Integration of an Adaptive Ground Contact Model into the Dynamic Simulation of Gait

by

C. David Remy

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science

(Mechanical Engineering)

at the

UNIVERSITY OF WISCONSIN–MADISON

2006
Abstract

Integration of an Adaptive Ground Contact Model into the Dynamic Simulation of Gait

C. David Remy

Under the supervision of Associate Professor Darryl G. Thelen
at the University of Wisconsin – Madison

Forward dynamic simulations are becoming an increasingly important tool in the analysis of human gait. Instead of generating only descriptive information, they allow to study the causal relationships in the generation of human motion. This process, that creates muscle excitations, muscle forces, joint torques and eventually movement, can only be investigated by methods that maintain this dynamic cause-effect sequence of the human body. Forward simulation is a tool that fulfills this requirement. After applying small changes to the model of the system and the control signals, the simulation is able to predict the effect of these changes, and therefore how they contribute to the overall motion.

The goal of this project was to improve and extend an existing environment for the simulation of human gait. First, a data processing algorithm was developed that was able to alter experimental motion data so that it could be used in forward dynamic simulations. Measured joint accelerations and ground reaction forces were used to estimate a set of generalized accelerations that was consistent with the overall equations of motion and optimally represented the measured quantities. The algorithm eliminated undesired residual forces at the pelvis, allowed the introduction of prescribed segment accelerations and created a motion that was dynamically consistent over time.

Second, a model of the dynamic foot-floor interactions was developed and included in the existing simulation framework that, to this point, relied on experimentally recorded movements. Such a ground contact model allows the simulation to run independently of any recorded data. This is especially important when the impact of changes in the model or the
controls is studied, and the simulated motion deviates from the recorded movement. Numerical parameter optimization was used to adapt the ground contact model to specific subjects and achieve better agreement of recorded and model predicted ground reaction forces.

In a final step these pieces were brought together. Foot trajectories were extracted from the experimentally recorded motion data of a young healthy adult. These trajectories were first used to adapt the ground contact model, and then altered in a control loop to achieve maximal agreement between model-predicted and recorded ground reaction forces. Joint accelerations were estimated that balanced the overall equations of motion and minimized the differences between actual and desired foot trajectories. A muscle driven forward simulation of gait was generated from the processed data using the Computed Muscle Control algorithm.
Jeder Schritt im Leben ist eine Folge der Fähigkeit beim Fallen wieder und wieder auf den eigenen Füßen zu landen.
Acknowledgements

I want to take this opportunity to thank all the people who helped me with this work.

First and foremost, I would like to thank Darryl Thelen for being such a great advisor. Especially when things didn’t work out as planned, he showed a lot of patience and gave me the help and guidance necessary to complete such a project.

I want to essentially thank everybody in the Neuromuscular Biomechanics Lab for making the time I spent here so enjoyable. It was wonderful to work here. In the context of this thesis, I especially want to thank Amy, Ben, and Brian who’s data I used in my project.

I gratefully acknowledge the financial support provided by the National Institutes of Health (NIH grant HG24276) as well as by ‘Simbios’, the NIH Center for Biomedical Computation at Stanford. Further founding was provided by generous scholarships of the German Academic Exchange Service and the German National Academic Foundation.

Ich möchte mich an dieser Stelle auch bei all denen bedanken, die meinen Aufenthalt in den USA, und damit diese Arbeit möglich gemacht haben. Besonderer Dank gilt in diesem Zusammenhang dem Institut für Werkzeugmaschinen und dem Akademischen Auslandsamt der Universität Stuttgart, und im besonderen Sibylle Krug. Bedanken möchte ich mich zudem bei Professor Zeitz und vor allem bei Professor Kistner die mich mit großem Engagement auf meinem akademischen Werdegang unterstützt haben und bedeutenden Anteil an meinem Erfolg haben.

# Contents

List of Figures vi

List of Tables x

1 Introduction 1

2 Dynamically consistent estimation of joint accelerations for forward dynamic simulations 9

2.1 Abstract ........................................... 9

2.2 Introduction ...................................... 10

2.3 Methods ........................................... 12

  Equations of Motion ................................ 12

  Cartesian and Generalized Coordinates ............ 12

  Linear Equations of Motion in Generalized Coordinates .. 13

  Residual Formulation ............................... 14

  Satisfying the Residual Formulation ............... 15

  Manual Selection of a Subset of Generalized Coordinates and Reaction Forces ... 15

  Least Square Minimization of $\vec{\delta}$ .......... 16

  Prescribing Cartesian Accelerations ............... 17

  Integration and Initial Conditions .................. 19

  Geometric Constraints for Initial Conditions ...... 21

  Evaluation with Experimental Data ................... 22

2.4 Results ........................................... 24

2.5 Discussion ........................................ 27
# List of Figures

1.1 The musculoskeletal model used in this work ............................................. 3  
1.2 The anatomical coordinate system of the shank is defined by the knee and ankle joints, which can be identified by retro reflective markers. Similar definitions, using anatomical landmarks, are used at the other segments. . . 4  
1.3 Motion is recorded in a gait lab by optical tracking of passive retro reflective markers. Force plates, embedded in the lab floor, simultaneously measure ground reaction forces. ................................................................. 5  
1.4 The Computed Muscle Control algorithm is essentially a dynamic observer that estimates an optimal set of muscle excitations necessary to align the model-predicted motion with the experimentally recorded movement (The- len and Anderson, 2006). ................................................................. 6  
1.5 Overall process for the creation of a muscle-driven forward dynamic simulation that replicates a recorded movement. ............................................... 7  

2.1 Residual forces and torques acting on the pelvis. The traditional inverse dynamics analysis produces substantial residuals (dashed red lines) that have no physical meaning and impair subsequent analysis. The proposed method eliminates these residuals completely (solid blue lines) by estimating accelerations and reaction forces consistent with the equations of motion. . . . 24  
2.2 Joint torques acting on the back and in the lower limbs. The results of the traditional inverse dynamics analysis (dashed red lines) is compared to torques computed after the residual forces have been eliminated (solid blue lines). . 26
3.1 The toe segment was modeled as a massless body. The rotation of this segment about the metatarsal axis, and the velocity of this rotation, was therefore statically determined by the implicit balance of torque created by the metatarsal axis itself (yellow) and torque created by the visco-elastic units (green) in the toe segment.

3.2 Stiffness and damping torques about the metatarsal axis were a function of rotation and rotational speed of the toe segment. Static balance requires these functions to be zero. They had to be solved numerically for metatarsal rotation, and metatarsal rotational velocity. (The functions are given for an exemplary configuration of the model right before toe-off.)

3.3 All damping coefficients $c$ were scaled according to the deflection of the corresponding unit using a piecewise cubic function. This smoothly increased damping from zero to $c_{max}$ (reached at deflection $x_{max}$) when units became involved in the contact process.

3.4 Top, side and front view of the ground contact model. The local reference frame (red) is placed in the center of gravity of the foot segment. The current translational and rotational velocities of the segment are represented by dark blue and light blue arrows, respectively. The units and the metatarsal axis are defined in relation to the bounding box (black) and illustrated by small spheres and a cylinder. The forces acting on the individual units are represented by grey arrows, the overall force acting on the origin of the local reference frame by a green arrow. The overall torque acting on the segment is shown in yellow.

3.5 Example of the convergence behavior of the Genetic Algorithm.

3.6 Optimization results for parameter $bb_{pos}$.

3.7 Optimization results for the combination of the parameters: $bb_{pos}$, $bb_{pos}$, and $meta_{pos}$.

4.1 This force controller zeros the differences between measured and model predicted ground reaction forces by adapting the foot kinematics. (Note: The state block diagram shown in this figure doesn’t distinguish between rotational and translational movements. All accelerations are denoted with $a$, all velocities with $v$, all positions/orientations with $x$, and forces and moments with $F$.)
4.2 Adding position and velocity feedback and an acceleration feed forward path to the force controller in Figure 4.1 keeps the trajectories of the ‘walking feet’ close to their measured kinematics while still creating foot trajectories that reproduce the recorded ground reaction forces.

4.3 Overview of the entire system. The measured foot trajectories and external reactions are altered in a control loop so that they balance the ground contact model equation (top). The results of this adaptation are included as geometric constraints in the balancing of the overall accelerations and the ground reaction forces. The motion created by this process is dynamically consistent and produces model based ground reaction forces that closely replicate the measured ones.

4.4 Due to a fully extended leg at heel-contact, the prescribed accelerations of the feet carry through to the pelvis. They can only be counterbalanced by large upper body movements (both arms swinging upwards) and extensive pelvis tilt, which cause the simulation to deviate significantly from the recorded motion.

4.5 If the foot trajectories are not prescribed precisely, small errors between expected ground reaction forces and those created by the ground contact model cause the simulated motion to deviate significantly from the recorded movement. Because of the dynamic instability of gait the model eventually tips and falls over.

4.6 The ground contact models after subject specific parameter optimization.

4.7 The ground contact was used in an closed loop simulation produced by the Computed Muscle Control algorithm (Thelen and Anderson, 2006). Putting feedback gains on joint angles to determine necessary generalized accelerations enables the simulation to run stable for about half a gait cycle.

4.8 Recorded external reaction forces (solid blue lines) are compared to the forces created by the ground contact model (dashed red lines). Even though there is no feedback associated with these forces, the model-based reactions tend to converge to the experimentally recorded forces. This suggests that the ground contact model is capable of producing stable and correct ground reaction forces.
## List of Tables

2.1 The full body model consisted of 12 independent segments (Pelvis, thighs, shanks, feet, trunk, upperarm and forearms). The model state was described by 29 generalized coordinates that mostly corresponded to joint angles. 23

2.2 The initial model configuration was determined by minimizing the weighted distance between computed and recorded marker trajectories over a full integration of the motion. These 38 markers contributed to this objective function. 23

2.3 Average differences between measured kinematics (obtained by inverse kinematics analysis) and the altered kinematics (obtained by integrating over the estimated accelerations). 25

3.1 Subjects in the study 38

3.2 Parameter values for the overall vertical position of the units in relation to the local reference frame 39

3.3 List of parameters adapted in combination with bb_pos_y. A detailed description of these parameters can be found in Appendix A. 40

3.4 Optimization results for pairs of parameters ( * The optimization ran into the boundaries for one or more trials) 42

3.5 Optimization results for the combination of the parameters: bb_pos_x, bb_pos_y, and meta_pos_x using a gradient based SQP optimization method. ( * The optimization ran into the boundaries for one or more trials) 44

3.6 Optimization results for the combination of the parameters: bb_pos_x, bb_pos_y, and meta_pos_x using a genetic algorithm. ( * The optimization ran into the boundaries for one or more trials) 44
4.1 Numerical optimization was performed to adapt the overall vertical and anterior/posterior position of the viso-elastic elements in the ground contact model as well as the position of the metatarsal axis. A description of these parameters and the values of the parameters not subject to optimization can be found in Appendix A.

4.2 Average differences between measured kinematics (obtained by inverse kinematics analysis on marker date) and kinematics created by the forward simulation.
Chapter 1

Introduction

With the rising power of computer hardware and software, dynamic simulation is becoming an increasingly important tool in the analysis of movement. With a comprehensive and accurate model of the musculoskeletal system, it is possible to look into the human body, estimate occurring forces, and assess load distributions in joints. Furthermore, forward dynamic simulation maintains the cause-and-effect relation in the generation of movement and is therefore a powerful tool for investigation. We are able to learn more about the activation patterns of muscles and improve our understanding of the underlying principles of movement. Most importantly, it is possible to predict how changes in the musculoskeletal system affect any of these attributes just by applying the same changes to the model. Simulation is therefore not only a powerful tool for analysis, but can be used to design implants, prevent injuries, plan surgery, and optimize rehabilitation. Recent examples used forward dynamic simulation to evaluate the benefit of backward pedaling in rehabilitation (Neptune, 2000), to predict contact forces in total knee replacements (Piazza and Delp, 2001), to estimate muscle-tendon lengths during crouch gait (Arnold et al., 2001), and to assess the implication of stiff-knee joints on gait (Goldberg et al., 2004).

The first attempts of creating computer models of the human body go back to the early eighties (Seireg and J., 1973; Hatze, 1981) and since then a multitude of researchers have devoted their attention to this topic. The vast majority of dynamic models incorporate the mechanical dynamics of rigid multi body systems. These include accelerations due to inertia, coriolis effects and gravity as well as externally and internally applied forces. The models can be distinguished by two criteria: 2D-models vs. 3D-models and joint torque driven models vs. muscle driven models. Besides this raw classification, they differ largely in the number of body segments, the number of degrees of freedom and the number of muscles.
Values range from six degrees of freedom to 23 and from eight muscles up to 92 (Cole et al., 1996; Anderson and Pandy, 2001; Thelen and Anderson, 2006).

Muscles are generally modeled as Hill-type muscle tendon units. The force created in the muscle component of these units is dependent on fiber length, fiber velocity and muscle activation which follows the muscle excitation with a first-order time delay. As fiber length and fiber velocity depend on the current joint angles and velocities, as well as on the interaction of the active muscle component with the passive elastic elements in the muscle and tendon, the resulting dynamic behavior of the muscles in the model can become very complex. A set of lumped parameters is used to account for individual differences between the different muscles in the musculoskeletal model. Among these parameters are: maximal isometric force, maximum contraction velocity, optimal fiber length, tendon length and penation angle. Many muscles in the human body do not connect segments in a straight line. Their path is constricted by other muscles, bones and ligaments. The rectus femoris, for example, wraps around the patella and over the vasti while connecting the tibia with the pelvis. To account for these restrictions in the model the muscle path can be explicitly defined by, so called, via points or constrained by simple geometrical shapes like cylinders and ellipsoids. According to the complexity of the model, small muscles, or those with similar functionality might be integrated into one single actuator, whereas bigger muscles like the glutei might be split into two or more muscles, to account for multiple attachment points.

Models, as described above, can be designed, visualized and simulated with the help of commercial software packages like SIMM (MusculoGraphics, Inc., Evanston, IL). A visualization of such a model is shown in Fig. 1.1.

An important step towards a successful simulation with meaningful results is the adaptation of the dynamic model to an individual subject. The physical properties of a certain subject in a clinical study (like joint locations, segment masses and muscle sizes) have to be reflected in corresponding model parameters to achieve maximal agreement between simulation and experiment. This process of adaptation does, however, not only include static parameters, but also the dynamic values of movement and muscular excitation. This is especially important when dealing with pathologic movement patterns. For example, in the cases of children with cerebral palsy or patients with neurological dysfunctions caused by a stroke. Their gait patterns deviate significantly from normal movement and it is important that the simulations reflect these differences.

A broad variety of methods exist to record and track a motion and to compute the necessary inputs for the model. The most common technique used in gait labs for recording movement
is measuring the three dimensional position of passive retro reflective markers with an optical tracking system. The markers are fixed on the subject’s skin at anatomic landmarks (joint locations, or palpable bones like the anterior superior iliac spine). These landmarks are used to define the anatomical coordinate system for the according segment (Fig. 1.2), to calculate joint angles, and to determine the dimensions of the individual segments (i.e. to scale the model to a given subject).

The greatest source of error in such a measurement is soft tissue movement, i.e. the motion of the skin (and therefore of the markers) in relation to the actual anatomical landmarks. Soft tissue movement can lead to position errors of individual markers as high as 10 mm.

Some of the anatomical markers would impede normal movement. Especially markers between the legs, like the medial knee and ankle markers have to be removed for the actual gait analysis. To deal with these limitations, additional tracking markers are placed at arbitrary positions on every segment. In a static trial, the relative positions of tracking markers and anatomical markers are determined. Under the assumption that the body segments are truly rigid, the position of these tracking markers can later be used to determine the position of
Figure 1.2: The anatomical coordinate system of the shank is defined by the knee and ankle joints, which can be identified by retro reflective markers. Similar definitions, using anatomical landmarks, are used at the other segments.

Simultaneously with the kinematic measurements, the ground reaction forces are recorded by force transducers (also called force plates) that are embedded in the lab floor. Figure 1.3 shows the data collection in a motion lab.

After the model is scaled according to the anatomic marker locations, the first step of transferring the experimentally recorded movement to the model is the computation of the generalized coordinates (the joint angles and the position and orientation of the pelvis) from marker trajectories. This process is called inverse kinematics analysis. Static optimization is used to generate generalized coordinates that minimize the distance between the measured and the model predicted marker positions.

The kinematic data and the ground reaction forces, as they are collected in most gait labs, are redundant measurements. The generalized coordinates can be used to calculate the center of mass acceleration, which is also fully determined by the reaction forces. In other words, the ground reaction forces could be determined by measuring only the marker positions. But while the kinetic data suffers from high measurement noise, soft tissue movement, and low

anatomical landmarks that can’t be measured directly.
Figure 1.3: Motion is recorded in a gait lab by optical tracking of passive retro reflective markers. Force plates, embedded in the lab floor, simultaneously measure ground reaction forces.

time resolution, the force plate data is generally very accurate and recorded at high sampling rates. Including this data will substantially increase the accuracy of the simulation. Various ways have been developed to work with both data sources without discarding information. One approach is tracking kinematic values and forces simultaneously and just weighting errors differently (Seth and Pandy, 2006). The Residual Elimination Algorithm (REA) developed by Thelen and Anderson (2006) chooses six variables of the kinematic data (position of the pelvis and relative orientation of the trunk) and adjusts them so that the overall center of mass movement agrees with the external forces.

When the movement is described as a function of generalized coordinates, the joint torques or muscle excitations that would produce this movement can be computed. This process is referred to as inverse dynamics analysis. While it is relatively straightforward to compute joint torques, the challenge of calculating excitation signals for muscles lies in the fact that these are highly ambiguous and also do not directly correspond to muscle force. The number of muscles is normally substantially bigger than the number of joints. This under-determined dynamic system can only be solved if the number of muscles in the model is greatly reduced
(i.e. by grouping various muscles into one) or if further conditions are provided. Additional constraints could be introduced by EMG-measurements which determine the excitation of some muscles, or by the assumption that the human body will try to minimize the consumption of metabolic energy.

In order to integrate all these restrictions, all previous studies used some sort of optimization to compute muscle activations. This can be an optimal Ricatti-controller (Seth and Pandy, 2006), where a weight-matrix is used to emphasize the accuracy of certain coordinates and to minimize the overall excitation of muscles, or it can be pure static optimization (Anderson and Pandy, 1999), where an objective function is used for the same purpose. In this case, the excitation pattern is predefined for the entire simulation, which is executed in every single optimization iteration. The simulation outcome is compared to the recorded motion, and the excitation pattern is adjusted until sufficient agreement between desired and calculated movement is achieved. The Computed Muscle Control (CMC) algorithm, introduced by Thelen and Anderson (2006) combines both approaches. A control loop in a time discrete observer compares experimental recordings to the simulation outcome and calculates desired general accelerations $\ddot{q}^{des}$. An subsequent optimization algorithm is used to calculate muscle excitations that generate these accelerations, while trying to minimize metabolic energy consumption. The process is repeated every 10 ms. It is shown as a schematic in Figure 1.4.

Figure 1.4: The Computed Muscle Control algorithm is essentially a dynamic observer that estimates an optimal set of muscle excitations necessary to align the model-predicted motion with the experimentally recorded movement (Thelen and Anderson, 2006).

6
The muscle excitations calculated by this algorithm can finally be used in forward dynamic simulations. The effect of perturbations in both, the model and the input, can be studied. Figure 1.5 shows an overview of the entire process, from motion capturing in the lab to the muscle-driven forward simulation.

Figure 1.5: Overall process for the creation of a muscle-driven forward dynamic simulation that replicates a recorded movement.

The goal of this project was to extend and improve the dynamic model and simulation environment developed by Thelen and Anderson (2006). Their study used a full 3D eight segment musculoskeletal model of the trunk and the lower limbs with 21 degrees of freedom. It is driven by 92 first order Hill-type muscles and used to replicate experimentally recorded motions. The REA and the CMC-algorithm, as introduced above, have been implemented in
order to track motion, and the whole system was successfully applied to human walking in 2005.

An adaptive ground contact model was developed and included in the existing simulation environment which previously relied on recorded external forces. This step detaches the simulation entirely from experimental data, and allows studying the true impact of changes in the model and the excitation pattern on the motion.

The following thesis is divided into three articles:

Chapter 2 describes a method to balance the generalized accelerations and external reaction forces with respect to the equations of motion. This step eliminates residual forces at the pelvis and allows the integration of prescribed segment accelerations. The estimated generalized accelerations are dynamically consistent over time and can therefore be used in forward dynamic simulations.

Chapter 3 introduces the adaptive ground contact model used in this study. Special attention is given to the process of parameter optimization and the creation of a subject specific ground contact model.

Chapter 4 shows how this ground contact model can be included in a forward dynamic simulation.

All three steps were implemented and successfully tested with experimental data.
Chapter 2

Dynamically consistent estimation of joint accelerations for forward dynamic simulations

2.1 Abstract

The model-based calculation of inter-segmental loads, joint torques, and muscle forces depends highly on the accurate estimation of joint accelerations. In the analysis of gait, these accelerations are commonly obtained by double differentiation of noisy position data which introduces large errors in the analysis. The quality of the estimation can be substantially improved by including measured ground reaction forces and utilizing model inherent information in the form of dynamic and geometric constraints. When the data is used in forward dynamic simulations (generated by integrating the equations of motion for given joint torques or muscle force trajectories) it is not sufficient to estimate these quantities on a ‘per frame’ basis. The joint angles, velocities and accelerations, and the external reactions have to be dynamically consistent over the entire motion.

In the present work, a weighted least-squares approach is used to solve for a set of generalized accelerations that satisfy the given constrains and that optimally reflect the experimental acceleration and force recordings. The accelerations are integrated over the entire movement to enforce the dynamic equations of motion over time. Numerical optimization is used to determine intitial generalized coordinates and speeds that produce a movement most consistent with the desired motion.

The proposed method is able to accurately reproduce a recorded motion without intro-
ducing residual forces at the pelvis. Integrated joint kinematics were within $1^\circ$ of recorded values when it was tested on the experimental data of a whole cycle of gait.

### 2.2 Introduction

Inverse dynamics analysis, i.e. the model-based calculation of inter-segmental forces and torques, is a primary tool in the clinical analysis of motion. The computed joint torques are also used in subsequent analyses, like the assessment of muscle and joint contact forces. They are, furthermore, a fundamental component in the generation of forward dynamic simulations from experimental data. In the latter case, it is important that the computed joint torque trajectories, when used as input of the simulation, closely replicate the desired motion. ‘Desired’—in this context—means usually ‘as recorded in an experiment’. However, a priori knowledge or model inherent restrictions can be included in the formulation of such a desired motion. While the recorded motion is defined by joint angle trajectories, these additional restrictions are often given in Cartesian coordinates. Examples are studies where the subject holds on to a handrail or where segments have to follow certain trajectories, for example when using exercise equipment. Such Cartesian restrictions can also be used to improve the quality of the recorded data, for example, by removing excessive foot motion during stance phase.

The conventional approach to the inverse dynamics problem is the iterative solution of the equations of motion for each segment (Zajec and Gordon, 1989). Starting with the topmost segment and moving downwards, the forces and torques that balance the measured accelerations are calculated at the distal joint and subsequently applied to the next segment. This approach is solely based on acceleration measurements. In clinical studies, these accelerations are usually obtained by double differentiation of noise polluted position information and are prone to large errors. To improve the quality of the estimation, additional measurements in the form of ground reaction forces can be included. While the kinetic data suffers from high measurement noise and low time resolution, such force plate data is very accurate and recorded at high sampling rates. Instead of starting at the base segment, the ‘bottom up’ approach (Winter, 1990) begins with the contact segments (i.e. the feet). It uses the measured ground reactions and accelerations to calculate proximal joint forces and works its way towards the pelvis. Joint moments tend to be more accurate when estimated with this method. However, due to measurement and model errors, the base segment will inevitably develop proximal forces and moments. These residual forces are undesired. They have no physical
meaning, as there is no subsequent segment or joint that can produce them. Their occurrence is a direct consequence of the redundancies in the measurement of force and acceleration.

A very promising approach of dealing with these redundancies incorporates both, kinematic and kinetic measurements, to improve the estimation of accelerations (Cahout et al., 2002; Kuo, 1998). A static least-squares optimization method is used to estimate a set of accelerations, torques and external reactions that are most consistent with all measurements while satisfying the dynamic constraints imposed by the equations of motion. Knowledge of error distribution in the measurements can be included in the optimization to obtain truly optimal estimations. It was shown that the more accurate estimation of accelerations leads to substantially improved inverse dynamics calculations.

The mentioned methods have in common that they only generate estimates on a ‘per frame’ basis. This is sufficient for clinical studies which are interested in the occurring forces and joint torques. However, if the data is being used in forward dynamic simulations, it is important that the estimates are dynamically consistent over time and that the computed accelerations, when integrated, actually reproduce the desired motion.

Dynamic optimization can be used to determine joint torque trajectories or muscle excitation patterns that satisfy this criterion and produce motion that is dynamically consistent over time (Chao and Rim, 1973; Anderson and Pandy, 2001). However, such a dynamic optimization suffers from practical limitations. The objective functions used are often ill-behaved, and the convergence behavior and the quality of the solution depends strongly on the accuracy of the initial guess. Furthermore, the inordinate amounts of computation time (Neptune et al., 2001) make the approach, especially on a subject-specific basis, impractical.

The aim of this study was to generate dynamically consistent acceleration and reaction force estimates that satisfy the dynamic and geometric constraints imposed by the equations of motion and potential prescribed Cartesian accelerations. The constrained equations of motion were solved at every integration step, using a static least-squares approach. To enforce dynamic consistency, the obtained accelerations were integrated over the entire motion and the computed positions and velocities were used in subsequent time steps. Numerical optimization determined the initial conditions for this integration so that the overall motion optimally reproduced a desired movement. The proposed method was tested on experimental gait data and closely reproduced the recorded joint trajectories.
2.3 Methods

Equations of Motion

Newton’s law requires that the sum of all external forces \( F_i \) balances the sum of the mass-acceleration products of the individual body segments (Greenwood, 1988). *(Note: Gravity, as a very special external force, has been moved to the right hand side of the equation.)*

\[
\sum_{i=1}^{N} F_i = \sum_{j=1}^{n} m_j \left( \ddot{r}_j - \ddot{g} \right)
\]  
(2.1)

Similarly, the equations of rotational motion about the origin of the coordinate system can be stated as:

\[
\sum_{i=1}^{N} \left( \vec{\rho}_i \times F_i + M_i \right) = \sum_{j=1}^{n} \vec{r}_j \times m_j \left( \ddot{r}_j - \ddot{g} \right) + \sum_{j=1}^{n} \left[ I_j \cdot \dot{\omega}_j + \vec{\omega}_j \times I_j \vec{\omega}_j \right]
\]  
(2.2)

\( F_i \) and \( M_i \) are external forces and moments respectively, \( \vec{\rho}_i \) is the location of a point along the external reaction force \( F_i \), \( \vec{r}_j \) points to the center of gravity, \( \vec{\omega}_j \) is the angular velocity, and \( m_j \) and \( I_j \) are the mass and the inertia dyadic of segment \( j \).

The number of segments is \( n \), the number of external reactions is \( N \). Internal reactions are in no way accounted for, which reduces the number of equations in this overall balance to six.

Cartesian and Generalized Coordinates

The equations of motion stated above need the individual segment positions, velocities, and accelerations in Cartesian coordinates. Unfortunately, the current state of a skeletal model is usually described in generalized coordinates \( q \) and their derivatives \( \dot{q} \) and \( \ddot{q} \). These generalized coordinates (that mostly correspond to joint angles) can be mapped to the Cartesian coordinates by a nonlinear transformation \( t \).

\[
\begin{bmatrix}
\vec{r} \\
\theta
\end{bmatrix} = t \left( \vec{\bar{q}} \right)
\]  
(2.3)

The instantaneous relationship between Cartesian speeds and the generalized speeds can be expressed with the Jacobian \( J \). *(Note: As the Jacobian \( J \) depends on the current state \( \vec{\bar{q}} \), it would be more appropriate to state \( J \left( \vec{\bar{q}} \right) \).*
\[
\begin{pmatrix}
\dot{\vec{r}} \\
\dot{\vec{\omega}}
\end{pmatrix} = J \dot{\vec{q}}
\] (2.4)

Differentiating this expression leads to:
\[
\begin{pmatrix}
\ddot{\vec{r}} \\
\ddot{\vec{\omega}}
\end{pmatrix} = J \ddot{\vec{q}} + J \dot{\omega}
\] (2.5)

**Linear Equations of Motion in Generalized Coordinates**

Equations (2.1) and (2.2) can be rearranged and put into matrix form forming the following linear equation.

\[
\begin{bmatrix}
-m_1 & 0 & 0 \\
0 & -\rho_1^z & \rho_1^y \\
0 & \rho_1^z & -\rho_1^x \\
-\rho_1^y & \rho_1^x & 0
\end{bmatrix}
\begin{pmatrix}
\vec{F}_1 \\
\vec{M}_1 \\
\ddot{\vec{r}}_1 \\
\ddot{\vec{\omega}}_1
\end{pmatrix}
= \sum_{j=1}^{n} \begin{bmatrix}
-m_j \vec{g} \\
-m_j (\vec{r}_j \times \vec{g}) + \vec{\omega}_j \times \vec{I}_j \vec{\omega}_j
\end{bmatrix}
\] (2.6)

or shorter

\[
A^{cart} (\vec{r})
\begin{pmatrix}
\vec{F} \\
\vec{M} \\
\ddot{\vec{r}} \\
\ddot{\vec{\omega}}
\end{pmatrix} = f^{cart} (\vec{r}, \vec{\omega})
\] (2.7)

Where \( A^{cart} \) is a matrix of the dimension \( 6 \times (n \cdot 6 + N \cdot 6) \) and \( f^{cart} (\vec{r}, \vec{\omega}) \) a six dimensional vector function. This equation states that the external forces and segment accelerations have to balance the gyroscopic and gravitational forces defined in \( f^{cart} (\vec{r}, \vec{\omega}) \). We will refer to this equation later, when geometric constraints are introduced to handle prescribed Cartesian accelerations. However, for the time being, we focus on the dynamic constraints, the balance of forces and accelerations, that is achieved by adaptation of the measured generalized accelerations and external reactions. Using (2.5), one can easily replace the Cartesian segment accelerations with the derivations of the generalized coordinates.
which can be used to state the equations of motion as a linear function of the generalized coordinates:

\[
A(\vec{q}) \begin{bmatrix} \vec{F} \\ \vec{M} \\ \frac{\ddot{r}}{\ddot{\omega}} \end{bmatrix} = f(\vec{q}, \dot{\vec{q}})
\] (2.9)

with

\[
A(\vec{q}) = A^{\text{cart}}(\vec{r}) \begin{bmatrix} [1] & 0 & 0 \\ 0 & [1] & 0 \\ 0 & 0 & [J] \end{bmatrix}
\] (2.10)

and

\[
f(\vec{q}, \dot{\vec{q}}) = \sum_{j=1}^{n} \begin{bmatrix} -m_j \ddot{g}_j \\
-m_j (\vec{r}_j \times \vec{g}) + \vec{\omega}_j \times I_j \vec{\omega}_j \end{bmatrix} - A^{\text{cart}}(\vec{r}) \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & [J] \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ \dot{\vec{q}} \end{bmatrix}
\] (2.11)

where the Cartesian coordinates and speeds have been calculated from the generalized coordinates using

\[
\begin{bmatrix} \vec{r} \\ \theta \end{bmatrix} = t(\vec{q}) \quad \text{and} \quad \begin{bmatrix} \frac{\ddot{r}}{\ddot{\omega}} \end{bmatrix} = J \dot{q}.
\]

The matrix \( A \) is of the dimension \( 6 \times (N \cdot 6 + n_q \cdot 6) \), where \( n_q \) denotes the number of generalized coordinates \( q \).

**Residual Formulation**

As a result of model and measurement errors, the experimental recordings of ground reactions and accelerations (denoted with a prime) will generally not satisfy these equations. Small residuals \( (\delta \vec{q}, \delta \vec{F}, \delta \vec{M}) \) have to be introduced to account for the differences in between model, measurement and reality:

\[
\begin{align*}
\vec{F} &= \delta \vec{F} + \vec{F}' \\
\vec{M} &= \delta \vec{M} + \vec{M}' \\
\dot{\vec{q}} &= \delta \dot{\vec{q}} + \dot{\vec{q}}'
\end{align*}
\] (2.12)
With these residuals, equation (2.9) can be written as

\[
A (\vec{q}) \begin{bmatrix}
\delta \vec{F} \\
\delta \vec{M} \\
\delta \ddot{\vec{q}}
\end{bmatrix} = f \left( \vec{q}, \dot{\vec{q}} \right) - A (\vec{q}) \begin{bmatrix}
\vec{F}' \\
\vec{M}' \\
\ddot{\vec{q}}'
\end{bmatrix}
\]  

(2.13)

or simply as the linear equation

\[
A \vec{\delta} = \vec{b}
\]  

(2.14)

**Satisfying the Residual Formulation**

As stated in the previous section, equation (2.9) does normally not agree with experimental measurements. Integrating the kinematic and kinetic measurements into one consistent set of data is important for two reasons:

- Many of the algorithms that are applied to such experimental data (computation of muscle forces or excitations, forward dynamic simulations,...) require the dynamic balance of reaction forces and accelerations with respect to the model (Thelen and Anderson, 2006).

- The redundancies in between kinematic and kinetic information lead to better acceleration estimates when they are combined with the information inherent to the model (Kuo, 1998).

The preprocessing of the data can be done by calculating a set of residuals \( \vec{\delta} \) from equation (2.14) and adding them to the experimental recordings according to equation (2.12). As equation (2.14) is an under-determined system of linear equations, generally no unique solution exist to this problem. The way chosen to determine a set of residuals will therefore depend on the individual application and might vary from one experimental setup to another.

**Manual Selection of a Subset of Generalized Coordinates and Reaction Forces**

By setting most of the \( \delta F, \delta M \) and \( \delta \dot{q} \) in equation (2.14) to zero, the effective number of columns in matrix \( A \) can be reduced until a unique solution of the linear equation exist. The selection of the six remaining residuals \( \delta \) depends on the experimental setup and the given problem and should take into account the following considerations:

1. The measurement of the contact forces is generally more precise then the measurement of the joint accelerations. The external reactions are recorded at a higher frame rate...
and don’t have to deal with the noise amplification during differentiation. Therefore, the reaction forces are normally kept as measured. The corresponding $\delta$ values are set to 0.

2. The selected generalized accelerations should have a relatively large impact on the external reactions. This guarantees that the resulting system of linear equations is not ill-conditioned, that the coefficient matrix $A$ is invertible, and that the algorithm is actually able to balance forces and accelerations. The pelvis translations are a good example for generalized coordinates that are well suited in this context.

3. Some generalized coordinates or external reactions are not measurable. One might think of a simplified model that represents the entire upper body (trunk, head, and arms) with only one rigid segment. In such a case, the lower back angles of the model do not represent measurable joint movements, but rather the overall motion of the entire upper body and the upper extremities of the subject. As the corresponding generalized coordinates do not represent actual joint angles, they can’t be measured in an experiment. Other experimental setups only allow the measurement of some of the six components of the external reactions, for example, when using insole pressure sensors instead of force plates. In cases like this, the residual formulation serves as a practical tool to estimate the missing generalized coordinates from the external reactions and vice versa.

As a coordinate is considered to be either accurately measured or completely unknown, the $\delta$-notation seems excessive and it would be more intuitive to base this method on equation (2.9) instead of equation (2.14). However, to remain consistent with the notation and the following methods, we assume that the measurement for an unknown quantity is 0, and the $\delta$-value for an exactly measured quantity is 0. This way, the remaining $\delta$-values represent the actual generalized accelerations.

**Least Square Minimization of $\vec{\delta}$**

Assuming that not all measurements are equally accurate, one can introduce a diagonal matrix $W$ of measurement standard deviations. This matrix represents the accuracy of the experimental estimates of the generalized coordinates and external reactions. Quantities that can be measured more precisely will obtain smaller values then quantities that are prone to errors. With this matrix equation (2.14) can be expanded to:
\[ AW W^{-1} \bar{\delta} = \bar{b} \]  

(2.15)

Using the right Moore-Penrose Matrix Inverse \( B^+ = B^T (B B^T)^{-1} \), equation (2.15) can be solved for a set of deltas

\[ \bar{\delta} = W (AW)^+ \bar{b}, \]  

(2.16)

that will minimize the Euclidean norm \( \| W^{-1} \bar{\delta} \| \) (Yamaguchi, 2002).

In other words: The more we trust a certain measurement, the smaller the associated residual \( \delta \) will be. The values of quantities that can be measured precisely will stay close to the recorded data, while others might undergo substantial alterations in order to balance external reactions and forces. If the values of certain quantities are known precisely, the according elements in \( W \) should be set to almost 0. If a quantity is unknown, the according element should be set to a very large value.

**Prescribing Cartesian Accelerations**

In a large variety of applications the movements of the studied subjects are not entirely free. Limitations might be given by the experimental setup or by the inherent nature of the motion. An example for an experimental setup with limited motion is pedaling. The feet are rigidly coupled to the pedals and have to follow a circular trajectory prescribed by the crank. Despite the many degrees of freedom of the leg, only two elemental motions remain possible: Rotation about the crank axis and rotation about the pedal axis. Other examples include experiments where some of the limbs are fixed, or where the subject holds on to objects that are fixed in space (handles, guardrails). Additionally, it is possible to use a priori knowledge of the movement to increase the accuracy of the measurement. It is known, for example, that during the stance phase of walking, the foot should remain motionless (at least approximately), and any movement recorded in spite of that, most likely reflects measurement errors.

If these constraints can be described analytically and remain effective during the entire duration of the experiment, a way to approach this problem is to reduce the number of degrees of freedom of the model. In the pedaling example this can be done by replacing the generalized coordinates for hip, knee, and ankle with the crank and pedal angle. This will reduce the degrees of freedom of the model to two for each leg. Within these degrees of freedom the movement is unlimited, no additional constraints have to be applied and we can use the methods mentioned above to balance motion and external forces.
However, this method doesn’t work if the constraints come in and out of effect during the motion. An example is walking, where the feet remain fixed only during the stance phase, but can move freely in the remainder of the movement. We also use prescribed accelerations of the feet to track the ground reaction forces produced by a ground contact model. In this case, the required foot motion is too complex to be described analytically and a different approach has to be found for prescribing the foot segment accelerations.

If accelerations are prescribed in terms of generalized coordinates, the prescribed accelerations \( \ddot{\mathbf{q}} \) can simply replace the measured ones \( \ddot{\mathbf{q}}' \). Only the remaining generalized accelerations are adapted to achieve dynamic balance. Unfortunately, in most of the cases, the accelerations are defined for individual segments and therefore prescribed in Cartesian coordinates. Such prescribed accelerations \( \ddot{\mathbf{r}} \) and \( \dot{\mathbf{ω}} \), don’t map directly into generalized accelerations \( \ddot{\mathbf{q}} \), but form complex geometric constraints. To include these constraints into the formulation derived above, we go back to equation (2.7):

\[
A^{\text{cart}} (\mathbf{r}) \begin{bmatrix} \ddot{\mathbf{F}} \\ \dot{\mathbf{M}} \\ \ddot{\mathbf{r}} \\ \dot{\mathbf{ω}} \end{bmatrix} = f^{\text{cart}} (\mathbf{r}, \mathbf{ω}) \tag{2.17}
\]

The Cartesian constraints can be introduced mathematically by expanding the matrix \( A^{\text{cart}} \). One row will be added for each constraint. Each of these rows contains a single 1 which maps a Cartesian acceleration to its prescribed value (denoted by a \( \hat{\cdot} \)) on the right hand side of the linear equation.

\[
\begin{bmatrix} A^{\text{cart}} (\mathbf{r}) \\ [0] [0] 0 \ldots 1 \ldots 0 \\ [0] [0] 0 \ldots 1 \ldots 0 \\ \vdots \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{F}} \\ \dot{\mathbf{M}} \\ \ddot{\mathbf{r}} \\ \dot{\mathbf{ω}} \end{bmatrix} = \begin{bmatrix} f^{\text{cart}} (\mathbf{r}, \mathbf{ω}) \\ \hat{\mathbf{r}}_i \\ \hat{\mathbf{ω}}_i \\ \vdots \end{bmatrix} \tag{2.18}
\]

\[
\hat{A}^{\text{cart}} (\mathbf{r}) \begin{bmatrix} \ddot{\mathbf{F}} \\ \dot{\mathbf{M}} \\ \ddot{\mathbf{r}} \\ \dot{\mathbf{ω}} \end{bmatrix} = \hat{f}^{\text{cart}} (\mathbf{r}, \mathbf{ω}, \hat{\mathbf{r}}, \hat{\mathbf{ω}}) \tag{2.19}
\]

This formulation has the advantage that it keeps a full vector of external reactions and Cartesian accelerations. It guarantees that the remaining problem formulation, as discussed above, can be kept as it is. It will inherently take care of the geometric constraints that result
when the prescribed Cartesian accelerations are mapped into generalized coordinates. We only have to replace the matrix $A^{\text{cart}}$ and the function $f^{\text{cart}}$ by its extended counterparts and base the residual formulation on these new matrices:

$$A(q) = \hat{A}^{\text{cart}}(r) = \begin{bmatrix} [1] & 0 & 0 \\ 0 & [1] & 0 \\ 0 & 0 & [J] \end{bmatrix}$$ (2.20)$$

and

$$f(\dot{q}, \ddot{q}) = \hat{f}^{\text{cart}}(\dot{r}, \ddot{\dot{r}}, \ddot{\dot{\omega}}) - \hat{A}^{\text{cart}}(\dot{r}) \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & [\dot{J}] \end{bmatrix} \begin{bmatrix} \dot{q} \\ \dot{\dot{q}} \end{bmatrix}$$ (2.21)$$

An alternative approach would be removing the prescribed Cartesian accelerations from the solution vector, and putting them on the right hand side of equation (2.14). But doing so, would only monitor the dynamic constraints, i.e. balance external reactions with the prescribed accelerations. A separate formulation for the actual geometric constraints would become necessary.

Without any geometric constraints we can assume that the system of linear equations (2.14) is under-determined, leaving some liberties in finding the "best" solution for a particular application. As soon as constraints are introduced, this assumption doesn’t hold anymore. Every constraint effectively removes one degree of freedom in the solution space. If more constraints then generalized coordinates are introduced $A$ will contain more rows then columns and (2.14) becomes overdetermined and all constraints can’t be simultaneously satisfied. But even when dealing with fewer constraints, one has to be careful. Singularities in $J$ map directly into $A$, and even a single constraint can cause (2.14) to become unsolvable for certain model configurations. Such singularities reflect instances in which two or more segments or joint axis align. The mobility of the model in the Cartesian space decreases in such a configuration. A fully extended arm - for example - restricts the hand to rotational movements about the shoulder joint. One has to be extremely careful when introducing constraints in a model that traverses such configurations during the recorded movement.

**Integration and Initial Conditions**

The processing of the kinetic and kinematic data as presented in the previous paragraphs has been expressed entirely in terms of accelerations. The equations of motion return no information about position and velocity. On the contrary, these values are necessary in the
coefficient matrix $A$ and on the right hand side of equation (2.14). They are needed to determine the current moment arms and the mapping from generalized coordinates into Cartesian coordinates, as well as for the calculation of gravitational and gyroscopic forces. Equation (2.14) represents a nonlinear ordinary differential equation. The only feasible way of solving it is by numerical integration of the generalized accelerations that satisfy the residual formulation and the geometric constraints.

A major challenge of this numerical integration is the determination of the initial conditions $\vec{q}_0$ and $\dot{\vec{q}}_0$ - the generalized coordinates and velocities at the beginning of the integration. The difficulty lies in the fact that gait is an inherently unstable dynamic process. Small perturbations in the initial conditions can lead to large deviations from the intended motion. Only a very complex feedback system enables humans to reject such perturbations, to keep balance and maintain the regular patterns of gait. A simple algorithm like the one presented has to live with the instabilities, rather then being able to reject them. For this reason, the initial conditions for the integration have to be selected with outmost care.

The agreement of recorded and calculated motion can be assessed by the squared distances between calculated and recorded generalized coordinates:

$$e(t)_i = (q(t)_i - q'(t)_i)^2 \quad (2.22)$$

Alternatively to expressing this agreement in terms of generalized coordinates, it is possible to define it in Cartesian space. A promising approach in this context is the summation over the differences between predicted and measured marker kinematics. By putting weights on this error function, the accuracy of the spatial tracking of certain segments can be emphasized, or a priori knowledge can be included. For example, by trying to minimize foot movement during the stance phase. Tracking markers directly has the additional advantage of removing the intermediate inverse kinematics step for comparison and closing the bridge to the experimental data. This allows to better quantify the source of errors, and might therefore help reducing them.

$$e(t)_i = (\vec{r}(t)_i - \vec{r}'(t)_i)^2 \quad (2.23)$$

Both error formulations can be used simultaneously. Provided with weights, summed up, and integrated over the entire motion, these values express a cost function (2.24) that maps a certain set of initial conditions to a measure of ‘agreement’ between the calculated and recorded motion.

20
\[
E \left( \vec{q}_o, \dot{\vec{q}}_o \right) = \int_{t_o}^{t_{\text{end}}} \left( \sum_{i=1}^{n_q} w^q_i \cdot e(t)_i^q + \sum_{i=1}^{n_r} w^r_i \cdot e(t)_i^r \right) dt
\]  

(2.24)

Numerical optimization is used to determine a set of initial positions and velocities that minimizes this objective function and yields maximal agreement. To achieve a truly optimal result, the initial positions and velocities of all generalized coordinates that are not prescribed have to be adapted simultaneously.

However, there are practical limitations. The computational costs of the optimization are immense, as the entire motion has to be integrated for every evaluation of equation (2.24). It is therefore important that the optimization converges as fast as possible. For this reason, only a small subset of the initial generalized coordinates might become subject to optimization and the remaining ones being set to their experimentally recorded values. The number of coordinates subject to optimization is a trade-off between computational costs and desired accuracy.

The approach is similar to strategies for the numerical solution of boundary value problems. It does not address the underlying problem of inherent instability, but for a whole cycle of gait, the calculated and recorded motions will be close enough for all practical purposes.

**Geometric Constraints for Initial Conditions**

Section 2.3 introduced ways of handling prescribed accelerations in the estimation process. Such prescribed accelerations are in virtually all cases a side effect of prescribed velocities and positions. However, as long as the initial positions \( \vec{q}_o \) and velocities \( \dot{\vec{q}}_o \) are set in accordance to the given constraints, prescribing the correct accelerations will ensure that the constraints remain fulfilled for the entire motion.

When determining initial conditions, it is therefore important to include the geometric constraints that result from prescribed positions or velocities. If \( n_c \) geometric constraints are given, only \( n_q - n_c \) of the initial conditions can be set freely. The remaining ones have to be determined by inversion of equations 2.3 and 2.4. A major problem of this inversion is that both equations might become unsolvable in certain geometric configurations of the model.

Let’s consider an example where the Cartesian positions and velocities of the right foot are prescribed. This defines three coordinates for the position and three coordinates for the orientation of the foot segment. \( n_c \) is 6. If the optimization routine is used to set the initial position, orientation and velocity of the pelvis, another 6 of the initial generalized positions and velocities are defined. This leaves \( n_q - 12 \) generalized coordinates that can
be set to recorded values. However, after setting the position and orientation of the right foot and the pelvis, the configuration of the right leg is determined entirely. Therefore, the remaining $n_q - 12$ generalized coordinates have to be set in the left leg and the upper body. All generalize coordinates for the right leg are already determined. Furthermore, the position of the pelvis itself is already subject to geometric bounds. The length of the leg limits the pelvis position to a spherical range around the foot. This is especially problematic as the pelvis position and velocity is provided by the optimization routine and can’t be controlled by the algorithm.

As a consequence, the difference between the prescribed and the actual Cartesian positions and velocities are only minimized numerically, instead of being enforced rigorously. This allows the algorithm to run stable, even if not all geometric constraints can be fulfilled. In such cases, the differences between desired and calculated motion will contribute to the cost function (2.24) of the overlaid optimization routine. The value of this function will be considerably higher if not all constrains can be fulfilled and the routine will hence converge to a solution where the initial conditions can fulfill all geometric constraints.

**Evaluation with Experimental Data**

The presented method was used to perform exemplary inverse dynamics analysis on the motion data of a young healthy adult. The data was collected experimentally for a full cycle of gait, using a passive, optical marker motion capture system (Motion Analysis Corporation, Santa Rosa, CA, USA). Ground reaction forces were collected simultaneously with three force plates embedded in the lab floor (Advanced Mechanical Technologies, Newton, MA). Marker kinematics and force plate data were filtered at 6 Hz and 20 Hz, respectively.

A three dimensional full body model consisting of 12 independent segments was used for the analysis. The generalized coordinates used corresponded to the 29 degrees-of-freedom that fully described the model’s state (Table 2.1). The model’s segment lengths and anthropometric parameters where determined by the estimated joint-to-joint lengths of the individual subjects, and the subject’s overall height and body mass. Inverse kinematics analysis was performed and the generalized coordinates extracted. These trajectories were spline-fitted with a cut-off frequency of 6 Hz and the corresponding accelerations determined from the second spline derivative.

All generalized accelerations and ground reactions were estimated with a least squares approach (Section 2.3). Measurement standard deviations of external Forces and Moments
Generalized Coords. | Description
--- | ---
Position of the pelvis | A 3-D translation vector and 3 body-fixed Z-X-Y rotation angles.
Lower limb joint angles | Hip flexion/extension, abduction/adduction, and rotation; Knee flexion/extension; Ankle dorsiflexion/plantarflexion and eversion/inversion.
Low back angles | Successive flexion/extension, lateral bending, and transverse rotation.
Upper limb joint angles | Shoulder flexion/extension, abduction/adduction, and rotation; Elbow flexion/extension.

Table 2.1: The full body model consisted of 12 independent segments (Pelvis, thighs, shanks, feet, trunk, upperarm and forearms). The model state was described by 29 generalized coordinates that mostly corresponded to joint angles.

were assumed to be negligible (1e-6 N and 1e-6 Nm, respectively). The standard deviation of measured generalized accelerations was set to $2.0 \text{rad/s}^2$ (Kuo, 1998). Optimization of the initial conditions was performed for the entire set of generalized coordinates and speeds. The kinematics of 38 Markers were used in the objective function and compared to their experimentally recorded trajectories (Table 2.2). Inverse dynamic analysis was performed with the original and with the processed acceleration estimates. The estimated accelerations and the resulting torques of both methods were compared.

<table>
<thead>
<tr>
<th>Segment</th>
<th>Nr of Markers</th>
<th>Weighting factor (relative)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pelvis</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>Trunk</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>Thigh (Tracking)</td>
<td>1 (2x)</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>3 (2x)</td>
<td>1</td>
</tr>
<tr>
<td>Shank (Tracking)</td>
<td>1 (2x)</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>4 (2x)</td>
<td>1</td>
</tr>
<tr>
<td>Foot</td>
<td>2 (2x)</td>
<td>5</td>
</tr>
<tr>
<td>Upper arm</td>
<td>2 (2x)</td>
<td>1</td>
</tr>
<tr>
<td>Fore arm</td>
<td>2 (2x)</td>
<td>1</td>
</tr>
<tr>
<td>Total</td>
<td>38</td>
<td>23</td>
</tr>
</tbody>
</table>

Table 2.2: The initial model configuration was determined by minimizing the weighted distance between computed and recorded marker trajectories over a full integration of the motion. These 38 markers contributed to this objective functions
The data processing was implemented in C, using the Dynamics Pipeline of SIMM (Motion Analysis Corporation, Santa Rosa, CA, USA). The equations of motion were derived with SD/FAST (Parametric Technology Corporation, Waltham, MA). All numerical optimizations were performed using a sequential quadratic programming method (FSQP; AEM Design, Tucker, GA).

2.4 Results

When performing ‘bottom up’ inverse dynamics analysis (Winter, 1990) on the unprocessed data, residual forces arise at the base segment (the pelvis). These virtual forces have no physical meaning and are a result of model errors and inaccurate measurements. The proposed algorithm correctly eliminated these residual forces by estimating external reaction forces and accelerations that balance the equations of motion. Figure 2.1 shows the residual forces at the pelvis, before and after processing of the acceleration data.

Figure 2.1: Residual forces and torques acting on the pelvis. The traditional inverse dynamics analysis produces substantial residuals (dashed red lines) that have no physical meaning and impair subsequent analysis. The proposed method eliminates these residuals completely (solid blue lines) by estimating accelerations and reaction forces consistent with the equations of motion.

Changes between generalized coordinates before and after balancing the external reactions and generalized accelerations were relatively small (Table 2.3). They were less then 20 mm
for pelvic positions and about 1 deg for pelvic orientation and joint angles. The average distance between recorded and model-based marker positions was $\theta 10.5 \pm 4.8$ mm after the inverse kinematics analysis. Adapting the motion to balance the equations of motion increased this error to $\theta 27.9 \pm 10.9$ mm.

The differences in joint torque estimates were most notable for joints close to the pelvis. Figure 2.2 shows a comparison for the joint torques at the back and in the lower limbs.

<table>
<thead>
<tr>
<th>Generalized Coordinate</th>
<th>Average Difference</th>
<th>Joint Torque Estimate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pelvic A-P Translation</td>
<td>$\theta 17.9 \pm 3.3$ mm</td>
<td>Hip Flexion (L) $\theta 0.7 \pm 0.7$ deg</td>
</tr>
<tr>
<td>Pelvic M-L Translation</td>
<td>$\theta 2.5 \pm 1.6$ mm</td>
<td>Hip Adduction (L) $\theta 0.9 \pm 0.8$ deg</td>
</tr>
<tr>
<td>Pelvic S-I Translation</td>
<td>$\theta 15.9 \pm 2.9$ mm</td>
<td>Hip Internal Rotation (L) $\theta 1.2 \pm 1.2$ deg</td>
</tr>
<tr>
<td>Pelvic Anterior Tilt</td>
<td>$\theta 1.5 \pm 0.7$ deg</td>
<td>Knee Flexion (L) $\theta 1.5 \pm 1.2$ deg</td>
</tr>
<tr>
<td>Pelvic Obliquity</td>
<td>$\theta 0.9 \pm 0.3$ deg</td>
<td>Ankle Flexion (L) $\theta 1.0 \pm 0.9$ deg</td>
</tr>
<tr>
<td>Pelvic Transverse Rotation</td>
<td>$\theta 0.7 \pm 0.4$ deg</td>
<td>Shoulder Flexion (R) $\theta 1.1 \pm 0.6$ deg</td>
</tr>
<tr>
<td>Lumbar Extension</td>
<td>$\theta 1.2 \pm 0.4$ deg</td>
<td>Shoulder Adduction (R) $\theta 1.1 \pm 1.4$ deg</td>
</tr>
<tr>
<td>Lumbar Lateral Bending</td>
<td>$\theta 1.1 \pm 0.7$ deg</td>
<td>Shoulder Internal Rotation (R) $\theta 1.1 \pm 1.4$ deg</td>
</tr>
<tr>
<td>Lumbar Rotation</td>
<td>$\theta 0.7 \pm 0.6$ deg</td>
<td>Elbow Flexion (R) $\theta 1.1 \pm 1.4$ deg</td>
</tr>
<tr>
<td>Hip Flexion (R)</td>
<td>$\theta 1.0 \pm 0.5$ deg</td>
<td>Shoulder Flexion (L) $\theta 1.3 \pm 1.2$ deg</td>
</tr>
<tr>
<td>Hip Adduction (R)</td>
<td>$\theta 0.9 \pm 0.6$ deg</td>
<td>Shoulder Adduction (L) $\theta 0.1 \pm 0.1$ deg</td>
</tr>
<tr>
<td>Hip Internal Rotation (R)</td>
<td>$\theta 1.2 \pm 1.3$ deg</td>
<td>Shoulder Internal Rotation (L) $\theta 0.6 \pm 0.4$ deg</td>
</tr>
<tr>
<td>Knee Flexion (R)</td>
<td>$\theta 0.9 \pm 0.6$ deg</td>
<td>Elbow Flexion (L) $\theta 1.3 \pm 1.3$ deg</td>
</tr>
<tr>
<td>Ankle Dorsiflexion (R)</td>
<td>$\theta 0.4 \pm 0.4$ deg</td>
<td></td>
</tr>
</tbody>
</table>

Table 2.3: Average differences between measured kinematics (obtained by inverse kinematics analysis) and the altered kinematics (obtained by integrating over the estimated accelerations).
Figure 2.2: Joint torques acting on the back and in the lower limbs. The results of the traditional inverse dynamics analysis (dashed red lines) is compared to torques computed after the residual forces have been eliminated (solid blue lines).
2.5 Discussion

This article proposed a method of eliminating residual forces in the inverse dynamic analysis of gait. Generalized accelerations, consistent with the overall equations of motion, were estimated with a least-squares method, while additional geometric constraints in the form of prescribed segment accelerations could be satisfied. The estimated accelerations were integrated over time, and numerical optimization was used to determine an initial model state that best replicated the recorded movement. The motion computed in this process is hence dynamically consistent over time. Joint torques, computed by an inverse dynamics analysis, will exactly reproduce the motion when used to drive a forward dynamic simulation.

Such forward dynamic simulations have, in contrast to inverse dynamic analyses, the advantage that they maintain the causal dynamic relationship of joint torques, accelerations and movement over time. Through this relationship, valuable information can be gained on how the human body creates locomotion. Forward simulations can qualitatively and quantitatively identify how certain joint torques and muscles contribute to the overall movement, and they can predict how this movement will change when modifications on the musculoskeletal or nervous system are made.

The proposed method closed two important gaps in the creation of such forward dynamic simulations. The first is the elimination of residual forces and torques at the pelvis. Any analysis of gait, that doesn’t eliminate these residuals, will eventually have to attribute some of the observed effects to them. As the residual forces and torques have no physical reality, their presence will only conceal other causal relationships. The second gap the method closed, was the step from ‘frame-based’ data analysis (Kuo, 1998; Cahout et al., 2002) towards data processing that is dynamically consistent over time. The huge benefit is that the created motion can be used directly in forward dynamic simulations. Feedback systems that were previously used to ensure that the simulated motion didn’t deviate from the recorded movement can now be omitted. It is possible to create the simulation in an ‘open loop’.

With this additional control, the tracking of certain joint angles, segments or markers can be emphasized over the entire motion, or over a certain time span. Some of the segment accelerations can even be determined accurately by prescribing accelerations. All this can be used to include inherent knowledge into the simulation. We are able to accurately replicate those parts of the motion that we are interested in, or of which we have the most accurate information.

The precise determination of initial model states also improves the simulation with regard
to the problem of the dynamic instability of gait. While feedback controllers are widely used to ensure proper tracking of joint angles (Seth and Pandy, 2006), they can’t stabilize the overall ‘inverted pendulum’ characteristics of human gait. Precise identification of the initial conditions is not solving the underlying problem either, but can retard its effect, so that a finite duration of a motion can be precisely simulated in spite of the instability.

The proposed method is getting closer to experimentally recorded marker data. Inverse kinematics analysis is still needed as a tool to create initial estimates of joint angles and generalized accelerations. Like every analysis, inverse kinematics introduces errors, and it might be worthwhile to try to find a way of entirely skipping this step and going directly from marker kinematics to estimating a consistent set of accelerations. Another possible way of improving the algorithm is by finding a more general way of formulating the dynamic constraints. The equations of motion that were used in this project, were reduced to the effects of external reactions onto the center of mass acceleration and angular momentum. Internal reactions were not accounted for. The main advantage of this formulation is its computational efficiency. However, the simplifying formulation conceals information about what happens within the system. While the overall equations of motion are being fulfilled, other physical constraint, like joint torque limits, can’t be monitored. When the equations of motion are stated in their full form, it might become possible to include further measurements or conditions in the estimation of accelerations. In such a formulation the requirement that the residual forces and torques at the pelvis are zero, might only be one of many constraints. Further limitations arise from the approximations of the model used. These include, for example, the assumptions that the body segments are truly rigid and that all movements happen within the given degrees of freedom. We also assumed that all model parameters can be estimated precisely. While other approaches used the redundancies of force and acceleration measurement to improve the assessment of model parameters (Vaughan et al., 1982), our method attributed all differences to errors in the force and acceleration measurements. Other error sources were completely ignored.
Chapter 3

Design and Implementation of an Adaptive Ground Contact Model

3.1 Abstract

Ground reaction forces are the driving element of human gait. They are an important factor in clinical studies and necessary in all types of gait analyses. Forward dynamic simulations, which are becoming an increasingly important tool in this area, need means of generating these reaction forces in order to work properly. An adaptive contact model, that can be used for this purpose, is presented in this work. It is based on a model published by Gilchrist and Winter (1996) and approximates the continuous contact process with punctiform viscoelastic units that are distributed under a two-segment model of the foot. Additional rotational dampers are used to smooth and stabilize the foot motion about all three spatial axes. A large set of parameters was used in the implementation to allow for tuning of the model and adaptation to subject specific differences.

Experimental gait data of five subjects was used to identify parameter combinations that are well suited for subject specific adaptation of the model; i.e. parameters that vary between different subjects, but remain constant for different trials of the same subject. Numerical optimization was used for the determination of individual parameters. For this optimization, both a gradient based method and a genetic algorithm were implemented and their performance compared.
3.2 Introduction

In forward dynamic simulations of human gait, a large number of strategies have been developed to account for the dynamic process of foot-floor interaction. The reaction forces that are produced by this process are extremely important for the performance of the simulation as they essentially determine the overall motion of the center of gravity. The easiest way of dealing with the foot-floor interactions is by excluding the entire contact process from the simulation. In this case, the experimentally recorded ground reaction forces are directly applied to the most distal segments during the forward dynamic simulation (Reinbolt et al., 2005; Thelen and Anderson, 2006). Another way of tackling this task is by fixing the stance foot on the ground. This can be done by reducing the degrees of freedom of the model and limiting the motion to the single foot stance-phase (Amirouche et al., 1990) or by imposing time variant geometric constraints (Hatze, 1981). However, it is often desired that ground reaction forces are an actual output of the simulation rather than an input or a constraint. So called ground contact models are used in these cases for the calculation of the reaction forces. These models describe the relation between the current position and velocity of the feet and the resulting external reactions. They can be described by:

\[
\begin{bmatrix}
\vec{F} \\
\vec{M}
\end{bmatrix} = f \left( \vec{q}, \dot{\vec{q}} \right)
\]

This equation resembles the formulation of a viscoelastic process. In fact, the real physical contact of foot and ground can be described by the interaction of a number of continuous viscoelastic elements. These elements are the ground surface and the soles of the shoes, but also the foot itself, with its ability to deform and absorb shocks through its bone structure, elastic ligaments, and soft tissue.

A common way of approximating the continuous elastic and viscous nature of these elements, is by placing discrete linear spring-damper units at the contact surface. These units embody the viscoelastic characteristics of a certain region and thus approximate the continuous surfaces. Many descriptions of such models can be found in the literature (Seth and Pandy, 2006; Anderson and Pandy, 1999; Neptune et al., 1999). The model developed in this project was largely based on the work of Gilchrist and Winter (1996). The complexity of their model seemed to be an optimal trade-off between computational efficiency and accuracy, particularly for the purpose of simulating gait. It was also one of the few models described in the literature that has been individually tested and validated.

Ground contact models, as described above, are such a crude approximation of the real
foot-shoe-floor system that it is impossible to reflect all physical properties of this system in the structure and parameters of the model. The models are rather designed empirically. In a first step, a general framework is established. Based on heuristic considerations and practical limitations, the design of such a framework determines, for example, the number of individual foot segments, the way they are connected, and how visco-elastic units are distributed underneath them. It might determine the overall nature of these units, define in which spatial directions they act, and if any additional elements, like rotational springs or dampers, are used. The details of the model, like the actual stiffness and damping properties of the units, are obtained in a second step by tuning the model on experimental data. In this process, the model parameters are altered until sufficient agreement between measured and predicted reaction forces results. A major problem of such an empirical formulation is the missing physical insight: How do changes in the model affect the predicted reaction forces? And, more importantly, how do subject specific properties (like subject size, or different kinds of footwear) translate into certain parameters of the model formulation?

To address these questions, the process of parameter adaptation was implemented as an optimization problem and thus automated. Adaptation was done for varying sets of parameters and with experimental gait data from different subjects. This allowed the identification of parameters that actually reflect subject specific properties, and to create a ground contact model capable of adapting to different subjects.

### 3.3 Methods

#### Design of the Model

The model of the foot consisted of two segments: The main part of the foot and a toe segment. Both parts were connected by a purely passive rotational spring/damper element at the metatarsal-phalangeal joint. The toe segment was given the dynamic properties of a massless body. Hence, no knowledge of the previous state of the metatarsal-phalangeal joint was necessary when evaluating the ground contact model in dynamic simulations. The model could be described as a static function that took the current position, orientation, translational velocity and rotational velocity of the main part of the foot as input. These properties (given in global coordinates) defined a local coordinate system for the model, which was used to transform the model description from local coordinates into the global reference frame. The external reaction forces and torques were calculated in global coordinates and passed to the
calling instance. The model implemented the following equation:

\[
\begin{bmatrix}
  F \\
  M
\end{bmatrix} = f \left( \vec{r}, \phi, \vec{r}, \vec{\omega} \right)
\]

where \( \vec{r} \) defines the position of the foot segment reference frame, \( \phi \) its orientation, and \( \vec{r} \) and \( \vec{\omega} \) its translational and rotational velocities. \( F \) is the force created at the origin of the foot segment (i.e. at the position \( \vec{r} \)), and \( M \) is a vector of torques about this point. All quantities are expressed in global coordinates.

![Figure 3.1: The toe segment was modeled as a massless body. The rotation of this segment about the metatarsal axis, and the velocity of this rotation, was therefore statically determined by the implicit balance of torque created by the metatarsal axis itself (yellow) and torque created by the visco-elastic units (green) in the toe segment.](image)

This static formulation facilitated the integration of the model into existing frameworks of forward dynamic simulations and allowed for a modular structure of the overall model. The drawback was that at every iteration the implicit equations imposed by the equilibrium of stiffness and damping torques about the metatarsal axis had to be solved. The balance of forces produced by the visco-elastic units in the toe segment and by the rotational spring and damper in the metatarsal axis determined the current rotation and rotational velocity of the toe segment (Fig. 3.1). Fortunately, the function that described the relation of rotation and torque was mathematically well behaved. The used numerical solver (a modification of the Regula Falsi method described in (Press et al., 1992)) converged rapidly and at little computational expense. Torque was a linear function of rotational velocity, which could
hence be determined analytically. Figure 3.2 shows an exemplary relationship of torque and rotation as well as torque and rotational velocity.

![Graph showing relationship between rotation and torque/damping](image)

**Figure 3.2: Stiffness and damping torques about the metatarsal axis were a function of rotation and rotational speed of the toe segment. Static balance requires these functions to be zero. They had to be solved numerically for metatarsal rotation, and metatarsal rotational velocity. (The functions are given for an exemplary configuration of the model right before toe-off.)**

Eleven punctiform visco-elastic units were placed linearly spaced underneath the two parts of the foot. They were distributed roughly along the course of the center of pressure during normal gait. The number of the units has been a trade-off between closely approximating the continuous surfaces of the contact elements and the high computational costs of too many units.

Additional rotational dampers along all three spatial axes ensured that the transition between contact elements was smooth and that only little rotation occurred about the anterior/posterior and vertical axes. Arranging the elements along only one dimension and using rotational damping for the others had the big advantage of greatly reducing the number of required elements from $\mathcal{O}(n^2)$ to $\mathcal{O}(n)$ while the center of pressure could still move unrestricted and in a natural way.

Each unit consisted of a linear spring-damper element which created a vertical force proportional to the magnitude and speed of penetration of the unit into the ground surface. Translational dampers, perpendicular to the vertical elements created viscous friction against
movements within the ground plane. No elastic elements were used for this purpose, which made it unnecessary to keep track of the position of initial ground contact.

The piecewise cubic function

\[
c(x) = \begin{cases} 
0 & x \leq 0 \\
c_{\text{max}} \cdot \left( \frac{x}{x_{\text{max}}} \right)^2 \cdot \left[ 3 - 2 \left( \frac{x}{x_{\text{max}}} \right) \right] & 0 \leq x \leq x_{\text{max}} \\
c_{\text{max}} & x \geq x_{\text{max}} 
\end{cases}
\]

was used to scale the damping coefficients \( c \) between 0 and \( c_{\text{max}} \) according to the deflection parameter \( x \). This was done to smooth the sudden rise of damping force that would occur when additional damping elements became involved in the contact process. The parameter \( x_{\text{max}} \) defined the point at which the maximal damping \( c_{\text{max}} \) was reached. This function is shown for exemplary values of \( x_{\text{max}} \) and \( c_{\text{max}} \) in Figure 3.3.

![Figure 3.3](image)

Figure 3.3: All damping coefficients \( c \) were scaled according to the deflection of the corresponding unit using a piecewise cubic function. This smoothly increased damping from zero to \( c_{\text{max}} \) (reached at deflection \( x_{\text{max}} \)) when units became involved in the contact process.

For each of the three damping components, translational damping of the units, rotational damping, and damping about the metatarsal axis, a separate function was defined. \( x \) and \( x_{\text{max}} \) were expressed in terms of penetration of individual units into the ground plane for the translational damping, in terms of average unit penetration for the rotational damping coefficients, and in terms of extension of the metatarsal-phalangeal joint for damping about the metatarsal axis.
The model was fully described by a set of parameters that defined:

- the number of units
- the position of these units
- the damping and stiffness properties of these units
- the position and orientation of the metatarsal axis, as well as the according stiffness and damping values
- the rotational damping coefficients
- the damping onset functions for translation and rotational damping, as well as for damping about the metatarsal axis.

Additional redundant parameters were introduced to allow setting the overall position, scaling and orientation of all units simultaneously. These parameters defined the position, orientation and dimensions of a bounding box in relation to the model reference frame. The absolute positions of the units and the metatarsal axis were defined in these bounding box coordinates, which were normalized to the interval $[-1...1]$. This concept allowed for easy adjustment of the model with respect to changes in the contact surfaces (for example because of varying sizes of footwear). It also facilitated the use of different foot segment reference frames without altering the entire model. Such reference frames are typically defined at the COG of the foot, at the most posterior point of the heel, or in the ankle joint.

Other parameters provided scaling factors for the stiffness and damping properties of all units, so that these properties could be changed simultaneously as well. For a detailed description of all parameters please refer to Appendix A. Figure 3.4 shows a three side view of the complete ground contact model.
Figure 3.4: Top, side and front view of the ground contact model. The local reference frame (red) is placed in the center of gravity of the foot segment. The current translational and rotational velocities of the segment are represented by dark blue and light blue arrows, respectively. The units and the metatarsal axis are defined in relation to the bounding box (black) and illustrated by small spheres and a cylinder. The forces acting on the individual units are represented by grey arrows, the overall force acting on the origin of the local reference frame by a green arrow. The overall torque acting on the segment is shown in yellow.
Parameter Optimization

The parameters of the model were initially set to values reported by Gilchrist and Winter (1996). In a adjacent step, numerical optimization was used to establish a set of parameters that minimized the difference between recorded and predicted ground reaction forces for a given foot motion. The objective function used by the optimization routine expressed the weighted average of squared differences between the predicted and measured forces and the positions of the centers of pressure. The weights were chosen so that an error of 30 N in force corresponded to an error of 1 cm in the center of pressure position. This reflected how much both quantities changed during regular ground contact. The forces within the ground plane and the vertical torque component did not contribute to the objective function, as the extracted foot motion showed relatively large noise for horizontal movement.

\[
E(\vec{p}) = \frac{1}{n_{frame}} \cdot \sum_{i=1}^{n_{frame}} \left[ w_F \cdot \left( \vec{F}_i - \vec{F}'_i \right)^2 + w_\rho \cdot \left( \vec{\rho}_i - \vec{\rho}'_i \right)^2 \right]
\]

\(n_{frame}\) is the number of frames, \(w_F\) a vector of weights for the differences in forces \(\vec{F}_i\), and \(w_\rho\) a vector of weights for the error in position of the center of pressure \(\vec{\rho}_i\). Experimental values are denoted with a prime.

The optimization was performed for various subsets of parameters and two different optimization methods were tested. These were a genetic algorithm (Houck et al., 1996), and a bounded sequential quadratic programming method (Mat, 2006). The latter seemed to be the most promising approach for this kind of problem.

The genetic algorithm maintained a population size of 1000 individuals (Goldberg et al., 1992). 300 generations were generated to allow for good estimation while reducing over fitting to noise. The algorithm used arithmetic crossover, uniform mutation and tournament selection for the creation of successor generations. Figure 3.5 shows an example of the convergence behavior of the algorithm.

Implementation using Experimental Gait Data

The experimental data was obtained from a full body gait analysis of five subjects wearing tennis shoes. Three of the subjects were female, two male (Table 3.1). Model-based inverse kinematics analysis was performed over a full cycle of gait and the position, orientation and velocities of the right foot segment were extracted. Ground reaction forces were recorded simultaneously. Both, kinematic and kinetic data was generated at a framerate of 100 Hz.
Every subject performed 15 trials. Five each at 80%, 100% and 120% of their comfortable walking speed.

<table>
<thead>
<tr>
<th>Subject</th>
<th>Gender</th>
<th>Size</th>
<th>Age</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Y4</td>
<td>F</td>
<td>1.57 m</td>
<td>24 years</td>
</tr>
<tr>
<td>2 Y5</td>
<td>F</td>
<td>1.65 m</td>
<td>23 years</td>
</tr>
<tr>
<td>3 Y10</td>
<td>M</td>
<td>1.73 m</td>
<td>32 years</td>
</tr>
<tr>
<td>4 Y12</td>
<td>M</td>
<td>1.85 m</td>
<td>21 years</td>
</tr>
<tr>
<td>5 Y15</td>
<td>F</td>
<td>1.68 m</td>
<td>24 years</td>
</tr>
<tr>
<td>Average</td>
<td>⊗1.67 ± 0.1 m</td>
<td>⊗24.8 ± 4.2 years</td>
<td></td>
</tr>
</tbody>
</table>

Table 3.1: Subjects in the study

The ground contact model was implemented in MATLAB and in C. The optimization was performed in MATLAB using the Matlab Optimization Toolbox and the Genetic Algorithms for Optimization Toolbox (GOAT) developed at the North Carolina State University. For details of the implementation, please refer to Appendix B.
3.4 Results

The overall vertical position of the units in relation to the local reference frame (parameter ‘bb_pos.y’) was determined first. This value is probably the most important parameter, as it essentially determines the magnitude of the vertical forces and the scaling of the damping coefficients. Optimization was limited to the interval [-8 cm, -2 cm]. The computed mean values and standard deviations for this parameter are listed in Table 3.2 and shown in Figure 3.6.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Subject 1</th>
<th>Subject 2</th>
<th>Subject 3</th>
<th>Subject 4</th>
<th>Subject 5</th>
<th>Overall</th>
</tr>
</thead>
<tbody>
<tr>
<td>b\textunderscore pos\textunderscore y [cm]</td>
<td>⊗ -6.126</td>
<td>-5.339</td>
<td>-6.310</td>
<td>-6.915</td>
<td>-6.294</td>
<td>-6.209</td>
</tr>
<tr>
<td>σ</td>
<td>0.071</td>
<td>0.047</td>
<td>0.116</td>
<td>0.073</td>
<td>0.094</td>
<td>0.510</td>
</tr>
</tbody>
</table>

Table 3.2: Parameter values for the overall vertical position of the units in relation to the local reference frame

Figure 3.6: Optimization results for parameter bb\_pos\_y

The fact that the standard deviations for different trials of the same subject were almost a magnitude smaller than the overall standard deviation, suggests that the approach actually extracts subject specific properties. In an subsequent step, additional parameter candidates
were examined. The optimization was run for pairs of parameters, with the overall vertical position of the units (parameter ‘bb_pos.y’) always being one of the two. The second parameter was one of the parameters listed in Table 3.3. The relative size of the subject specific standard deviations and the overall standard deviation was used as means to determine which of these parameters were possible candidates for subject specific adaptation of a ground contact model. The results of the optimization process are summarized in Table 3.4.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Optim. Interval</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>bb_pos_x</td>
<td>[0 cm, 10 cm]</td>
<td>Overall position of the units in x direction</td>
</tr>
<tr>
<td>bb_pos_z</td>
<td>[-6 cm, 6 cm]</td>
<td>Overall position of the units in z direction</td>
</tr>
<tr>
<td>bb_rot_3</td>
<td>[-0.5 rad, 0.5 rad]</td>
<td>Overall rotation of the units about the body fixed 3(z)-axis</td>
</tr>
<tr>
<td>bb_rot_1</td>
<td>[-0.7 rad, 0.7 rad]</td>
<td>Overall rotation of the units about the body fixed 1(x)-axis</td>
</tr>
<tr>
<td>bb_rot_2</td>
<td>[-0.5 rad, 0.5 rad]</td>
<td>Overall rotation of the units about the body fixed 2(y)-axis</td>
</tr>
<tr>
<td>bb_scale_x</td>
<td>[0 cm, 30 cm]</td>
<td>Scalefactor of the bounding box coordinate system in x direction</td>
</tr>
<tr>
<td>bb_scale_z</td>
<td>[-10 cm, 10 cm]</td>
<td>Scalefactor of the bounding box coordinate system in z direction</td>
</tr>
<tr>
<td>meta_pos_x</td>
<td>[0 , 1]</td>
<td>Position of the metatarsal axis in bounding box x-coordinates</td>
</tr>
<tr>
<td>meta_stiff</td>
<td>[0 Nrad, 150 Nrad]</td>
<td>Stiffness of the metatarsal axis</td>
</tr>
<tr>
<td>meta_damp</td>
<td>[0 Nrad, 25 Nrad]</td>
<td>Damping coefficient of the metatarsal axis</td>
</tr>
<tr>
<td>trans_damp_x</td>
<td>[0 Nm, 2000 Nm]</td>
<td>Translational damping coefficient in the x direction</td>
</tr>
<tr>
<td>trans_damp_y</td>
<td>[0 Nm, 1000 Nm]</td>
<td>Translational damping coefficient in the y direction</td>
</tr>
<tr>
<td>trans_damp_z</td>
<td>[0 Nm, 2000 Nm]</td>
<td>Translational damping coefficient in the z direction</td>
</tr>
<tr>
<td>rot_damp_x</td>
<td>[0 Nrad, 100 Nrad]</td>
<td>Rotational damping coefficient about the x axis</td>
</tr>
<tr>
<td>rot_damp_z</td>
<td>[0 Nrad, 5 Nrad]</td>
<td>Rotational damping coefficient about the z axis</td>
</tr>
</tbody>
</table>

Table 3.3: List of parameters adapted in combination with bb_pos.y. A detailed description of these parameters can be found in Appendix A.
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Subject 1</th>
<th>Subject 2</th>
<th>Subject 3</th>
<th>Subject 4</th>
<th>Subject 5</th>
<th>Overall</th>
</tr>
</thead>
<tbody>
<tr>
<td>bb_pos_x [cm]</td>
<td>3.804</td>
<td>4.631</td>
<td>7.563</td>
<td>7.469</td>
<td>6.005</td>
<td>5.911</td>
</tr>
<tr>
<td>bb_pos_z [cm]</td>
<td>-5.314</td>
<td>-6.000</td>
<td>6.000</td>
<td>-3.600</td>
<td>-6.000</td>
<td>-2.942</td>
</tr>
<tr>
<td>bb_pos_y [cm]</td>
<td>-6.392</td>
<td>-5.940</td>
<td>-6.038</td>
<td>-7.204</td>
<td>-7.001</td>
<td>-6.524</td>
</tr>
<tr>
<td>bb_rot_3 [rad]</td>
<td>-0.099</td>
<td>-0.429</td>
<td>0.480</td>
<td>-0.398</td>
<td>0.388</td>
<td>-0.006</td>
</tr>
<tr>
<td>bb_rot_1 [rad]</td>
<td>0.576</td>
<td>-0.600</td>
<td>-0.700</td>
<td>0.700</td>
<td>-0.626</td>
<td>-0.124</td>
</tr>
<tr>
<td>bb_pos_y [cm]</td>
<td>-5.736</td>
<td>-5.781</td>
<td>-7.023</td>
<td>-5.725</td>
<td>-6.758</td>
<td>-6.089</td>
</tr>
<tr>
<td>bb_rot_2 [rad]</td>
<td>0.314</td>
<td>0.016</td>
<td>0.487</td>
<td>-0.284</td>
<td>0.407</td>
<td>0.190</td>
</tr>
<tr>
<td>bb_scale_x [cm]</td>
<td>9.475</td>
<td>0</td>
<td>13.275</td>
<td>18.135</td>
<td>11.335</td>
<td>10.585</td>
</tr>
<tr>
<td>bb_pos_y [cm]</td>
<td>-6.090</td>
<td>-4.853</td>
<td>-6.271</td>
<td>-6.968</td>
<td>-6.103</td>
<td>-6.073</td>
</tr>
<tr>
<td>bb_pos_y [cm]</td>
<td>0.108</td>
<td>1.214*</td>
<td>0.130</td>
<td>0.106</td>
<td>0.252</td>
<td>0.751</td>
</tr>
<tr>
<td>bb_pos_y [cm]</td>
<td>9.475</td>
<td>0</td>
<td>13.275</td>
<td>18.135</td>
<td>11.335</td>
<td>10.585</td>
</tr>
<tr>
<td>bb_pos_y [cm]</td>
<td>-6.176</td>
<td>-2.000</td>
<td>-6.276</td>
<td>-6.898</td>
<td>-6.359</td>
<td>-5.590</td>
</tr>
<tr>
<td>meta_pos_X [m]</td>
<td>0.567</td>
<td>0.622</td>
<td>0.772</td>
<td>0.768</td>
<td>0.648</td>
<td>0.676</td>
</tr>
<tr>
<td>meta_stiff [N/°]</td>
<td>32.507</td>
<td>59.650</td>
<td>138.07</td>
<td>150.00</td>
<td>58.274</td>
<td>88.078</td>
</tr>
</tbody>
</table>

Continued on the next page
<table>
<thead>
<tr>
<th>Parameter cont.</th>
<th>Subject 1</th>
<th>Subject 2</th>
<th>Subject 3</th>
<th>Subject 4</th>
<th>Subject 5</th>
<th>Overall</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\sigma$</td>
<td>6.869</td>
<td>20.876</td>
<td>14.232$^*$</td>
<td>10.768</td>
<td>49.202$^*$</td>
<td></td>
</tr>
<tr>
<td>$\sigma$</td>
<td>0.070</td>
<td>0.896</td>
<td>0.114</td>
<td>0.083</td>
<td>0.092</td>
<td>0.695</td>
</tr>
<tr>
<td>$\sigma$</td>
<td>8.241$^*$</td>
<td>0.000$^*$</td>
<td>0.006$^*$</td>
<td>*</td>
<td>8.797$^*$</td>
<td>4.674$^*$</td>
</tr>
<tr>
<td>$\beta_{pos,y}$ [cm]</td>
<td>-6.127</td>
<td>-4.967</td>
<td>-6.310</td>
<td>-6.915</td>
<td>-6.296</td>
<td>-6.139</td>
</tr>
<tr>
<td>$\sigma$</td>
<td>0.070</td>
<td>0.497$^*$</td>
<td>0.118</td>
<td>0.072</td>
<td>0.094</td>
<td>0.118</td>
</tr>
<tr>
<td>$\sigma$</td>
<td>0.071</td>
<td>0.048</td>
<td>0.116</td>
<td>0.072</td>
<td>0.096</td>
<td>0.504</td>
</tr>
<tr>
<td>$\beta_{pos,y}$ [cm]</td>
<td>-6.126</td>
<td>-5.339</td>
<td>-6.310</td>
<td>-6.915</td>
<td>-6.294</td>
<td>-6.209</td>
</tr>
<tr>
<td>$\sigma$</td>
<td>0.071</td>
<td>0.047</td>
<td>0.116</td>
<td>0.072</td>
<td>0.095</td>
<td>0.510</td>
</tr>
<tr>
<td>$\beta_{pos,y}$ [cm]</td>
<td>-6.127</td>
<td>-5.339</td>
<td>-6.310</td>
<td>-6.915</td>
<td>-6.294</td>
<td>-6.209</td>
</tr>
<tr>
<td>$\sigma$</td>
<td>0.070</td>
<td>0.047</td>
<td>0.116</td>
<td>0.072</td>
<td>0.095</td>
<td>0.509</td>
</tr>
<tr>
<td>$\beta_{pos,y}$ [cm]</td>
<td>-6.126</td>
<td>-5.339</td>
<td>-6.310</td>
<td>-6.915</td>
<td>-6.294</td>
<td>-6.209</td>
</tr>
<tr>
<td>$\sigma$</td>
<td>0.070</td>
<td>0.047</td>
<td>0.116</td>
<td>0.072</td>
<td>0.095</td>
<td>0.509</td>
</tr>
<tr>
<td>$\beta_{pos,y}$ [cm]</td>
<td>-6.127</td>
<td>-5.339</td>
<td>-6.310</td>
<td>-6.915</td>
<td>-6.294</td>
<td>-6.209</td>
</tr>
<tr>
<td>$\sigma$</td>
<td>0.070</td>
<td>0.047</td>
<td>0.116</td>
<td>0.072</td>
<td>0.095</td>
<td>0.509</td>
</tr>
<tr>
<td>$\beta_{pos,y}$ [cm]</td>
<td>-6.126</td>
<td>-5.339</td>
<td>-6.310</td>
<td>-6.915</td>
<td>-6.294</td>
<td>-6.209</td>
</tr>
<tr>
<td>$\sigma$</td>
<td>0.070</td>
<td>0.047</td>
<td>0.116</td>
<td>0.072</td>
<td>0.095</td>
<td>0.509</td>
</tr>
<tr>
<td>$\beta_{pos,y}$ [cm]</td>
<td>-6.126</td>
<td>-5.339</td>
<td>-6.310</td>
<td>-6.915</td>
<td>-6.294</td>
<td>-6.209</td>
</tr>
<tr>
<td>$\sigma$</td>
<td>0.070</td>
<td>0.047</td>
<td>0.116</td>
<td>0.072</td>
<td>0.095</td>
<td>0.509</td>
</tr>
</tbody>
</table>

Table 3.4: Optimization results for pairs of parameters

( * The optimization ran into the boundaries for one or more trials)

Looking at this data reveals two trends: First, all damping parameters showed extremely large variations when optimized and many of the trials produced damping parameters outside
the given estimation range. Second, parameters that primarily acted in the anterior posterior direction showed much better results than parameters that affected the other directions. Best results were achieved for combinations with the parameters $bb_{pos,x}$ and $meta_{pos,x}$ which defined the anterior/posterior position of the units and of the metatarsal axis. Acceptable results, for at least some of the subjects were achieved by the parameters $bb_{scale,x}$, $bb_{rot,3}$ and $meta_{stiff}$.

Based on this result, further analysis was performed on a set containing the three parameters $bb_{pos,x}$, $bb_{pos,y}$, and $meta_{pos,x}$. Appendix A contains a description of these parameters. The optimizations for these trials were performed with the gradient based SQP-algorithm and the genetic algorithm. Results are listed in Tables 3.5 and 3.6 and shown in Figure 3.7.

![Optimization results](image)

Figure 3.7: Optimization results for the combination of the parameters: $bb_{pos,x}$, $bb_{pos,y}$, and $meta_{pos,x}$

The average value of the objective function was $\Theta 265 \pm 92.5$ with the original parameters (75 trials; each over a full cycle of gait). This is equivalent to a force error of 265 N, or an error in the position of the center of pressure of 8.8 cm. The local optimization method was
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Subject 1</th>
<th>Subject 2</th>
<th>Subject 3</th>
<th>Subject 4</th>
<th>Subject 5</th>
<th>Overall</th>
</tr>
</thead>
<tbody>
<tr>
<td>( bb_{pos_y} ) [cm]</td>
<td>( -6.292 )</td>
<td>( -6.360 )</td>
<td>( -6.554 )</td>
<td>( -6.818 )</td>
<td>( -7.489 )</td>
<td>( -6.707 )</td>
</tr>
<tr>
<td>( \sigma )</td>
<td>0.170</td>
<td>0.300</td>
<td>0.130</td>
<td>0.127</td>
<td>0.447*</td>
<td>0.508*</td>
</tr>
<tr>
<td>( bb_{pos_y} ) [cm]</td>
<td>2.774</td>
<td>5.851</td>
<td>3.768</td>
<td>4.420</td>
<td>6.662</td>
<td>4.680</td>
</tr>
<tr>
<td>( \sigma )</td>
<td>1.409*</td>
<td>1.830</td>
<td>0.985</td>
<td>1.102</td>
<td>2.920</td>
<td>2.243*</td>
</tr>
<tr>
<td>( bb_{pos_y} ) [cm]</td>
<td>0.402</td>
<td>0.203</td>
<td>0.590</td>
<td>0.539</td>
<td>0.181</td>
<td>0.385</td>
</tr>
<tr>
<td>( \sigma )</td>
<td>0.109</td>
<td>0.168*</td>
<td>0.075</td>
<td>0.092</td>
<td>0.269*</td>
<td>0.229*</td>
</tr>
</tbody>
</table>

Table 3.5: Optimization results for the combination of the parameters: \( bb_{pos_x} \), \( bb_{pos_y} \), and \( meta_{pos_x} \) using a gradient based SQP optimization method. (* The optimization ran into the boundaries for one or more trials)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Subject 1</th>
<th>Subject 2</th>
<th>Subject 3</th>
<th>Subject 4</th>
<th>Subject 5</th>
<th>Overall</th>
</tr>
</thead>
<tbody>
<tr>
<td>( bb_{pos_y} ) [cm]</td>
<td>( -6.261 )</td>
<td>( -6.414 )</td>
<td>( -6.563 )</td>
<td>( -6.673 )</td>
<td>( -7.312 )</td>
<td>( -6.648 )</td>
</tr>
<tr>
<td>( \sigma )</td>
<td>0.140</td>
<td>0.226</td>
<td>0.331</td>
<td>0.266</td>
<td>0.408</td>
<td>0.461</td>
</tr>
<tr>
<td>( bb_{pos_y} ) [cm]</td>
<td>2.401</td>
<td>6.272</td>
<td>3.756</td>
<td>2.862</td>
<td>5.113</td>
<td>4.051</td>
</tr>
<tr>
<td>( \sigma )</td>
<td>1.318</td>
<td>1.797</td>
<td>2.562*</td>
<td>2.078*</td>
<td>2.693</td>
<td>2.536*</td>
</tr>
<tr>
<td>( bb_{pos_y} ) [cm]</td>
<td>0.432</td>
<td>0.158</td>
<td>0.597</td>
<td>0.670</td>
<td>0.304</td>
<td>0.436</td>
</tr>
<tr>
<td>( \sigma )</td>
<td>0.094</td>
<td>0.154*</td>
<td>0.228</td>
<td>0.174</td>
<td>0.246</td>
<td>0.261*</td>
</tr>
</tbody>
</table>

Table 3.6: Optimization results for the combination of the parameters: \( bb_{pos_x} \), \( bb_{pos_y} \), and \( meta_{pos_x} \) using a genetic algorithm. (* The optimization ran into the boundaries for one or more trials)

able to significantly reduce this error to \( \varnothing 113 \pm 27.0 \). The genetic algorithm produced almost the same results, the error values was \( \varnothing 114 \pm 28.3 \).

3.5 Discussion

The goal of this study was to design and implement a model of the dynamic foot-floor interactions during human gait. Special attention was paid to the issue of subject specific adaptation. Numerical optimization was used to test various model parameters, and finally establish a set of parameters that was well suited for adaptation.

Ground contact models are mostly developed empirically. The sophisticated structure of the human foot and the complex contact process make it difficult to create a physical model
that is accurate enough to represent the actual physics of the ground contact. Due to the coarse approximations in such models, it is uncertain if and how model parameters related to actual physical properties. The gait data of five different subjects (each performing 15 trials at different walking speeds) was used to investigate in how far certain model parameters reflected subject specific properties.

The following three parameters were identified as being subject specific:

- The overall vertical position of the visco-elastic units.
- The overall anterior-posterior position of the visco-elastic units.
- The position of the metatarsal axis.

It was shown that numerical optimization of these three parameters can substantially improve the model-based prediction of ground reaction forces. The error between predicted and recorded ground reaction forces was reduced by 55% by the parameter optimization.

The study indicated that especially damping parameters were ill suited for optimization. This can be attributed to two effects. First, damping is used in the model primarily to stabilize movement. No physical equivalent exists for the rotational dampers, and the translational dampers are, even if they can be interpreted physically, primarily used to smoothen the motion and to eliminate the need for elastic elements within the ground plane. Second, damping forces are a function of velocity, which is obtained by differentiation of position data and therefore contains more noise which degrades the optimization.

It also became evident that it was easier to estimate parameters that affected the model properties in the sagittal plane. This is due to the fact, that most of the foot motion happens in this plane, which significantly improves the signal-to-noise ratio for motion and force data in this plane.

Any attempts to include more then three parameters into the optimization failed. The objective functions used in the numerical optimization are not exactly well behaved and reproduced mostly noise artifacts when optimizing in too many dimensions. Using a genetic algorithm couldn’t really improve the results. It seemed to be more stable and less affected by the multitude of local minima at the limits of the objective function, but couldn’t improved the overall performance. Problematic in the context of adapting many parameters at once, is also the fact that many combinations of parameters are partially redundant. A rotation about the anterior-posterior axis and lateral displacement contribute to the vertical displacement. Moving all units closer to the ground has the same effect as increasing the
vertical stiffness. And even changes in the overall length of the contact model and the stiffness of the metatarsal joint have a similar impact on the created forces. In many of these cases numerical optimization –especially the gradient based approach – created parameters far away from any reasonable value.

For the studied parameter combinations, the gradient based SQP algorithm showed a better overall performance. As generally a priori knowledge exists about the model, it is easy to provide a good initial guess for the optimization, which converges substantially faster then the genetic algorithm.

The biggest flaw in the analysis was probably the fact that it was limited to walking data. A good ground contact model should be able to replicate other movements like running or jumping, and including these movements into the analysis might provide further evidence that the ground contact model actually represents the physical reality and is not only a mathematical construct.
Chapter 4

Integration of the Ground Contact Model in the current Simulation Framework

4.1 Abstract/Introduction

Integrating a ground contact model into a forward dynamic simulation of gait is challenging for a couple of reasons. First of all, it is very important that the reaction forces created by the ground contact model closely replicate recorded external reactions. They are the driving element behind the overall motion and essentially determine the motion of the center of mass of the model. This is a difficult task, as the ground contact model used in this study (see Chapter 3) has the properties of a very stiff and heavily damped visco-elastic system that relates the positions and velocities of the feet to external reaction forces. Small perturbations in position or velocity lead to large errors in ground reaction force. The motion of the feet have to be determined as accurate as possible so that the desired and model-based ground reaction forces are as similar as possible. A main problem in this adaptation process is that the ground contact model is described by a nonlinear equation. It can’t be inverted analytically, and even numerical inversion is infeasible as it would lead to very ambiguous results. Various configurations of the model can produce the same reaction forces. For example, a model standing on its heels and a model standing on its toes are able to produce undistinguishable reaction forces. Furthermore, the input variables position and velocity are dynamically linked and have to be a smooth function over time. They can’t be computed independently, but have to be integrated from continuous acceleration values.

In this section, a dynamic controller is presented that minimizes the differences between recorded and model-predicted ground reaction forces and, at the same time, keeps the motion
of the feet as close to their recorded trajectories as possible. The foot trajectories created by this process are then incorporated as geometric constraints in the acceleration estimations introduced in Chapter 2. The Computed Muscle Control algorithm, developed by Thelen and Anderson (2006) was applied to the processed data and a muscle drive forward simulation of a half cycle of gait was computed.

4.2 Methods

Contact Model Formulation

The contact model that was integrated in the existing simulation environment was based on modeling the contact as a visco-elastic process (rather than on spatial constraints). Models of this kind can be described mathematically as:

$$\begin{bmatrix} \vec{F} \\ \vec{M} \end{bmatrix} = f \left( \vec{q}, \dot{\vec{q}} \right)$$  \hspace{1cm} (4.1)

Depending on the implementation of a contact model the nature of $f \left( \vec{q}, \dot{\vec{q}} \right)$ varies. The specific model used in this project implemented equation (4.1) as a superposition of individual linear spring/damper units that come in and out of contact as the foot touches and leaves the ground. Hence the function the algorithm has to deal with is globally nonlinear, continuous, surjective but not bijective, and can be locally approximated as a linear function of foot position and velocity. Under these circumstances, an analytical inversion of the contact equation (4.1) is generally impossible.

Adapting External Reactions and Foot Movement

As soon as foot-floor interactions are included in the model, an additional redundancy is introduced. Accelerations and external reactions are now not only coupled by the equations of motion (2.1) and (2.2), but also by the ground contact model. Equation (4.1) relates foot positions and velocities to ground reaction forces. Because of errors in the model and in the experimental measurement, the recorded motion and reaction forces normally do not agree with respect to this equation.

Without a possibility of inverting the contact model equation, there are no direct means of adapting the movement of the feet to the measured forces. The only way of dealing with the measurement redundancies would be by disregarding the experimental reaction forces
and replacing them by those received when evaluating the ground contact model along the experimentally recorded foot trajectories. This is not desirable, as errors in the model and in the kinematic measurements would lead to huge and undesired differences between model predicted and recorded external reactions.

Incorporating both measurements, kinematic and kinetic into one consistent set of data significantly improves the quality of the simulation. The following section introduces a way of adapting both measured quantities - the position of the feet as well as the external reactions. This process optimally integrates both measurements and produces foot trajectories and external reactions that are consistent with respect to the ground contact model. It creates a set of Cartesian accelerations that describe the trajectories the feet have to follow in order to replicate the desired external reactions. This can be seen as generating a pair of “Walking Feet”, as no predictions of the movement of the rest of the body are made. The motion of these feet has to be replicated later (or at least closely approximated) by the overall motion of the model. Chapter 2 introduced possibilities of enforcing such a Cartesian motion by prescribing the segment accelerations (i.e. as geometric constraints), or approximating them by including the position of the segments in the objective function of the numerical search for optimal initial conditions.

**Controlling the Ground Reactions**

When creating such foot trajectories, it is important to consider the dynamic relationship between \( r \) and \( \dot{r} \), as well as between \( \phi \) and \( \omega \). These values can’t be set individually, but have to be integrated numerically from their respective accelerations \( \ddot{r} \) and \( \ddot{\omega} \). Computing the desired accelerations as a function of the difference between measured and model based reaction forces, yields a control loop as shown in Figure 4.1. It has the structure of a linear Luenberg Observer (Luenberger, D.G.: Observing the state of a linear system. IEEE Trans. MIL-8(1964),pp.74-80).

A simple proportional gain is sufficient as controller. There is no need for an integral part, as the system under control has integrating properties itself. The entire controller can be described by one single constant \( k_f \). The units of \( k_f \) are \( m/Ns^2 \) and \( 1/Nms^2 \) for translational and rotational movements respectively. They become more intuitive, when looked at as inverse units of mass (\( 1 \, m/Ns^2 = 1 \, 1/Kg \)) and inertia (\( 1 \, Nms^2 = 1 \, m^2/Kg \)). This provides another way of interpreting the control loop. It can be seen as a dynamic simulation of a rigid body on which the differences between measured and calculated ground reactions are acting.
Controlling the Foot Position

The effects of the nonlinearity of equation (4.1) become crucial as soon as the foot leaves the ground. During the swing phase of gait, the effective stiffness of the ground contact model goes to zero, no forces are acting on the foot, and the feedback path in the force controller vanishes. As a consequence, the accelerations calculated by this controller go to zero and the foot follows a linear trajectory for the remainder of swing phase.

To prevent this from happening, a position controller is superposed as shown in Figure 4.2. It has linear feedback gains for velocity ($k_v$) and position ($k_p$), as well as an acceleration feed forward path. During swing phase (when no forces are acting) solely this controller is active and guides the ‘walking feet’ along their experimental trajectories. During stance phase, the force and position controllers try to achieve competing goals as it is generally not possible to achieve agreement in position and forces at the same time. For a given steady state error in force [N], a certain steady state error in position [m] / [rad] and velocity [m/s] / [rad/s] has to be tolerated, and vice versa. The ratios between these errors should reflect the quality of the underlying measurements and can be expressed as follows:

\[
k_e = \frac{e_f}{e_x} = \frac{k_e}{k_f} \\
b_e = \frac{e_f}{e_v} = \frac{k_e}{k_f}
\]

They have units of stiffness and damping, and can be physically interpreted as spring-
damper elements that connect the experimental trajectories with the simulated ones.

Figure 4.2: Adding position and velocity feedback and an acceleration feed forward path to the force controller in Figure 4.1 keeps the trajectories of the ‘walking feet’ close to their measured kinematics while still creating foot trajectories that reproduce the recorded ground reaction forces.

Controller Parameters

The position controller is tuned so that errors between recorded and calculated positions fall to zero with a time constant of 0.1 s. I.e. position errors vanish after about 20 % of the swing phase. This can be achieved by a position feedback gain $k_p$ of $100 \, 1/s^2$. The velocity feedback gain is set to $k_v = 2\sqrt{k_p} = 20 \, 1/s$ so that the controller shows the transient behavior of a critically damped system. There is no need to distinguish between translational and rotational motion.

As the stiffness and damping properties of the ground contact model strongly depend on the current model configuration (i.e. orientation of the foot, number of units in action,...) it’s not feasible to use established linear controller design methods to determine the force feedback gain $k_f$. Instead, we can estimate the steady state error ratios $k_e$ and $b_e$. A good heuristic is setting them to the ratio of the standard deviations of force and position measurements. Which are about 1 N per 1 cm ($k_e^{\text{trans}} = 100 \, N/m$) for translational movement and 1 N per 0.05 rad ($k_e^{\text{rot}} = 20 \, N/rad$) for rotations. After setting the error ratios $k_e$ and $b_e$, the gain for
the force feedback $k_f$ can be set accordingly:

$$k_f^{\text{trans}} = \frac{1}{\text{kg}}$$
$$k_f^{\text{rot}} = 5 \text{ m}^2/\text{kg}$$

**Integrating the ground contact model in an open-loop simulation**

After a pair of ‘walking feet’ has been created, the overall motion of the model has to be adapted so that the feet of the model actually follow their designated trajectories. This was done by including the desired foot motion as prescribed accelerations in the balance of the overall equations of motion (Section 2.3). To compensate for numerical errors and ensure that desired and computed foot motion doesn’t deviate over the course of the simulation, feedback gains have been put on errors in foot position and velocity and the output of this controller was added to the prescribed accelerations. The gains were set to $400 \frac{1}{\text{s}^2}$ for position errors and $40 \frac{1}{\text{s}}$ for velocity errors and added to the desired accelerations. Figure 4.3 shows an overview of the entire system.

Unfortunately, the geometric constraints caused the motion to deviate significantly from the originally recorded movement and eventually ran into singularities. At heel strike, when the leg is fully extended the prescribed accelerations of the feet carry right through to the pelvis, where they essentially determine the motion of the center of gravity. To compensate for this undesired acceleration and to keep the ground reaction forces at their desired values, the upper body of the model performs large counter movements that eventually reach the bounds of the joint angles. The simulation deviates from the recorded motion and terminates in a model state far off from the original movement (Figure 4.4).
Figure 4.3: Overview of the entire system. The measured foot trajectories and external reactions are altered in a control loop so that they balance the ground contact model equation (top). The results of this adaptation are included as geometric constraints in the balancing of the overall accelerations and the ground reaction forces. The motion created by this process is dynamically consistent and produces model based ground reaction forces that closely replicate the measured ones.
Figure 4.4: Due to a fully extended leg at heel-contact, the prescribed accelerations of the feet carry through to the pelvis. They can only be counterbalanced by large upper body movements (both arms swinging upwards) and extensive pelvis tilt, which cause the simulation to deviate significantly from the recorded motion.
**Integrating the ground contact model in a closed loop simulation**

Without being able to strictly prescribe the foot trajectories, the model-based ground reaction forces will inevitably deviate from their desired values. In an open loop simulation, such errors in the external reactions are unacceptable. They will sum up over the course of the simulation and eventually corrupt the entire motion. It is interesting to note that the ground contact model itself, is dynamically stable in this context. Ground reaction forces that are too big, will push the entire model away from the ground and thus decrease the forces until equilibrium is established, and vice versa. The problem is rather that the overall motion is unstable. Gait coarsely represents the dynamics of an inverted pendulum. This is why any model in an open loop simulation will, at some point, start to tip over and fall. When the motion is recorded accurately and the initial model state has been determined carefully, it is still possible to simulate over a couple of steps before this effect becomes apparent. However, any errors, even small ones, speed up the deviation process substantially. If, for example, the ground contact model is included in the unprocessed data, all similarities between recorded and simulated motion vanish after less than 50% of the gait cycle (Figure 4.5).

Figure 4.5: If the foot trajectories are not prescribed precisely, small errors between expected ground reaction forces and those created by the ground contact model cause the simulated motion to deviate from the recorded movement. Because of the dynamic instability of gait the model eventually tips and falls over.

However, if dynamic feedback is used in the simulation, it is possible to recover from the deviations introduced by the errors in the reaction forces. The Computed Muscle Control (CMC) algorithm, developed by Thelen and Anderson (2006) can be used for this purpose. It adjusts muscle forces according to the differences between desired and currently simulated
generalized coordinates (Figure 1.4) and leads the segments back to their designated trajectories. By this, errors introduced by the ground contact model were rejected and the forward dynamics in the simulation could be run without applying recorded ground reaction forces.

Some pre-processing of the data was still necessary to get foot motions that were at least close to the desired trajectories. This was achieved by putting large weights on the Cartesian tracking error of the feet during the determination of the initial conditions (Section 2.3).

Creating a forward simulation from experimentally collected gait data

A simulation of a half cycle of gait was performed, based on the recorded gait data of a young healthy adult at his preferred walking speed.

After the model was scaled according to anatomical marker locations measured in a static motion trial, inverse kinematics analysis was performed. The trajectories of the feet and the external reaction forces were extracted and used in the optimization of the ground contact model (Chapter 3). Only the vertical and anterior/posterior overall positions of the units were adapted, together with the position of the metatarsal axis. The gradient based SQP method was used for optimization. The base parameter file used in this optimization is identical to the one given in Appendix A. The values obtained by the optimization are stated in Table 4.1. The optimization was performed for the right foot only, the left foot was created by reflecting the parameter values along the sagittal plane. The ground contact models for both feet are shown in Figure 4.6.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>bb_pos_x</td>
<td>3.64 cm</td>
</tr>
<tr>
<td>bb_pos_y</td>
<td>-6.39 cm</td>
</tr>
<tr>
<td>meta_pos_x</td>
<td>0.364 mm</td>
</tr>
</tbody>
</table>

Table 4.1: Numerical optimization was performed to adapt the overall vertical and anterior/posterior position of the viso-elastic elements in the ground contact model as well as the position of the metatarsal axis. A description of these parameters and the values of the parameters not subject to optimization can be found in Appendix A.

The experimentally recorded foot motion was adapted to achieve maximal agreement of recorded and model based external reactions (Section 4.2). The positions of the foot markers that were predicted by this process were included in the acceleration estimation when balancing the equations of motion. Errors in the tracking of these markers were weighted
substantially higher than errors in the tracking of the generalized coordinates. A marker tracking error of 1 mm contributed as much to the objective function as about a 6 degree error in joint angle tracking.

The resulting motion data was dynamically balanced, replicated the recorded motion and had foot trajectories that were as close as possible to the desired ones without actually prescribing them. This motion data was passed to the Computed Muscle Control algorithm, were the muscle excitations for a closed loop, muscle driven forward simulations of a half cycle of gait were computed.

4.3 Results

A muscle driven forward dynamic simulation was created based on the experimental gait data of a young healthy adult. Figure 4.7 shows five frames of the resulting motion.

The use of the ground contact model introduced errors in the tracking of the generalized coordinates. Most notable in the translational coordinates of the pelvis (Table 4.2). While the other coordinates could be adjusted very well by changes in the muscle excitation, there are no muscles available that directly alter the pelvic position. The pelvic translation are essentially created by the ground reaction forces, which explains the relatively big deviations that occur as soon as a ground contact model becomes involved.

Figure 4.8 shows the three components of the ground reaction forces that were created by the ground contact model. Even though there are big errors especially at the beginning of the simulation and at heel strike, the forces tend to converge against the experimentally recorded trajectories. This is surprising, as these forces are not part of the feedback loops in the CMC
The ground contact was used in an closed loop simulation produced by the Computed Muscle Control algorithm (Thelen and Anderson, 2006). Putting feedback gains on joint angles to determine necessary generalized accelerations enables the simulation to run stable for about half a gait cycle.

The ‘correct’ ground reaction forces are hence a product of a ‘correct’ motion combined with the dynamically stable properties of the ground contact model. This provides evidence that the ground contact model is absolutely capable of producing the necessary reaction forces, and that the problem is rather to stabilize the motion and recover from the errors the ground contact model introduced.

Figure 4.8: Recorded external reaction forces (solid blue lines) are compared to the forces created by the ground contact model (dashed red lines). Even though there is no feedback associated with these forces, the model-based reactions tend to converge to the experimentally recorded forces. This suggests that the ground contact model is capable of producing stable and correct ground reaction forces.
Table 4.2: Average differences between measured kinematics (obtained by inverse kinematics analysis on marker data) and kinematics created by the forward simulation.
4.4 Discussion

In this final part of the project, the adaptive ground contact model presented in Chapter 3 was included in an existing simulation framework and tested on experimental gait data. A dynamic controller was developed that enabled the adaptation of the recorded foot motion such that the external reaction force produced by the ground contact model and the experimentally recorded forces agreed as much as possible. The methods of Chapter 2 were used to include this desired foot motion in the estimation of the joint accelerations. Finally, the CMC algorithm was applied to this data and muscle excitation were computed that replicated the recorded motion in a forward dynamic simulation.

The project succeeded in its main goal, the simulative replication of human gait that is not dependent on recorded ground reaction forces, but creates them online with a ground contact model. However, one of the main problems, the instability of gait, has been avoided rather then addressed. This became most evident when the initial approach of using prescribed accelerations to include the ‘walking feet’ in the overall simulation failed, and alternative solutions had to be found.

Open loop forward simulations of gait are limited by the inherent dynamic instability of gait itself. They can be created when outmost care is devoted to consistency, errors, and initial conditions, and as long as the motion is limited to a couple of steps. When it becomes impossible or infeasible to avoid errors, open loop simulations are doomed to failure. Errors that have been introduced are augmented by the instable nature of gait and will eventually corrupt the entire motion.

The attempt to create a consistent and error free simulation by prescribing the accelerations of the feet failed. The alternative way of generating a simulation with a closed loop method, produced acceptable, yet improvable results.

I currently see two ways for further improvement. The first are improvements of the generation of foot trajectories that are used as prescribed accelerations. The crux during the generation of ‘walking feet’ was the heel strike. During swingphase, these feet follow the recorded trajectories, lead by the position controller, which created two problems at the instance of heel contact:

- The initial ground reaction force at heel contact deviated significantly from the desired forces. This is due to the fact, that at this point no corrections have been made by the force controller. The controller needs some time to react and can establish the correct forces only slightly after the first ground contact happened.
• One way of dealing with the first problem, is the amplification of the feedback gain in the force controller. This is the only way to ensure that the model based ground reaction forces converge to their desired values as fast as possible. The practical limitation of increasing this gain, is the fact that a higher gain inevitably creates higher accelerations that have to be replicated by the simulation. Things are getting worse as these accelerations are applied at heel strike, and hence to an almost fully extended leg. While it is relatively easy to create acceleration of the toes by dorsiflexion of the ankle joint, the only way to create heel motion is by moving the entire foot. And, if the leg is already fully extended, the only way of creating such a motion, is by moving the pelvis. Such a pelvic motion has a large impact on the motion of the center of gravity which has to be counterbalanced by the rest of the upper body. In other words: Due to the specific model configuration at heel strike, even small perturbations from the original foot trajectories have a big impact on overall motion.

A way of getting out of this dilemma would be the use of a more foresighted method to generate the ‘walking feet’. It is in the nature of every controller, that it can only reject errors after they already occurred. What is needed instead is a method that avoids such errors in advance. A method that leads the heel to the right spot for the contact instance instead of acting after the contact already happened. One possibility in this context is the adaptation of the foot motion after the controller finished its work. When such alterations are limited to the swing phase, the changes will not affect the created external reactions, but can remove unwanted high accelerations.

The alternative and more fundamental approach of improving the simulation would be to tackle the underlying problem of instability. This can only be done by including a sophisticated controller that estimates accelerations not only based on the experimental measurements, but also on the needs of the ground contact model and the overall movement. If this controller is able to replicate experimentally recorded motion while rejecting disturbances and model flaws, and to stabilize the movement over a larger number of steps, it would also give valuable insight into the tasks and processes of the human nervous system.
Appendix A

Parameter used in the GCM

The ground contact model was described entirely by a small set of parameters. They are listed below.

Number of units (n):

- The number of spring damper units. It is very important that the rest of the parameter definition agrees with this value. I.e. the length of the vector parameters has to be $n$.

Geometrie:

- Overall position and orientation of a bounding box in relation to the local coordinate system ($bb_{pos_x}, bb_{pos_y}, bb_{pos_z}, bb_{rot_3}, bb_{rot_1}, bb_{rot_2}$). The bounding box defines the orientation and the scaling of the entire contact model. The geometrical position of the spring/damper units and the metatarsal axis is given in values in the range of $\pm 1$, describing only their relative position within this box. The orientation of the bounding box is given by three angles describing a 312 rotation in relation to the local coordinate system.

- Scaling of the bounding box ($bb_{scale_x}, bb_{scale_y}, bb_{scale_z}$). These 3 parameters define the length, height and width of the bounding box. *(Note: As values within the bounding box are normalized to the range of $\pm 1$, the overall dimensions of the box are actually twice the values given here.)*

- Displacement of the units ($vec_{disp_x}, vec_{disp_y}, vec_{disp_z}$) in relation to the bounding box. The units should be arranged along the course of the center of pressure.
Generally, it should not be necessary to adapt these values to certain motion trials. However, to better simulate pathologic gait or try different configurations, this provides an easy way to adapt the model. All values are in the range of ±1, referring to the bounding box coordinate system.

Properties of the units:

- Scaling of the vertical stiffness (stiff_y). A global factor to adapt the stiffness of all elements at once.

- Vertical stiffness of the individual units (vec_stiff_y). The stiffness reported here will be multiplied with the scaling factor defined above. Values should therefore be in, but are not limited to the range [0..1].

- Damping translational scaling for all three spatial directions (trans_damp_x, trans_damp_y, trans_damp_z).

- Individual translational damping for all directions (vec_damp_x, vec_damp_y, vec_damp_z). The damping reported here will be multiplied with the scaling factor defined above. Values should therefore be in, but are not limited to the range [0..1].

Rotational damping:

- Three parameters describe the rotational damping around all three spatial axis (rot_damp_x, rot_damp_y, rot_damp_z).

The metatarsal axis:

- Position of the metatarsal axis (meta_pos_x, meta_pos_y). Two parameters define the vertical and anterior-posterior placement of the axis.

- Orientation of the metatarsal axis (meta_rot_2). Angle of rotation within the transverse plane.

- Rotational stiffness and damping coefficients of the metatarsal axis (meta_stiff_3 and meta_damp_3).

Onset of the damping:
• There are three onset functions which describe the relation between damping and compression for the translational movement of the units, the metatarsal axis, and the overall rotation. Each of the functions is defined by one parameter $x_{\text{max}}$ - the compression at which damping becomes constant. The damping coefficient $c$ depends on the compression $x$ as follows:

$$c(x) = \begin{cases} 
0 & x \leq 0 \\
c_{\text{max}} \cdot \left(\frac{x}{x_{\text{max}}}\right)^2 \cdot \left[3 - 2 \left(\frac{x}{x_{\text{max}}}\right)\right] & 0 \leq x \leq x_{\text{max}} \\
c_{\text{max}} & x \geq x_{\text{max}}
\end{cases}$$

$c_{\text{max}}$ is equal to the individual damping coefficient of each unit. For the different damping properties, $x_{\text{max}}$ is described by the parameters (const_damp_trans, const_damp_rot, const_damp_meta).

This parameter file primarily describes the geometric and viscoelastic properties of the contact points. Parameters not listed here are set to their hard coded standard values in the initialization routine. A star after a value denotes that a certain parameter is free. An optimization routine can adapt this parameter to achieve better agreement of experimental and calculated reactions during a given motion trial. Even in this case a value must be provided which serves as an initial guess for the optimization routine.

If a certain parameter starts with the keyword "vec\_", it is assumed that this parameter is provided for each individual spring/damper unit. Instead of a single value, a whole vector of values has to be entered.

Sometimes not all components of the force measurements are equally good or some of them are missing entirely. It is therefore possible to state weights for the individual components. Missing components in the measurement can be ignored by setting the corresponding weight to 0. If weights are not given at all they are set to 1 by default.

The parameter file also contains the specification of the optimization routine. The user can select between a local, gradient based approach and a genetic algorithm. The later also has to be provided with the number of individuals per generation and the number of generations.

```c
/** ************************************************** ************* */
/** This parameter file of a foot contact model contains a list */
/* of all parameters. Most of them are scalar and have only one */
/* value. However, some of them (beginning with "vec\_") are */
/* vectors and should contain one value for each spring/damper */
```
/* unit. The number of units is given in the very special */
/* parameter "n". Free parameters (i.e. parameters that should */
/* be adapted later to match given motion and force data) are */
/* denoted with a "*". In this case the given value serves as */
/* an initial guess for an optimization routine. Additionally a */
/* range of possible values can be given for the optimization */
/* routine. */
/* The contribution of every component of the reactions towards */
/* the cost function of the optimization can be determined by */
/* specifying weights for the different components. */
/***************************************************************************/

beginparameters

/* Form: parameter value [* minVal maxVal (optional)]*/
/* number of units */
n 11

/* scalar parameters */
bb_pos_x +0.03 * 0.00 0.1 /* [m] */
bb_pos_y -0.06 * -0.08 -0.02 /* [m] */
bb_pos_z 0.0 /* [m] */
bb_rot_3 -0.1 /* [rad] */
bb_rot_1 0.0 /* [rad] */
bb_rot_2 0.0 /* [rad] */
bb_scale_x 0.12 /* [m] */
bb_scale_y 0.01 /* [m] */
bb_scale_z 0.03 /* [m] */

stiff_y 40000 /* [N/m ] */
trans_damp_x 400 /* [Ns/m] */
trans_damp_y 200 /* [Ns/m] */
trans_damp_z 400 /* [Ns/m] */
rot_damp_x 20 /* [Nms/rad] */
rot_damp_y 10 /* [Nms/rad] */
rot_damp_z 0.5 /* [Nms/rad] */

meta_pos_x 0.4 * 0.0 1 /* [] */
meta_pos_y 0 /* [] */
meta_rot_2 -0.3 /* [] */
meta_stiff_3 12 /* [Nm/rad] */
/* vector parameters */
/* These parameters only scale the values defined above. They come all without physical unit */
vec_disp_x -1.0000 -0.8000 -0.6000 -0.4000 ...
vec_disp_y 0 0 0 0 ...
vec_disp_z 0.2000 0.2000 0.3000 0.5000 ...
vec_stiff_y 1 1 1 1 ...
vec_damp_x 1 1 1 1 ...
vec_damp_y 1 1 1 1 ...
vec_damp_z 1 1 1 1 ...

endparameters

beginweights
/* All weights not mentioned here are set to 0 */
/* Form: component weight*/
force_x 0 /* Do not account for shear forces */
force_y 1
force_z 0 /* Do not account for shear forces */
torque_x 0 /* Not considering Torques */
torque_y 0
torque_z 0
cop_x 3000 /* we use the cop instead */
cop_z 3000
/* the ratio of force and cop weighting is based on */
/* the following consideration: */
/* - COP changes about 30cm during one step */
/* - Forces change about 1000 N during one step */
/* - one N should be equal to 0.3mm */
/* - one meter has to be weighted 3000 times as high*/
/* as one N */

endweights

beginoptimset
/* declare the optimization method*/
/* can be local or genetic*/
local

/*settings for the optimization routine */
popSize 1000 /* Size of the population. */

numGenerations 300 /* Number of generations evaluated */

endoptimset
Appendix B

Implementation of the GCM

The main interface of a contact model is a C-function representing \[ \begin{bmatrix} F \\ M \end{bmatrix} = f \left( \vec{r}, \phi, \dot{\vec{r}}, \dot{\omega} \right) \].

This function is provided with the actual position and orientation of the contact segment (which defines the model coordinate system) as well as its translational and rotational velocity (all values given in global coordinates). The function returns a force and a torque vector (again, defined in global coordinates). The force acts in the origin of the contact element. The definition is consistent with the SDFAST routines sdpos(), sdvel(), sdorient(), and sdangvel().

/* Evaluates the ground contact model at the given configuration for a certain foot, where 'foot' is either RIGHT or LEFT. All values are 3 dimensional vectors in the global reference frame. Only "orientation" is a 3x3 matrix - the direction cosine as provided by the the SDFAST function sdorient(). The resulting torque and force are stored in the variables force and torque. */
void evalGCM(double position[3], double orientation[3][3], double transVelocity[3], double rotVelocity[3], int foot, double* force, double* torque);

A contact model is initialized by passing it two text files, which list the parameters for the left and right foot respectively. See Appendix ?? for a description of the format of such a parameter file and the individual parameters.

/* LoadGCMParams uses 'filename' to open the associated parameter file, reads the parameters given there and writes them into a set of global variables. 'foot' can be either RIGHT or LEFT and defines in which set of variables the results are stored. */
void loadGCMParams(char* filename, int foot);
To speed up the evaluation of the ground contact model, a couple of pre-calculations are performed in the function ‘initGCM’, which has to be called after ‘loadGCMParameters’, and before ‘evalGCM’.

/* This function does a couple of pre-calculations, to speed up the evaluation of the GCM-function. 'foot' can be either RIGHT or LEFT and defines on which set of variables the function works. */
void initGCM(int foot);

The Matlab routine runGCMOptimization reads in a base parameter file and a data file of recorded spatial trajectories and forces. Parameters marked with a '*' are considered 'free' and will be adapted in an optimization routine until the experimentally measured and the calculated forces agree as much as possible. The routine will write out a parameter file for each foot containing the altered parameters.

The format of the data given to this routine is close to the definition of the contact model’s main function: Trajectories are given in the global coordinate system and forces and moments are defined to act along these trajectories. More than one motion trial can be incorporated into these trajectories. As the data is evaluated at every time step individually, arrays can simply be concatenated without worrying about dynamic consistency. The program "make_foot_traj", can be used to read in a SIMM *.jnt file, and a *.mot file, and to create a data file used by the optimization routine.

The cost function of the optimization routine is simply the sum of the squared differences in between measured (given) and calculated reactions averaged over all data points. Each of the components is multiplied with the corresponding weight, which is also defined in the input parameter file. Special attention should be paid to the weighting difference between forces and moments, which should in some way represent the average moment arm of the units at the contact element.

The resulting output parameter file will be complete. It will even contain values for parameters not specifically given in the input parameter file. Such undefined parameters will be set to the hard coded standard values and adapted if specified by these standards.

runGCMOptimization.mat:

% This script loads a parameter file, and a motion data file as % created by 'createFootTrajectories.exe’. It finds a set of % optimal parameters for each foot, by minimizing a cost function % which describes the difference between the calculated and the % measured ground reaction forces. % Everything necessary for the optimization is specified in the
% input parameter file.

% This script creates the following files in an output directory specified by the user:
% - gcm_parameter_right.txt -> the adapted parameters for the right foot
% - gcm_parameter_left.txt -> the adapted parameters for the left foot
% - input_parameter.txt -> a copy of the input parameter file
% - input_data.txt -> a copy of the input motion data file
% - input_data_settings.mat -> a .mat file, which stores the data selection of the user (frame range, forceplate-foot mapping,...)

The Matlab routine runGCMAdaptation reads in a parameter file for each foot and a data file of recorded spatial trajectories and forces. A control loop adapts these forces and spatial trajectories, to achieve equilibrium with respect to the GCM. The routine will append this information to a SIMM *.mot file. The spatial trajectories are expressed by the position of the foot markers, as they are later used in REA.

runGCMAdaptation.mat:

% This script loads a parameter file for each foot, and a motion data file as created by ’make_foot_traj.exe’.
% A control loop is used to track the measured reaction forces with the reaction forces simulated by the ground contact model.
%
% This script will also load a SIMM *.mot file, in which it will alter the columns for the foot markers (so that they correspond to the calculated foot motion) and the external reaction forces.
%
Bibliography


