mathbf{g} + \mathsf{S}\tau + \mathsf{J}_c^t \mathbf{f}_c \qquad (1)$$

where $\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}, \mathsf{M}(\mathbf{q}), \mathsf{N}(\mathbf{q}, \dot{\mathbf{q}}), \mathbf{g}(\mathbf{q}), \mathsf{S}, \tau, \mathsf{J}_c(\mathbf{q}), \mathbf{f}_c$ are respectively the generalized position, velocity and acceleration vectors, the generalized inertia matrix, the Coriolis and non-linear effects matrix, the gravity vector, the actuation matrix, the torque vector, the contact points Jacobian and finally the contact forces vector. Actuation matrix S allows each degree of freedom to be actuated or not. This equation is linearized around $(\mathbf{q}, \dot{\mathbf{q}})$ to fit the LQP, and it uses $\ddot{\mathbf{q}}, \tau, \mathbf{f}_c$ as variables.

2.2 Joint and Actuation Limits

As written above, each joint has a limited motion range and the related actuator can generate bounded torque. Furthermore, the linearization of Eq. 1 around $(\mathbf{q}, \dot{\mathbf{q}})$ should be ensured with bounded joint velocities. These constraints are described in the LQP as follows:

$$\tau_{min} \leq \tau \leq \tau_{max} \tag{2}$$

$$\mathbf{q}_{min} \leq \mathbf{q} + \dot{\mathbf{q}}h + \ddot{\mathbf{q}}h^2/2 \leq \mathbf{q}_{max} \tag{3}$$

$$\dot{\mathbf{q}}_{min} \leq \dot{\mathbf{q}} + \ddot{\mathbf{q}}dt \leq \dot{\mathbf{q}}_{max} \tag{4}$$

where $\tau_{(min,max)}, \mathbf{q}_{(min,max)}, \dot{\mathbf{q}}_{(min,max)}$ are torque bounds, joint position and velocity limits, h is the horizon of anticipation, and dt the sampling time. Anticipation avoids sudden braking and collisions with joint mechanical limits. The horizon of anticipation used in Eq. 3 has to be tuned to brake sooner or later.

2.3   Contact and Kinematic Closure

For all the humanoid bodies interacting with the environment, contacts are described as a set of punctual interactions with friction. Each contact point i has a velocity $\mathbf{v}_{ci} = \mathsf{J}_{ci}(\mathbf{q})\dot{\mathbf{q}} \in \mathbb{R}^3$, and each point develops a force denoted $\mathbf{f}_{ci} \in \mathbb{R}^3$. Four cases may happen

- the contact remains persistent, $\mathbf{v}_{ci} = \mathbf{0}$ and $\mathbf{f}_{ci}$ lies inside the Coulomb cone,
- the contact is lifting, $\mathbf{v}_{ci}.\mathbf{n} > 0$ and $\mathbf{f}_{ci} = \mathbf{0}$,
- there is no contact, $\mathbf{v}_{ci} \in \mathbb{R}^3$ and $\mathbf{f}_{ci} = \mathbf{0}$,
- the contact is sliding $\mathbf{v}_{ci} \times \mathbf{n} \neq \mathbf{0}$ and $\mathbf{f}_{ci} \neq \mathbf{0}$,

where $\mathbf{n}$ is the normal vector of contact. In order to be integrated in the LQP-based dynamic controller, Contact point constraints are expressed in terms of joint accelerations and forces constraints lying inside linearized cones. For instance, the two first cases are expressed as follows:

$$case\ 1\ \mathbf{v}_{ci} = \mathbf{0}\ :\quad \mathsf{J}_{ci}\ddot{\mathbf{q}} + \dot{\mathsf{J}}_{ci}\dot{\mathbf{q}} = \mathbf{0} \tag{5}$$

$$\mathsf{C}\mathbf{f}_{ci} \leq \mathbf{0} \tag{6}$$

$$case\ 2\ \mathbf{v}_{ci}.\mathbf{n} > \mathbf{0}\ :\ \mathsf{J}_{ci}\ddot{\mathbf{q}} + \dot{\mathsf{J}}_{ci}\dot{\mathbf{q}}.\mathbf{n} > \mathbf{0}. \tag{7}$$

In the same way, kinematic loop closure is described in the LQP as a constraint. Two points $i1$ and $i2$ from two kinematic chains are linked to form the kinematic loop, so $\mathbf{v}_{i1} = \mathbf{v}_{i2} \in \mathbb{R}^3$ is expressed in the controller as follows:

$$(\mathsf{J}_{i1} - \mathsf{J}_{i2})\ddot{\mathbf{q}} + (\dot{\mathsf{J}}_{i1} - \dot{\mathsf{J}}_{i2})\dot{\mathbf{q}} = \mathbf{0}. \tag{8}$$

3   Definition of Cost Function

The quadratic cost function of the LQP represents the objective that the controller may satisfy. Many objectives may lead to conflicting situations, so the selection of a strategy to cope with this issue is discussed in Sec. 4. Here, one objective has to be completed, and its related task is the set composed of the controlled part of the robot, the goal and the way to achieve it.

3.1 Task Definition

Each task i is associated to a functional element of the robot. It can be a joint, a frame linked to the robot bodies, the Center of Mass (CoM), etc. One of the main concern in the task definition is to find the appropriate Jacobian matrix J_i and its derivative (with regard to time) $\dot{\mathsf{J}}_i$ related to the task. Hence, the velocity $\mathbf{v}_i$ and acceleration $\dot{\mathbf{v}}_i$ of the task i can be computed as well:

$$\mathbf{v}_i = \mathsf{J}_i\dot{\mathbf{q}} \tag{9}$$
$$\dot{\mathbf{v}}_i = \mathsf{J}_i\ddot{\mathbf{q}} + \dot{\mathsf{J}}_i\dot{\mathbf{q}}. \tag{10}$$

Since the whole-body motion control is dynamic, at each time step a desired acceleration $\dot{\mathbf{v}}_i^{des}$ is computed beforehand, and the cost function of the task becomes $(\delta_i)^2 = \|\dot{\mathbf{v}}_i^{des} - \dot{\mathbf{v}}_i\|^2$.

3.2 Task Controller

A task controller computes the desired acceleration mentioned above. A simple one is the tracking of a predefined trajectory. If for the task i the desired pose, velocities and acceleration of reference over the time are respectively $\mathbf{pos}_i^{ref}$, $\mathbf{v}_i^{ref}$ and $\dot{\mathbf{v}}_i^{ref}$, its controller gives the desired acceleration as follows:

$$\dot{\mathbf{v}}_i^{des} = \dot{\mathbf{v}}_i^{ref} + \mathsf{K}_p\mathbf{err}_p + \mathsf{K}_d\mathbf{err}_d \tag{11}$$

where $\mathsf{K}_p, \mathsf{K}_d$ are respectively the stiffness and the damping of the trajectory tracking, and $\mathbf{err}_p, \mathbf{err}_d$ are the proportional and derivative errors. $\mathbf{err}_d$ is the difference between the reference and actual velocities $\mathbf{err}_d = \mathbf{v}_i^{ref} - \mathbf{v}_i$, and $\mathbf{err}_p$ is the difference between the reference and actual pose. Its computation relies on the controlled element and the nature of the movement. For example a joint with linear configuration (like a hinge), or the relative position of a point give a simple error $\mathbf{err}_p = \mathbf{pos}_i^{ref} - \mathbf{pos}_i$. If it is only a translation, the error is computed as an Euclidean distance, but if it incorporates a rotation it becomes more complicated. The authors use the imaginary part of quaternions to compute the proportional rotational error.

Another interesting way to achieve an objective is to use a preview control. It is exploited here to control the center of mass of the robot to generate a gait pattern using a preview control of the zero-moment point (ZMP) [15]. Indeed, the predictive controller returns an optimal path according to a given criterion along a predefined horizon. For example, the use of the inverted pendulum as the approximation of a humanoid explained in [6, 14] allows to predict the trajectory of the center of mass function to a control vector. The first component of this vector is used as the desired acceleration, and the solution is updated at each time step. This preview approach can be extended to other tasks such as compliant motions.

3.3 LQP-Based Controller

At each time step, the dynamic LQP-based controller solves the following problem:

$$\min_{(\ddot{\mathbf{q}}, \tau, \mathbf{f}_c)} ((\delta_i)^2 + \alpha_\tau (\tau)^2 + \alpha_{\mathbf{f}_c} (\mathbf{f}_c)^2) \tag{12}$$

$$s.t. : \qquad Eqs. \; 1 - 8 \tag{13}$$

where $\alpha_\tau, \alpha_{\mathbf{f}_c}$ are weights. The given torque vector is applied to the robot, and due to the minimization of its norm, the controller generates smooth motions. This method illustrated in Fig. 2 can be compared to optimal quadratic control, where there is a trade-off between the state-feedback and the control vector of the robot. Here the state of the robot is $\ddot{\mathbf{q}}$, the control vector regroups $\tau, \mathbf{f}_c$, and the ratio is given by α_τ and $\alpha_{\mathbf{f}_c}$.
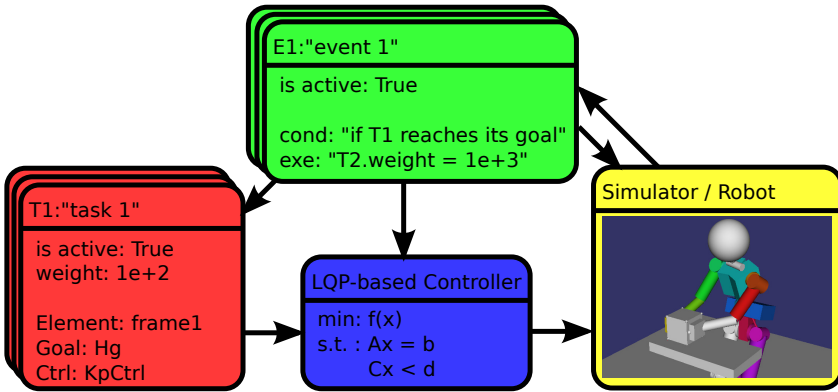
Fig. 2 Relationships between the different parts to perform the control of the humanoid robot

4 Importance, Hierarchy and Transition

One of the main difficulties for the implementation of coordination between several tasks on a system with high redundancy is the importance between the various tasks. Indeed, a strict hierarchy should be used when higher priority tasks need to be performed with no error, but a simple trade-off between weighted objectives may offer more flexibility.

4.1 Weighting of Tasks

The easiest way to perform several tasks into the LQP-based controller is to optimize a sum of weighted costs which replaces Eq. 12 by $(\Sigma_i(\alpha_i(\delta_i)^2) + \alpha_\tau(\tau)^2 + \alpha_{\mathbf{f}_c}(\mathbf{f}_c)^2)$

where α_i is the weight of task i, and $\Sigma_i \alpha_i = 1$ to keep the same ratio with the command described above. It is a good way to quickly realize several tasks at the same time with one LQP at each time step. As a task j with a bigger weight $\alpha_j \gg \alpha_i \forall i$ will be roughly fulfilled, this method allows to set up a sort of smooth hierarchy between tasks.

4.2 Task Hierarchy

A way to perform strict hierarchy is to solve many LQP one after another. Assume that tasks have been sorted by importance: the first level realizes task 1, the second level realizes tasks 2 and 3 according to their weighting coefficients, the third level realizes task 4, and so on. When one level is solved, the related solution is inserted as a new constraints in the next level. For example, assume the first level of the hierarchy is solved by the LQP in Sec. 3.3, hence the optimal solution of task 1 is δ_1^*, the related constraint is $\delta_1 = \delta_1^*$, and the second level becomes:

$$\min_{(\ddot{\mathbf{q}}, \tau, \mathbf{f}_c)} (\alpha_2 (\delta_2)^2 + \alpha_3 (\delta_3)^2 + \alpha_\tau (\tau)^2 + \alpha_{\mathbf{f}_c} (\mathbf{f}_c)^2)$$
$$s.t.: \qquad Eqs.\ 1-8$$
$$\delta_1 = \delta_1^*$$

which gives solutions for tasks 2 and 3 and provides new constraints to the next level.

4.3 Tasks Transition over Time

When performing the synthesis of complex tasks, the set of objectives, their relative importances, their hierarchy and their goals may change during time. A strategy must be set to ensure stable transitions and avoid peaks in the control.

For example, the humanoid robot has to grab a object on a table, it must bend to reach its objective, then it gets up to put the object elsewhere. The set of tasks is:

1. keep robot balance,
2. keep the back upright,
3. grab the object,
4. displace the object.

The balancing task has the most importance (or the highest priority) throughout the example, but grabbing the object is more important than keeping the back upright. When the object is caught, keeping the back upright becomes more important, and the object may finally be displaced. If a hierarchy is set as in Sec. 4.2, the task which keeps the back upright is not continuous because one of its constraint related to higher tasks suddenly vanishes and affects the control. If the importance between coefficients is considered, the transition is done by sliding smoothly their values to

their new relative importances, so the trade-offs change continuously over time. This example is described and validated in Sec. 5.2.

5   Experiments

The validation of the method is done through some experiments with the model of the iCub robot. The simulations has been carried out with arboris-python [3], a simulator developed in the ISIR-UPMC laboratory. It aims to perform dynamic simulations quickly and easily with tree-shaped structures under some physical constraints as frictional contacts, joint limits and ball joints. The LQP is solved with cvxopt [5], a convex optimization python library. The sampling time is set to $dt = 0.02\ s$ for all simulations.

5.1   Joint Limit Avoidance

Here, the robot reaches an objective in front of it with its left hand. The motion is constrained by joint limits, so the controller needs some anticipation to avoid hard braking. Figure 3 shows the displacement of the left arm.
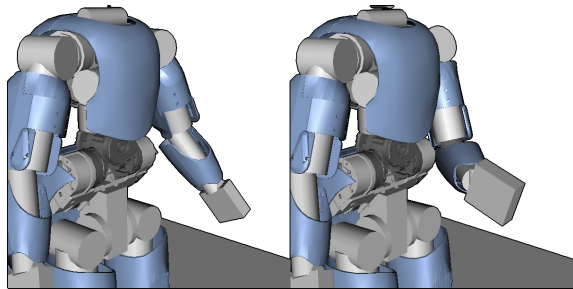

Fig. 3 The initial (left) and final (right) positions of the left hand

During this simulation, the pitch and roll joints of the shoulder approach their limits. The method presented in Eq. 3 bounds the joint acceleration on a predefined horizon. This approach reduces the sharp evolutions of the torques as shown in Fig. 4, where some peaks appears before $t = 0.5\ s$ due to a short horizon of prediction.


5.2   Box Displacement

This experiment is described above. The robot grabs a 3 kg box and displaces it from one place to another, as shown in Fig. 5.

Here, torques and velocities of the torso joints have been recorded and the results are given in Fig. 6. As the box is relatively heavy — the robot weighs about 20 kg
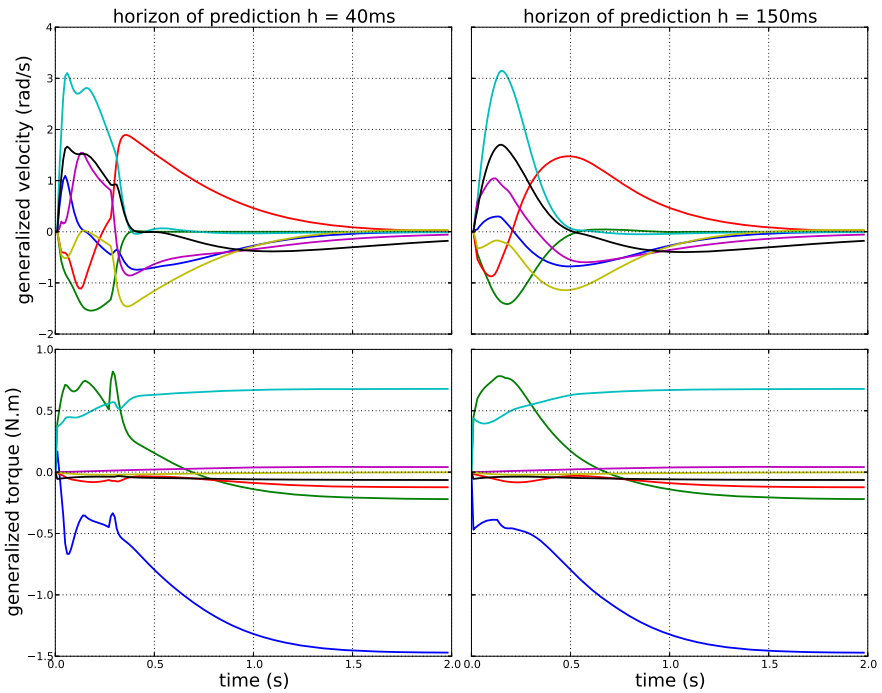
Fig. 4 Left arm behavior when subject to joint limit constraints for different horizons of prediction h
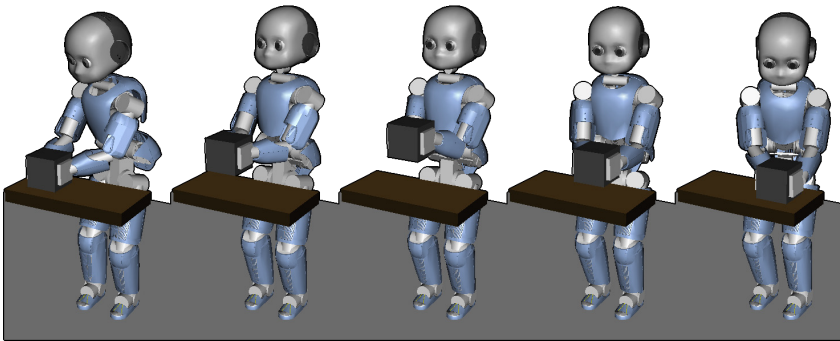
Fig. 5 Sequence of the robot grabbing the box

— it cannot be neglected in the dynamic behavior and the postural balance of the humanoid. At $t = 1.5\ s$ and $t = 4.8\ s$, there is two discontinuities in torques applied to the robot, when it lifts and drops the box. Elsewhere, the coefficient of the task which keeps the back upright evolves smoothly to make the grasping task easier, and no sudden change appears during the transition periods.
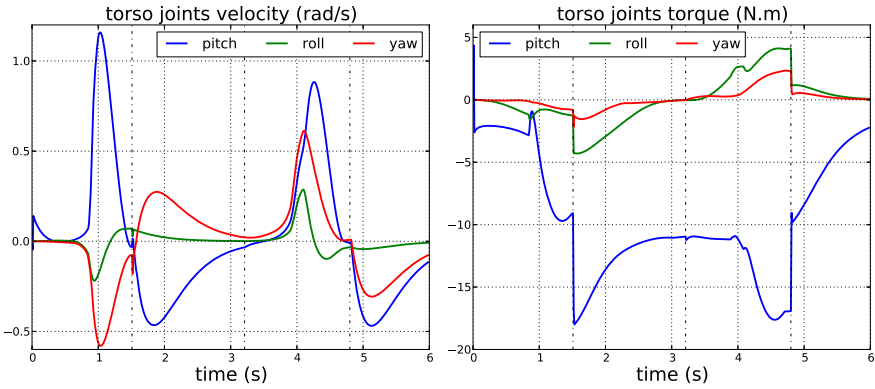
Fig. 6 Torso and velocity joints evolution over the time

5.3 Gait Pattern

As explain in Sec. 3.2, the approximation of the robot with an inverted pendulum gives an approximated ZMP, which can be controlled with this method. The trajectory of this point is defined a priori and the tasks which realize the steps are synchronized with its path.
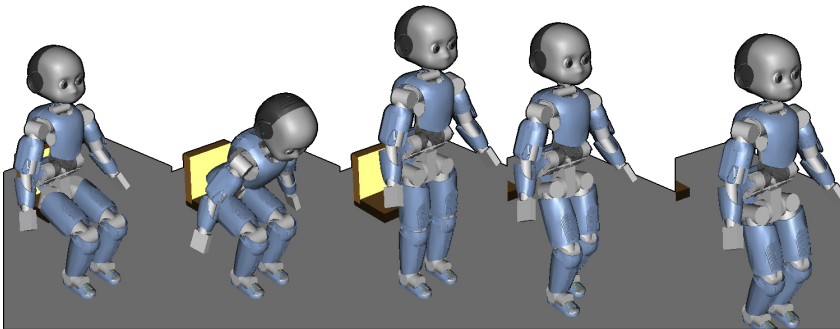

Fig. 7 Sequence of the walk

Here, the robot gets up from a seated position, initiates a gait pattern, walks $0.5\,m$ and stops. Figure 7 shows the snapshots of this sequence. The duration of one foot step is about $0.8\,s$ and the entire simulation is done in $8\,s$. To perform this simulation, all joint limits are actives and the horizon of prediction is set to $h = 0.15\,s$. To prevent the knees to bend in the wrong way, their limits have been reduced and their positions remain in an admissible set. Furthermore, the feet get closer during the walk, hence the CoM trajectory has less amplitude which minimizes the swing of the upper body.
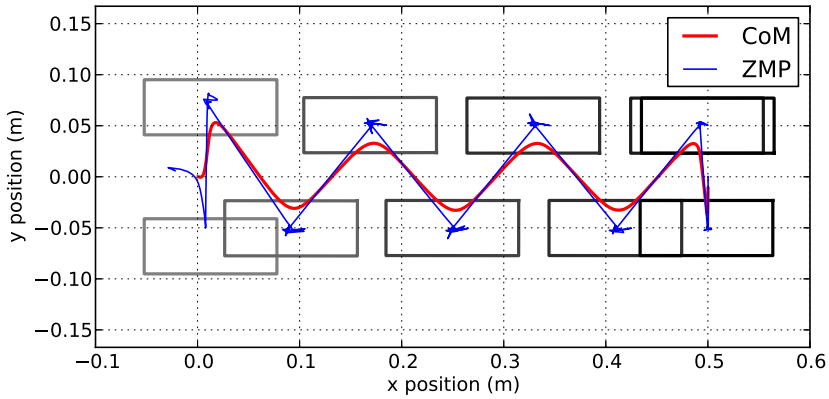
Fig. 8 CoM and ZMP trajectories, with the footprints

The trajectories of the CoM and ZMP from the initiation of the gait pattern to the end are shown in Fig. 8.

6   Conclusions

This controller allows to perform the synthesis of several dynamic tasks on a humanoid robot while interacting with its environment. The robot is subjected to both internal and external constraints, which are described as linear constraints on a Linear Quadratic Program. The tasks are described as a quadratic weighted-sum objective, and the solution of the problem is a general trade-off. The smooth evolution of the weights over the time prevents sharp evolutions of torques applied to the robot. In the future, the authors would like to take into account joint acceleration limits in the solution of the problem. It implies new constraints and some checks should be done to ensure that they are not overabundant with the joint position and velocity limits.

References

1. Abe, Y., da Silva, M., Popovic, J.: Multiobjective Control with Frictional Contacts. In: Symposium on Computer Animation (SCA) (2007)
2. Baerlocher, P., Boulic, R.: An inverse kinematics architecture enforcing an arbitrary number of strict priority levels. Visual Computer 20(6), 402–417 (2004)
3. Barthélemy, S., Salini, J., Micaelli, A.: Arboris-python,
   https://github.com/salini/arboris-pyhton

4. Collette, C., Micaelli, A., Andriot, C., Lemerle, P.: Dynamic Balance Control of Humanoids for Multiple Grasps and non Coplanar Frictional Contacts. In: Humanoids 2007 (2007)
5. Dahl, J., Vandenberghe, L.: Cvxopt - python software for convex optimization, http://abel.ee.ucla.edu/cvxopt/
6. Kajita, S., Kanehiro, F., Kaneko, K., Fujiwara, K., Harada, K., Yokoi, K., Hirukawa, H.: Biped walking pattern generation by using preview control of zero-moment point. In: Proceedings of the 2003 IEEE International Conference on Robotics and Automation Taipei, Taiwan (2003)
7. Kanoun, O.: Contribution à la planification de mouvement pour robots humanoïdes. Ph.D. thesis, Université Toulouse III (2009)
8. Metta, G., Sandini, G., Vernon, D., Natale, L., Nori, F.: The iCub humanoid robot: an open platform for research in embodied cognition. In: Permis: Performance Metrics for Intelligent Systems Workshop, Washington DC, USA (2008)
9. Padois, V.: Enchaînements dynamiques de tâches pour des manipulateurs mobiles à roues. Ph.D. thesis, Institut National Polytechnique, Toulouse, France (2005)
10. Park, J.: Control strategies for robots in contact. Ph.D. thesis, Stanford University (2006)
11. Sentis, L.: Synthesis and control of whole-body behaviors in humanoid systems. Ph.D. thesis, Stanford University (2007)
12. Siciliano, B., Slotine, J.-J.: A general framework for managing multiple tasks in highly redundant robotic systems. In: ICAR 1991, vol. 2, pp. 1211–1215 (1991)
13. da Silva, M., Abe, Y., Popovic, J.: Simulation of human motion data using short-horizon model-predictive control. In: Eurographics (2008)
14. Wieber, P.B.: Trajectory free linear model predictive control for stable walking in the presence of strong perturbations. In: IEEE-RAS International Conference on Humanoid Robots, Genova, Italy (2006)
15. Sardain, P., Bessonnet, G.: Forces Acting on a Biped Robot. Center of Pressure - Zero Moment Point. IEEE Transactions on Systems, Man, and Cybernetics, Part A 34(5), 630–637 (2004)

Walking and Running: How Leg Compliance Shapes the Way We Move

Andre Seyfarth, Susanne Lipfert, Jürgen Rummel,
Moritz Maus, and Daniel Maykranz

Abstract. The function of the human leg during walking and running is complex. One issue is the segmented structure of the leg, which consists of thigh, shank and foot. The situation is further challenged by the parallel arrangement of muscles spanning a single or multiple leg joints. How is the leg function organized to make typical movements such as walking and running possible and easily accessible? In this paper, we review a number of biomechanical models based on the spring-mass model, which may help to better understand how compliant leg function can be used and properly adjusted to selected movement tasks. This includes the emergence and stabilization of walking and running patterns. One general characteristic of movements based on compliant leg function is the functional redundancy in the leg adjustment, i.e. at a given speed, walking or running can be achieved with different leg strategies. This principle of redundant leg adjustments fulfilling the same general goal of movement is a key for understanding the organization of human locomotion.

1 Dynamics of Running and Walking

The leg force generated during the stance phase of running is remarkably similar to that of a linear telescopic spring. This behavior originates from the contribution of elastic energy recoil to the positive work during the stretch-shortening cycle of the

Andre Seyfarth
Lauflabor Locomotion Laboratory, Bereich Sportbiomechanik,
Institut für Sportwissenschaft, Technische Universität Darmstadt,
Magdalenenstr. 27, D-64289 Darmstadt
e-mail: seyfarth@sport.tu-darmstadt.de

Susanne Lipfert · Jürgen Rummel · Moritz Maus · Daniel Maykranz
Lauflabor Locomotion Laboratory
e-mail: {s.lipf,juergen.rummel,daniel.maykranz}@uni-jena.de,
    mmaus@sport.tu-darmstadt.de

muscle-tendon units [4]. When plotting the leg force F_{leg} against the leg length l_{leg}, defined as the distance from the leg's point of force application on the ground to the center of mass (COM), this results in an almost linear relationship with maximum leg force being achieved at maximum leg compression (Fig. 1, [11]). The slope of the force-length curve represents the leg stiffness k_{leg}. This observation of a linear elastic leg function motivates the spring-mass model [1, 17] as a conceptual framework to investigate the dynamics of running.

With such a simple mechanical model periodic trajectories of the COM and patterns of the leg force $F_{leg}(t)$ similar to those found during legged locomotion can be predicted. For instance, when selecting a certain leg stiffness k_{leg}, appropriate leg angles of attack α_0 can be identified to result in a steady state running pattern. Some of these periodic solutions are passively stable [21], i.e. they are approached from adjacent COM trajectories (e.g. trajectories with different apex heights) without changing the leg parameters. This attractive property of spring-mass running can be described as 'self-stability', as the fixed point is stable only due to the mechanical nature of the system without requiring any additional control effort to return to the periodic running pattern.

Fig. 1 Mean leg force - leg length $F_{leg}(l_{leg})$ tracings during human walking and running averaged for 21 subjects on an instrumented treadmill at 1.55 m/s [11]. The progression of time is indicated by arrows. In both gaits, maximum leg forces are found at maximum leg compressions (smallest leg lengths). Leg length at touch-down (TD) is shorter than at take-off (TO). Index 'c' indicates the contralateral leg, BW = body weight, l_0 is the height of the COM with respect to the ground during standing upright. Adapted from Lipfert SW. Kinematic and dynamic similarities between walking and running. Verlag Dr. Kovac 2010. Used with permission.
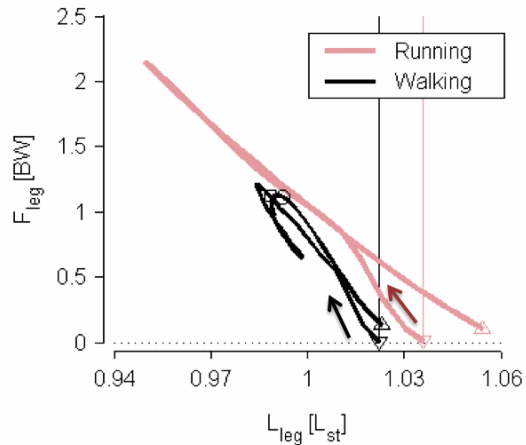
In human walking, leg function seems to be fundamentally different as compared to running. At about mid-stance of walking, the knee joint is almost straight in contrast to maximum knee flexion in running. The opposite is true for the instant

of take-off. Here the knee joint is clearly bent in walking and almost straight in running. Surprisingly, at speeds close to the preferred walking speed the similarity of walking and running on the global leg level may be larger than expected as indicated by the relation between leg force and leg length (Fig. 1).

For a linear leg spring, we would expect an increase in leg force F_{leg} proportional to the amount of leg compression during stance phase. Indeed, for the speed shown in Fig. 1 this is the case for running (red graph) and walking (black graph). At the beginning of contact phase, the slope of the force-length graphs is steeper than during the remaining contact. In walking, the situation is slightly different as compared to running. There are two maxima in leg force and two minima in leg length. However, the force-length dependency in walking at 1.55 m/s is remarkably similar to the one in running (Fig. 1).

These experimental findings support the previously proposed extension of the spring-mass model with a second leg spring [6]. This bipedal spring-mass model is capable of predicting both walking and running motions with similar COM trajectories and GRF profiles as found experimentally [11]. For fast speeds, running gaits with single humped GRF patterns are predicted by the model, similar to predictions from the original spring-mass model [21]. At lower speeds, stable walking patterns are predicted, which are characterized by double support and single support phases as in human gait. Not only does the model qualitatively predict the kinematics of the COM, for the first time it also predicts the double-humped GRF patterns in human walking [6]. However, leg stiffness in human walking and running is not completely constant (Fig. 1) and there are systematic deviations between COM trajectories predicted by the model compared to experimental data. This indicates that a more detailed analysis and representation of human leg function than provided by this model is required.

It is important to note that the bipedal spring-mass model (as well as the underlying spring-mass model) fails in describing the shift of the leg's point of force application during stance phase of walking and heel-toe running. This deficit leads to an overestimation of vertical excursions of the COM in comparison to experimental data. By introducing the forward shift of the point of force application in the spring-mass model [2], the vertical impulse and the mechanical work on the COM observed during stance phase of human running can be described more realistically [3].

Another striking difference between the concept of a simple leg spring and the experimentally observed force-length curves (Fig. 1) is that the experimentally observed leg length is shorter at touch-down as compared to take-off. This difference in leg length is increasing with speed (Fig. 2). Within the speed range $(0.5 - 2.6\,m/s)$ where both walking and running can be observed, the force-length curve remains spring-like in running with maximum leg force occurring at the instant of maximum leg compression. Except for the initial landing phase, the slope of the force-length curve remains quite constant indicating a spring-like leg behavior in running. The leg force-leg length function in walking, however, appears to be more complex. Here, the initial loading of the leg occurs at a higher leg stiffness than unloading before take-off. At speeds higher than 75% PTS, the second loading phase of the leg
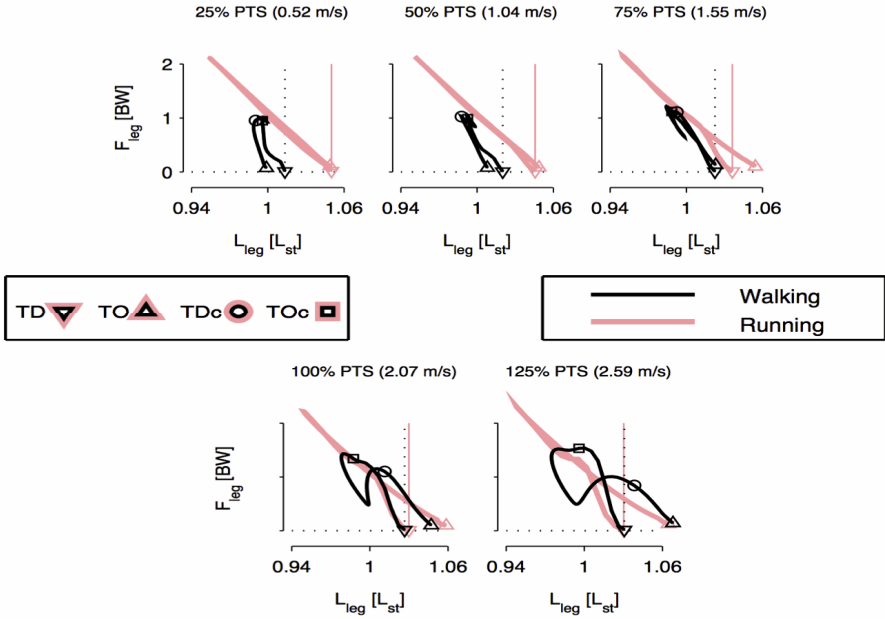
Fig. 2 Leg force-leg length tracings measured on an instrumented treadmill at different walking and running speeds ($0.52 - 2.59$ m/s) corresponding to 25% $PTS - 125\%$ PTS. 100% PTS = individual preferred transition speed between walking an running [11]. Instances of touch-down and take-off are indicated as TD and TO. Index 'c' indicates the contralateral leg. Adapted from Lipfert SW. Kinematic and dynamic similarities between walking and running. Verlag Dr. Kovac 2010. Used with permission.

(during mid-stance) occurs even during leg lengthening. This cannot be explained by a simple spring-like leg function as assumed in the spring-mass model.

In section 3 we will give an example of how the geometry of the segmented human leg and especially the function of an elastically coupled foot segment can contribute to a change in the overall force-length relation of the leg during stance phase. But before enhancing the complexity of the leg model we would like to discuss to what extent a model with two spring-like legs can select between walking and running gait patterns. In particular, we are interested to see whether the regions of walking and running are separate in the space of leg parameters or whether continuous transitions between both gaits are possible. For this, we will analyze in the next section the predicted solutions of the bipedal spring-mass model at system energies[1], where both gaits can naturally occur.

---

[1] With 'system energy' we mean the mechanical energy of the system, i.e. the sum of kinetic and potential energies including the energy stored in elastic structures.

2   Are Walking and Running Discrete Gaits?

In the simulation work of Geyer et al. [6] it was shown that both walking and running can be predicted by the bipedal spring-mass model when the leg stiffness k_{LEG} and the angle of attack α_0 are adjusted appropriately to match the system energy. Between the regions of stable walking and running, a speed gap was observed indicating that a certain amount of energy needs to be injected to change from walking to running. In the study of Rummel et al. [20] we asked whether a transition between both gaits could also be predicted for the same system energy. More specifically, we analyzed the required changes in the leg adjustment and the COM patterns to make the transition. To simplify the analysis of the bipedal spring-mass model we selected a constant leg stiffness ($k = 15\ kN/m$), a rest length of the leg ($l_0 = 1\ m$) and different system energies spanning the range of human walking and running.

To map walking and running solutions, the COM height and the orientation of the vertical COM velocity at the instant of vertical leg orientation (VLO) during mid-stance were used. As the bipedal spring-mass model only describes gaits with constant system energy, it was sufficient to proof that the VLO height y_0 and the vertical COM velocity v_{y0} are the same for two subsequent VLO's to identify a periodic solution.

The search for periodic solutions of the bipedal spring-mass model (Fig. 3) reveals a large region of walking patterns for VLO heights y_0 larger than the landing height $y_{TD} = l_0 \sin\alpha_0$ (rest length of the leg l_0, angle of attack α_0) and running patterns with $y_0 < y_{TD}$. Within a small range of touch-down angles (angle of attack around $\alpha_0 = 70\ deg$) and VLO heights y_0 above 0.9 m, there is a small region of running solutions with single humped force patterns but no flight phases (see example with corresponding COM trajectory and GRF pattern in lower panel in Fig. 3). This new gait pattern can be described as grounded running and bridges the gap between walking and running (with flight phases). Within the regions of these three gait patterns (walking, grounded running and running) it is possible to continuously change from walking to running for a given system energy (e.g. thick black line in upper panel of Fig. 3 indicates $E = 820\ J$) with appropriately selected angles of attack.

Another finding of this study was the predicted shift in the angle of attack α_0 with speed (system energy E). For running, the angle of attack is shifted towards flatter angles with increasing speed. This is different to walking. Here, two independent walking solutions exist, which move in different directions with increasing energy as indicated in the upper panel of Fig. 3. For a given angle of attack, different periodic walking patterns can exist at different VLO heights. Alternatively, for a given VLO height, walking solutions can be found for different angles of attack. Walking patterns with VLO heights y_0 close to the touch-down height y_{TD} are characterized by trapezoid single humped force patterns with flat plateaus (for details see [20]).

In summary, we found that within the framework of the bipedal spring-mass model, walking and running solutions can be found in large ranges of angles of attack α_0 and VLO heights y_0. Walking and running patterns are separated by the line $y_0 = y_{TD}$. This means that for spring-like legs, walking can be found when the
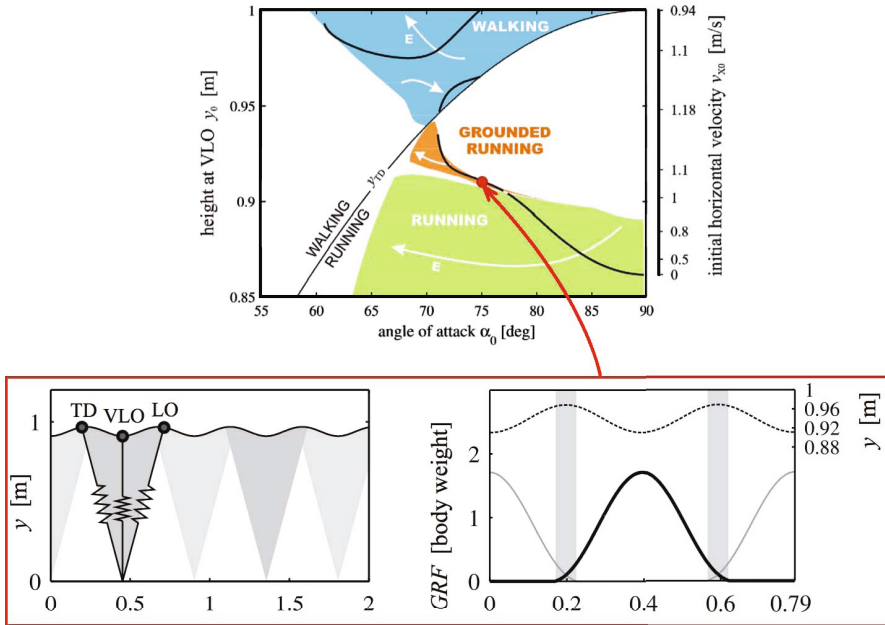
Fig. 3 Upper panel: Regions of walking, running and grounded running depending on the angle of attack α_0 and COM height y_0 at vertical leg orientation (VLO) during mid-stance. The right scale of initial horizontal velocities v_{x0} (at VLO) corresponds to a selected system energy of $E = 820\ J$, denoted by thick black lines. Lower panel: COM trajectory and GRF pattern of a selected 'grounded running' solution (red dot in the upper panel, around $\alpha_0 = 75\ deg$) [20].

COM position is lifted with respect to the landing condition ($y_0 > y_{TD}$). In contrast, running requires a lowering of the COM during mid-stance with respect to touch-down. Within a small range of angles of attack (α_0 around 70 deg, Fig. 3), however, both walking and running solutions approach each other at the separating line $y_0 = y_{TD}$. Hence, at moderate speeds (e.g. around 1.1 m/s for $k_{LEG} = 15\ kN/m$) a smooth transition from walking via grounded running to running with flight phases could be performed with spring-like legs even without changing the system energy. It remains for further studies to investigate, whether such transitions can also be observed experimentally. Here, deviations from a linear spring-like leg function (as in fast human walking, Fig. 2) or other physiological constraints (e.g. internal work required to swing the leg forward before next touch-down) might play an important role.

The transition between walking and running has attracted the interest of many researchers within the last years. Here, the function of the foot and the foot extensor muscles [12] is considered to be crucial to explain the walk-run transition (WRT). With the help of detailed computer simulation models, a dramatic increase in plantar flexor forces was predicted when the WRT was performed [18]. This could be due to the more synchronized function of knee and ankle joint in running as compared

to walking [11]. In the following section we will investigate how the geometry of the foot could affect the overall function of a spring-like operating leg. For this, a rigid foot segment will be coupled to the leg by a rotational spring representing the function of muscles spanning the ankle joint.

3   Effects of the Foot on the Global Leg Function

During walking and running[2], the point of force application on the ground [2] is shifted from heel to toe. As a result, the leg forces do not intersect in a fixed foot point as assumed in the spring-mass model. In order to represent this shift of the center of pressure during ground contact as a natural effect of the roll-over function of the foot, we introduce a rigid foot segment (length l_f) attached to the linear leg
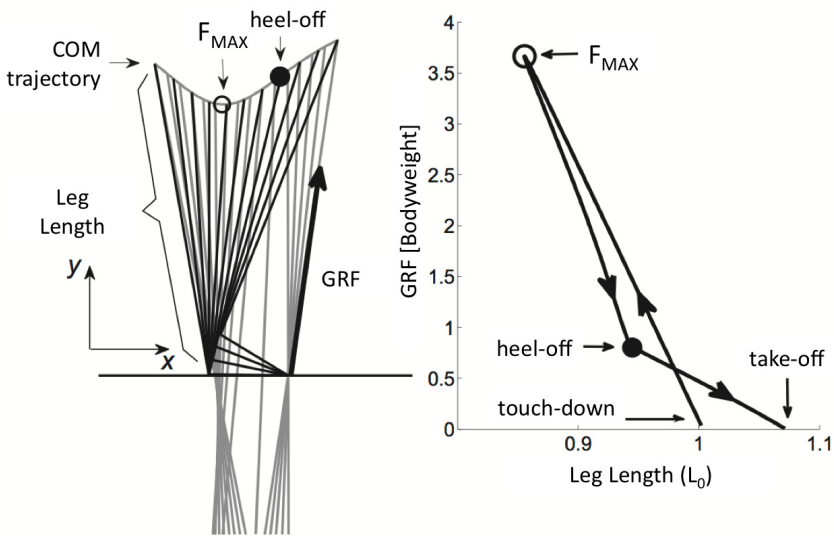
Fig. 4 Left panel: Contact phase of a spring-mass model extended with an elastically coupled rigid foot segment [16]. The compression and extension of the elastic leg (prismatic spring) and the elastic foot joint (rotational spring) may occur at different time scales. As a result, maximum leg force F_{MAX} and heel-off do not occur simultaneously. Gray lines indicate the alignment of GRF with a forward shift of the point of force application on the ground. Right panel: During leg loading, there is a steady rise in leg force with a parallel loading of both springs. Between maximum leg force F_{MAX} and heel-off, the leg is very stiff. Here both springs are counter-acting resulting in a rather constant leg length. After heel-off, both springs are unloading and the leg becomes more compliant. Leg length is defined as the distance from center of pressure to center of mass.

---

[2] Except for fore-foot running.

spring of the spring-mass model [16]. The action of the muscles spanning the ankle joint is represented by a rotational spring (joint stiffness c, rest angle ϕ_0). There are several studies indicating an elastic function of the ankle joint both in walking [8, 11] and running [7, 11].

The combination of a linear leg spring and a rotational foot spring (Fig. 4) can lead to a modulation of the overall leg force-leg length relation during stance phase (right panel). This may result in a higher leg stiffness during leg compression compared to leg extension. Additionally, the leg length at take-off becomes larger than at touch-down. These two effects are also indicated in the force-length curves of the human leg during walking and running (Figs. 1, 2).

These simulation results indicate that a spring-like joint function does not necessarily lead to a spring-like leg function. The asymmetry of the leg with the foot pointing forward and initial heel contact leads to higher leg stiffness during leg loading as compared to leg extension. In recent studies on human running the observation that the leg is more stiff in landing than during take-off [1] was attributed to the different responses of the muscle-tendon units to stretching and shortening [5]. Here, we demonstrated that this landing-take-off asymmetry could also exist even when the muscle-tendon units would operate fully elastically. To better understand the contribution of joint function to the global leg function during locomotion, a more detailed representation of the segmented structure of the human body is required. In the following sections, we will separately focus on the two more proximal joints, namely the knee and the hip joint. The effects of the knee joint within the segmented leg will be discussed in section 4 to reveal its potential function more specifically.

4   Leg Segmentation with Upper and Lower Leg

Maybe the most striking structure of the human leg is the segmentation with thigh and shank. Within such a two-segmented leg, spring-like leg function can be attributed to a rotational stiffness of the knee joint [19]. The segmentation of the leg offers variable gearing between joint function and leg function by adjusting the nominal joint angle (i.e. the rest angle of the rotational spring). Joint stiffness behavior can be adapted to speed and to the amount of joint flexion. This is used in human locomotion and may help to guarantee gait stability at different speeds and gaits.

Elastic knee joint function can be found in human walking [11] and running [7]. In running, the rotational stiffness of the knee increases with running speed [10] in contrast to ankle stiffness, which remains rather constant. Assuming a linear knee joint stiffness for running with a two-segmented leg, the predicted minimum speed required for stable running [21] can be clearly reduced. Also, the robustness of running with respect to changes in the angle of attack is largely enhanced. This comes at the cost of a reduced tolerance in speed, which must now be compensated by joint stiffening. Such an adaptation of knee stiffness with running speed is in agreement with experimental observations [19]. This adaptation of joint stiffness may

compensate for the changed leg geometry (larger joint flexion with higher speed) and by this contribute to an approximately constant leg stiffness at different running speeds [11].

Translating spring-like leg function into joint function offers the opportunity to adapt joint stiffness and nominal joint angles with speed. By this the stability and robustness of running can be increased further potentially reducing the required control effort during locomotion. Knee flexion not only contributes to changes in leg length but also effects the amount of ankle joint flexion when the foot is flat on the ground. This kinematic coupling between knee and ankle joint within the segmented leg is lost after heel-off and may be replaced by a coupling through the bi-articular gastrocnemius muscle spanning the knee and ankle joint. In this muscle, different strategies are found between walking and running, with almost constant muscle fibre lengths in walking and continuous fibre shortening in running [9]. With this coupling of knee and ankle joint, the function of the foot in shaping the global leg function as described in section 3 will be further pronounced. A more detailed representation of the human leg with three segments (foot-shank-thigh, [22]) is required to better describe the gait-specific interplay of elastically operating knee and ankle joints during dynamic loading and unloading.

5 Postural Control Based on Compliant Leg Function

In the last two sections, we focused on the role of the foot and leg segmentation on gait and gait stability. Similar as in the spring-mass model, the supported body was reduced to a point mass. In this section we aim at extending the model by an upper body in order to study mechanisms for stabilizing the upright trunk posture in human locomotion. Similar to [13], we use a rigid segment as a replacement of the point mass in order to describe walking and running [14, 15]. Without any additional means, the upper body cannot be stabilized in an upright posture, as the COM is located above the hip joint. In order to align the trunk upright during locomotion, hip torques acting between the telescopic leg spring and the trunk are required. A typical approach is to measure the orientation of the trunk with respect to the vertical axis. This is not aimed here. Instead, we ask whether there is a simple strategy to stabilize the trunk without any external information (e.g. direction of gravitation as a reference to align the trunk).

One mechanically favorable solution would be that the hip joint is located above the COM. In this configuration, the COM would have a naturally stable equilibrium position[3]. Here, we aim at mimicking this situation by placing a virtual pivot point (VPP) instead of shifting the location of the hip. The idea is to redirect the leg forces from the original leg axis (represented by a telescopic leg spring attached to the trunk at the hip joint) to the VPP by applying appropriate hip torques (Fig. 5). To some extent, this mimics the situation in which the leg spring would be attached

---

[3] This is similar to a pendulum, which would asymptotically converge to this configuration in the case of some damping.
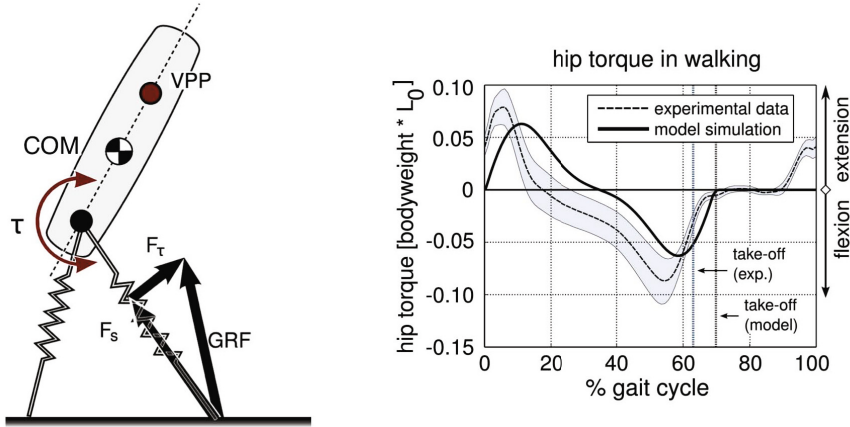
Fig. 5 Left panel: Extended spring-mass model with a rigid upper body. The leg spring is attached at the hip, which is located below the center of mass (COM). The force of the leg spring is deviated by hip torque τ to point towards the virtual pivot point (VPP), located above the COM. Right panel: By applying this simple hip torque control [14], stable walking can be predicted with hip torque patterns similar to those found in human walking [15].

to VPP instead of the hip joint. The required hip torque is calculated based on the spring force and the inner hip angle. Introducing this hip torque results in a non-conservative model, as the total system energy is not necessarily constant any more. By implementing this simple concept, it is possible to predict both walking and running with a stabilized upright trunk posture. For this, the VPP needs to be located in a certain range above the center of mass. The predicted torque patterns are in good agreement with experimental hip torque data on human walking (right panel in Fig. 5).

The presented gait model with an upright trunk demonstrates that the bipedal spring-mass model is not restricted to simplifying body representations such as a point mass. The previously identified gait dynamics for walking and running can be preserved while adding additional functionality such as an upright trunk posture to the model. For walking, the trunk can even be used as a reference frame to align the leg angle in preparation of touch-down resulting in stable gait patterns. This further reduces the need of global sensory information and may facilitate the transfer of these conceptual gait models to technical systems (e.g. legged robots, prostheses).

With this presentation of the VPP concept as a strategy to achieve an upright trunk posture during walking and running we conclude our review of conceptional models for describing human walking and running. So far, these models are restricted to selected features of the human body (leg segmentation, upright trunk) taking the overall spring-like leg function into account. At all model levels, it was possible to generate COM movements and GRF patterns which resemble those observed in human walking and running. However, it must be stated that the models are still far from describing the detailed dynamics of human gait. The interplay between

the different structural levels of the segmented human body during locomotion is still only poorly understood. We hope that by carefully increasing the model complexity we can finally achieve more realistic representations of the dynamics of the human body, which can help us to understand the underlying mechanisms of legged locomotion.

References

1. Blickhan, R.: The spring-mass model for running and hopping. J. Biomech. 22, 1217–1227 (1989)
2. Bullimore, S.R., Burn, J.F.: Consequences of forward translation of the point of force application for the mechanics of running. J. Theor. Biol. 238, 211–219 (2006)
3. Bullimore, S.R., Burn, J.F.: Ability of the planar spring-mass model to predict mechanical parameters in running humans. J. Theor. Biol. 248, 686–695 (2007)
4. Cavagna, G.A., Saibene, F.P., Margaria, R.: Mechanical work in running. J. Appl. Physiol. 19, 249–256 (1964)
5. Cavagna, G.A.: The landing-take-off asymmetry in human running. J. Exp. Biol. 209, 4051–4060 (2006)
6. Geyer, H., Seyfarth, A., Blickhan, R.: Compliant leg behaviour explains basic dynamics of walking and running. Proc. R. Soc. B. 273(1603), 2861–2867 (2006)
7. Günther, M., Blickhan, R.: Joint stiffness of the ankle and the knee in running. J. Biomech. 35, 1459–1474 (2002)
8. Hansen, A.H., Childress, D.S., Miff, S.C., Gard, S.A., Mesplay, K.P.: The human ankle during walking: implications for design of biomimetic ankle prostheses. J. Biomech. 37, 1467–1474 (2004)
9. Ishikawa, M., Komi, P.V.: The role of the stretch reflex in the gastrocnemius muscle during human locomotion at various speeds. J. Appl. Physiol. 103, 1030–1036 (2007)
10. Kuitunen, S., Komi, P.V., Kyröläinen, H.: Knee and ankle joint stiffness in sprint running. Medicine & Science in Sports & Exercise 34, 166–173 (2002)
11. Lipfert, S.W.: Kinematic and dynamic similarities between walking and running. Verlag Dr. Kovac (2010)
12. Malcolm, P., Fiers, P., Segers, V., Van Caekenberghe, I., Lenoir, M., De Clercq, D.: Experimental study on the role of the ankle push off in the walk-to-run transition by means of a powered ankle-foot-exoskeleton. Gait & Posture 30, 322–327 (2009)
13. Poulakakis, I., Grizzle, J.: Formal embedding of the spring loaded inverted pendulum in an asymmetric hopper. In: Proc. of the European Control Conference, Kos, Greece (2007)
14. Maus, H.M., Rummel, J., Blum, Y., Seyfarth, A.: Stable upright walking and running using a simple pendulum based control scheme. In: Marques, L., Almeida, A., Tokhi, M.O., Virk, G.S. (eds.) Advances in Mobile Robotics: Proc. of 11th CLAWAR, pp. 623–629. World Scientific (2008)

15. Maus, H.M., Lipfert, S.W., Gross, M., Rummel, J., Seyfarth, A.: Upright human gait did not provide a major mechanical challenge for our ancestors. Nature Communications (2010), doi:10.1038/ncomms1073
16. Maykranz, D., Grimmer, S., Lipfert, S.W., Seyfarth, A.: Foot function in spring mass running. In: Dillmann, R., Beyerer, J., Stiller, C., Zöllner, J.M., Gindele, T. (eds.) Autonome Mobile Systeme 2009, pp. 81–88. Springer (2009)
17. McMahon, T.A., Cheng, G.C.: The mechanics of running: how does stiffness couple with speed? Journal of Biomechanics 23(1), 65–78 (1990)
18. Neptune, R.R., Sasaki, K.: Ankle plantar flexor force production is an important determinant of the preferred walk-to-run transition speed. J. Exp. Biol. 208, 799–808 (2005)
19. Rummel, J., Seyfarth, A.: Stable running with segmented legs. Int. J. Robot. Res. 27(8), 919–934 (2008)
20. Rummel, J., Blum, Y., Seyfarth, A.: From walking to running. In: Dillmann, R., Beyerer, J., Stiller, C., Zöllner, J.M., Gindele, T. (eds.) Autonome Mobile Systeme 2009, pp. 89–96. Springer (2009); Broy, M., Dener, E. (eds.) Software Pioneers, pp. 10–13. Springer, Heidelberg (2002)
21. Seyfarth, A., Geyer, H., Günther, M., Blickhan, R.: A movement criterion for running. J. Biomech. 35, 649–655 (2002)
22. Seyfarth, A., Günther, M., Blickhan, R.: Stable operation of an elastic three-segment leg. Biological Cybernetics 84, 365–382 (2001)

Modeling and Simulation of Walking with a Mobile Gait Rehabilitation System Using Markerless Motion Data

S. Slavnić, A. Leu, D. Ristić-Durrant, and A. Graeser

Abstract. Research and development of gait rehabilitation systems and devices such as orthosis, prosthesis and wearable robots are complex processes in which simulation techniques are exploited in order to accelerate development process, reduce development costs, optimize the proposed solution, analyse the interaction between the system and human, etc. The modelling and simulation results can give valuable insights in the functionality of the system and directions for optimization and improvement of the researched system. Within the frame of the RoboWalker project a concept of a mobile robotic gait rehabilitation system, which will improve gait rehabilitation through several novel system features, was investigated. The system consists of a mobile platform with integrated active exoskeleton. In this paper, the modelling and simulation approaches utilized in designing and analysing the concept of mobile gait rehabilitation system are presented together with a novel markerless motion capture system that was used for collecting human motion data for simulation purposes.

1 Introduction

Models of human body of different complexity have been used in different fields such as humanoid robotics, biomechanics of human walking, gait rehabilitation and computer animation to model and simulate biped's walking. The complexity of a model depends of its intended use. The simplest model that can represent some locomotion activities is the planar inverted pendulum model (single-mass model) [5]. This model consists of mass-less legs and concentrated mass point that represents the humanoid 's upper body. The further extension of this model was done by including legs with variable lengths that models the knee functionality [7], [8] or by

S. Slavnić, A. Leu, D. Ristić-Durrant, A. Graeser
Institute of Automation, University of Bremen, Germany
e-mail: {slavnic,leu,ristic,ag}@iat.uni-bremen.de

extending the planar model into spatial (3D) model [6]. The single-mass model, due to its simplicity, is mainly used for analysing dynamic walking of bipeds [5], real-time robot control [8], and real-time motion generation [20]. However, a number of locomotion effects such as posture stability and control, knee functionality, joint forces and moments, dynamic stability and different types of walking could not be analysed and investigated using simple models such as single-mass models. Hence, more complex and advanced models have been deployed. First, the planar models of different complexity (consisting of three, five, seven or nine links) have been used mainly to study standing, walking, running, jumping. Further, very detailed and complex multi-body mechanicals models with large number of DoF (Degrees of Freedom) has been used in humanoid robotics for walking synthesis and control development. In the field of biomechanics, very detailed musculoskeletal models of the human body have been developed [12], [4] for the usage for modelling and simulation of human walking. High computing power of today's computers enables effective usage of these models, through simulation software such as SIMM [19], AnyBody [1], OpenSim [13] and LifeMod [9]. In the work presented in this paper the focus is on modelling and simulation of walking of human models (models of patients) inside the robotic gait rehabilitation system. The developed models and simulation results should give valuable information on improvement and optimization of the mechanical construction of the system, interaction forces/moments between user and machine, dynamic stability of the system and other. Also, the same developed models should be appropriate for research and development of control algorithms and approaches for gait rehabilitation systems. The complexity of the components of the system to be simulated demanded combination of the software tolls as explained in Section 2. This combination, that is simulation environment developed as optimal to model all components of the considered complex system, represents a novelty in rehabilitation robotics. The paper is organized as follows. Sections 2, 3 and 4 present respectively simulation environment, a concept of mobile gait rehabilitation system and mechanical models of the human body and the rehabilitation system used in simulations. Section 5 describes the markerless motion capture system used for collecting human motion data for simulations. Finally, the Section 6 presents simulations results obtained. The work is summarized and conclusion is given in the Section 7.

2   Simulation Environment

Figure 1 shows simulation tools used, and illustrates simulation process setup. MSC Adams [11], LifeMod [9] and MATLAB [10] were used for simulation of human walking inside the rehabilitation system consisting of mobile platform and an exoskeleton. Mechanical models of human body and the rehabilitation system were built using simulation software MSC Adams. For this, the mechanical CAD construction of the mobile platform and the exoskeleton are imported in MSC Adams providing geometrical and mass properties of the system parts. Ground-tire contacts
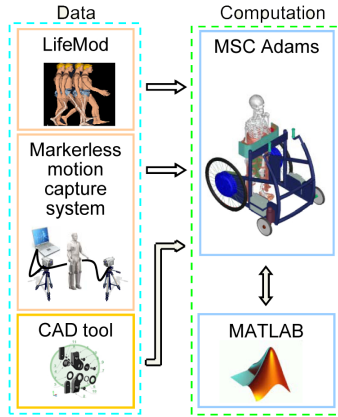
Fig. 1 Software tools and simulation environment

are modelled using MSC Tire module. Human motion data for "driving" the simulation model were provided by the markerless motion capture system that will be described in Section 5. Mechanical model of the human body was initially generated using the LifeMod software database and later was refined so that walking of particular subject, for which motion data are captured, can be simulated.

3 Mobile Gait Rehabilitation System

Figure 2 shows the proposed gait rehabilitation system that consists of a mobile platform with wheels and a powered exoskeleton attached to mobile platform through two revolute joints in frontal plane as marked with (A) on the image. Rear wheels are drivable while front wheels are used for steering. Exoskeleton consists of thigh, shank and pelvis segments that are interconnected by revolute joints. Actuated joints on the exoskeleton are revolute joints of the hip and knee in sagittal plane. An actuated translational DoF, which exists in contact between the exoskeleton and the mobile platform, allows translation of human body in vertical direction. Motions around remaining rotational axis (frontal and transverse) of the exoskeleton's hip and knee joints are restricted by mechanical construction of the exoskeleton. Since motions of the human in the system are allowed only in sagittal and frontal plane, the motion capture system that is described in Section 5 has to provide motion data only for those two planes. The selected number of DoF enables "human like" walking and was chosen to be the same as in the case of the existing stationary gait rehabilitation systems [3], [21]. Several different mechanical designs were modelled and simulated during the RoboWalker[1] project. Figure 2 shows the latest design that

---

[1] The work was performed within the RoboWalker project founded by The Federal Ministry of Economics and Technology of Germany.

(a)   (b)
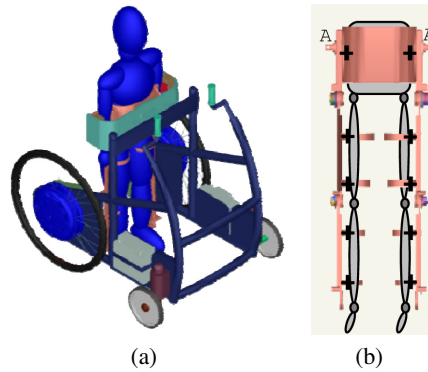
Fig. 2 The model of mobile gait rehabilitation system with the human inside it used in the simulation study (a). Connection points between the exoskeleton and the human body (b).

is outcome of simulation results based optimization of the system (stability, weight, manoeuvrability and etc.).

4   Mechanical Models of Human Body and the Rehabilitation System

Mechanical model of the human body used in presented simulation process consists of 19 segments (links) that are interconnected with spherical joints. Spherical joints were modelled as sections of three revolute joints connecting "virtual" segments of neglectable masses and inertial properties such that they do not affect dynamics of the mechanism. In contrast to other joints, knee joints were considered as having one DoF, rotation about sagittal axis, as two other rotations, around transverse and frontal axis, were locked . Modelling of the ground-foot contacts was done using 11 contact points per each foot segment defining 11 contact ellipsoids. This number of contact ellipsoids enables accurate foot-floor contact force generation. The normal forces are generated according to the penetration of the ellipsoids into the ground surface. Mechanical models of the human body and the rehabilitation system were interconnected using connection elements with stiffness-damping characteristics. Distances and rotations between coordinate systems attached to every contact point on the exoskeleton and body segments were measured and action-reaction forces, acting on exoskeleton and human body, were generated according to the following equations:

$$\mathbf{F_c} = \mathbf{k} * \mathbf{d} + \mathbf{c} * \mathbf{d} \tag{1}$$

$$\mathbf{T_c} = \mathbf{k} * \mathbf{r} + \mathbf{c} * \mathbf{r} \tag{2}$$

where **Fc** and **Tc** are respectively action-reaction forces and moments, **k** and **c** are stiffness and damping coefficients, **d** is distance vector between contact points on

human body and exoskeleton and **r** is vector containing rotation angles between local coordinate systems that reside in contact points. Two of these connection points exist per each leg's segment and several of them exist on pelvis segment. Stiffness and damping coefficients are experimentally obtained, such that small motion between human body and exoskeleton is allowed. Human motion capture data used in simulations were obtained by markerless motion capture system developed at the Institute of Automation (IAT), University of Bremen, which is described below.

5 Markerless Motion Capture System

The markerless motion capture system used in the presented work, as shown in Figure 3, is a chain consisting of three main blocks: image acquisition, image data processing and human gait analysis.
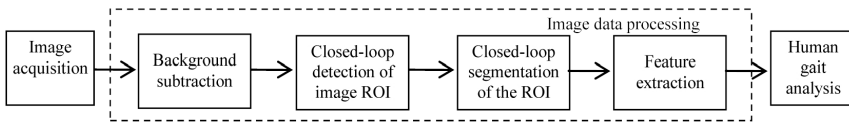

Fig. 3 Block-diagram of the presented vision based gait analysis system

The ultimate goal of the image processing is the reliable extraction of gait features that can be used for gait analysis for the purpose of gait rehabilitation. The extracted features are joint angles of human limbs, which are an effective means for characterizing human gait [14], [23]. Joint angles are extracted from two planes: frontal and sagittal. These angles are illustrated in Figure 4 (b, c) showing models of a human body in the frontal and sagittal planes in the form of stick figures, consisting of seven and five segments respectively, that are connected respectively with the five and three joints. For the sake of clarity, Figure 4 (a, d) shows two example video frames of a person walking in the frontal and sagittal plane respectively, overlaid with the extracted stick figures. This extraction of the human body skeleton in the presented system is the basis for the joint angle calculations. As can be seen in Figure 4 (b, c), the considered joint angles θ_{Fi} and θ_{Si}, are angles between the vertical and the line segments of the human body skeleton in the frontal and sagittal plane respectively. The indexes denote the following angles: 1 - torso angle; 2, 4 - left and right thigh angles; 3, 5 - left and right shank angles. Beside the above described angles it is of particular interest for gait analysis [14], [23] to extract knee angles θ_{K1} and θ_{K2}, the angles between thighs and shanks segments in sagittal plane, as illustrated in Figure 4 (c). In the presented system, the extraction of the human body skeleton, that is the localization of joint angles in 2D images, which is predecessor to joint angles calculation, is based on the segmented image of the human body. The image obtained by segmentation of the original image of a person walking in the frontal plane shown in Figure 4 (a) is given in Figure 5. The
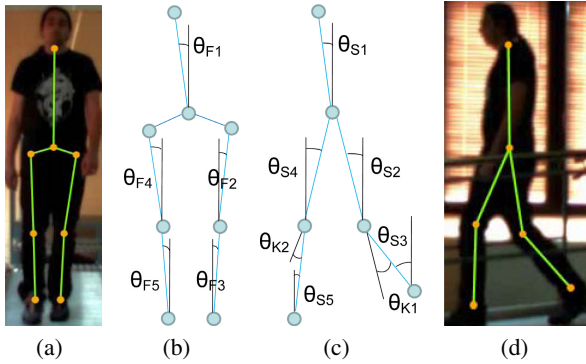
Fig. 4 Original images of a person walking in frontal (a) and sagittal (d) plane overlaid with stick figures. Human body models in frontal (b) and sagittal (c) plane in the form of stick figures with the joint angles.
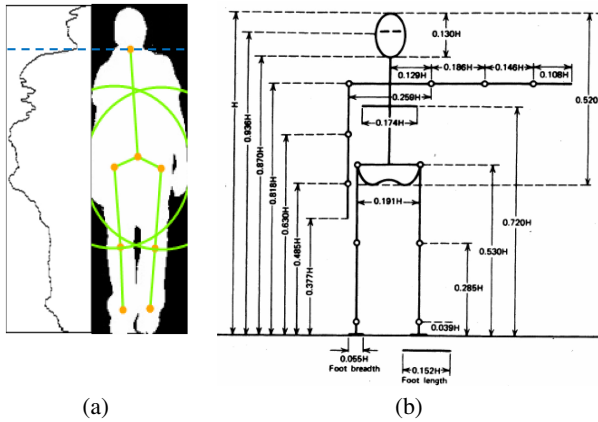
Fig. 5 Joints localization based on both the segmented image of the human body and vertical projection (a). Statistical data on body segments lengths [23] (b).

graph displaying the number of segmented (white) pixels in each image row, known as the vertical projection, is shown in Figure 5 together with the segmented image. This vertical projection is used for determining the location of the first joint, the neck joint. Namely, as illustrated in Figure 5, the neck joint corresponds to the top located global minimum of the vertical projection. The locations of other joints are determined by combining the segmented image data, the extracted location of the neck joint and the statistical anatomical measures of the human body segments, which can be found in [23], where all body segment lengths are represented as a percentage of body height H. Hence, at first the height H is estimated from the segmented image as the height of the bounding box of segmented human body. Then, starting from the extracted neck joint and using statistical information about human skeleton

segments lengths, the locations of hip joints are determined. As illustrated in Figure 5, the hip joints are considered as the centres of circles having radiuses equal to the length of the thighs, that is, the distance between the hip and the knee joint. Thus, the knee joints are defined by the intersections of the circles and the segmented body parts identified as the legs. Since the location of the joints in 2D images rely heavily on the segmented image, a segmented image of "good" quality, containing full "well shaped" segmented regions of the human body which is fully separated from the background, is essential to the presented system. The image which is the subject of segmentation in the presented system is obtained by background subtraction. Ideally, the image resulting from this background subtraction should contain only pixels belonging to the human body. Such an image would be an "ideal" input to the segmentation by simple thresholding [17]. However, background subtraction is very sensitive to different external influences, and in real-world applications often leads to subtracted images containing background pixels besides the human body pixels. Therefore, in order to achieve a good segmented image as necessary for reliable feature extraction, it is essential to adapt the segmentation parameters to illumination and background conditions. For automatic adjustment of the segmentation parameters, in the presented system the inclusion of a sequential control structure at the image segmentation level is proposed as indicated in Figure 3. The control structure consists of two sequential closed-loops: closed-loop control of image region of interest (ROI) definition and closed-loop control of ROI segmentation. The main idea behind the inclusion of closed-loop control structures is to change the processing parameters in a closed-loop manner so that the current image processing result at a particular processing level is driven to a desired result independently of external influences. The so-called two-dimensional (2D) entropy of segmented pixels [15] is considered as the measure of the quality of image processing result in both closed-loops. Applying an appropriate extremum seeking control in both closed-loops the processing (segmentation) parameter is determined which assures minimum of 2D entropy and so desired connectivity of segmented pixels in human body region [16].

6   Simulation Results

Simulation of walking inside the gait rehabilitation system using motion data obtained by the described markerless motion captured system was performed. The motion data of healthy subject was not suitable for simulating human walking inside the system, because human body model in the system can use only limited number of DoF during walking in contrast to healthy humans that use all available DoF. Hence, a healthy human subject (178cm, 91kg), whose walking inside the system was modelled, imitated walking inside the system by limiting motions in the joints which motion is restricted by mechanical construction of the system. Motion data of the ankle, knee and hip joints were extracted from frontal and sagittal planes. Extracted motion data were used to drive kinematically human's hip and knee joints in sagittal plane

and human's ankle joints in frontal and sagittal plane. Figure 6 (a) depicts right hip and knee angles changes used for simulation. Transitional motion of the pelvis was calculated according to the lengths of left and right legs during walking, such that at least one foot is always in contact with the ground. Figure 6 (b) shows ground reaction force (GRF) during walking of the human model inside the rehabilitation system. The GRF shape is similar as in case of "normal" walking [24]. Nonetheless, at some time periods during the swing phase, both legs are in contact with ground. This is a consequence of limitation of lateral pelvis translation, due to the mechanical construction of the system. Contrary, humans translate pelvis laterally in order to preserve the dynamic balance [22]. The results imply that walking when the horizontal translation of the pelvis is not allowed is possible inside the system, because the support platform provides balance during walking. However, walking with the same gait pattern would not be feasible without the support. Therefore, additional degrees of freedom in pelvis and hips have to be introduced in order to allow walking with dynamically balanced gait patterns.
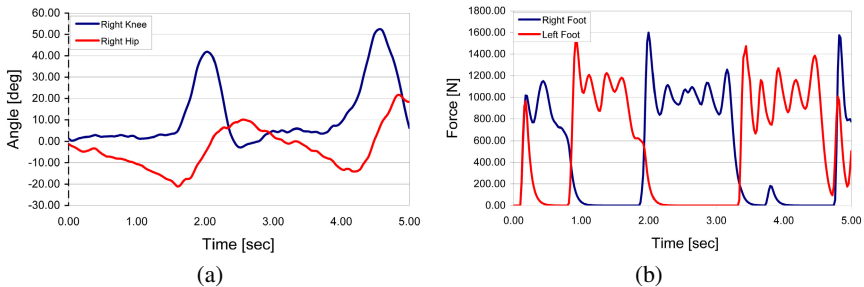
Fig. 6 Angle changes in hip and knee extracted from images and used in simulation (a). Ground reaction forces during walking inside the rehabilitation system obtained by simulation (b).

7 Conclusions

In this paper, an approach for modelling and simulation of human walking inside the robotic mobile gait rehabilitation robotic system, consisting of mobile platform and powered exoskeleton, is presented. Software tools that are usually used in modelling and simulation of complex mechanical systems were combined and exploited for modelling human walking and human-robot mechanical interaction. Motion data for driving mechanical model of human body was obtained using novel markerless motion capture system. The results of simulation and performance evaluation of the markerless motion capture system show that the selected simulation environment and the developed markerless motion capture system can be effectively used for simulating of human walking inside the rehabilitation system. According to the simulation results, it is decided to improve the mechanical construction of the

system by introducing an additional DoF that will allow lateral translation of pelvis. This will allow practicing of dynamic balanced walking.

References

1. AnyBody, `http://www.anybodytech.com`
2. Biometrics Ltd., `http://www.biometricsltd.com`
3. Colombo, G., Joerg, M., Schreier, R., Dietz, V.: Treadmill training of paraplegic patients using a robotic orthosis. Journal of Rehabilitation Research & Development 37(6), 693–700 (2000)
4. Damsgaard, M., et al.: Analysis of musculoskeletal systems in the AnyBody Modeling System. Simulation Modelling Practice and Theory 14, 1100–1111 (2006)
5. Hemami, H., Weimer, F.C., Kwzekanani, S.H.: Some aspects of the inverted pendulum problem for modeling of locomotion systems. IEEE Trans. Automat. Contr. AC-18, 658461 (1973)
6. Kajita, S., Matsumoto, O., Saigo, M.: Real-time 3D walking pattern generation for biped robot with telescopic legs. In: 2001 IEEE International Conference on Robotics and Automation, pp. 2299–2306 (2001)
7. Kajita, S., Tani, K.: Study of dynamic biped locomotion on rugged terrain - Derivation and Application of the Linear Inverted Pendulum Mode. In: Proceedings of the 1991 IEEE International Conference on Robotics and Automation, pp. 1405–1411 (1991)
8. Kajita, S., Yamaura, T., Kobayashi, A.: Dynamic walking control of a biped robot along a potential-energy conserving orbit. IEEE Trans. Robot. Autom. 8(4), 431–438 (1992)
9. LifeMod, `http://www.lifemodeler.com`
10. MATLAB, `http://www.mathworks.com`
11. MSC Adams, `http://www.mscsoftware.com`
12. Nakamura, Y., Yamane, K., Suzuki, I., Fujita, Y.: Dynamics Computation of Musculo-Skeletal Human Model Based on Efficient Algorithm for Closed Kinematic Chains. In: Proceedings of the 2nd International Symposium on Adaptive Motion of Animals and Machines (2003)
13. OpenSim, `https://simtk.org`
14. Perry, J.: Gait Analysis: Normal and Pathological Function. SLACK Incorporated (1992)
15. Ristic, D., Greaser, A.: Performance measure as feedback variable in image processing. EURASIP Journal on Applied Signal Processing (2006)
16. Ristic-Durrant, D., Leu, A., Slavnic, S., Greaser, A.: Markerless Vision-Based Human Gait Analysis System for Gait Rehabilitation. In: The 3rd International Congress on Gait & Mental Function, Washington DC, USA (2010)
17. Sezgin, M., Sankur, B.: Survey over image thresholding techniques and quantitative performance evaluation. Journal of Electronic Imaging 13(1), 146–165 (2004)
18. Shiratsu, A., Coury, H.J.C.G.: Reliability and accuracy of different sensors of a flexible electrogoniometer. Clinical Biomechanics 18, 682–684 (2003)
19. SIMM, `http://www.musculographics.com`
20. Sugihara, T., Nakamura, Y., Inoue, H.: Real-time humanoid motion generation through ZMP manipulation based on inverted pendulum control. In: Proceedings of IEEE International Conference on Robotics and Automation, vol. 2, pp. 1404–1409 (2002)

21. Veneman, J.F., Kruidhof, R., Hekman, E.E.G., et al.: Design and Evaluation of the LOPES Exoskeleton Robot for Interactive Gait Rehabilitation. IEEE Trans. Neural Syst. Rehabil. Eng. 15(3), 379–386 (2007)
22. Vukobratovic, M., Borovac, B.: Zero-moment point - thirty five years of its life. International Journal of Humanoid Robotics 1(1), 157–173 (2004)
23. Winter, D.A.: Biomechanics and Motor Control of Human Movement, 3rd edn. John Wiley & Sons (2005)
24. Zatsiorsky, V.M.: Kinetics of human motion, Champaign IL (2002)

Optimization and Imitation Problems for Humanoid Robots

Wael Suleiman, Eiichi Yoshida, Fumio Kanehiro,
Jean-Paul Laumond, and André Monin

Abstract. In this paper, the problems of humanoid robot motion optimization and human motion imitation by a humanoid robot are investigated. At first, we propose a unified framework for the optimization of humanoid robot motions. This framework is based on an efficient dynamics algorithm which allows the calculation of the gradient function with respect to the control parameters analytically. We show the efficiency of the framework through an example of smoothing a pre-calculated humanoid motion by minimizing the exerted torques, and at the same time improving the stability of the humanoid robot during the execution of the motion. Furthermore, we give insights into the problem of imitating human capture motions by a humanoid robot. We point out that the imitation problem can be formulated as an optimization problem under the constraints of physical limits and balance. The experimental results conducted on the humanoid robot HRP-2 have pointed out the efficiency of the framework of optimization to smooth humanoid robot motions and to generate imitated motions that preserve the salient characteristics of the original human captured motions. Moreover the experiments showed that the optimization procedure is well converging thanks to the analytical computation of the gradient function.

Wael Suleiman
Electrical Engineering and Computer Engineering Department, Faculty of Engineering,
Université de Sherbrooke, 2500 boul. Universite, Sherbrooke Quebec (Canada) J1K 2R1
e-mail: Wael.Suleiman@USherbrooke.ca

Eiichi Yoshida · Fumio Kanehiro
CNRS-AIST JRL (Joint Robotics Laboratory), UMI3218/CRT,
National Institute of Advanced Industrial Science and Technology (AIST),
Tsukuba Central 2, 1-1-1 Umezono, Tsukuba, Ibaraki, 305-8568 Japan
e-mail: {e.yoshida,f-kanehiro}@aist.go.jp

Jean-Paul Laumond · André Monin
LAAS-CNRS, 7 Avenue du Colonel Roche, 31077 Toulouse Cedex 4, France
e-mail: {jpl,monin}@laas.fr

1   Introduction

Few years ago, talking about humanoid robots was some kind of science fiction. The recent technological advancement not only transformed this dream into a reality, but also the ability of actual humanoid robots to execute complex tasks is increasing rapidly. The latest trends in humanoid research are to increase their autonomous behavior as well as improving the stability and the smoothness of their motions.

Motion optimization is becoming an active research topic in the recent years in computer graphics as well as in robotics. In virtual reality, a method based on optimal control theory within a recursive dynamics framework is proposed in [13]. The objective of their work is to simulate dynamically-correct astronaut motions by minimizing joint torques. In [24], a method for modeling and optimally control a human-like running motion is presented. The problem of modeling and control is transformed into a numerical optimization problem that has been solved using the optimization package MUSCOD-II [12, 11].

In robotic research, the optimization of manipulability trajectories has been investigated in [3]. The obtained solution is kinematically optimal and therefore does not take into account the dynamic constraints.

An overview of motion optimization in robotic using inverse dynamic model is given in [27]. Using Newton and quasi-Newton optimization algorithms for dynamics-based robot motion generation is proposed in [10], their algorithm makes use of an analytical formulation of the dynamic equation that is based on Lie group and Lie algebra.

A whole-body control framework for humanoids is proposed in [25]. This framework integrates task-oriented dynamic control while complying with humanoid physical constraints. The controller is obtained within the operational space [9] at multiple levels. The controller provides the torques which should be applied on each joint, that means the humanoid robot should be controlled by joint torques. Although the joint positions can be obtained by deploying forward dynamics methods, these methods are time consuming and not numerically efficient. It is worth mentioning that the humanoid robot HRP-2 and many other humanoid robots are controlled by joint positions.

The imitation of human motions by a humanoid robot is another research topic that recently begins receiving attention. In the cooperative tasks involving human beings and humanoid robots, the humanoid robot is expected to act like human beings. To this end, the humanoid robots should be able to imitate human motions. The imitation task can be done interactively by observing human motions or offline by using motion capture data. A method to generate human natural behavior through abstracting the human being's behavior into symbols by using mimesis theory is proposed in [4]. The observed symbols are analyzed into self motion elements which is regarded as a series of behavior. A hidden Markov models is then used for the description of the relation between the sequence of motion patterns and primitive symbols, and a natural behavior can be generated and applied on a humanoid robot.

Another method to transform a dance captured motion into a motion that the humanoid robot can properly execute is proposed in [18]. A whole body control of the

humanoid robot to imitate Jongara-Bushi dance, which is a traditional Japanese folk dance, is realized in [16]. To maintain the dynamical stability of humanoid robot, the previous method controls the trajectory of Zero Moment Point (ZMP) [31] to be always inside of the polygon of support. A user interface for creating whole body motions of biped humanoid robots is proposed in [14]. Similarly to computer graphics, the interface enables a user to edit variety of motions in an intuitive and flexible way without paying attention to the balance or self-collision issues. This interface has been validated on the humanoid robot HRP4-C [8] through an impressive dance performance. Other research related to imitating dance motions have been as well realized [15]. A pre-recorded human motion is used to generate optimal motion of the upper body of *Sarcos* humanoid robot [23]. The function to be minimized is the difference between the recorded and the executed motion by the robot. However, the previous methods do not consider some physical limits of humanoid robot, e.g. torque limits.

A method to optimize upper body motion of humanoid robot in order to imitate a human captured motion is proposed in [22]. Their objective function preserves the main characteristics of the original motion, and at the same time it respects the physical constraints of the humanoid robot. However, the authors mentioned that the resulting trajectories might not respect the physical limits of the humanoid robot, and in reality they are often violated. This is because their method considers the velocity and the force limits separately.

The present paper gives insights into the main contribution of our papers [30, 29]. Those contributions are:

- Developing an optimization framework for humanoid robot motions. This framework takes as input a pre-calculated motion that is provided by motion planning techniques. The output is an optimized and a stable motion.
- The proposed method uses the inverse dynamic formulation and the parameters to be optimized are the joint positions. As a consequence, the humanoid robot can be controlled directly in the joint space and not in torque control space. The method is therefore well adapted for a position controlled humanoid robot such as HRP-2 platform.
- Providing an optimization framework to generate humanoid robot motions from human motion capture data. The generated motion imitates the original human captured motion, and at the same time it respects the physical limits of humanoid robot.
- The validation of the proposed methods on the humanoid robot HRP-2 pointed out their efficiency and robustness.

2   Optimization of Humanoid Motions

A general optimization framework of humanoid robot motion can be formulated as follows

$$\min_{\mathbf{q_t},\dot{\mathbf{q}}_\mathbf{t},\ddot{\mathbf{q}}_\mathbf{t}} J\left(\mathbf{q_t},\dot{\mathbf{q}}_\mathbf{t},\ddot{\mathbf{q}}_\mathbf{t}\right)$$

(1)

subject to

$$S^T \mathbf{F_t} = \tau_\mathbf{t} \qquad \text{(Dynamic equation)}$$

$$\mathbf{q_{t_0}} = \mathbf{q_0}, \dot{\mathbf{q}}_{\mathbf{t_0}} = \mathbf{0}, \ddot{\mathbf{q}}_{\mathbf{t_0}} = \mathbf{0} \qquad \text{(Initial configuration constraints)}$$

$$\mathbf{q_{t_f}} = \mathbf{q_f}, \dot{\mathbf{q}}_{\mathbf{t_f}} = \mathbf{0}, \ddot{\mathbf{q}}_{\mathbf{t_f}} = \mathbf{0} \qquad \text{(Final configuration constraints)}$$

$$\tau^- \leq \tau_t \leq \tau^+ \qquad \text{(Torque limits)}$$

$$\dot{\mathbf{q}}^- \leq \dot{\mathbf{q}}_\mathbf{t} \leq \dot{\mathbf{q}}^+ \qquad \text{(Velocity limits)}$$

$$\mathbf{q}^- \leq \mathbf{q_t} \leq \mathbf{q}^+ \qquad \text{(Joint limits)}$$

$$\mathbf{p}_t^- \leq \mathbf{p_t} \leq \mathbf{p}_t^+ \qquad \text{(Stability criterion)}$$

$$\mathscr{G}_{foot} = \mathscr{G}_{foot}^{ref} : \mathbf{t_{c^1}}, \cdots, \mathbf{t_{c^p}} \qquad \text{(Foot prints)}$$

where $J\left(\mathbf{q_t},\dot{\mathbf{q}}_\mathbf{t},\ddot{\mathbf{q}}_\mathbf{t}\right)$ is the objective function to be minimized. The stability of the humanoid robot is assured by guaranteeing that the ZMP is always inside of the polygon of support.

- $\tau_\mathbf{t}$ is the vector of the applied torques on the joints
- $\mathbf{F_t}$ is the vector of the applied forces on the joints
- $\tau_{i,t}$ and $F_{i,t}$ denote the value of the applied torque and force on the joint i respectively
- $\mathbf{q}_t$ denotes the configuration vector which includes the joint positions of the humanoid robot and the 6 degree of freedoms of the free-flyer (pelvis joint). That means for a humanoid robot of n degree of freedoms $\mathbf{q}_t \in \mathbb{R}^{n+6}$
- x^- and x^+ denote, respectively, the minimal and the maximal values of vector x
- $\mathbf{p_t} \in \mathbb{R}^2$ is the projection of the ZMP on the floor
- $\mathbf{p}_t^+$ and $\mathbf{p}_t^-$ design the polygon of support
- $\mathscr{G}_{foot} \in \mathbb{R}^6$ denotes the configuration (position and orientation) of the ankle joint in the euclidean space
- t_{c^i} denotes the instant of the foot impact with the floor

Studying human movements [32, 33] has brought out a connection between minimizing energy dissipation and forces, and the smoothness of human movements. On account of the complexity of calculating the energy dissipation, one can use an approximative prediction of it. A good predictor of human's metabolic energy is proven to be the joint torques [26]. As the humanoid robot is supposed to realize human-like motions, we chose therefore minimizing the integral of the Euclidean norm of joint torques $J\left(\mathbf{q_t},\dot{\mathbf{q}}_\mathbf{t},\ddot{\mathbf{q}}_\mathbf{t}\right) = \int_{t_0}^{t_f} \tau_\mathbf{t}^T \tau_\mathbf{t} dt$.

In order to transform the optimization problem (1) into a classical optimization problem, let us define

$$X_t = \begin{bmatrix} \mathbf{q_t}^T & \dot{\mathbf{q}}_\mathbf{t}^T & \ddot{\mathbf{q}}_\mathbf{t}^T \end{bmatrix}^T, \quad L(X_t) = \int_{t_0}^{t_f} \tau_\mathbf{t}^T \tau_\mathbf{t} dt$$

(2)

$$G(X_t) = \begin{bmatrix} \tau_t \\ \mathbf{q}_t \\ \dot{\mathbf{q}}_t \\ \mathbf{p}_t \end{bmatrix}, \quad G^- = \begin{bmatrix} \tau^- \\ \mathbf{q}^- \\ \dot{\mathbf{q}}^- \\ \mathbf{p}_t^- \end{bmatrix}, \quad G^+ = \begin{bmatrix} \tau^+ \\ \mathbf{q}^+ \\ \dot{\mathbf{q}}^+ \\ \mathbf{p}_t^+ \end{bmatrix}, \quad H(X_t) = \begin{bmatrix} \tau_t - S^T \mathbf{F_t} \\ \mathbf{q_{t_0}} - \mathbf{q_0} \\ \dot{\mathbf{q}}_{t_0} \\ \ddot{\mathbf{q}}_{t_0} \\ \mathbf{q_{t_f}} - \mathbf{q_f} \\ \dot{\mathbf{q}}_{t_f} \\ \ddot{\mathbf{q}}_{t_f} \\ \mathscr{G}_{foot} - \mathscr{G}_{foot}^{ref} \end{bmatrix} \quad (3)$$

Thus the optimization problem (1) can be transformed into the following classical form:

$$\min_{X_t} L(X_t)$$

subject to

$$H(X_t) = 0$$
$$G^- \leq G(X_t) \leq G^+$$

(4)

The above optimization problem has been extremely studied in the literature of optimization theory. To solve this optimization problem, one can use the augmented Lagrange multiplier method, which is a very efficient and reliable method [19, 13, 30].

The implementation of augmented Lagrange multiplier method, and other deterministic optimization methods, requires the computation of the gradient functions $\frac{\partial L(X_t)}{\partial X_t}, \frac{\partial H(X_t)}{\partial X_t}$ and $\frac{\partial G(X_t)}{\partial X_t}$. Approximating those gradient functions by a numerical difference method is usually used in practice. However, this approach is not only a time consuming on account of the evaluation of the gradient functions, but also might does not converge well because of the approximation.

To overcome this difficulty, we decided to use the recursive dynamic algorithm proposed in [17]. The main advantage of this algorithm is that the gradient function of the dynamic equation is calculated analytically in a recursive way. For more details refer to [17, 30, 29].

However, it is well known that the space of the admissible solutions of the minimization problem (4) is very large. In order to transform this infinite dimensional space to a finite one, one can use a basis of shape functions. To this end, we chose a basis of *cubic B-spline functions*. For more details refer to [30, 29].

3 Imitating Human Motions

The main purpose of the imitation is to reproduce a human motion by an avatar or a humanoid robot. The inputs of the imitation procedure are often human motion capture data, these motions are provided by a motion capture system as a skeleton of a virtual actor and a sequence of the values of virtual actor's joints.

The virtual actor has generally more degree of freedoms than the humanoid robot and its links' lengths are different from those of the humanoid robot. The imitation problem, from a kinematic point of view, is well known in computer graphics and it is called motion retargeting [2]. The motion retargeting problem is formulated as follows

$$\min_{\mathbf{q_t}} \int_{t_0}^{t_f} \left\{ (\mathbf{q_t} - \mathbf{q_t^c})^T (\mathbf{q_t} - \mathbf{q_t^c}) + \sigma (P_t - P_t^c)^T (P_t - P_t^c) \right\} dt$$

subject to

$$\begin{cases} \mathbf{q_{t_0}} = \mathbf{q_0} \\ \mathbf{q_{t_f}} = \mathbf{q_f} \\ \mathbf{q}^- \leq \mathbf{q_t} \leq \mathbf{q}^+ \end{cases} \tag{5}$$

where σ is a user defined constant. $\mathbf{q_t}$ and $\mathbf{q_t^c}$ are the joints position vectors of the humanoid robot and the virtual actor respectively. P_t and P_t^c are the Cartesian positions of the head, hands and feet of the humanoid robot and those of the virtual actor. These positions are usually expressed in the pelvis frame, and the vector P_t is defined as follows:

$$P_t = \begin{bmatrix} P_{\text{head}}^T & P_{\text{right hand}}^T & P_{\text{left hand}}^T & P_{\text{right foot}}^T & P_{\text{left foot}}^T \end{bmatrix}^T \tag{6}$$

If the lengths of the virtual actor's links are largely different from those of the humanoid robot, the vector P_t^c can be scaled to fit for the humanoid robot size. The retargeting problem has been extremely studied in computer graphics during the last years, and there is many commercial graphic software that can solve it efficiently.

However, in the case of human motion imitation by a humanoid robot additional difficulties arise such as the joints velocity and torque limits. Moreover, another challenging issue in the imitation of fast human motions because of the actuators limits of the humanoid robot. A simple solution could be slowing down the captured motion.

Let us consider that we have a captured motion which consists of N samples and the sampling frequency of this motion is f (e.g. $f = 120$ Hz). Let $\mathbf{q}^c(n)$ denotes the vector of joint values which corresponds to the sample number n. An algorithm to transform the human captured motion into a motion within the joint velocity limits of the humanoid robot is given by the following pseudo code.

Algorithm 1. Time re-parameterization of the human captured motion

---

Input: $n \leftarrow 1$
while $n \leq N-1$ **do**
    Calculate $\Delta\mathbf{q}^c \leftarrow |\mathbf{q}^c(n+1) - \mathbf{q}^c(n)|$;
    Input: $\Delta t \leftarrow \frac{1}{f}$
    while $\frac{\Delta\mathbf{q}^c}{\Delta t} > \dot{\mathbf{q}}^+$ **do**
        $\Delta t \leftarrow \Delta t + \frac{1}{f}$;
    end
    $n \leftarrow n+1$;
end

Recall that $\dot{\mathbf{q}}^+$ is the maximal value of humanoid robot's joint velocity.

Algorithm 1 provides the new associated timing (t) to each $\mathbf{q}^c(n)$. Fig. 5 shows that a motion which originally lasts around 12 seconds is transformed into a motion lasts around 16 seconds.

After modifying the original human motion and taking into account the additional physical constraints, the imitation problem becomes

$$\min_{\mathbf{q_t},\dot{\mathbf{q}}_\mathbf{t},\ddot{\mathbf{q}}_\mathbf{t}} \int_{t_0}^{t_f} \left\{ (\mathbf{q_t} - \mathbf{q_t^c})^T (\mathbf{q_t} - \mathbf{q_t^c}) + \sigma (P_t - P_t^c)^T (P_t - P_t^c) \right\} dt \tag{7}$$

subject to

$$S^T \mathbf{F_t} = \tau_\mathbf{t} \qquad \text{(Dynamic equation)}$$
$$\mathbf{q_{t_0}} = \mathbf{q_0}, \dot{\mathbf{q}}_{\mathbf{t_0}} = \mathbf{0}, \ddot{\mathbf{q}}_{\mathbf{t_0}} = \mathbf{0} \qquad \text{(Initial configuration constraints)}$$
$$\mathbf{q_{t_f}} = \mathbf{q_f}, \dot{\mathbf{q}}_{\mathbf{t_f}} = \mathbf{0}, \ddot{\mathbf{q}}_{\mathbf{t_f}} = \mathbf{0} \qquad \text{(Final configuration constraints)}$$
$$\tau^- \leq \tau_\mathbf{t} \leq \tau^+ \qquad \text{(Torque limits)}$$
$$\dot{\mathbf{q}}^- \leq \dot{\mathbf{q}}_\mathbf{t} \leq \dot{\mathbf{q}}^+ \qquad \text{(Velocity limits)}$$
$$\mathbf{q}^- \leq \mathbf{q_t} \leq \mathbf{q}^+ \qquad \text{(Joint limits)}$$
$$\mathbf{p}_t^- \leq \mathbf{p_t} \leq \mathbf{p}_t^+ \qquad \text{(Stability criterion)}$$

where $\mathbf{q}_t^c$ is the modified motion (time re-parameterization) and the other parameters are similar to those defined in Section 2.

Analogously to the previous case, let us define

$$X_t = \begin{bmatrix} \mathbf{q_t}^T & \dot{\mathbf{q}}_\mathbf{t}^T & \ddot{\mathbf{q}}_\mathbf{t}^T \end{bmatrix}^T$$
$$L(X_t) = \int_{t_0}^{t_f} \left\{ (\mathbf{q_t} - \mathbf{q_t^c})^T (\mathbf{q_t} - \mathbf{q_t^c}) + \sigma (P_t - P_t^c)^T (P_t - P_t^c) \right\} dt \tag{8}$$

$$G(X_t) = \begin{bmatrix} \tau_\mathbf{t} \\ \mathbf{q}_t \\ \dot{\mathbf{q}}_t \\ \mathbf{p}_t \end{bmatrix}, \quad G^- = \begin{bmatrix} \tau^- \\ \mathbf{q}^- \\ \dot{\mathbf{q}}^- \\ \mathbf{p}_t^- \end{bmatrix}, \quad G^+ = \begin{bmatrix} \tau^+ \\ \mathbf{q}^+ \\ \dot{\mathbf{q}}^+ \\ \mathbf{p}_t^+ \end{bmatrix}, \quad H(X_t) = \begin{bmatrix} \tau_\mathbf{t} - S^T \mathbf{F_t} \\ \mathbf{q_{t_0}} - \mathbf{q_0} \\ \dot{\mathbf{q}}_{\mathbf{t_0}} \\ \ddot{\mathbf{q}}_{\mathbf{t_0}} \\ \mathbf{q_{t_f}} - \mathbf{q_f} \\ \dot{\mathbf{q}}_{\mathbf{t_f}} \\ \ddot{\mathbf{q}}_{\mathbf{t_f}} \end{bmatrix} \tag{9}$$

Thus, the optimization problem (7) can be transformed into the following classical form

$$\min_{X_t} L(X_t)$$

subject to

$$H(X_t) = 0 \tag{10}$$
$$G^- \leq G(X_t) \leq G^+$$

As one can easily notice that the above optimization problem is similar to the optimization problem (4) and it can be solved by an analogues logic, i.e. using the augmented Lagrange multiplier method, the analytical calculation of the gradient function and discretizing the configuration space in order to transform the optimization problem into a finite dimensional problem.

4 Experimental Results

The kinematic structure of the humanoid robot HRP-2 [7] is given in Fig. 1. In this structure the degree of freedoms are presented by cylinders. The structure contains 30 degree of freedoms.
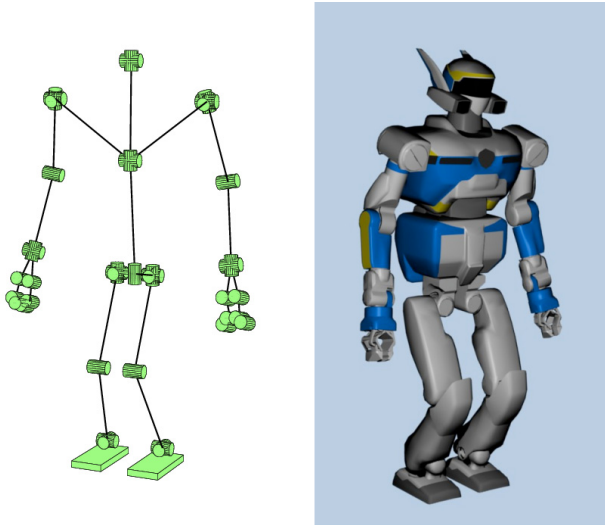
Fig. 1 Description of HRP-2 kinematic structure

4.1 Humanoid Motion Optimization

The experimental scenario that we have tested to validate the proposed method for optimizing humanoid motions is the following:

1. The robot carries a bar with its right hand.
2. The robot starts walking while lowering the vertical position of its pelvis, and at the same time it lifts up the carried bar.

The characteristics of the carried bar are: length= 2 m, weight= 0.7 Kg and cylindrical form with uniform density distribution. The robot grasps the bar at 0.35 m

from its end. The main purpose of adding the bar is creating an asymmetry in the kinematical structure of the humanoid robot. It is worth mentioning that generating the above-mentioned motion by considering only kinematic constraints yields an unstable walking pattern and the robot falls down in the middle of the experiment. Snapshots of the conducted motion are presented in Fig. 2.

The augmented Lagrange multiplier method required *43 iterations* to converge and provide the optimal trajectory. We have chosen to report the number of iteration instead of the computational time, this is because we are using MATALB language to solve the optimization problem. However, the number of iterations of the augmented Lagrange multiplier method gives a good idea of the computational time [20, 21, 1]. In fact, the augmented Lagrange multiplier method is a fast and reliable method because it does not require the inversion of matrices which have, in our case, very huge dimensions.

Fig. 2 Optimizing humanoid motion: snapshots of the conducted experiment

To compare the obtained results with the results obtained by the method presented in [5], we use the real measurements provided by the sensors of the humanoid robot HRP-2.

Fig. 3 shows the x coordinate of ZMP, the $x-$axis is the direction of the motion. The ground reaction force applied on the left foot are presented in Fig. 4(a). Fig. 4(b) shows the applied torque on the left knee.

We conclude from Fig. 3 and Fig. 4 that:

1. The optimization method smoothes the shape of ZMP, and the oscillations have been avoided. Note that the ZMP trajectory of the optimized motion does not
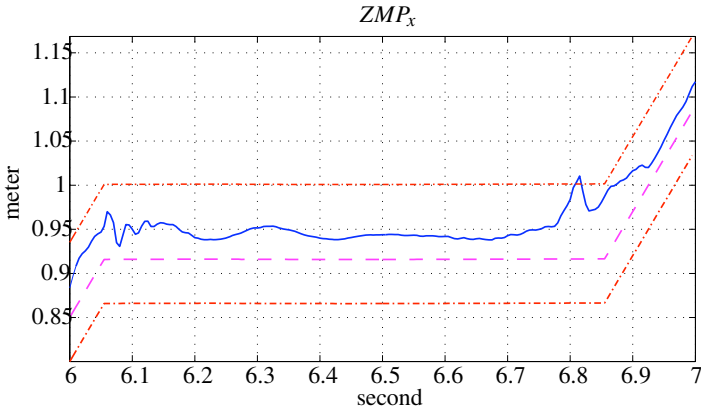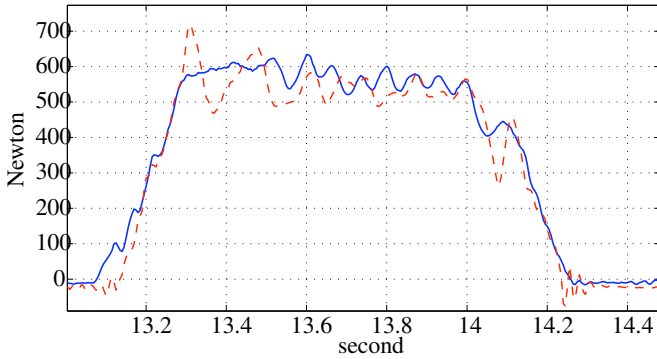
(a) Before optimization.
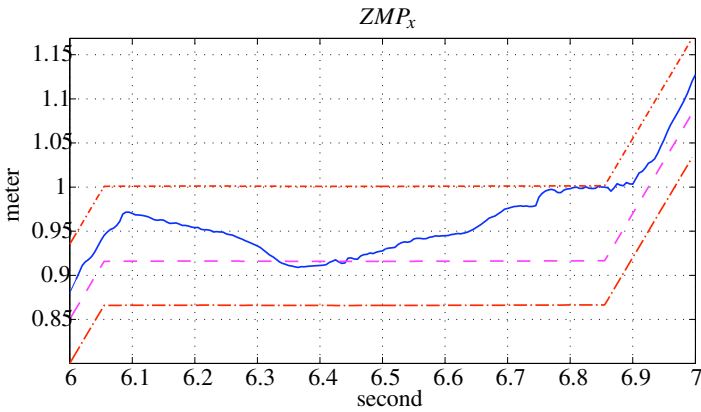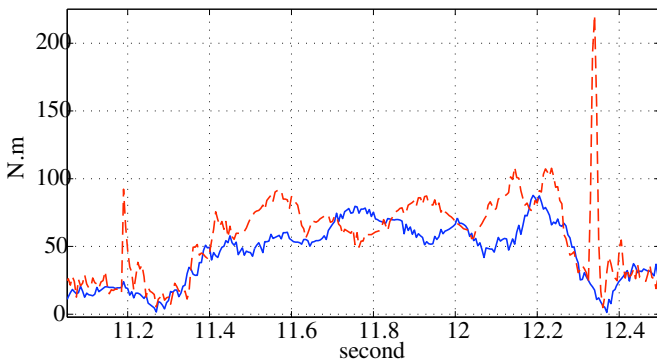

(b) After Optimization.

Fig. 3 x coordinate of ZMP: the solid line is the measured ZMP_x, the dash-dotted lines design the safe stability zone and the dashed line denotes the designed reference. The $x-$axis is the direction of the motion. The oscillations around 6.1 seconds disappear after the optimization as well as the peak at 6.81 second

follow the designed ZMP trajectory. This is because the optimization method assures the stability of humanoid robot by guaranteeing that the ZMP is always inside of the polygon of support.

2. Using the optimization method avoids the peaks in the profile of ground reaction forces applied on the feet.

3. Using the optimization method not only minimizes the joint torques, but also avoids the peaks.

(a) Applied ground reaction force on the left foot: solid line for the optimization method and dashed line for the method proposed in [5]. The peak at 13.3 second disappears after optimization.

(b) Applied torque on the left knee: solid line for the optimization method and dashed line for the method proposed in [5]. The peaks at 11.2 and 12.35 seconds disappear after the optimization.

**Fig. 4** Applied ground reaction force and torque

4.2 Human Motion Imitation

We chose a boxing captured motion to validate our proposed method. The imitation of a boxing motion by a humanoid robot is really a challenging issue. Fig. 5(a) shows the angular position trajectory of the virtual actor's right elbow, the optimized trajectory for the humanoid robot is given in Fig. 5(b). The optimized trajectory respects the physical limits of HRP-2 humanoid robot, which are the joint limits as well as the joint velocity and torque limits. In order to obtain the optimal trajectory, the augmented Lagrange multiplier method required *29 iterations*.
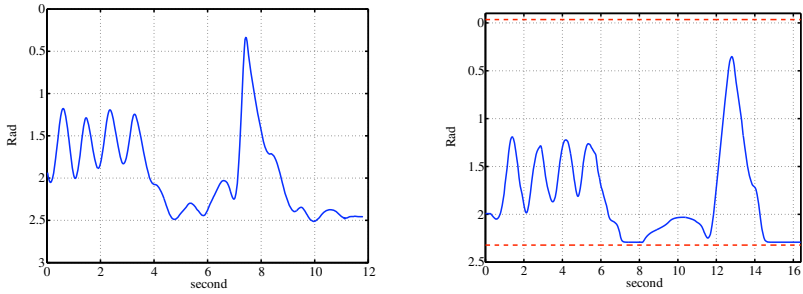
(a) Original human captured motion.

(b) Obtained motion after the time reparameterization and optimization under the constraints of physical limits of the humanoid robot (HRP-2). Dash lines denote the limits of elbow joint.

Fig. 5 Angular position of the right elbow
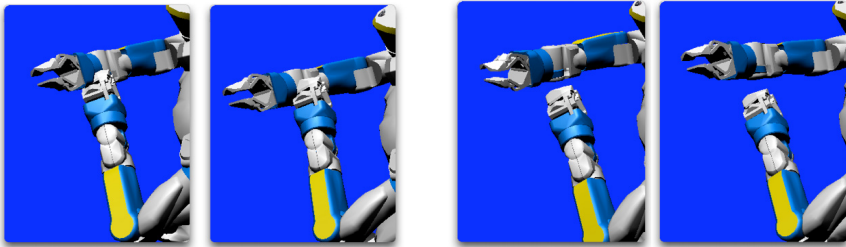
(a) Obtained motion after optimization.

(b) Applying self-collision avoidance algorithm [6] as post-processing task.

Fig. 6 Self-collision avoidance

However, the self-collision problem is not considered in this work as shown in Fig. 6(a). Although, approximating the humanoid robot's links by cylinders and spheres and consider the distance between them as an additional constraint can solve the problem of self-collision, this procedure might yield an imitated motion largely different from the original human captured motion on account of the approximation.

F. Kanehiro *et al* [6] proposed an efficient collision-avoidance method for a non-strictly convex objects. The method makes use of non-strictly convex polyhedra as geometric models of the robot and the environment. Applying this method as post-processing task can solve the problem of self-collision, and it yields a collision-free motion. Fig. 6(b) shows the self-collision avoidance of the humanoid robot's hands by applying the method proposed in [6]. Snapshots of the conducted motion using the humanoid robot HRP-2 are given in Fig. 7.
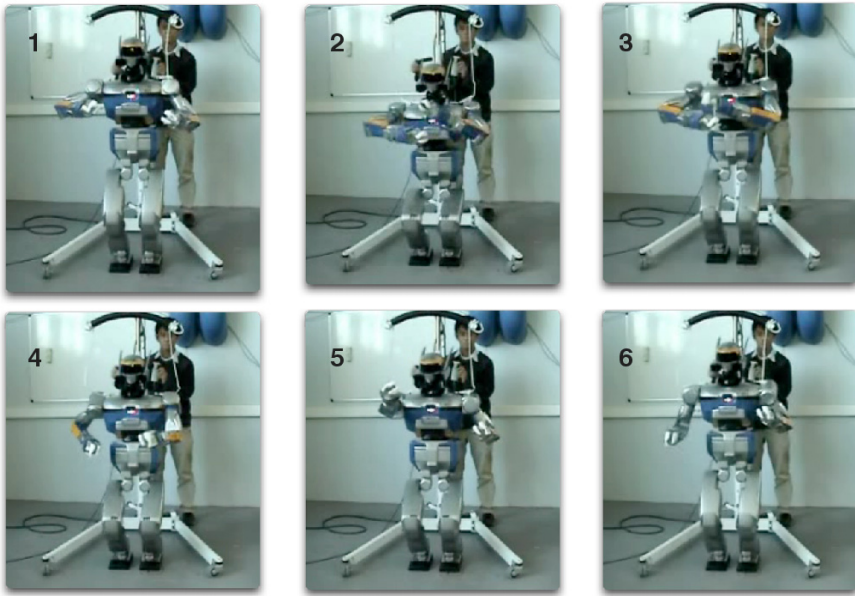
Fig. 7 Imitation of boxing motion: snapshots of the conducted motion

5   Discussion and Conclusion

In this paper, we present the methodology that we have developed to optimize humanoid robot motions and to imitate human motions by a humanoid robot. To this end, we proposed a unified framework for the optimization of humanoid robot motions under the constraints of the robot's physical limits and balance. This framework is based on an efficient dynamics algorithm, which allows computing the gradient function with respect to the control parameters analytically. Furthermore, we pointed out that the optimization framework can be deployed to efficiently solve the imitation of human motions by a humanoid robot.

The experimental results have pointed out the efficiency of the proposed methods, and they have been successfully validated on the humanoid robot HRP-2.

Further improvements of the proposed method can be done by considering the following issues:

1. Adding the jerk function of the joint trajectories in the objective function could yield a smoother motion.
2. In motion planning, deforming the planned trajectory is sometimes required. Therefore, the optimization framework can be enhanced by including the distance between the robot and the obstacles as an additional constraint.
3. The integration of self-collision avoidance into the optimization problem as an additional constraint.

4. Replacing the time re-parameterization algorithm by a more efficient algorithm such as the algorithm that we developed in [28].
5. Considering real-time application. This issue is very challenging because it requires developing new and highly efficient optimization algorithms.

References

1. Bertsekas, D.P.: Nonlinear Programming. Athena Scientific (1995)
2. Gleicher, M.: Retargetting Motion to New Characters. In: ACM SIGGRAPH, pp. 33–42 (1998)
3. Guilamo, L., Kuffner, J., Nishiwaki, K., Kagami, S.: Manipulability Optimization for Trajectory Generation. In: IEEE International Conference on Robotics and Automation, pp. 2017–2022 (2006)
4. Inamura, T., Nakamura, Y., Ezaki, H., Toshima, I.: Imitation and primitive Symbol Acquisition of Humanoids by the Integrated Mimesis Loop. In: IEEE International Conference on Robotics and Automation, pp. 4208–4213 (2001)
5. Kajita, S., Kanehiro, F., Kaneko, K., Fujiwara, K., Harada, K., Yokoi, K., Hirukawa, H.: Biped Walking Pattern Generation by using Preview Control of Zero-Moment Point. In: Proc. IEEE International Conference on Robotics and Automation, Taipei, Taiwan, pp. 1620–1626 (2003)
6. Kanehiro, F., Lamiraux, F., Kanoun, O., Yoshida, E., Laumond, J.P.: A Local Collision Avoidance Method for Non-strictly Convex Objects. In: 2008 Robotics: Science and Systems Conference, Zurich, Switzerland (2008)
7. Kaneko, K., Kanehiro, F., Kajita, S., Hirukawa, H., Kawasaki, T., Hirata, M., Akachi, K., Isozumi, T.: Humanoid Robot HRP-2. In: IEEE International Conference on Robotics and Automation, New Orleans, LA, USA, pp. 1083–1090 (2004)
8. Kaneko, K., Kanehiro, F., Morisawa, M., Miura, K., Kajita, S.: Cybernetic Human HRP-4C. In: IEEE-RAS International Conference on Humanoid Robots, Paris, France (2009)
9. Khatib, O.: A Unified Approach for Motion and Force Control of Robot Manipulators: The Operational Space Formulation. IEEE Journal of Robotics and Automation 3(1), 43–53 (1987)
10. Lee, S.H., Kim, J., Park, F.C., Kim, M., Bobrow, J.E.: Newton-type algorithms for dynamics-based robot movement optimization. IEEE Trans. Robotics 21(4), 657–667 (2005)
11. Leineweber, D.: The theory of muscod in a nutshell. Tech. rep., Universitat Heidelberg, IWR-Preprint 96-19 (1996)
12. Leineweber, D.B., Bauer, I., Bock, H.G., Schloder, J.P.: An efficient multiple shooting based reduced SQP strategy for large-scale dynamic process optimization - Part I: Theoretical aspects. Comput. Chem. Eng. 27(157-166) (2003)
13. Lo, J., Metaxas, D.: Recursive Dynamics and Optimal Control Techniques for Human Motion Planning. Computer Animation, 220–234 (1999)
14. Nakaoka, S., Kajita, S., Yokoi, K.: Intuitive and flexible user interface for creating whole body motions of biped humanoid robots. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 1675–1682 (2010)

15. Nakaoka, S., Nakazawa, A., Kanehiro, F., Kaneko, K., Morisawa, M., Ikeuchi, K.: Task Model of Lower Body Motion for a Biped Humanoid Robot to Imitate Human Dances. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 3157–3162 (2005)
16. Nakaoka, S., Nakazawa, A., Yokoi, K., Hirukawa, H., Ikeuch, K.: Generating Whole Body Motions for a Biped Humanoid Robot from Captured Human Dances. In: IEEE International Conference on Robotics and Automation, pp. 3905–3910 (2003)
17. Park, F.C., Bobrow, J.E., Ploen, S.R.: A Lie Group Formulation of Robot Dynamics. International Journal of Robotics Research 14(6), 1130–1135 (1995)
18. Pollard, N., Hodgins, J., Riley, M., Atkeson, C.: Adapting Human Motion for the Control of a Humanoid Robot. In: IEEE International Conference on Robotics and Automation (2002)
19. Rockafellar, R.: Augmented Lagrange Multiplier Functions and Duality in Nonconvex Programming. SIAM J. Control 12(2) (1974)
20. Rockafellar, R.T.: Lagrange multipliers in optimization. SIAM-AMS Proceedings 9, 145–168 (1976)
21. Rockafellar, R.T.: Lagrange multipliers and optimality. SIAM Review 35(2), 183–238 (1993)
22. Ruchanurucks, M., Nakaoka, S., Kudoh, S., Ikeuchi, K.: Humanoid Robot Motion Generation with Sequential Physical Constraints. In: Proc. IEEE International Conference on Robotics and Automation, pp. 2649–2654 (2006)
23. Safonova, A., Pollard, N., Hodgins, J.: Optimizing Human Motion for the Control of a Humanoid Robot. In: Proc. Applied Mathematics and Applications of Mathematics (2003)
24. Schultz, G., Mombaur, K.: Modeling and optimal control of human-like running. IEEE/ASME Transactions on Mechatronics 15(5), 783–792 (2010)
25. Sentis, L., Khatib, O.: A Whole-Body Control Framework for Humanoids Operating in Human Environments. In: Proceedings of the IEEE Internactional Conference in Robotics and Automation (2006)
26. Skrinar, A., Burdett, R., Simon, S.: Comparison of Mechanical and Metabolic Energy Consumption During Normal Gait. Journal of Orthopedic Research 1(1), 63–72 (1983)
27. Steinbach, M.C.: Optimal motion design using inverse dynamics. Tech. rep., Konrad-Zuse-Zentrum fur Informationstechnik Berlin, Deutschland (1997)
28. Suleiman, W., Kanehiro, F., Yoshida, E., Laumond, J.P., Monin, A.: Time parameterization of humanoid-robot paths. IEEE Transactions on Robotics 26(3), 458–468 (2010)
29. Suleiman, W., Yoshida, E., Kanehiro, F., Laumond, J.P., Monin, A.: On Human Motion Imitation by Humanoid Robot. In: IEEE International Conference on Robotics and Automation (ICRA), pp. 2697–2704 (2008)
30. Suleiman, W., Yoshida, E., Laumond, J.P., Monin, A.: On Humanoid Motion Optimization. In: IEEE-RAS 7th International Conference on Humanoid Robots, Pittsburgh, Pennsylvania, USA, pp. 180–187 (2007)
31. Vukobratović, M., Borovac, B.: Zero-Moment Point—Thirty Five Years of its Life. International Journal of Humanoid Robotics 1(1), 157–173 (2004)
32. Winter, D.: Biomechanics and Motor Control of Human Movement, 2nd edn. John Wiley (1990)
33. Yamaguchi, G.: Performing Whole-body simulations of Gait with 3D. In: Multiple Muscle Systems: Biomechanics and Movement Organization. Springer (1990)

Motor Control and Spinal Pattern Generators in Humans

Heiko Wagner, Arne Wulf, Sook-Yee Chong, and Thomas Wulf

Abstract. Spinal pattern generators can produce cyclic and acyclic muscular activation patterns. Here we introduce a model, which describes the interaction between spinal pattern generators and the musculoskeletal system. The behavior of this model will be demonstrated in three different examples, i.e. the reflexive behavior, the behavior during locomotion and during coupled arm movements. The results show that the model was capable of reproducing complex reflex patterns. For locomotion, it demonstrated the ability in adapting to changes and selecting appropriate afferents, so as to enable step-like motion to occur. Even phase transition during coupled arm movements could be described by the model. Our results support the hypothesis, that depending on the movement task, humans are able to change the coupling within the spinal pattern generator, which provides an extremely effizient motor control strategy.

1 Introduction

Neural oscillators, also known as spinal pattern generators (SPG) are neural networks that can produce complex muscular activation patterns ([13], [19]), i.e. cyclic and acyclic movements, based on simple central commands and feedback from proprioceptive sensors. To study the interaction between the neural network within the spinal column and the musculoskeletal system, we introduce a simple model, which describes the basic function of spinal pattern generators. Humans possess a flexible coupling of thoracolumbar and cervical centers for the motor control of movement tasks [21]. They can use the upper limbs for manipulative and skilled movements or alternatively for locomotor tasks. Depending on the envisioned task, humans

Heiko Wagner · Arne Wulf · Sook-Yee Chong · Thomas Wulf
Motion Science and Center for Nonlinear Science CeNoS, University of Muenster,
48149 Muenster, Germany
e-mail: `heiko.wagner@uni-muenster.de`

seemed to be able to manipulate the gating of neuronal circuits controlling lower and upper limb muscles [3]. Dietz et al. demonstrated that coordinated electromyographic (EMG) patterns could be induced in patients with either complete or incomplete paraplegia on a moving treadmill. This could only be achieved when loading and unloading of the limbs were done in combination with cyclic movements. Furthermore, to control bipedal walking or complex motions, it is advantageous to use an intelligent mechanical system, i.e. the musculoskeletal system, which by itself should be self-stable with respect to small perturbations. On this basis, the interaction between the musculoskeletal system and the SPG might support the stabilization of cyclic and acyclic motions. Our interest is focused on how these self-stabilizing properties interact with the motor control system. In these studies, we show that the use of a simple SPG model based on Matsouka [18] and combined with a forward-dynamic musculoskeletal model can describe several different situations of human movement. In the following, we will introduce a neuromusculoskeletal model which is based on a simple spinal pattern generator. The behavior of this model will be demonstrated in three different examples, i.e. the reflexive behavior, the behavior during locomotion and during coupled arm movements.

2 Model

The musculoskeletal system (4)–(5) together with the SPG (2)–(3) and the sensory integration (1) includes a system of antagonistic muscles, described by a Hill-type muscle model. The SPG is based on a model that describes the interaction of a network of connected neurons.

$$\dot{s}_i \quad = \quad r(F_i - s_i) \tag{1}$$

$$\tau\dot{x}_i \quad = \quad \sum_{j=1}^{n}(a_{ij}A_j + W_{ij}s_j + W_{ij}^v\dot{s}_j) - Bv_i - x_i \tag{2}$$

$$T\dot{v}_i \quad = \quad A_i - v_i \tag{3}$$

$$\dot{\alpha} \quad = \quad \omega \tag{4}$$

$$J\dot{\omega} \quad = \quad \sum_{k=1}^{m}A_k h_k f_k^l f_k^v + M_{\text{ext}} \tag{5}$$

3 Example A: Long-Latency Reflexes

The emergence of long-latency reflexes is still not understood [16, 24]. Many studies showed that reflexes and locomotion influence one another. Dietz et al. [3] made use of task-dependent reflex modulation to test coupling of upper- and lower-limb movements. The transmission of a reflex from tibialis nerve (foot) to triceps brachii mucle (arm) is interpreted as evidence for interlimb coupling by SPG during locomotion. The neural networks of reflex pathways are thus involved in those of SPG for

locomotion. Reflexes and SPG both consist of spinal neural networks with recipro-cal and recurrent connections and mechanisms like self-inhibition (e.g. mediated by Renshaw cells) [11, 26]. Therefore, we assumed that reflexes may be described by the SPG concept [27]. To test the hypothesis that reflexes can be described by SPG, we simulated reflex generation with a SPG model for two neurons and compared the simulation output to reflex signals measured by EMG. The model parameters were adjusted for each trial by an optimisation algorithm such that the model matched the measured EMG time course.

Fig. 1 Experimental setup: Sudden loading pertur-bations are applied to a wooden handle (A). The sensory afferents are input to the spinal pattern gen-erator, SPG (B). The SPG output is the activation of left and right abdominal ex-ternal oblique muscles (C). The measured reflexive ac-tivation pattern is compared to the simulated reflex (D).
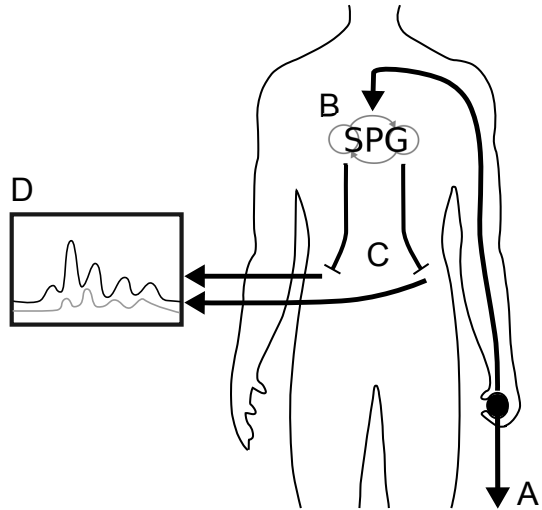
10 healthy women (23.6±1.4 years, 55.4±3.7 kg) volunteered for this study. They held a handle in one hand, in which sudden vertical loading perturbations were applied. The subjects were instructed to stand upright during the perturbations. Ran-domized perturbations with loads of 180 N and durations 200 ms and 400 ms were presented five times in both left and right sides. The experimental setup is shown in Figure 1. We implemented (1)–(3) as a SIMULINK model (MATLAB 7.5.0, 2007b, The MathWorks, Inc.). Load force was measured by a dynamometer inserted in the rope between motor and handle. The rope force data was passed to the model as input signal F, delayed by a transport delay Td. Gain factors for sensory inputs and change in sensory inputs were W_c, W_c^v for the contralateral and W_{ip}, W_{ip}^v for the ipsilateral neuron, respectively. Reciprocal and recurrent connections were chosen symmetri-cally as a_{other} and a_{self}. Simulation time steps met the recording frequency of the measurement. Integration initial values were taken from measured data for $t = 0$. Depending on the model parameters, the output y_i of the model neurons was the simulated reflex response to the actual force input. Bipolar surface EMG of left and right abdominal external oblique muscles (EO) were taken with sensor application following [22]. The EMG signals and force data were digitised at 2000 Hz. Stan-dard procedures were applied for EMG data preparation. The force data was not

processed. The time of perturbation was triggered by the force signal first exceeding the static force level by more than 2.5 standard deviations. To allow relaxation of the simulation after initial interferences, datasets were trimmed to 300 ms (600 samples) of pre-trigger signal and 400 ms post-trigger signal (1401 overall samples). For the parameter optimisation, the MATLAB curve fitting toolbox was used with a nonlinear least-squares algorithm. The EMG data showed that the system dynamics was mostly characterised by the contralateral EMG. So, firstly, only the parameters for the contralateral neuron were optimised (W_c, W_c^v, R, Td, a_{self}, b, T), and the contralateral simulation was matched to the contralateral EMG data. Secondly, the adjusted parameters from the contralateral neuron were used as initial values for the ipsilateral neuron. Only a_{other}, W_{ip} and W_{ip}^v were optimised by matching ipsilateral signals. Thirdly, all parameters were optimised with initial values provided by the previous step. Both ipsi- and contralateral channels were fitted.
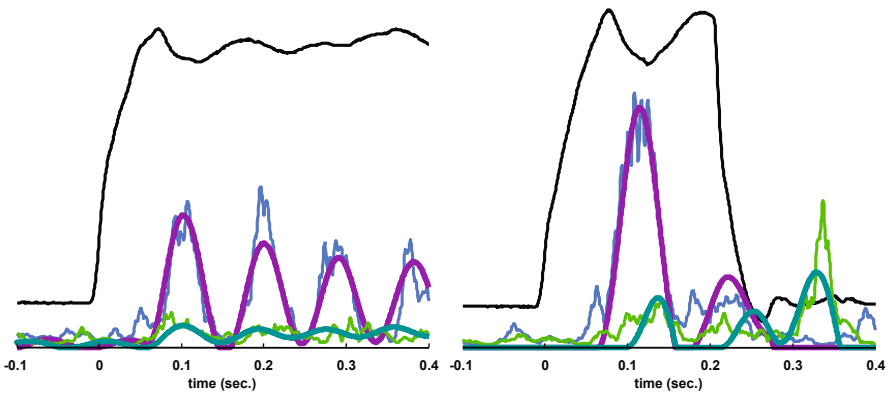

Fig. 2 Measured and optimised simulated activation time courses. Black: rope forces, blue: contralateral EMG, light green: ipsilateral EMG, purple: contralateral simulation, dark green: ipsilateral simulation. Left: long perturbation (2 s), right: short perturbation (200 ms).

The same set of initial parameters was used for all subjects in all trials. Since this set might not be suitable for every individual trial, the resulting simulations were visually inspected to eliminate trials that would not follow the measured EMG qualitatively. Temporal pattern with relative amplitudes can be compared between two datasets with the determination coefficient r^2 which represents the fraction of common variance in both datasets. The simulation with optimised parameters was compared to its corresponding EMG trial by r^2. For a measure of reflex variability, r^2 is calculated pairwise for all EMG measurement trials per subject, for the left and right trials separately.

In more than 85% of the cases, the optimisation algorithm delivered sets of parameters capable of producing plausible simulations. The quality of the reflex simulations for both perturbation types produced by the optimisation was high. Figure 2 shows representative measured EMG data and corresponding optimised

simulation for both types of perturbation. The simulations with fitted parameters resembled their corresponding EMG data with determination coefficients of $r^2 = 0.75$ and $r^2 = 0.70$ for the short and long perturbations, respectively. This resemblance was clearly stronger ($p < 0.001$) than between EMG trials. The determination coefficients for EMG show only limited intra-individual reproducibility of reflexes with $r^2 = 0.43$ and $r^2 = 0.34$ for short and long perturbations, respectively (see Figure 3).

Fig. 3 Determination coefficients r^2 for optimised simulation and corresponding EMG (left) and for EMG trial-to-trial resemblance (right). Results for short perturbations are shown in blue, results for long perturbations are shown in red.
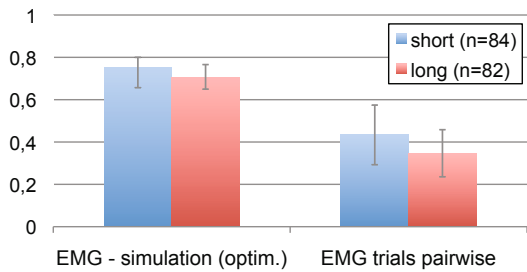
The results show that the SPG model was capable of reproducing complex reflex patterns. The perturbation signal was the only input to the model that represented a neural network on a single spinal segment. This way, the long-latency reflex responses were generated as SPG activation patterns. These findings attenuate the need for long-loop explanations for late reflex peaks. The close fit of simulation and measurement during considerable reflex variability suggested that the connection properties of the network generating the reflex pattern would change from trial to trial. It is reasonable to have supraspinal influence on SPG, in order to achieve the versatile demands of motor control tasks.

4 Example B: Rhythmic Motion at the Ankle

Dietz et al. have demonstrated that coordinated EMG patterns could be induced in human subjects with incomplete or complete paraplegia on a treadmill [4]. Further studies would demonstrate that loading and unloading of the limbs in combination with cyclic movements were necessary for these rhythmic EMG activities [6]. Thus, walking could be attributed to locomotor SPG in humans. This section presents a method which used neural oscillators to generate and examine the effects of different inputs on the rhythmic motion at the ankle [2].

7 male and 3 female healthy subjects (26.5 ± 4.3 years, 72.8 ± 10.5 kg) took part in this study. They were thoroughly informed of the procedures and gave their consent. Each subject was requested to walk at their normal self-selected speed (4.8 ± 0.5 km/h) on a treadmill. Forces were measured from in-sole pressure sensors (Gesellschaft für Biomechanik Münster, Germany) at 200 Hz. They were calculated as a summation of pressure acting on the entire area of the insole. Motion

capture was performed using Qualisys (Gothenburg, Sweden) 3D motion capture system. The motion analysis system used 6 infra-red cameras which tracked 15 retro-reflective markers attached to the following body landmarks: anterior superior iliac spine (left and right), sacrum, lateral and medial knee (left and right), 4 tracking markers on each thigh. The markers were captured at a sampling rate of 100 Hz. Muscle activation of the soleus (Sol) and the tibialis anterior (TA) was captured using bipolar electrodes (Biovision, Wehrheim, Germany) at 2000 Hz. The SPG model consisted of a simple oscillator made up of two neurons [17] [18]; One represented the tibialis anterior and the other, the soleus. Therefore, the outputs of the oscillator represented the EMG generated by each muscle. While neural oscillators are simple and effective, the parameters required to produce the required oscillations are difficult to tune. Therefore, a nonlinear least squares fitting algorithm was used to determine the set of parameters that would provide the best fitting between the output and experimental EMG data. Since the 2 main afferent sources are related to loading and hip position [5] [23], the inputs used in the oscillator were in-sole forces and hip angles. To ensure that the model produced stable oscillations, 6 consecutive steps were analysed in this model.

With two inputs, the SPG model successfully generated outputs that fit experimental data ($R = 0.81$, Figure 4). It is possible that these afferent inputs which play a role in influencing motoneurons are largely under the control of SPG. In addition, this agreed with previous studies that the 2 main afferent inputs related to walking is loading and hip position [5].
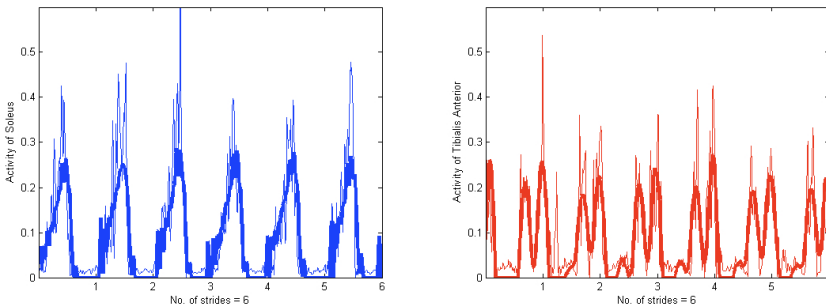
Fig. 4 Muscle activation of the tibialis anterior and soleus of one subject with in-sole force and hip angles as inputs (thin lines represent the experimental data, bold lines represent the outputs from the SPG model)

Since there seemed to be a close relation between ground reaction forces and ankle extensor activity in decerebrate cats [8], afferent input from the hip was removed to study the effects of loading alone, and to determine if it will be sufficient to activate the soleus. As shown in Figure 5, the use of different parameters enabled the model to generate another fit with loading as the sole input ($R = 0.78$). However, the resulting fit is not as good as when both inputs were used. This agreed with
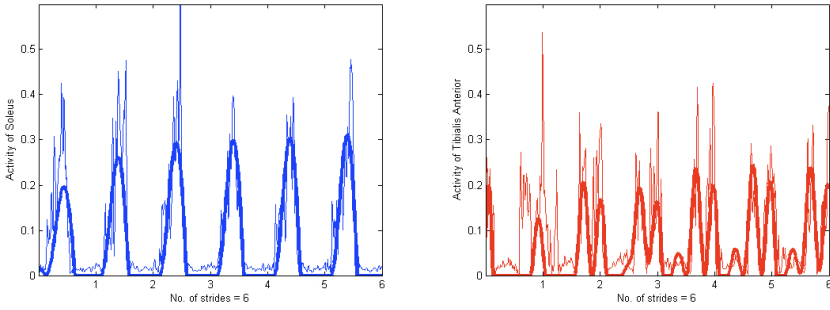
Fig. 5 Muscle activation of the tibialis anterior and soleus with loading as sole input (thin line represents the output from the SPG model, bold line represents the experimental data)

studies that activation of the ankle extensors are affected by loading of the limb during stance phase [25] [8]. However, the activation in the TA was unexpected as the effect of loading on the TA in normal and spinal-cord-injured subjects was unclear [12]. It might be also possible that loading and unloading has little or no effect on the TA [15]. However, these results showed that loading alone could produce a good activation of the Sol ($R = 0.86$), but less so in the TA ($R = 0.71$). In such a case, afferent input from the hip would be required for a better fit as extension of the hip would signal the start of swing, and thus, an activation of the TA for foot clearance.

When hip angles were used as a sole input, a different set of parameters allowed the model to also generate another output ($R = 0.76$, Figure 6). This might explain how [7] successfully induce locomotor-like EMG activity in patients (with complete spinal cord injury) while in a supine position. So while inputs from the receptors at the soles of the feet were absent, neural locomotor activity could still be strongly regulated by afferents and muscles around the hip.
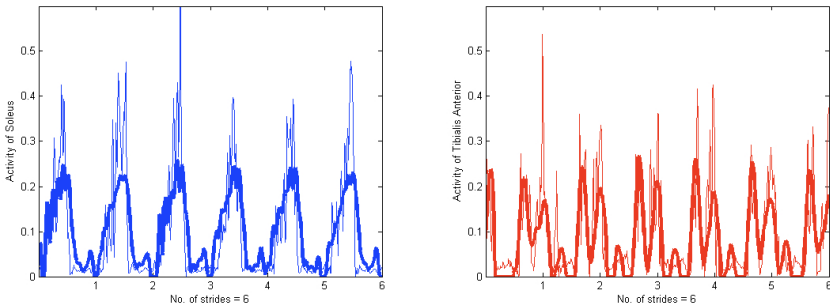

Fig. 6 Muscle activation of the tibialis anterior and soleus with hip angles as input (thin line represents the output from the SPG model, bold line represents the experimental data)

In both cases when only either input was used, changes to the parameters allowed the model to generate better fitting outputs. However, the resulting fit was not as good as using a combination of both inputs. This clearly demonstrated the ability of the SPG in adapting to changes and selecting appropriate afferents, so as to enable step-like motion to occur.

5   Example C: Phase Transition in Cyclic Arm Movements

Under certain circumstances, animals show sudden changes in motion pattern, i.e. the transition from trot to gallop. These so called phase transitions between two possible states of locomotion do not seem to be caused by conscious intervention into the movement task. In fact, there are several control parameters which determine such a transition. The phenomenon of phase transition in locomotion is not restricted to animals, there are also examples in human coordination. For example, the participants of a study done by Kelso [14] had shown an unintended phase transition from an asymmetric to symmetric finger movement while increasing the frequency.

There is evidence, that animals use SPG for locomotion. The claim that even humans are using SPG for simple cyclic movements is well accepted and can be supported by several investigations i.e. Dietz and Minassian [3][19]. If SPG can be responsible for human movement patterns, i.e. for cyclic movement, then it might be possible that phase transitions can be explained by specific properties of the neuro-muscular system containing SPG. To investigate whether phase transitions between asymmetric and symmetric movements can be explained by the cooperation between the SPG and the mechanical system, we simulated the movement of the forearm in the horizontal plane. The advantage of this model is that we can simulate the movement quite accurately with only two muscles (flexor, extensor).
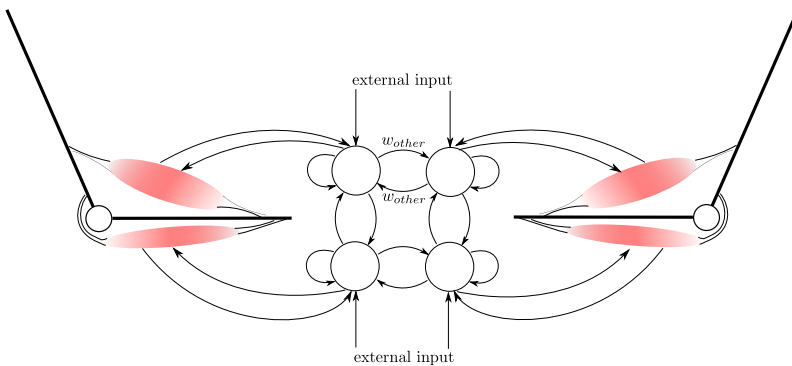

Fig. 7 Neuro-muscular model consists of 2 antagonistic pairs of muscles, 2 revolute joints and 4 mutual coupled neurons. Each neuron obtains external input u and sensory information s. The coupling between left and right neurons is called W_{other}. The activation of each muscle is given by the output of a neuron.

Furthermore, we were able to show a phase transition for increasing frequency, similar to Kelso's finger experiment, for this movement task in preliminary experiments.

To simulate the transition, we used a neuro-muscular model (see Figure 7) and evaluated the system behaviour for different parameter settings accountable for frequency changes. For the neural activation of the muscles, we used the model described above for four reciprocally and recurrently coupled neurons, representing a cluster of neurons. The output of each neuron acted as an activation on each respective muscle and thus produced a torque at the given joints. The mechanical model consisted of two elbow joints and two pairs of antagonistic muscles (flexors and extensors). In the model, the forearm motion was restricted to the horizontal plane to exclude gravity in the simulation. We neglected other external moments. The produced muscle torques consisted of lever arms, muscle activations, force-length and force-velocity functions. The distance between joint centre and muscle line of action was taken as the flexor lever arm. For the extensors, the regression equation by Gerbeaux [9] was used. The force-length function for extensors was set to 1 as an approximation (see [20]). For the flexor force-length relation, we used the term described by Giesel et al. [10]. Force-velocity relations were given by Hill's function. The parameters of the neuron model were chosen to produce a physiological arm swing in frequency and amplitude. The system behaviour (phase, frequency) was analysed for varied model parameters. Following Kelso's experiment, the analysis was focused on parameters that influenced the frequency as shown in Figure 8 for an example of two parameters. Increasing the adaptation b lead to an increase in frequency while the phase relation between the movements of both arms remained unchanged ($\Delta\varphi = \pi$). Changing the reciprocal coupling W_{other} from a negative to
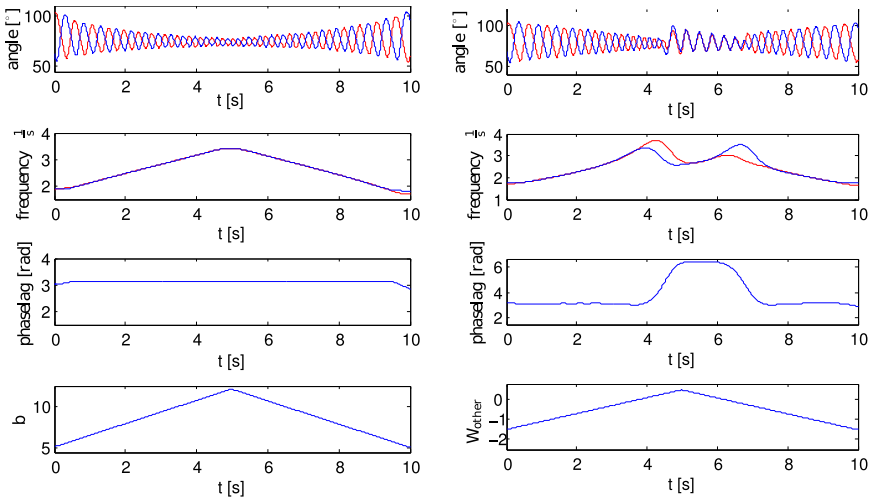
Fig. 8 Left: Changes in angle time course (right arm=red, left arm=blue) frequency and phase lag during variation of b. Right: Only variation of W_{other} from inhibition to excitation causes a transition from asymmetric to symmetric arm movements.

a positive value induced a transition in the range of $W_{\text{other}} = 0$. The arm movement shifted from asymmetric to symmetric coordination within a few cycles. Since no other parameter in equations (1)–(5) can produce a transition in a physiological range of frequency, a change in W_{other} might be the manner in which humans control the frequency. This assumption needs the evidence of both inhibitory and excitatory connections between the left and right sides of the body. Inhibitory connections have been found in the isolated spinal cord of the lamprey [1]. Evidence for excitatory interconnections would support the assumption that the increase in frequency is achieved by changing the coupling parameter.

6   Conclusion

Our results support the hypothesis, that depending on the movement task, humans—and some animals—are able to change the coupling between the neurons within the spinal pattern generator. This provides an extremely simple way to control a large variety of cyclic and acyclic movements, based on a small number of control parameters. Thus, the stability of the system is guaranteed based on a stable SPG, a mechanical self-stabilization and the coupling between these two systems.

References

 1. Buchanan, J.T., Grillner, S.: Newly identified 'glutamate interneurons' and their role in locomotion in the lamprey spinal cord. Science 236(4799), 312–314 (1987)
 2. Chong, S.Y., Wagner, H., Wulf, A.: Neural oscillators triggered by loading and hip orientation can generate activation patterns at the ankle during walking in humans. Medical and Biological Engineering and Computing 50, 917–923 (2012), 10.1007/s11517-012-0944-2
 3. Dietz, V.: Do human bipeds use quadrupedal coordination? Trends Neurosci. 25(9), 462–467 (2002)
 4. Dietz, V., Colombo, G., Jensen, L., Baumgartner, L.: Locomotor capacity of spinal cord in paraplegic patients. Annals of Neurology 37(2), 574–582 (1995)
 5. Dietz, V., Duysens, J.: Significance of load receptor input during locomotion: A review. Gait and Posture 11(6), 102–110 (2000)
 6. Dietz, V., Mueller, R., Colombo, G.: Locomotor activity in spinal man: Significance of afferent input from joint and load receptors. Brain 125(3), 2626–2634 (2002)
 7. Dimitrijevic, M., Gerasimenko, Y., Pinter, M.: Evidence for a spinal central pattern generator in humans. Annals New York Academy of Sciences 860(12), 360–376 (1998)
 8. Donelan, J., Pearson, K.: Contribution of force feedback to ankle extensor activity in decerebrate walking cats. Journal of Neurophysiology 92(8), 2093–2104 (2004)
 9. Gerbeaux, M., Turpin, E., Lensel-Corbeil, G.: Musculo-articular modelling of the triceps brachii. Journal of Biomechanics 29(2), 171–180 (1996)
10. Giesl, P., Meisel, D., Scheurle, J., Wagner, H.: Stability analysis of the elbow with a load. Journal of Theoretical Biology 228(1), 115–125 (2004)

11. Grillner, S.: Biological pattern generation: the cellular and computational logic of networks in motion. Neuron 52(5), 751–766 (2006)
12. Harkema, S., Hurley, S., Patel, U., Requejo, P., Dobkin, B., Edgerton, V.: Human lumbosacral spinal cord interprets loading during stepping. Journal of Neurophysiology 77(10), 797–811 (1997)
13. Ijspeert, A.J., Crespi, A., Ryczko, D., Cabelguen, J.M.: From swimming to walking with a salamander robot driven by a spinal cord model. Science 315(5817), 1416–1420 (2007)
14. Kelso, J.A.S., Holt, K.G., Rubin, P., Kugler, P.N.: Patterns of human interlimb coordination emerge from properties of Non-Linear, limit cycle oscillation processes: Theory and data. Journal of Motor Behavior 13(4), 226–261 (1981)
15. Klint, R., Nielsen, J., Sinkjaer, T., Grey, M.: Sudden drop in ground support produces force-related unload response in human overground walking. Journal of Neurophysiology 101(11), 1075–1012 (2009)
16. Lewis, G.N., Perreault, E.J., MacKinnon, C.D.: The influence of perturbation duration and velocity on the long-latency response to stretch in the biceps muscle. Exp. Brain Res. 163(3), 361–369 (2005)
17. Matsuoka, K.: Sustained oscillations generated by mutually inhibiting neurons with adaptation. Biological Cybernetics 52(4), 367–376 (1985)
18. Matsuoka, K.: Mechanisms of frequency and pattern control in the neural rhythm generators. Biological Cybernetics 56(5), 345–353 (1987)
19. Minassian, K., Jilge, B., Rattay, F., Pinter, M.M., Binder, H., Gerstenbrand, F., Dimitrijevic, M.R.: Stepping-like movements in humans with complete spinal cord injury induced by epidural stimulation of the lumbar cord: electromyographic study of compound muscle action potentials. Spinal Cord 42(7), 401–416 (2004)
20. Murray, W.M., Buchanan, T.S., Delp, S.L.: The isometric functional capacity of muscles that cross the elbow. Journal of Biomechanics 33(8), 943–952 (2000)
21. Nakajima, K., Maier, M.A., Kirkwood, P.A., Lemon, R.N.: Striking differences in transmission of corticospinal excitation to upper limb motoneurons in two primate species. J. Neurophysiol. 84(2), 698–709 (2000)
22. Ng, J.K., Kippers, V., Richardson, C.A.: Muscle fibre orientation of abdominal muscles and suggested surface emg electrode positions. Electromyogr. Clin. Neurophysiol. 38(1), 51–58 (1998)
23. Pang, M., Yang, J.: The initiation of the swing phase in human infant stepping: Importance of hip position and leg loading. Journal of Physiology 528(7), 389–404 (2000)
24. Schuurmans, J., de Vlugt, E., Schouten, A.C., Meskers, C.G.M., de Groot, J.H., van der Helm, F.C.T.: The monosynaptic ia afferent pathway can largely explain the stretch duration effect of the long latency m2 response. Exp. Brain Res. 193(4), 491–500 (2009)
25. Sinkjaer, T., Andersen, J., Ladouceur, M., Christensen, L., Nielsen, J.: Major role for sensory feedback in soleus emg activity in the stance phase of walking in man. Journal of Physiology 523(9), 817–827 (2000)
26. Windhorst, U.: Muscle proprioceptive feedback and spinal networks. Brain Res. Bull. 73(4-6), 155–202 (2007)
27. Wulf, A., Wagner, H., Wulf, T., Schinowski, D., Puta, C., Anders, C., Chong, S.Y.: Phasic bursting pattern of postural responses reflect internal dynamics: Simulation of trunk reflexes with a neural oscillator model. J. Biomech. 45(15), 2645–2650 (2012), doi:http://dx.doi.org/10.1016/j.jbiomech.2012.08.010

Modeling Human-Like Joint Behavior with Mechanical and Active Stiffness

Thomas Wahl and Karsten Berns

1 Introduction

The bipedal locomotion has become a research topic of great interest over the last decades. Several groups have realized dynamic walking on afore known terrain. Our goal is to realize dynamic walking and running in rough terrain. Beside the control of the highly dynamic and unstable motion, the actuator itself is still an open research topic. The main task of such an actuator is to supply the system with energy to compensate losses due to internal friction and environmental impact. In slow walking speeds this share is very low. Several so-called passive walkers have shown that there is no need for an actuator, if the energy loss is compensated by transforming potential energy into kinetic energy [13, 4]. But in more dynamic motions like running and jumping, however, a powerful actuator is essential.

Beside the energy supply, the actuator is required to control the robot's movements. One approach solves this by mere permutation of the control signals from higher behaviors. Most of the today's bipedal robots use the actuator in that way. A stiff actuator is combined with a gear box that possesses a high gear ratio like e.g. harmonic drives. The drive follows their pre-calculated joint trajectories, which are based on multi-body dynamics and Zero-Moment Point considerations [9, 16]. The resulting walking and pseudo running gaits often look unnatural, are highly uneconomical and the computational overhead is tremendous.

Dynamic walking in rough terrain requires an adjustable compliant actuator. An elastic behavior is required to avoid hard impacts at touchdown and a fast adaption to the uneven terrain. Besides that in normal walking gaits the leg should be able to rely on the mechanical dynamics by just relaxing the joint. On the other hand the joint has to be very stiff when the leg is in the support phase.

Thomas Wahl · Karsten Berns
Robotics Research Lab, University of Kaiserslautern, D-67663 Kaiserslautern, Germany
e-mail: {t_wahl,berns}@informatik.uni-kl.de

A more biological motivated approach allows the system to use the inherent dynamics and self stabilizing effects. This is achieved by an innately compliant joint, like it can be observed in all mammals. The loose joint can be stiffened by contracting the muscles. Their typical characteristic is described by the hill-type muscle model [7].
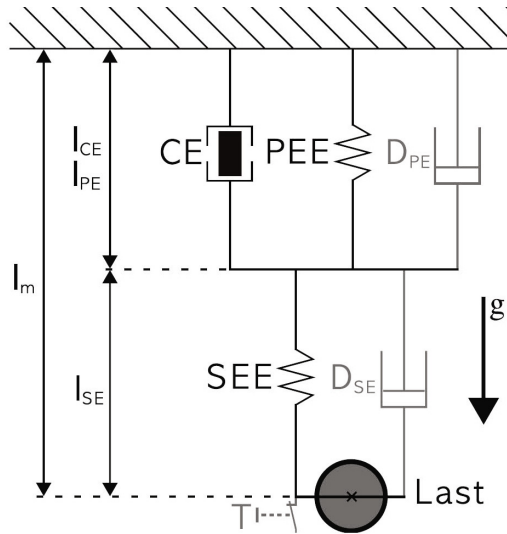

Fig. 1 Hill-type muscle model

In this antagonistic principle the nonlinear behavior of the muscles and tendons allows to adjust the stiffness. Their characteristic is described by the serial spring in the hill-type muscle model. Furthermore, the fascia of the muscle provides a kind of parallel spring. That allows all vertebrates to store energy in cyclic motions (Fig 1). Especially during running and jumping the amount of stored energy is quite high. Several groups have tried to copy and rebuild this behavior with electric motors and mechanical springs [5, 15, 6] or with artificial muscles [8]. Up to now, no satisfying and energy efficient solution for this problem could be observed.

2 Demonstrators

Based on this biologically motivated actuator concept Luksch [11] developed a control strategy for a whole biped (Fig 2). The control strategy is supported by the self stabilizing effects of the actuator and does not require a complete model of the biped. The backbone of the control is a kind of pattern generator that is triggered by external events. This external trigger allows the robot to handle uneven terrain without any parameter adaption or environment cognition. The idea behind this approach is

Fig. 2 Simulated biped walking up a step [12]

explained in more detail in the paper of Tobias Luksch that can be also found in this book and in [12].

The real demonstrator was built based on two biologically motivated actuators (Fig 3). One represents the hip, the other stands for the knee and is assisted by a parallel spring. The leg has a total weight of 17 kg and has a height of ca. 1 m. The high weight is a result of the modular concept with aluminum tubes. The modular concept allows equipping the leg with different foot setups.

It utilizes sparse sensor equipment as it can be found in biology. At each joint an optical encoder measures the actual position of the leg. These optical sensors are comparable to the *muscle spindles*. The spindles are located inside the muscle in
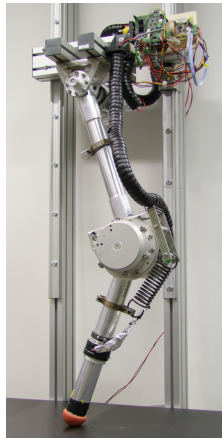
Fig. 3 Demonstrator with rubber sphere

parallel to the muscle fibers. The muscle fibers themselves have also sensory neurons. They send information about the muscle deflection to the spinal cord.

A load cell is placed in the tibia to measure the force in the z direction and the torque in x and y direction. The load cell in combination with the current measurement represents the *golgi tendon organ*. The organ measures the tension inside the muscle. It is located in parallel to a passive element between the muscle and the tendon that connects the muscle to the bone. The output of the connected sensory neuron is proportional to the load of the muscle.

There is no need for an inertial system to estimate the pose of the jumping leg, because it is fixed to a slider. The *vestibular system* in mammals is the counter part to the technical inertial system. It is located in the inner ear, measures angular velocity and linear acceleration, including gravity. This sensor is an integral part of the sense of balance in humans. It is supported by the vision system.

3   Mechanics

The actuated joints consist of a DC motor, a gearbox with low gear ratio and a parallel elastic element (Fig 4).

A rotor disc motor is chosen (Heinzmann SL120-2NFB), because it offers very good dynamic properties due to its low inertia ($J = 3,5kgcm^2$) and high peak torque. The selected model offers an absolute zero motion torque of approximately $13.72\,Nm$. The supply voltage U_1 is 48 V and the maximum output power at 3164 rpm is 430 W.

Equipped with a gear ratio of 32 : 1 it results in a realistic maximum obtainable torque of approx. $150\,Nm$. The low gear ratio allows for a free swing phase in comparison to the normally used harmonic drives. Tests in the simulation have shown that a maximum peak torque of 150 Nm is enough for jumping with a single leg [2] and walking with a human sized robot [10].

Every motor has an optical encoder with a resolution of 720 lines per rotation. Every line on the encoder wheel leads to four ticks in the counter unit. While the encoder is placed between the motor and the gearbox, the resulting 2880 ticks have to be multiplied by the gear ratio. Hence the resolution of the output knee angle is 256 ticks per degree. This high resolution is not required for an exact positioning but rather for the underlying speed controller.

To store energy during the squat phase the knee is extended by a parallel spring. There are two different approaches to realize this parallel elastic behavior. The first one is a pneumatic rotary spring. Two shunts inside a casing compress the air when the knee is flexed. The nonlinear behavior of the rotary spring is similar to the fascia. Problems during the assembly process led to a more damping-like characteristic.

Furthermore, the weight of the spring (due to the high pressured air, the casing is thick) is very high ($> 1.5kg$). To overcome these drawbacks a mechanical spring is used instead.

The hip is not equipped with such a parallel spring. This is because tests in the simulation have shown that there is no benefit. The energy that is saved in the squat
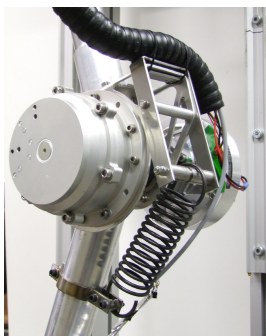
Fig. 4 Knee joint with spring in front and actuator in the back

and jump off phase is wasted again in the air. It requires a lot of energy to reach and hold the landing position against the parallel spring.

The serial spring in the Hill-type muscle model is implemented in the foot setup and the compliant behavior of the actuator. The foot setup consists of a simple rubber sphere, a mechanical foot with serial springs or a foot made of carbon fibers. The foot is important because at the moment it is the only serial elasticity, which can absorb hard shocks at touch down. If the first shocks are damped by the mechanical serial elasticity the physical stress for the gearbox and the motor are reduced. Due to the low inertia of the motor in combination with the gearbox no active compliance is required.

4 Electronics

The motor is controlled via a DSP-board (Digital Signal Processor) with two motor amplifiers (Fig 5). The Active stiffness controller is implemented on a *freescale 56F8357* DSP. The DSP is connected to a CPLD to receive a modular electronic concept that can be used in almost arbitrary systems. The CPLD allows rerouting some data lines to the IO-ports. The digital IO-ports are used to connect the optical encoders and the load cell. Furthermore the setup allows to switch some of the digital IO-ports to analog inputs and to implement some smaller sensor pre processing. For example a quadrature decoder for the optical encoders or the communication with the load-cell is implemented.

One amplifier consists of four MOSFETs and a controller chip from Vishay. The DSP-board further offers a PWM-synchronous (Pulse Width Modulation) current measurement. This information is essential for the implemented torque controller. The PWM-frequency is 39 kHz which is too fast for the internal A/D-converter of the DSP to measure every cycle. Therefore, every fourth cycle the current is measured over an instrument shunt $R2$ (Fig 6). The current version of the DSP-board can hold a continuous current of 15 A.
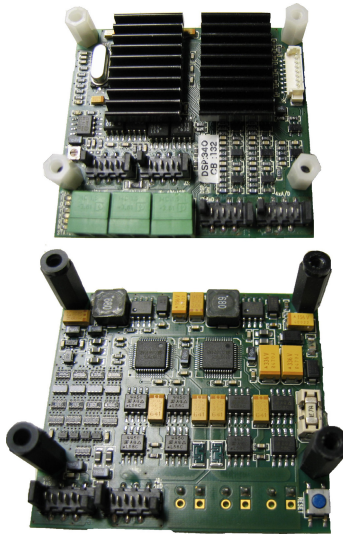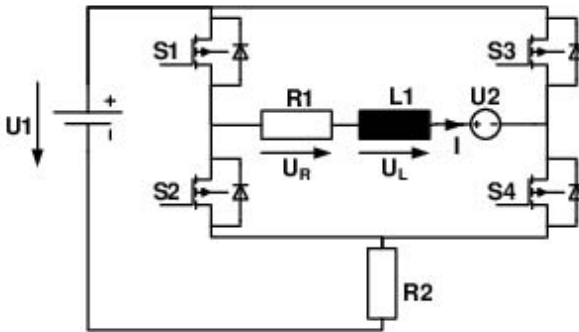
Fig. 5 DSP-board with motor amplifiers

Fig. 6 Standard DC-motor model with MOSFETs and instrument shunt $R2$

The board is connected to the PC via CAN-bus. This topology allows implementing higher behaviors on a PC, which need more computational power and are not time critical. This is also comparable to what is found in nature. The brain on the one hand has a lot of computational power, but it takes several milliseconds until the command is executed by the muscles. The reflexes on the other hand have short sensor actuator couplings. Their behavior is modulated in the spinal cord and doesn't require any decisions from the brain. The brain only has the possibility to influence the behavior by stimulating or inhibiting the whole reflex.

The communication between the available boards and the PC is handled automatically in the initialization phase at start up. It is not required to adapt any parameter,

if a second board is connected. Therefore, it is easy to attach further DSP-boards if the degrees of freedom are increased.

The joint controller adjusts the stiffness of the joint. To perform this highly dynamic task the controller is built up hierarchically.
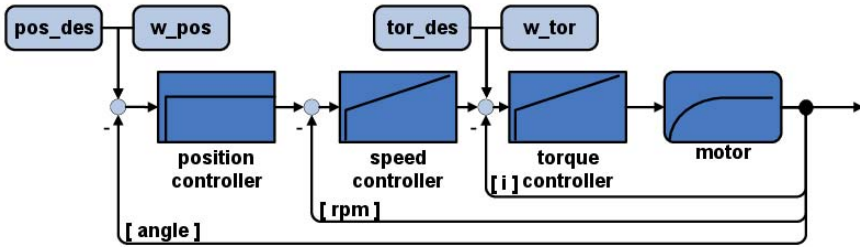

Fig. 7 Schematic view of the hierarchical closed loop controller

The innermost loop is a fast torque controller with a cycle time of $1\,ms$. The electrical time constant of the used motor is $0.27\,ms$. That is three times faster then the loop time of the controller. To overcome this disproportion the current is measured every $0.10\,ms$ (see above). A first order low pass filter is used to ensure the sampling theorem holds:

$$y(n) = (1 - 2^{-\alpha})y(n-1) + 2^{-\alpha}x(n) \text{ with } \alpha \in \mathscr{Z}$$

Due to the fast switching frequency of the PWM signal, there are two different risen characteristics of the current. In the first case the duty cycle is below 50% and the current returns to zero in the off-phase. Because of the large inductance the current doesn't reach the saturation of $I_{max} = U_1/R_{motor}$. This leads to a quadratic relation between the current and the duty cycle (Fig 8). While the voltage at the inductance U_L is nearly constant for the short period the slope of the triangle is given by:

$$U_L = L\frac{dI}{dt} \Rightarrow I = \frac{1}{L}\int U_L \approx \frac{U_1}{L} \cdot t$$

When the motor starts to turn this relation does not hold, since the voltage at the inductance is reduced by the back e.m.f. U_2.

In the second case the duty cycle is above 50% and the current does not return to zero in the off-phase. That leads to an increasing current from cycle to cycle until the current reaches the saturation (Fig 9). Although the current is shrinking in the off-phase the electrical time constant does not change until the motor reaches the saturation voltage.

In both cases the average current reaches the persistent value faster then the loop time of the controller. That is the background why 1 ms loop time is enough for the torque controller. From cycle to cycle the new value is the persistent value for the
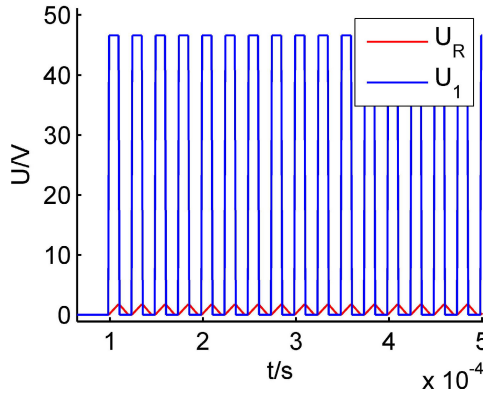
Fig. 8 PWM duty cycle step from 0% to 42%: blue line modulated motor input voltage U_1, red line voltage U_R at the resistance $R1$ proportional to current I
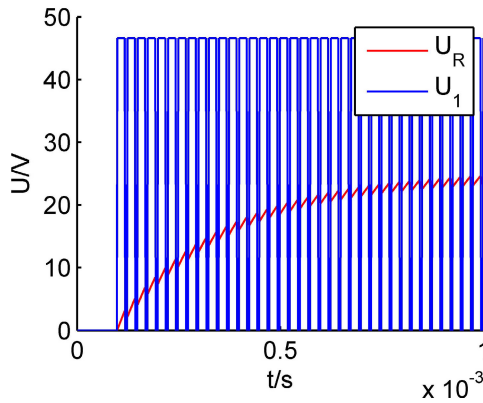

Fig. 9 PWM duty cycle step from 0% to 73%: blue line modulated motor input voltage U_1, red line voltage U_R at the resistance $R1$ proportional to current I

given PWM output from the last cycle. The controller itself is realized as a standard PI-controller.

Based on this torque controller a speed and a position controller are implemented. A schematic layout and the interfaces of the controller are presented in Fig 7. The mechanical time constant $\tau_{mec} = 5.8\,ms$ is by more than a magnitude slower than the electrical time constant. Therefore, the cascaded controller structure is possible, because the closed loop controller of the current can be approximated as a static system.

Fig 10 shows a step of the motor's input voltage from 0% to 94% PWM duty cycle. The current represented by the red line rises very fast (0.27 ms) until the motor starts to turn. With the motor speed the back e.m.f. U_2 increases and lowers the
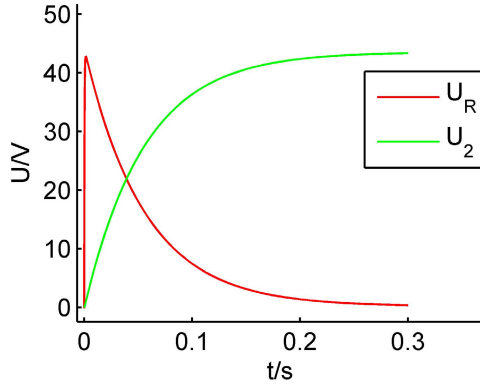
Fig. 10 PWM duty cycle step from 0% to 94%: green line is equivalent to the motor speed represented by back e.m.f. U_2, red line voltage U_R at the resistance $R1$ proportional to current I

resulting voltage on the resistance U_R. A persistent motor speed is reached at $U_2 \approx U_1$. The speed controller is also implemented as a standard PI-controller and has no interface to the higher reflex layer. Biology shows that there is no need for an exact speed control.

The outermost controller is the position controller. The position controller requires only a simple P-controller, because the integral portion is given by the system. Therefore, the combination of position and speed controller is not time critical as it is between speed and current controller.

The desired position and the desired torque impact of the controller output can be set using respective weight parameters named w_{pos} and w_{tor}:

$$current_{des} = \frac{w_{pos}^2 * tor_{pos} + w_{tor}^2 * tor_{des}}{w_{pos} + w_{tor}} \tag{1}$$

By decreasing w_{pos} the stiffness at a desired position is reduced, because the influence on the applied torque is reduced. w_{tor} is proportional to the influence of a desired torque. This is e.g. the case during the push-off phase: there is no need for an exact position since the maximum torque is required ($w_{pos} = 0$ and $w_{tor} = 1$). To hold a fixed desired position $w_{pos} = 1$ and $w_{tor} = 0$.

5 Result of Test

To verify the capabilities of the actuator several tests were performed. A result of one test is shown in Fig 11. A sinus was applied to the inputs of the position controller of hip and knee. This leads to a squat movement with a short flight phase. The position could be controlled in a very accurate way (blue), while the underlying controller

show more and more noise (red=speed, green = current). This is due to the fact that the lower controllers must be faster to allow cascading of integral closed loop controllers [14].
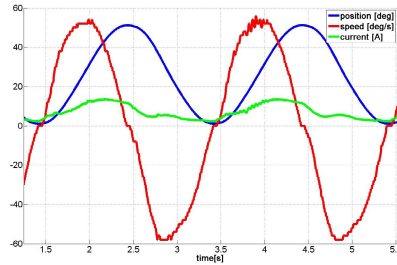
Fig. 11 Real data from the knee with sinus on input

6 Outlook

Tests in biology have shown that there is a coupling between joints on two layers [1]. On the lowest layer a coupling by bi-articular muscles is given (e.g. m. rectus femoris, m. gastrocnemius). This connection allows executing complex motions by activating a single muscle. Furthermore, the connection of more than one joint allows an easy compensation of perturbations. For example if the knee is flexed the gastrocnemius is responsible for an extension of the ankle joint, which compensates the leg length deviation.

In addition to the classic biomechanics several groups have shown that there is a neural representation for kinematic elements on a higher level. At the level of the spinal cord in the central nervous system specific populations of neurons represent the whole limb geometry [3]. This becomes apparent when for example a flexion perturbation is applied to the knee. It stretches the quadriceps muscle which triggers a heteronymous excitation of the triceps surea muscle group. The result of this reflex is a compensatory knee extension.

In the current version of our prototype leg the only coupling between the two joints is given by the PC. This connection is very slow (worst case >50 ms). That is too slow for the fast reflexes, in comparison to the mono-synaptic reflexes in extremities of humans.

At the moment the topology does not allow data transfer between DSP notes. For this reason, first tests with the *flexray* interface are in progress. The real-time capabilities of the interface allow implementing an electronic coupling between the joints. A cycle time of fewer than five milliseconds should be enough to realize the bi-articular coupling of several joints.

On the mechanical side the prototype leg is too heavy due to the aluminum tubes and connectors, the gearbox and the slider. By exchanging the tubes through carbon fibers tubes a weight reduction of up to 1 kg is possible. The weight of 1.5 kg for one

gearbox is also quite heavy. By replacing the compact gearbox with a wire driven actuator the weight of the leg is reduced. The wire driven joint has some further advantages. It allows an easy integration of a serial spring as it can be found in the Hill-type muscle model. And the inertia of the whole leg is reduced, because the heavy knee actuator is placed at the hip. The slider is built up on standard aluminum profiles and holds further potential for a weight reduction.

On the electrical side the enhancement of the amplifiers is in progress. In the next amplifier generation the continuous current is doubled. That leads to a four times higher output power.

7   Conclusion

In this paper an actuator is presented that models human-like joint behavior with mechanical and active stiffness. It consists of a DC-motor with a high torque output and a gearbox with a low ratio. This innately compliant joint allows a passive swing phase like it can be found in mammals. Based on this mechanical compliant actuator a controller is presented which has the possibility to adapt the stiffness with one single parameter.

The stability of the cascaded controller is given by the significant different time constants of electrical and mechanical part. This criterion is proven with a theoretical model of the motor. The capabilities of the real actuator are tested with a prototype leg. Tests have shown that the actuator is powerful enough to jump with the weight of ca. 17 kg (about 10 cm).

References

1. Auyang, A.G., Yen, J.T., Chang, Y.H.: Neuromechanical stabilization of leg length and orientation through interjoint compensation during human hopping. Experimental Brain Research 192(2), 253–264 (2009)
2. Blank, S., Wahl, T., Luksch, T., Berns, K.: Biologically inspired compliant control of a monopod designed for highly dynamic applications. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, St. Louis (MO), USA, pp. 148–153 (2009)
3. Bosco, G., Poppele, R.E.: Proprioception from a spinocerebellar perspective. Physiol. Rev. 81(2), 539–568 (2001)
4. Collins, S., Ruina, A., Tedrake, R., Wisse, M.: Efficient bipedal robots based on passive-dynamic walkers. Science 307, 1082–1085 (2005)
5. Curran, S., Orin, D.: Evolution of a jump in an articualted leg with series-elastic actuation. In: IEEE International Conference on Robotics and Automation (ICRA), Pasadena, CA, USA, pp. 978–971 (2008) ISSN 978-1-4244-1647-9
6. Ham, R.V., Damme, M.V., Vanderborght, B., Verrelst, B., Lefeber, D.: Maccepa, the mechanically adjustable compliance and controllable equilibrium position actuator. In: Proceedings of the 10th International Conference on New Actuators (ACTUATOR), Bremen, Germany (2006)

7. Hill, A.: The mechanics of active muscles. Proceedings of the Royal Society of London 141, 104–117 (1952)
8. Kerscher, T., Zöllner, J., Dillmann, R., Stella, G., Caporaletti, G.: Model and control of compliant joints driven by fluidic muscles. In: ACODUASIS-Workshop, Torino, Italy (2005)
9. Loeffler, K., Gienger, M., Pfeiffer, F., Ulbrich, H.: Sensors and control concept of a biped robot. IEEE Transactions on Industrial Electronics 51(5), 278–246 (2004) ISSN 0278-0046
10. Luksch, T., Berns, K.: Controlling dynamic motions of biped robots with reflexes and motor patterns. In: Fourth International Symposium on Adaptive Motion of Animals and Machines (AMAM), Cleveland, USA, pp. 115–116 (2008)
11. Luksch, T., Berns, K.: Initiating normal walking of a dynamic biped with a biologically motivated control. In: Proceedings of the 11th International Conference on Climbing and Walking Robots (CLAWAR), Coimbra, Portugal (2008)
12. Luksch, T.: Human-like Control of Dynamically Walking Bipedal Robots. Verlag Dr. Hut (2010) ISBN 978-3-86853-607-2
13. McGeer, T.: Passive dynamic walking. International Journal of Robotics Research 1(9), 62–82 (1990)
14. Pfaff, G., Meier, C.: Regelung elektrischer Antriebe II, 3rd edn. Oldenbourg Verlag (1992) ISBN 3-486-22376-3
15. Pratt, J., Krupp, B., Morse, C.: Series elastic actuators for high fidelity force control. Industrial Robot Journal 29(3), 234–241 (2002)
16. Takenaka, T., Matsumoto, T., Yoshiike, T., Shirokura, S.: Real time motion generation and control for biped robot - 2nd report: Running gait pattern generation. In: Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 1092–1099. IEEE Press (2009)

Geometry and Biomechanics for Locomotion Synthesis and Control

Katsu Yamane

Abstract. This paper summarizes two pieces of work related to human locomotion. The common hypothesis underlying these works is that human locomotion is characterized by, and possibly optimized to, the inherent mechanical and sensory-motor network structures. The first work investigates the effect of foot geometry on the walking speed and efficiency. Inspired from passive walk, we consider a foot shape with circular toe and heel segments, and optimize the gait for different toe and heel radii. We then compare the optimized gaits and demonstrate that round foot realizes faster and more efficient gaits. The second work focuses on the time delay of the human somatosensory reflex. Humans can walk robustly despite the tens of milliseconds of latency between the sensor input and motor output. To investigate how time delay affect the reflex model, we build somatosensory reflex models assuming different latency values and perform cross validation across multiple motions. The result shows that the network model using the physiologically realistic latency value better generalizes to a wide variety of motions, suggesting that the network is optimized to the inherent latency of the neural system.

1 Introduction

Human-like motions and robust controllers are essential for many applications of biped locomotion. Although biped robots with locomotion capability are not uncommon these days, it is still challenging to realize locomotion comparable to human in terms of efficiency, speed, robustness, adaptability and smoothness.

A typical approach to realize active dynamic walking is to first generate a physically feasible motion based on a simplified robot model and predefined foot trajectories, and then apply a joint servo controller to track the generated motion. A

Katsu Yamane
Disney Research, Pittsburgh, USA
e-mail: kyamane@disneyresearch.com

---

balance controller is usually added to maintain balance under modeling errors and disturbances. Using this approach, researchers have realized locomotion in various environments including rough terrains [9] and stairs [7].

This approach has been proved to be an effective and practical engineering solution to realize many biped motions. However, the gaits tend to be less efficient compared to human locomotion because of the high gains used to track the trajectory. In addition, the locomotion style is usually very different from human because, for example, the knee joints are bent to improve the robustness and the feet are maintained parallel to the ground to make balancing easier.

Inspired by biomechanical observations, some researchers have investigated bipedal locomotion with entirely different approaches. One of such approaches is passive walk [12], where the key idea is to realize efficient locomotion with no or little control by exploiting the geometry and mechanism of the leg and foot. While passive walkers often realize human-like and efficient gaits, they are not robust or adaptable enough to be applied to general humanoid robots.

Another group of researchers have tried to incorporate sensory-motor network of animals to improve the robustness and adaptability. For example, the pioneering work by Taga et al. [20] used central pattern generators to generate cyclic movements that are highly adaptive to environment changes. Recent work in graphics [23] also suggests that simple, reflexive control can realize robust and adaptive control.

This paper summarizes two pieces of work related to human locomotion. The common hypothesis underlying these works is that human locomotion is characterized by, and possibly optimized to, the inherent mechanical and sensory-motor network structures.

The first work investigates the effect of foot geometry on the gait and efficiency of bipedal locomotion [22]. Inspired from passive walk, we consider a foot shape with round toe and heel segments with a flat section in the middle, and optimize the gait for different toe and heel radii. We then compare the gaits and demonstrate that round foot realizes faster and more efficient gaits.

The second work focuses on the time delay of the human somatosensory reflex [15]. Humans can walk robustly despite the tens of milliseconds of latency between the sensor input and reflexive reaction. To investigate how time delay affect the reflex model, we build somatosensory reflex models assuming different latency values and perform cross validation across multiple motions. The result shows that the network model using the physiologically valid reflex latency value better generalizes across a wide variety of motions, suggesting that the network is optimized to the inherent latency of the neural system.

2   Foot Geometry and Locomotion [22]

Foot shape and leg mechanism are one of the key design issues in passive walk. Their feet typically have curved shapes [12, 6] or flat shape with torsional spring at the ankles [21]. Kwan and Hubbard [10] discussed the optimal foot shape considering

point, flat and fully curved feet. In biomechanics, a study showed the advantages of the curved, flexible sole in humans, such as less metabolic cost for arcs with larger radius [1].

One of the problems of passive walk is that it cannot be applied to general-purpose humanoid robots directly because of the low adaptability to different environments or walk parameters. In this section, we investigate the effect of foot sole shapes on the locomotion of planar active biped robots. For this purpose, we consider a foot composed of two curved sections at the toe and heel, connected by a flat section in the middle (Fig. 1). We develop an algorithm to obtain the gait that minimizes the squared ankle torque for a given foot shape. Our results suggest that having curved toe and heel realizes more efficient locomotion at speeds comparable to human.
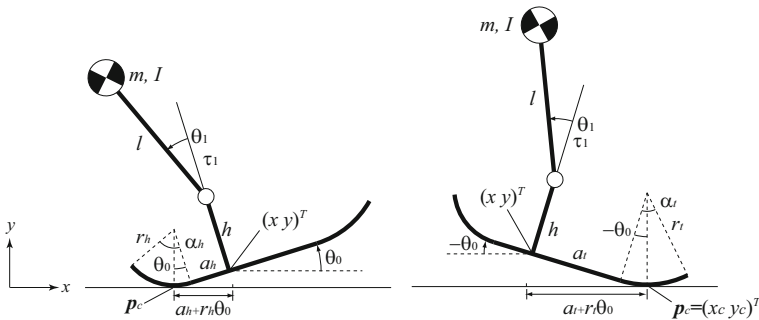

Fig. 1 The foot model used in our analysis

2.1 The Biped Model

We consider a 2D biped robot whose foot consists of circular segments at the toe and heel, and a flat section in the middle. This choice is inspired by the circular feet used in passive walking robots and passive toe joints used in some fully-actuated biped robots. The curved toe and heel parts have a similar effect to adding a passive joint because the ground reaction torque around the toe will be proportional to the angle of the foot with respect to the ground.

To simplify the optimization problem, we employ a simple bipedal robot model. We represent the mass of the entire robot as a single rigid body with mass m and inertia I. Each leg has one joint at the ankle but no knee joint. The rationale behind not having a knee joint is that the knee joint is likely to be locked at the joint limit in the supporting leg, and the knee joint torque is always zero in the free leg because we ignore the mass of the foot and leg.

The variables used in the rest of the paper are summarized in Fig. 1 and reviewed in the rest of this subsection.

The configuration of a single leg is fully determined by four parameters: (x, y) representing the position of a point fixed to the foot, θ_0 the angle of the flat section with respect to the horizontal ground, and θ_1 the ankle joint angle. We can therefore define the generalized coordinates as $q = (x\ y\ \theta_0\ \theta_1)^T$. The generalized coordinates are subject to one of the kinematic constraints corresponding to heel ($\theta_0 > 0$), toe ($\theta_0 < 0$) and flat ($\theta_0 = 0$) contact states.

We can also define the corresponding generalized forces as $\tau = (f_x\ f_y\ \tau_0\ \tau_1)^T$, where $(f_x\ f_y)^T$ is the force applied to the foot link, τ_0 is the torque applied to the foot link around (x, y), and τ_1 is the ankle joint torque. The only element of τ that can be actively selected by a controller is τ_1, while other forces are provided by the contact force.

If we assume that the foot is always in contact with the ground, y is uniquely determined by θ_0 and θ_1 due to the contact constraint. If we further assume that there is no slip, x is also uniquely determined by the trajectories of θ_0 and θ_1 for a given initial x. Therefore, the motion of a leg is fully determined by the trajectories of θ_0 and θ_1.

Using Lagrange's equation of motion, we can derive the equation of motion of a single leg in the following form:

$$\tau = M\ddot{q} + c + g \tag{1}$$

where M is the mass matrix, c denotes the centrifugal and Coriolis forces, and g denotes the gravitational force.

It should be noted that not all combinations of $\ddot{\theta}_0$ and $\ddot{\theta}_1$ are physically feasible at a particular state because the system is underactuated. We use $\Delta\tau$, the torque around the contact point $(x_c\ y_c)^T$ required to realize the given $\ddot{\theta}_0$ and $\ddot{\theta}_1$, as the measure of physical infeasibility of the motion because the point contacts at heel and toe cannot provide $\Delta\tau$. We can calculate $\Delta\tau$ by first computing τ using Eq.(1) and then computing the equivalent torque around the contact point by

$$\Delta\tau = \tau_0 + (y_c - y)f_x - (x_c - x)f_y \tag{2}$$

in the toe or heel contact case. If the foot is in flat contact ($\theta_0 = 0$), the motion is physically feasible if the center of pressure is in the flat section of the foot. $\Delta\tau$ therefore becomes

$$\Delta\tau = \begin{cases} \tau_0 - a_t f_y & \text{if } \tau_0 - a_t f_y > 0 \\ \tau_0 + a_h f_y & \text{if } \tau_0 + a_h f_y < 0 \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

2.2 Collisions

There are two types of collisions in each walk cycle of our foot model (Fig. 2): a) a swing leg makes a new contact with the ground, and b) a foot in heel contact
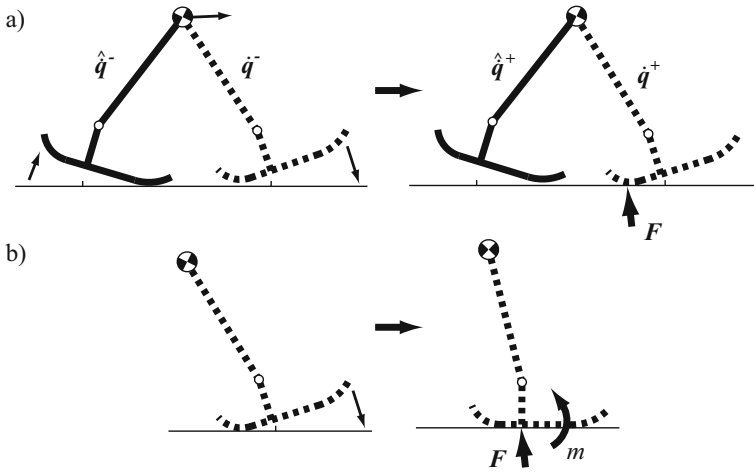
Fig. 2 Two types of collision: a) swing leg collides with the ground, and b) a foot makes flat contact

transitions to flat contact. We assume that all collisions are completely rigid and inelastic, i.e. the foot and ground do not deform and the velocity of the colliding point turns to zero after the collision. We use the collision models to derive the boundary conditions for the optimization because we divide a walk cycle into two phases and optimize each phase separately, as described in the next section.

It turns out that we do not have to enforce a boundary condition at collision a) because any post-collision velocity of the colliding leg can be realized by modifying its velocity before the collision. At collision b), on the other hand, we have the boundary condition

$$m_1 \dot{q}^- = m_1 \dot{q}^+ \tag{4}$$

where m_1 is the bottom row of M, and $\dot{q}^-$ and $\dot{q}^+$ are the joint velocities before and after the collision respectively. Intuitively, this condition means that the angular momentum around joint 1 must be the same before and after the collision because the impact does not provide a moment around the joint.

2.3 Gait Optimization

2.3.1 Representation of a Gait

Figure 3 depicts the three events that occur during each step of locomotion. Accordingly, we divide a step into two phases: *Phase 1* (from flat contact to touchdown) and *Phase 2* (from touchdown to flat contact). Due to the discontinuity at the
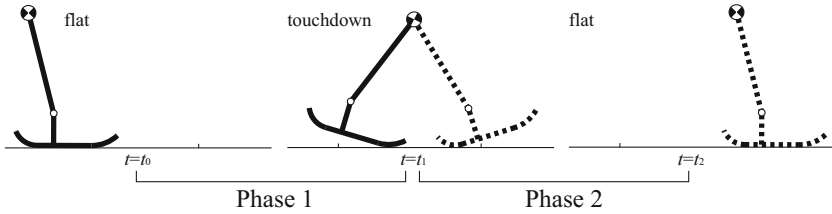
Fig. 3 Three events during a step and two phases for optimization

touchdown, the two phases have to be optimized separately with the following three boundary conditions:

- The swing and colliding legs must share the same mass position at $t = t_1$.
- The configurations at $t = t_0$ and $t = t_2$ must be the same.
- The velocity at the end of *Phase 2* must satisfy the collision boundary condition.

As shown in Section 2.1, the configuration of the biped robot is fully determined by θ_0 and θ_1 due to the contact constraints. We therefore represent the motion by two spline curves for each phase representing the trajectories of θ_0 and θ_1, and obtain the knot points that minimize a cost function Z by applying the conjugate gradient method.

Our choice of the cost function is

$$Z = \frac{1}{2} \sum \tau_1^2 + \frac{w_1}{2} \sum \Delta \tau^2 + \frac{w_2}{2} \left(m_1 (\dot{q}^- - \dot{q}^+) \right)^2 \tag{5}$$

where w_1 and w_2 are constant weights. The first term tries to minimize the ankle torque, the second term is intended to improve the physical feasibility of the motion, and the third term enforces the boundary condition (4). While $\Delta \tau = 0$ should be an equality constraint to ensure that the optimized motion is physically feasible, we relax this constraint because strictly feasible motion may not be represented by splines. We maintain the feasibility as much as possible by using relatively large w_1 and by making sure that $\Delta \tau$ is sufficiently small after the optimization.

2.3.2 Gait Optimization

Table 1 summarizes how the initial and final states are given for each phase. Because the contact foot is in flat contact at $t = t_0$ and t_2, we have $\theta_0(t_0) = \theta_0(t_2) = 0$. We also know $\dot{\theta}_0(t_0) = 0$ due to the inelastic collision assumption. For simplicity, we consider the case where the rest of the initial states, i.e. $\theta_1(t_0) = \theta_1(t_2)$ and $\dot{\theta}_1(t_0)$, are obtained from human motion capture data. The joint angles at t_1 are sampled from a uniform two-dimensional grid and the best sample is chosen after optimizing the gait for each sample.

Table 1 Initial and final values for θ_0 and θ_1 in each phase

	t_0	t_1	t_2
θ_0	0	sampled	0
$\dot\theta_0$	0	optimized	optimized
θ_1	given	sampled	given
$\dot\theta_1$	given	optimized	optimized

We obtain the optimal gait by the following steps:

1. Sample $\theta_0(t_1)$ and $\theta_1(t_1)$ from the grid.
2. Obtain the optimal gait (represented by joint trajectories) for each sample.
3. Find the gait with the lowest cost among the trajectories obtained in step 2.

Step 2 is further divided into the following three steps:

2-1. For *Phase 1*, repeat:

    1)   Randomly sample knot points for θ_0 and θ_1.
    2)   Optimize the knot points using a gradient-based algorithm.

2-2. Perform the same process for *Phase 2*.
2-3. Among all pairs of trajectories from *Phase 1* and *Phase 2*, find the one with the minimum total cost.

2.4 Results

We chose three foot models with different radii for the curved sections and calculated the optimal locomotion pattern for each model. Table 2 summarizes the model parameters used for the experiments. The parameters were chosen so that the total length of the foot is constant for all models (0.1 m for the heel side and 0.2 m for the toe side).

We used the gait parameters shown in Table 3. The parameters were extracted from motion capture clips of walking motions with three different speeds randomly selected from a human motion capture database [5]. The joint angles of the supporting leg at the end of *Phase 1* were sampled from a uniform grid in the θ_0–θ_1 space with the intervals of 0.05 rad for $-0.5 \le \theta_0 \le 0$ and 0.08 rad for $-0.8 \le \theta_1 \le 0.8$. Note that θ_0 must be negative to make toe contact. The total number of grid points is 231. The trajectories of θ_0 and θ_1 were represented by a spline curve with five knot points including the start and end points. We started the optimization from 100 random initial knot points for each sample.

Table 3 summarizes the numerical optimization results. Although we sampled 231 points for the touchdown joint angles, the number of gaits may be smaller because not all of the samples have valid inverse kinematics solution for the touchdown leg. In addition, we rejected the optimization result if any of the joint angles exceed their motion range. We did not impose joint velocity, acceleration or torque limits.

**Table 2** Model parameters used in the numerical experiments

	Flat	Curved 1	Curved 2	Curved 3
m (kg)		6	0	
I (kgm^2/s)		1.	0	
h (m)		0.	1	
l (m)		0.	7	
max/min θ_0 (rad)		-0.5/	0.5	
max/min θ_1 (rad)		-0.8/	0.8	
r_h (m)	0.001	0.05	0.1	0.15
α_h (rad)	0.5	0.5	0.5	0.5
a_h (m)	0.0995	0.075	0.05	0.025
r_t (m)	0.001	0.1	0.2	0.3
α_t (rad)	0.5	0.5	0.5	0.5
a_t (m)	0.1995	0.15	0.1	0.05
shape				

Table 3 Gait parameters and summary of results. L: step length (m), v: body velocity (m/s), $\sharp$: number of gaits found, (θ_0, θ_1): the joint angles of the supporting leg at touchdown for the minimum-torque gait, $\int \tau_1^2 dt$: squared ankle torque integrated over a step.

gait	L	$t_1 - t_0$	$t_2 - t_1$	v	foot	$\sharp$	(θ_0, θ_1)	$\int \tau_1^2 dt$
slow	0.94	0.83	0.17	0.62	Flat	11	(0, -0.64)	1.85×10^4
					Curved 1	12	(0, -0.64)	1.53×10^4
					Curved 2	11	(0, -0.72)	1.46×10^4
					Curved 3	8	(-0.1, -0.56)	1.62×10^4
normal	0.96	0.46	0.14	0.94	Flat	1	(0, -0.72)	1.57×10^4
					Curved 1	0	–	–
					Curved 2	9	(-0.45, -0.24)	1.31×10^4
					Curved 3	10	(-0.45, -0.24)	1.30×10^4
fast	0.99	0.31	0.08	1.26	Flat	0	–	–
					Curved 1	0	–	–
					Curved 2	1	(-0.4, -0.32)	9.57×10^3
					Curved 3	4	(-0.4, -0.32)	9.51×10^3

We observe that there is a large variation in the number of gaits obtained for the combinations. All foot shapes had a number of possible gaits for slow walk, while some foot shapes did not have a gait for normal and fast walk. In general, it was easier to find a gait for the curved feet.

The foot angle (θ_0) of the supporting leg at touchdown also varied across walk speed and foot shapes. Most foot shapes achieved slow walk keeping the supporting foot flat, with the exception of *Curved 3* feet that showed slight toe contact. The optimal gaits for both normal and fast walks had significant toe contact phase except for the only solution for the *Flat* feet. The squared ankle torque was also smaller with curved feet by 12–21%.
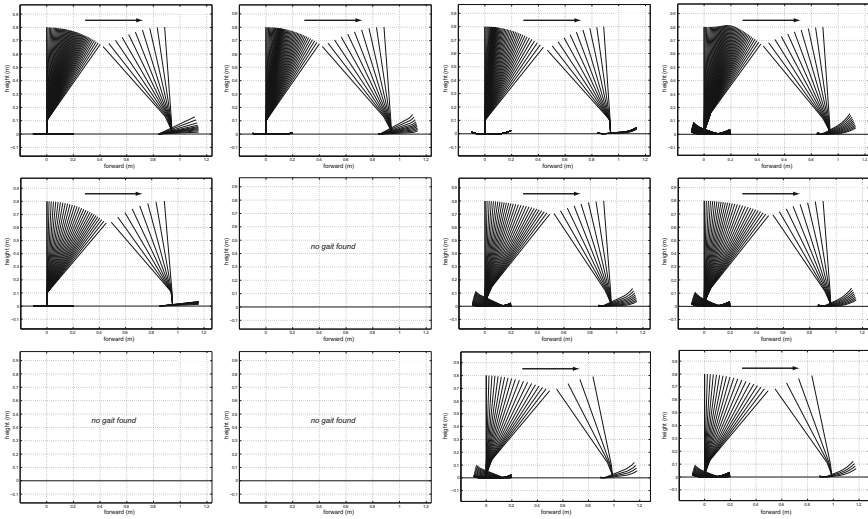
Fig. 4 Optimized gaits. From top to bottom row: slow, normal, and fast walk. From left to right column: *Flat*, *Curved 1*, *Curved 2* and *Curved 3*.

Figure 4 shows the stick figure representations of the optimal gaits obtained for each combination of walk speed and foot shape.

2.5 Discussion

In this section, we investigated the effect of foot shape on biped locomotion. We focused on a foot shape with circular sections in the toe and heel connected by a flat section, and obtained optimal walk patterns for given foot shapes and walk parameters using a numerical optimization technique. The optimization is based on the rigid-body and collision dynamics of the simplified leg model. We also divided a step into two phases to allow discontinuity at collisions.

Comparison of flat feet and three curved feet suggested that having curved toes and heels realizes more efficient locomotion at speeds comparable to human. Even in cases where the optimization could not find a gait for flat feet, it was able to find one or more gaits for curved feet. In addition, optimal gaits for curved feet utilize toe contact of the supporting leg. They also require smaller ankle torque than flat feet.

As future work, it would be interesting to consider more general shapes and add passive elements to the foot. Optimization of the shape parameters, rather than the gait, is also an interesting research direction. We also have to solve the control problem in toe and heel contacts in order to apply our result to real robots.

3 Time Delay and Human Somatosensory Reflex [15]

3.1 Introduction

Understanding the mechanism for generating and coordinating human motions is a longstanding research issue. It is commonly considered that the human motor control system has a hierarchical structure comprising the reflex behavior, the emotional behavior, and the rational behavior, in accordance with the hierarchical brain structure shown by MacLean [11]. Similar structures are often adopted in humanoid control systems.

One of the significant differences, however, is that the human motor control system has a very large response time delay. In contrast to many low-level humanoid controllers that run at around 1 kHz, even the fastest reflex loop can have a latency of 30 ms. It is surprising that such controllers can robustly control agile motions.

In this section, we investigate the effect of time delay in human reflex, based on our previous work on modeling and identifying the human somatosensory reflex system [13, 14]. More specifically, we identify the parameters of the human reflex system model assuming different latencies, using the data collected and computed from a stepping motion. We then perform cross validations with other motions including stepping with different speeds, squat, and jump. In our experiments, the models with time delays similar to the physiologically validated value show better cross-validation results. This result suggests that the human somatosensory reflex system is optimized to the inherent delay of the neural system.

3.2 Somatosensory Reflex Model [13, 14]

Fig. 5 shows our neuromuscular network model, which is essentially a six-layered neural network. The part enclosed by the dashed rectangle represents the somatosensory reflex network model. In this part, each layer consists of:

1. $N_{NJ,i}$ (filled circles) representing the neuromuscular junctions on the muscles, where n_m is the number of muscles included in the musculoskeletal model. This layer receives and integrates the motion command signal from the α motor neuron in the spinal nerve ramus. The integrated signal activates the muscle, which produces tension.
2. $N_{MS,i}$ (filled squares) representing the muscle spindles that measure the muscle length and its velocity. In our model, these values are computed by forward or inverse kinematics computation using the musculoskeletal model [17].
3. $N_{GT,i}$ (filled triangles) representing the Golgi tendon organs that measure the muscle tensions. Their outputs can be computed from the muscle activity using the Hill-Stroeve muscle model [8, 19] or from the inverse kinematics and dynamics computation using the musculoskeletal model [17].
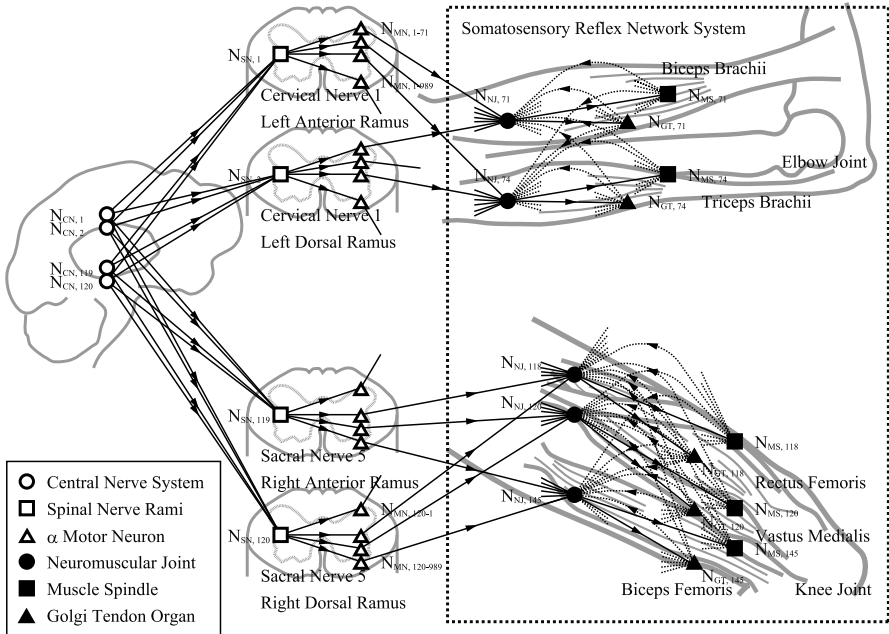
Fig. 5 The neuromuscular network modeled with six-layered neural network. Each layer represents central nerve system, spinal nerve rami, α motor neuron, neuromuscular joint, muscle spindle and Golgi tendon organ. The part enclosed by dashed rectangle represents the somatosensory reflex network model.

The somatosensory reflex network consists of the following connections among these layers:

1. N_{NJ} is connected to N_{MS} and N_{GT} (solid line) by the descending connections that represent the conversion from the motion command signal to the muscle length and tension. The former conversion can be simulated through the kinematics and dynamics computations of the musculoskeletal model [17]. The latter can be computed by a physiological muscle model.
2. N_{MS} and N_{GT} are connected to N_{NJ} (dashed line) by the ascending connections representing the reflex arc between proprioceptive receptor and muscle via the interneurons and α motor neurons in the spinal nerve rami. These connections are the main part of this model and modeled in detail following the anatomical nerve structure [4, 2]. The neuronal bindings between the spinal nerve rami and the muscles are also investigated in the anatomy field [18, 16].

The weight parameters of the somatosensory reflex model are identified using experimental human motion data through a standard back-propagation algorithm [3]

so that it outputs the computed muscle activity at N_{NJ} when the somatosensory information is fed back to N_{MS} and N_{GT} with a specific time delay.

3.3 Somatosensory Reflex and Time Delay

One of the critical characteristics of the somatosensory reflex is the time delay caused by the nerve signal transmission. The reflex arc consists of the proprioceptive sensory receptors, the Ia and II nerve fibers, the interneurons and α motor neuron, and the α motor fiber. A large portion of the time delay (δT) is caused by the signal transmission through the nerve fiber, and can be computed by dividing the fiber length by the transmission speed. The rest (δt) is a collection of response time of electrical and chemical reactions, such as potential discharge and synaptic transmission and has been measured *in-vivo* for some muscles. In Quadriceps, for example, δT and δt are reported to be 16 ms and 9–14 ms respectively. The total time delay of monosynaptic extension reflex of Quadriceps is therefore 25–30 msec, as often observed as the latency of knee-jerk reflex.

Different time delays obviously lead to different model parameters. Another interesting question is whether the parameters identified using the data from a specific behavior generalize to other behaviors. In the next subsection, we compare the identification and cross-validation results of the models with $\delta t = 0, 5, 10, 15, 30, 60$ and 120 ms.

3.4 Experimental Results

We use an optical motion capture system to measure the motion, two force plates to measure the ground contact force, and a wireless electromyograph (EMG) system with 16 electrodes to measure the activities of 8 representative leg muscles on each side. The following three types of motions are measured for the analysis:

1. Step motion in 100 step/min by Subject A (DATA$_{100}$).
2. Step motion in 170 step/min by Subject A (DATA$_{170}$).
3. Step motion with its speed change gradually from 120 step/min to 150 step/min in 6 sec by Subject A (DATA$_{120-150}$).
4. Jump motion by Subject B (DATA$_{jump}$).
5. Squat motion by Subject B (DATA$_{squat}$).

The speed of stepping is controlled with a metronome.

First, we train the model with seven different time delays using the motion data DATA$_{120-150}$. Then we apply these somatosensory reflex network models to the other motion data DATA$_{100}$, DATA$_{170}$, DATA$_{jump}$, and DATA$_{squat}$ for cross validation. The cross validation is performed for each of the seven time delays and the resulting errors are evaluated. Figure 6 represents the average and variance of error between computed and reconstructed muscle activity for cross validation.

3.5 Discussion

These experimental result can be summarized as follows:

1. The identification consistently results in an error of only 2–4 % for all δt, implying that the difference of time delay has little impact on the muscle activity reconstruction capability of the model.
2. The cross validations using $DATA_{100}$ and $DATA_{170}$ show slightly different results depending on the time delay: the error is under 2 % if δt is less than 10 ms, but increase with larger values.
3. The result of cross validations using the motion data $DATA_{jump}$, $DATA_{squat}$ show that our model can reproduce the muscle activity with 5 % error when δt is 5 ms, although these motions are significantly different from stepping and the usage of agonist and antagonist muscles and the pattern of co-contraction are expected to be entirely different. The error is significantly larger with the models with larger time delays.

Considering the fact that the experimentally obtained δt is 9–14 ms, this result suggests that the human somatosensory reflex system is optimized for the time delay enforced by the electric and chemical properties of the elements building the muscles, nerves, and sensory organs.
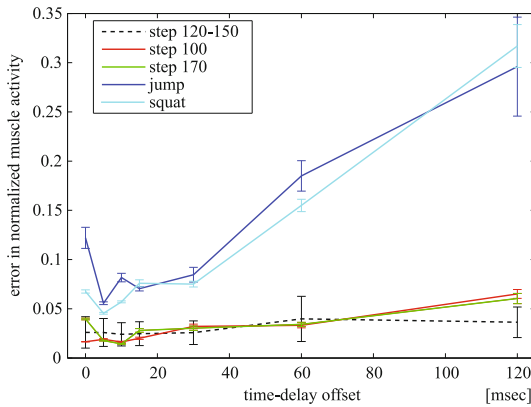
Fig. 6 Errors and their variances between computed and estimated muscle activities. Black dashed line: $DATA_{120-150}$, Red solid line: $DATA_{100}$, green solid line: $DATA_{170}$, blue solid line: $DATA_{jump}$, cyan solid line: $DATA_{squat}$.

4 Conclusion

This paper presented our work related to human locomotion based on the hypothesis that human locomotion is characterized by the inherent mechanical and sensory-motor network structures.

The first piece of work was inspired by the body of work in passive walk where researchers have exploited various foot shapes and leg mechanisms to realize efficient locomotion with no or little control. We focused on the foot geometry and considered foot shapes consisting of a curved toe and heel with a flat section in the middle. We developed an algorithm to optimize the gait for a given foot shape, and demonstrated that curved toe and heel enable faster and more efficient locomotion.

The second work concerned the time delay in human somatosensory reflex. An interesting property of human motor control is that it realizes robust and adaptive motions despite the very large time delay between sensory input and motor output. Based on our somatosensory reflex model, we investigated how the time delay affects the performance of reflex and demonstrated that models identified with time delays close to the physiologically valid value exhibit better generalization across different motions. This result suggests that human somatosensory network is optimized to the latency that inherently exists in our system.

References

1. Adamczyk, P.G., Collins, S.H., Kuo, A.D.: The advantages of a rolling foot in human walking. J. Exp. Biol. 209(20), 3953–3963 (2006),
   `http://jeb.biologists.org/cgi/content/abstract/209/20/3953`,
   doi:10.1242/jeb.02455
2. Agur, A.: Grant's atlas of anatomy. Williams & Wilkins, Baltimore (1991)
3. Bryson, A., Ho, Y.C.: Applied Optimal Control. Blaisdell, New York (1969)
4. Clemente, C.: Gray's Anatomy ed 30. Lea & Febiger, Phyladellphia (1985)
5. CMU graphics lab motion capture database, `http://mocap.cs.cmu.edu/`
6. Collins, S.H., Wisse, M., Ruina, A.: A three-dimensional passive-dynamic walking robot with two legs and knees. I. J. Robotic Res. 20(7), 607–615 (2001)
7. Gutmann, J.S., Fukuchi, M., Fujita, M.: Stair climbing for humanoid robots using stereo vision. In: Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 1407–1413 (2004)
8. Hill, A.: The heat of shortening and the dynamic constants of muscle. Proceeding of the Royal Society of London B126, 136–195 (1938)
9. Hirukawa, H., Hattori, S., Kajita, S., Harada, K., Kaneko, K., Kanehiro, F., Morisawa, M., Nakaoka, S.: A pattern generator of humanoid robots walking on a rough terrain. In: Proceedings of IEEE International Conference on Robotics and Automation, pp. 2181–2187 (2007)
10. Kwan, M., Hubbard, M.: Optimal foot shape for a passive dynamic biped. Journal of Theoretical Biology 248, 331–339 (2007)
11. MacLean, P.: The Triune Brain in Evolution: Role in Paleocerebral Functions. Plenum Press (1990)

12. McGeer, T.: Passive dynamic walking. International Journal of Robotics Research 9(2), 62–82 (1990)
13. Murai, A., Yamane, K., Nakamura, Y.: Modeling and Identifying the Somatic Reflex Network of the Human Neuromuscular System. In: Proceedings of the 29th IEEE EMBS Annual International Conference, Lyon, France (2007)
14. Murai, A., Yamane, K., Nakamura, Y.: Modeling and Identification of Human Neuro-musculoskeletal Network Based on Biomechanical Property of Muscle. In: Proceedings of the 30th IEEE EMBS Annual International Conference, Vancouver, Canada (2008)
15. Murai, A., Yamane, K., Nakamura, Y.: Effects of nerve signal transmission delay in somatosensory reflex modeling based on inverse dynamics and optimization. In: Proceedings of International Conference on Robotics and Automation (2010)
16. Naito, A., Shinozaki, K., Kobayashi, S., Fujii, H., Sato, T., Miyasaka, T., Shindo, M.: Excitatory projections of muscle afferents between the brachioradialis and extensor carpi radialis in humans. Neuroscience Research (suppl. 25), S87 (2001)
17. Nakamura, Y., Yamane, K., Fujita, Y., Suzuki, I.: Somatosensory Computation for Man-Machine Interface from Motion Capture Data and Musculoskeletal Human Model. IEEE Transactions on Robotics 21, 58–66 (2005)
18. Sato, T., Fujii, H., Naito, A., Tonosaki, A., Kobayashi, S., Shinozaki, K., Miyasaka, T.: Inhibition of muscle afferents from the brahioradialis to triceps brachii motorneurones ub gynabs: Centrak oathway. Acta Anato Nippon 77(suppl.), H511 (2002)
19. Stroeve, S.: Impedance Characteristics of a Neuro-Musculoskeletal Model of the Human Arm I: Posture Control. Journal of Biological Cybernetics 81, 475–494 (1999)
20. Taga, G., Yamaguchi, Y., Shimizu, H.: Self-organized control of bipedal locomotion by neural oscillators in unpredictable environment. Biological Cybernetics 65, 147–159 (1991)
21. Wisse, M., Hobbelen, D., Rottevell, R., Anderson, S., Zeglin, G.: Ankle springs instead of arc-shaped feet for passive dynamic walking. In: Proceedings of the 6th IEEE-RAS International Conference on Humanoid Robots, pp. 110–116 (2006)
22. Yamane, K., Trutoiu, L.: Effect of foot shape on locomotion of active biped robots. In: Proceedings of IEEE-RAS International Conference on Humanoid Robots (2009)
23. Yin, K., Loken, K., van de Panne, M.: SIMBICON. ACM Transactions on Graphics 26(3), 105 (2007)

Author Index