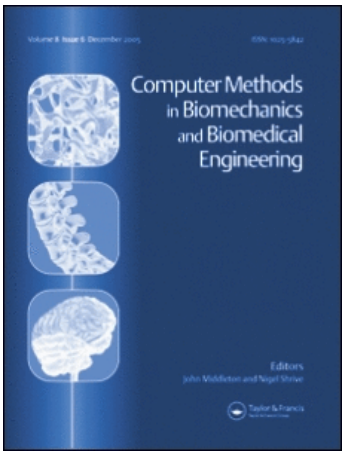

Computer Methods in Biomechanics and Biomedical Engineering
Publication details, including instructions for authors and subscription information:
http://www.informaworld.com/smpp/title~content=t713455284

Kinematic analysis of over-determinate biomechanical systems

M. S. Andersen [a]; M. Damsgaard [b]; J. Rasmussen [a]
[a] Department of Mechanical Engineering, Aalborg University, Aalborg, Denmark [b] AnyBody Technology A/S, Aalborg, Denmark

First Published:August2009

To cite this Article Andersen, M. S., Damsgaard, M. and Rasmussen, J.(2009)'Kinematic analysis of over-determinate biomechanical systems',Computer Methods in Biomechanics and Biomedical Engineering,12:4,371 — 384
To link to this Article: DOI: 10.1080/10255840802459412
URL: http://dx.doi.org/10.1080/10255840802459412

Computer Methods in Biomechanics and Biomedical Engineering
Vol. 12, No. 4, August 2009, 371–384

Taylor & Francis
Taylor & Francis Group

Kinematic analysis of over-determinate biomechanical systems

M.S. Andersen[a]*, M. Damsgaard[b] and J. Rasmussen[a]

[a]*Department of Mechanical Engineering, Aalborg University, Aalborg, Denmark;* [b]*AnyBody Technology A/S, Aalborg, Denmark*

(*Received 9 November 2007; final version received 5 September 2008*)

In this paper, we introduce a new general method for kinematic analysis of rigid multi body systems subject to holonomic constraints. The method extends the standard analysis of kinematically determinate rigid multi body systems to the over-determinate case. This is accomplished by introducing a constrained optimisation problem with the objective function given as a function of the set of system equations that are allowed to be violated while the remaining equations define the feasible set.

We show that exact velocity and acceleration analysis can also be performed by solving linear sets of equations, originating from differentiation of the Karush–Kuhn–Tucker optimality conditions.

The method is applied to the analysis of an 18 degrees-of-freedom gait model where the kinematical drivers are prescribed with data from a motion capture experiment.

The results show that significant differences are obtained between applying standard kinematic analysis or minimising the least-square errors on the two fully equivalent 3D gait models with only the way the experimental data is processed being different.

Keywords: kinematic analysis; over-determinate; motion capture

1. Introduction

Computer simulation models of the human musculoskeletal system hold the potential of revolutionising the design of equipment with a human interface, such as chairs, car seats, workspaces, etc., for diagnosing a wide range of musculoskeletal dysfunctions and to prescribe surgical or rehabilitative treatments in the health care industry.

Biomechanical modelling in general covers a large variety of mechanical modelling disciplines and hereunder numerical methods for dealing with the mathematical models arising. The narrower field of musculoskeletal modelling deals with modelling of the skeletal bones and the connecting tissue, both passive and active, i.e. ligaments and muscles. Typically, such models in the literature today are rigid-body dynamics models (also referred to as multibody dynamics models). Naturally, this implies a number of assumptions and simplifications in order to represent a musculoskeletal system such as the human body as a multibody dynamics model. We shall not go into details about these assumptions and simplifications, but the method presented in this paper is aimed at producing kinematical data for such musculoskeletal multibody models based on experimentally recorded movements.

Two approaches can be taken when performing simulations of the human musculoskeletal system, namely forward dynamics (Anderson and Pandy 1999, 2001) and inverse dynamics (Crowninshield 1978; Crowninshield and Brand 1981; Dul et al. 1984a, 1984b; Rasmussen et al. 2001; Erdemir et al. 2007). In inverse dynamics, the motion of each rigid segment (we use the term 'segment' instead of 'rigid body' to avoid confusion with the 'human body') in the model is prescribed together with the boundary conditions and the internal forces, i.e. the joint reaction and muscle forces, are calculated. This is usually accomplished by treating kinematic and kinetic analysis separately. Kinematic analysis is performed to obtain both the linear and rotational accelerations of each model segment. The calculated accelerations together with the boundary conditions can subsequently be used to form the equations of motion only having unknown muscle and reaction forces (Rasmussen et al. 2001). From this, the required muscle and reaction forces to balance the external loads are calculated by solving the so-called muscle recruitment problem (Crowninshield 1978; Crowninshield and Brand 1981; Dul et al. 1984a, 1984b; Rasmussen et al. 2001; Erdemir et al. 2007).

Despite the rapid development of musculoskeletal simulation techniques, the control of the model movement remains a difficult task. Since the movement is input to an inverse dynamic simulation, accurate motion information is important to obtain good analysis results. Especially, the accelerations are important since they appear directly in the equilibrium equations.

*Corresponding author. Email: msa@ime.aau.dk

ISSN 1025-5842 print/ISSN 1476-8259 online

DOI: 10.1080/10255840802459412
http://www.informaworld.com

The method we present in this paper is particularly aimed at providing the kinematical data needed for inverse dynamics analysis. Indeed, the method we present is for performing kinematical analysis of a multibody model with prescribed motion input, where the motion input can come from a measurement. The prevailing technology for providing such motion input for human motion is motion capture technology, which typically determines trajectories of markers (points) attached to the real system. This is for instance done by tracking small reflective balls by means of multiple synchronised video cameras and reconstructing the motion of the points from the images. We shall use such marker trajectory data as input, but the method can handle more general data formats as well.

The main complications associated with creating the link between measured marker trajectories and the multibody model are as follows:

(1) *Noise*. As with all measurements, marker position recording is susceptible to noise and deviations. The main source of noise in marker-based motion capture is soft tissue artefacts (STA), a phenomenon caused by the fact that markers tend to slide with the skin relative to the bones. Clinical assessment of the nature of STA has been subject to a lot of research in recent years (Cappozzo et al. 1996; Manal et al. 2000; Tayler et al. 2005; Stagni et al. 2005; Benoit et al. 2006).
(2) *Kinematic over-determinacy.* Each motion-captured segment typically has a minimum of three markers attached. This creates nine measured degrees of freedom (DOF), while a segment only has six independent DOF. Joint constraints further increase the over-determinacy.
(3) *Kinematic under-determinacy.* Although some parts of the measured human are kinematically over-determinate, the positions of others may be left under-determinate by the motion capture data. This primarily happens when the model contains 'internal' bones whose motion is not observable from the motion capture markers such as the shoulder blades.
(4) *Missing marker visibility.* A common problem with video-based motion capture is that markers can be occluded for periods of time during the experiment. This primarily happens when a body part or an obstacle is blocking a marker from being recorded by enough video cameras to reconstruct the 3D marker position from the 2D camera images.

The state-of-the-art motion reconstruction methods from markers mounted on the skin (called skin markers) split into two groups: (1) a group of methods that consider each body segment individually (Veldpaus et al. 1988; Söderkvist and Wedin 1993; Cheze et al. 1995; Cappozzo et al. 1997). For this to be possible, these methods require that at least three or more markers are placed on each motion-captured segment; (2) a group of methods that assume an underlying model and use an optimisation algorithm or some sort of Kalman filter to find the motion of the model (Jung and Wohn 1997; Lu and O'Connor 1999; Cerveri et al. 2003, 2005; Zakotnok et al., 2004; Wang et al. 2005; Ausejo et al. 2006).

The inclusion of kinematical models has been shown to reduce STA significantly but also restricts the recovered motion to the assumed DOF of the model (Lu and O'Connor 1999, Cerveri et al. 2005). Except the recently developed methods described by Wang et al. (2005) and Ausejo et al. (2006), which were designed to handle models described using natural coordinates (Garcia de Jalon et al. 1986), the methods in the second group are all based on formulating a model using a minimal set of generalised coordinates. All these methods are designed for position analysis only and do not provide a computational framework to obtain velocities and accelerations except by simple finite differences. Such approximations introduce an unavoidable discrepancy between calculated positions, velocities and accelerations even in the cases where the system equations are analytically known.

The reason why standard kinematical analysis methods for kinematically determinate systems, such as described in Nikravesh (1988) and reviewed in Section 2 for completeness, cannot be applied is that the kinematical constraints together with the measured marker trajectories result in an over-determinate set of equations. When the kinematical constraint equations are over-determinate (due to the number of measured points on the mechanism) and a non-minimal set of coordinates is desired, no general kinematical analysis method for performing position, velocity and acceleration analysis currently exists, and this is what the present work addresses.

The general formulation given in the following enables analysis of any determinate or over-determinate system subject to holonomic constraints formulated in any suitable choice of coordinates, whether it is a minimal or non-minimal set. Notice also that the position analysis given here reduces to those of Lu and O'Connor (1999) and Ausejo et al. (2006) when generalised coordinates and natural coordinates are used together with a least-square objective function, respectively. Holonomic constraints include the typical joint constraints such as spherical joints, revolute joints, etc. but in general they are simply nonlinear functions of the system coordinates and time.

2. Motivation and terminology

The first step in inverse dynamics is always to perform kinematic analysis to find the positions, velocities and accelerations of the time-dependent system coordinates, $q(t) \in Q$ (defined on some appropriate manifold), i.e. given some system description, we wish to find $q(t)$, $\dot{q}(t)$,

and $\ddot{q}(t)$. We shall define the dimension of $q(t)$ as n. In the rest of the paper, we shall only write q and leave it understood that it is a time-dependent variable.

For systems subject to holonomic constraints, position analysis can be formulated as solving a set of m equations (Nikravesh 1988):

$$\Gamma \equiv \Gamma(q, t) = 0. \tag{1}$$

These independent constraint equations are usually composed of constraints describing joints between segments and constraints that describe the motion (also known as kinematical drivers). We shall assume that these equations are sufficiently differentiable. Explicit dependency of time, t, is only the case for kinematical drivers whereas constraints that simply connect two or more segments, such as joint constraints, are only functions of q.

Whereas, kinematical driver functions can have any dependence of q and t, they typically have the following form (Nikravesh 1988):

$$\Pi(q) - \Omega(t) = 0, \tag{2}$$

where $\Pi(q)$ is a function of the system coordinates only and $\Omega(t)$ is a vector function that only depends on time. This type of constraint equation has the property that it will force the $\Pi(q)$ function to attain the value prescribed by the function $\Omega(t)$.

If there are as many constraint equations, m, as unknowns, n, Equation (1) can be solved numerically using, e.g. the Newton–Raphson method (Nikravesh 1988). Additionally, the velocity and acceleration equations can be derived using the chain rule on Equation (1), resulting in linear sets of equations. This linear set of equations for velocity analysis is (Nikravesh 1988):

$$\Gamma_q \dot{q} + \Gamma_t = 0, \tag{3}$$

where the subscript q denotes the partial derivative with respect to q and the subscript t denotes the partial derivative with respect to time.

Differentiation of the velocity equations one more time gives the acceleration equations (Nikravesh 1988):

$$\Gamma_q \ddot{q} + (\Gamma_q \dot{q})_q \dot{q} + 2\Gamma_{qt}\dot{q} + \Gamma_{tt} = 0. \tag{4}$$

As with the velocity analysis, if the positions and velocities are known, these are linear equations in the unknown accelerations. This method for kinematical analysis has been extensively used in many flavours in multi body dynamics, and here we shall refer to it as the standard method for kinematic analysis of kinematically determinate systems.

2.1 Motivation example – slider–crank mechanism

To motivate the idea behind the present work, we shall start with a simple 2D example, where the standard analysis method as described above cannot be applied directly. We shall look at a slider–crank mechanism as illustrated in Figure 1. The mechanism is composed of three rigid segments, connected by three revolute joints and one translational joint. This creates a mechanism with only one DOF. We assume that measurements of the motion of two points in the mechanism are available, i.e. trajectories of the points

$$s_1 = (\, s_{1x} \quad s_{1y} \,)^{\mathrm{T}},$$

and

$$s_2 = (\, s_{1x} \quad s_{1y} \,)^{\mathrm{T}},$$

in global coordinates, which we shall denote

$$m_{s_1} = (\, m_{s_{1x}} \quad m_{s_{1y}} \,)^{\mathrm{T}},$$

and

$$m_{s_2} = (\, m_{s_{2x}} \quad m_{s_{2y}} \,)^{\mathrm{T}}.$$

To model the system, we shall use a non-minimal set of coordinates:

$$q = \begin{pmatrix} q_0 \\ q_1 \\ q_2 \end{pmatrix} = \begin{pmatrix} \theta_1 \\ \theta_2 \\ x \end{pmatrix}. \tag{5}$$

The system equations can then be derived as follows:

$$\Gamma = \begin{pmatrix} l_1 \cos(q_0) + l_2 \cos(q_1) - q_2 \\ l_1 \sin(q_0) - l_2 \sin(q_1) \\ f(q, t) \end{pmatrix} = 0, \tag{6}$$

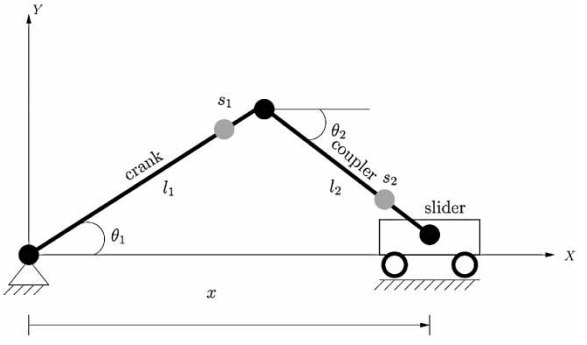

Figure 1. The slider-crank mechanism. θ_1, θ_2 and x specify the system coordinates, and l_1 and l_2 denote the length of the crank and coupler, respectively. s_1 and s_2 denote the local coordinates of two measured points.

where $f(q, t)$ specifies the kinematical driver function. The question is now how this function should be specified. We know that there are four measurements available (two for each point in the model) but only one model DOF. This means that only one driver equation can be specified in order to apply Equations (1), (3) and (4) for position, velocity and acceleration analysis. Additionally, since only one measurement equation can be used, this choice may result in singularities in the formulation. Suppose the following driver function is chosen:

$$f(q,t) = s_{1x}\cos(q_0) - m_{s_{1x}}(t). \tag{7}$$

Such a choice has several undesired implications: (1) three measured coordinates have been omitted, i.e. measurement data are being neglected. (2) Even if this loss of data is accepted, we have no guarantee that the best data were selected, implying that accuracy may have been sacrificed. (3) Worst of all, a singularity has been introduced when θ_1 is close to zero (i.e. the Jacobian matrix of Equation (6) will be close to singular). With the specified driver equation, which drives the mechanism along the x-axis, it is impossible to determine whether the mechanism reached this configuration during a clockwise or counter-clockwise rotation and it is therefore also impossible to determine how the motion will continue from here on. If more than one driver equation is provided, the system is over-determinate, and noise in the measured trajectories of the two points may cause the equations to not have a solution.

3. Kinematic analysis – an optimisation-based approach

In the following, we shall provide a framework for performing position, velocity and acceleration analysis of any over-determinate mechanical system subject to holonomic constraints.

The idea behind the method is to allow all system equations to be specified in Equation (1), which generally may result in an over-determinate system, i.e. $m \geq n$, as illustrated by the slider–crank example, where $m = 6$ and $n = 3$, if all four marker driver equations are included. The solution to the problem is then obtained by re-formulating the equations into a constrained optimisation problem.

To accommodate this over-determinacy, we shall presume that it is possible to split Equation (1) into two sets:

$$\Gamma(q,t) = \begin{pmatrix} \Psi(q,t) \\ \Phi(q,t) \end{pmatrix}, \tag{8}$$

where $\Psi \equiv \Psi(q, t)$ is a set of equations that only has to be solved 'as well as possible' in some sense and the remaining $\Phi \equiv \Phi(q, t)$ equations have to be fulfilled exactly. We shall define the dimension of Ψ and Φ as n_Ψ and n_Φ, respectively. For instance when driving a musculoskeletal model from motion capture data, an obvious choice of these sets would be that the experimental data belongs to Ψ and joint constraints and additional driver equations to Φ. Other choices are of course also valid as long as the solution set of $\Phi(q, t) = 0$ is non-empty. This idea can be cast as the following constrained optimisation problem:

$$\begin{aligned} \min_q \quad & G(\Psi(q,t)) \\ s{\cdot}t \quad & \Phi(q,t) = 0, \end{aligned} \tag{9}$$

where we have introduced the scalar objective function $G \equiv G(\Psi(q, t))$ as a function of the constraint equations that are allowed to be violated. This is necessary since we are faced with a multi objective optimisation problem. This optimisation problem is in general nonlinear and non-convex, which implies that for large-scale systems, such as a full human body model, only local minimisation is feasible. However, with a proper starting guess on q, it is possible to iterate towards the desired solution. Notice that this optimisation problem reduces to the global optimisation with joint constraints method of Lu and O'Connor (1999) when the system is specified using generalised coordinates and the kinematic constraints are given in $\Psi(q, t)$.

The question is now how this objective function could be chosen. One objective function that has previously been used is a weighted least-square with a time-varying weight matrix:

$$G(\Psi(q,t)) = \frac{1}{2}\Psi(q,t)^{\mathrm{T}} W(t) \Psi(q,t), \tag{10}$$

where $W(t)$ is a differentiable, time-dependent weight matrix. Ausejo et al. (2006) used a diagonal time-varying weight matrix while Lu and O'Connor (1999) used a constant diagonal weight matrix. The time-dependency in the weight matrix can be used to vary the weights on the measurements differently along the motion, e.g. when a measurement cannot be trusted within a period of the motion, its weight can be reduced, or if a measurement is invalid for a period of time, it can be assigned weight zero. However, in order to derive equations for velocity and acceleration analysis, we have to require that the objective function is differentiable sufficiently many times, implying that the weight matrix must be designed with care.

For the rest of this section, we shall not presume any specific form of the objective function G when the velocity and acceleration equations are derived.

When the optimisation problem in Equation (9) has been solved, the system coordinates, q, will be known for the discrete time steps where the optimisation problem is solved. However, velocities and accelerations still remain

to be found. Although an approximation of the velocities and accelerations could be found by finite differences, it is indeed possible to derive exact equations for these as we shall show in the following.

From local optimisation theory, it is known that the Karush–Kuhn–Tucker (KKT) conditions are the necessary conditions for optimality (Boyd and Vandenberghe 2004):

$$\begin{aligned} G_q^T + \Phi_q^T \lambda = 0 \\ \Phi = 0, \end{aligned} \tag{11}$$

where λ is a vector of so-called Lagrange multipliers. It is important to remember that this equation will hold for any time step in which the optimisation problem is solved.

As all involved functions are assumed to be differentiable, the velocity equations can be derived by differentiation of Equation (11) with respect to time using the chain rule and re-writing to matrix form:

$$\begin{pmatrix} G_{qq}^T + \left(\Phi_q^T \lambda\right)_q & \Phi_q^T \\ \Phi_q & 0 \end{pmatrix} \begin{pmatrix} \dot{q} \\ \dot{\lambda} \end{pmatrix} = \begin{pmatrix} -G_{qt}^T - \Phi_{qt}^T \lambda \\ -\Phi_t \end{pmatrix}. \tag{12}$$

Similarly, the acceleration equations can be derived by differentiation once again with respect to time:

$$\begin{pmatrix} G_{qq}^T + \left(\Phi_q^T \lambda\right)_q & \Phi_q^T \\ \Phi_q & 0 \end{pmatrix} \begin{pmatrix} \ddot{q} \\ \ddot{\lambda} \end{pmatrix} = \begin{pmatrix} \gamma_1 \\ \gamma_2 \end{pmatrix}, \tag{13}$$

where:

$$\begin{aligned} \gamma_1 &= -2\left(\Phi_q^T \dot{\lambda}\right)_q \dot{q} - 2\Phi_{qt}^T \dot{\lambda} - \left(G_{qq}^T \dot{q}\right)_q \dot{q} - 2G_{qqt}^T \dot{q} - G_{qtt}^T \\ &\quad - 2\left(\Phi_{qt}^T \lambda\right)_q \dot{q} - \Phi_{qtt}^T \lambda - \left(\left(\Phi_q^T \lambda\right)_q \dot{q}\right)_q \dot{q}, \\ \gamma_2 &= -\left(\Phi_q \dot{q}\right)_q \dot{q} - 2\Phi_{qt} \dot{q} - \Phi_{tt}. \end{aligned} \tag{14}$$

Hereby, the required equations are obtained. These equations are valid for any differentiable objective function. One should notice that the main difference in these equations compared to the equations for analysis of determinate systems is that the triple partial derivatives are now required.

In the following, we shall take a closer look at the case with a diagonally weighted least-squares objective function.

4. Elements for a general implementation

A general implementation of the method presented in the previous section requires derivation of the specific terms in the equations for any choice of objective function. Here, we restrict the treatise to a least-square objective function with a constant diagonal weight matrix. As shall be clear, by knowledge about the Ψ and Φ functions and a particular set of their partial derivatives, it is possible construct the terms involving the subject-to constraints and the objective function.

By inspection of Equations (11)–(14), the required functions and partial derivatives can be found. In Table 1, the necessary function information is specified, where we have used the function name Λ to indicate that this information must be available for both the Ψ and Φ functions. Notice in the table that some of the derivatives result in tensor structures. For instance, Λ_{qqq} means a structure containing the triple partial derivatives with respect to q of the vector function Λ.

First, we shall take a closer look at the required terms originating from the subject-to constraints. We shall use the notation $[A]_{ij}$ to denote the ijth element of the matrix A. Additionally, by the notation $\Phi_{k_{q_i}}$, we specify the partial derivative of Φ_k with respect to the ith coordinate, q_i:

$$\begin{aligned} \left[\left(\Phi_q^T \lambda\right)_q\right]_{ij} &= \sum_{k=1}^{n_\Phi} \Phi_{k_{q_i q_j}} \lambda_k \\ \left[\left(\Phi_q^T \dot{\lambda}\right)_q\right]_{ij} &= \sum_{k=1}^{n_\Phi} \Phi_{k_{q_i q_j}} \dot{\lambda}_k \\ \left[\left(\Phi_q \dot{q}\right)_q\right]_{hj} &= \sum_{k=1}^{n_\Phi} \Phi_{h_{q_k q_j}} \dot{q}_k \\ \left[\left(\left(\Phi_q^T \lambda\right)_q \dot{q}\right)_q\right]_{ij} &= \sum_{l=1}^{n} \sum_{k=1}^{n_\Phi} \Phi_{k_{q_i q_l q_j}} \lambda_k \dot{q}_l \\ \left[\left(\Phi_{qt}^T \lambda\right)_q\right]_{ij} &= \sum_{k=1}^{n_\Phi} \Phi_{k_{q_i q_j t}} \lambda_k, \end{aligned} \tag{15}$$

for $i = 1, 2, \ldots, n$, $j = 1, 2, \ldots, n$ and $h = 1, 2, \ldots, n_\Phi$.

From these equations, it should be clear that all the terms only involve sums of products between the different partial derivatives of the constraint equations and either the Lagrange multipliers or the system coordinates (or derivatives hereof). This enables all the expressions to be calculated by a simple loop over all the non-zero elements of the different tensors in Table 1.

Similar expressions can also be derived for the terms originating from the objective function. Here, we shall only show this in the case of a least-square objective function with a constant diagonal weight matrix,

Table 1. List of required functions and derivatives where Λ denote both Ψ and Φ.

Λ	Λ_q	Λ_{qq}
Λ_{qqq}	Λ_t	Λ_{tt}
Λ_{qt}	Λ_{qtt}	Λ_{qqt}

$D = \mathrm{diag}(W_1, W_2, \ldots, W_{n_\Phi})$. Additionally, we shall use the following index notation $D_i = W_i$:

$$
\begin{aligned}
G &= \frac{1}{2}\Psi^{\mathrm{T}} D \Psi, \\
G_q^T &= \Psi_q^T D \Psi, \\
G_{qt}^T &= \Psi_{qt}^T D \Psi + \Psi_q^T D \Psi_t, \\
G_{qtt}^T &= \Psi_{qtt}^T D \Psi + \Psi_q^T D \Psi_t + \Psi_{qt}^T D \Psi_t + \Psi_q^T D \Psi_{tt}, \\
\left[\left(G_{qq}^T \dot{q}\right)_q\right]_{ij} &= \sum_{l=1}^{n} \sum_{k=1}^{n_\Psi} \left(\Psi_{k_{q_i q_l q_j}} D_k \Psi_k + \Psi_{k_{q_i q_l}} D_k \Psi_{k_{q_j}} \right. \\
&\quad \left. + \Psi_{k_{q_i q_j}} D_k \Psi_{k_{q_l}} + \Psi_{k_{q_i}} D_k \Psi_{k q_l q_j} \right) \dot{q}_l, \\
\left[G_{qq}^T\right]_{ij} &= \sum_{k=1}^{n_\Psi} \left(\Psi_{k_{q_i q_j}} D_k \Psi_k + \Psi_{k_{q_i}} D_k \Psi_{k_{q_j}} \right), \\
\left[G_{qqt}^T\right]_{ij} &= \sum_{k=1}^{n_\Psi} \left(\Psi_{k_{q_i q_j t}} D_k \Psi_k + \Psi_{k_{q_i q_j}} D_k \Psi_{k_t} \right. \\
&\quad \left. + \Psi_{k_{q_i t}} D_k \Psi_{k_{q_j}} + \Psi_{k_{q_i}} D_k \Psi_{k_{q_j t}} \right),
\end{aligned} \tag{16}
$$

for $i = 1, 2, \ldots, n$ and $j = 1, 2, \ldots, n$. In this case, all terms involve products between the constraint functions Ψ, various partial derivatives of these, and the diagonal weight matrix. This means that special attention must be paid to how the data is stored such that the sparsity of the various terms can be utilised, while still allowing fast indexing.

Hereby, all the required information is established, and the analysis can be performed. Similarly to the analysis of kinematically determinate systems, for each time step, position analysis has to be performed first, then velocity analysis and finally acceleration analysis using Equations (9), (12) and (13), respectively. The optimisation problem in (9) can be solved by any standard algorithm for nonlinear programming with equality constraints. We apply a standard Newton algorithm (Boyd and Vandenberghe 2004) to find a local minimiser from a given starting guess. For each timestep, we use the previous solution as the starting guess. The linear equations for velocity and acceleration analysis are solved by LU factorising the matrix on the left hand side and forward backward substituting the vector on the right hand side. Notice that it is the same matrix that appears on the left hand side of the velocity and acceleration equations, so it is only necessary to LU factorise it once per time step.

5. Example and results

The core problem associated with driving a kinematical model with measured marker trajectories is that it will lead to an over-determinate system of equations with no solution. Therefore, some compromise has to be made on the equations. However, it is not immediately clear whether the better compromise is to combine a number of the equations in an objective function as done in Equation (9) and effectively end up not satisfying any of these equations, or if it is better, or if it even makes a difference, to exclude just enough equations from consideration, such that all equations can be satisfied as done in Equation (1).

In order to analyse this, both methods have been applied to an 18 DOF lower extremity gait model comprised of seven segments: pelvis, left thigh, left shank, left foot, right thigh, right shank, and right foot. The hip joints are modelled as spherical joints, the knees as revolute joints and the ankles as universal joints as outlined in Table 2.

The kinematical drivers are taken from a motion capture experiment. This is accomplished by treating the differences between the measured marker trajectories and the segment-fixed markers as kinematical driver constraints. We use the 'Man' marker set from Vaughan et al. (1992), where the markers have been placed on bony landmarks in accordance with the Vaughan marker set definition (Vaughan et al. 1992). The placement of the markers and the corresponding labels are shown in Figure 2. The Vaughan marker set also specifies markers on the left and right trochanter major, but these have been excluded in this model. This is because the best combination of kinematical drivers in the standard kinematic analysis approach we could come up with did not include the trochanter major markers and these were therefore also left out in the optimisation-based approach in the interest of a fair comparison. As the objective function, we use a weighted least-square with the matrix equal to the identity matrix for all times (Equation (16)).

For comparison, standard kinematic analysis is also applied to the gait model by using Equations (1), (3) and (4). In order to do this, some of the measured marker coordinates are excluded from consideration as discussed earlier. The included coordinates are listed in Table 3. The two gait models are illustrated in Figure 3.

5.1 Gait model using a full set of Cartesian coordinates

Both models are formulated using a full Cartesian formulation: the coordinate vector q is composed of the translational and rotational coordinates for each segment,

$$q = (\, q_1 \quad q_2 \quad \cdots \quad q_7 \,)^{\mathrm{T}},$$

Table 2. Definition of joints.

Joint	Type
Left hip	Spherical joint
Left knee	Revolute joint
Left ankle	Universal joint
Right hip	Spherical joint
Right knee	Revolute joint
Right ankle	Universal joint

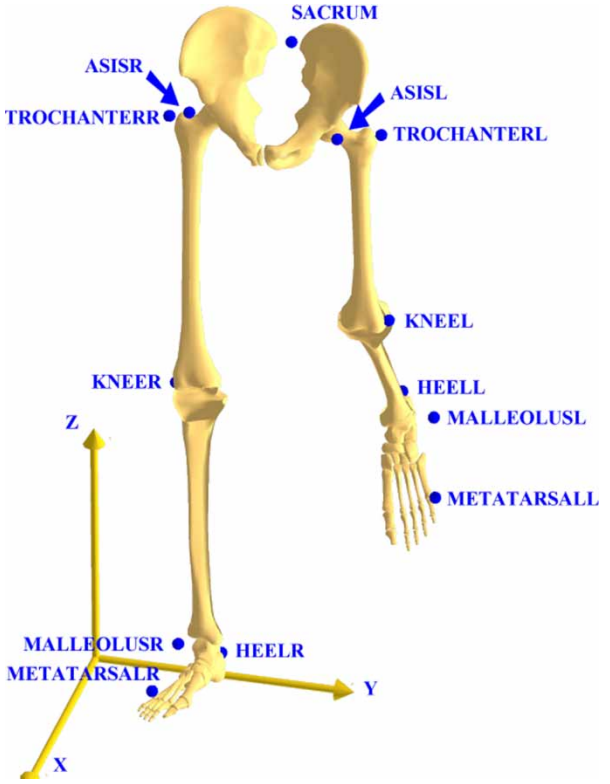

Figure 2. Illustration of the marker placement and their labels. Notice that the postfix 'L' and 'R' means left and right, respectively.

where

$$q_i = (\, r_i \quad p_i \,)^{\mathrm{T}},$$

with

$$r_i = (\, x_i \quad y_i \quad z_i \,)^{\mathrm{T}},$$

Table 3. Definition of kinematical drivers for the gait model using standard kinematic analysis.

Marker	Driver
SACRUM	y
ASISL	x
ASISR	x
KNEEL	x
KNEER	x
MALLEOLUSL	y
MALLEOLUSR	y
METATARSALL	x and z
METATARSALR	x and z
HEELL	x, y and z
HEELR	x, y and z
ASISL, ASISR and SACRUM	z coordinates combined in a sum

The directions refer to which coordinates in the calculation of the differences in the segment-fixed markers and the measured marker trajectories, calculated in segment-fixed coordinate system, that are used as kinematical drivers (see Equation (22)).

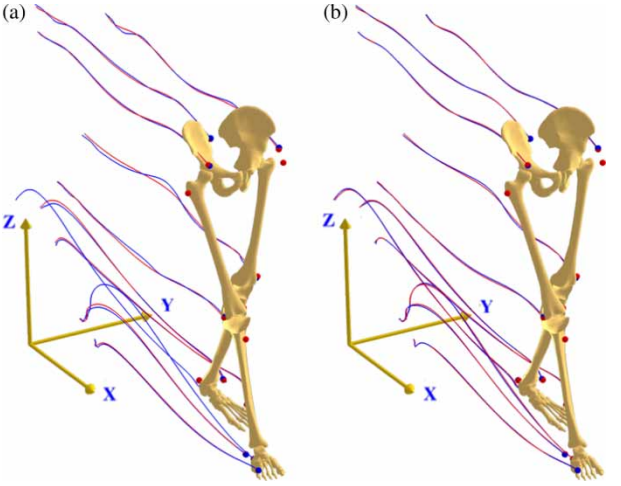

Figure 3. Illustration of the two gait models. Blue trajectories show the motion of the points in the model and the red trajectories are measured by motion capture. (a) Shows the result using standard kinematic analysis, where some of the measured marker coordinates are neglected. (b) Shows the result with the optimisation-based approach.

being the global position and

$$p_i = (\, e_{0_i} \quad e_{1_i} \quad e_{2_i} \quad e_{3_i} \,)^{\mathrm{T}},$$

the Euler parameters for the segments. Additionally, we shall use the notation

$$A(p_i) = (\, a_{1_i} \quad a_{2_i} \quad a_{3_i} \,),$$

to denote the rotation matrix associated with the Euler parameters p_i, where a_{1_i}, a_{2_i} and a_{3_i} denote the first, second and third column of the matrix, respectively. The joint constraints are created in the standard manner as outlined below.

Let s'_i and s'_j denote the local coordinates of the joint in the ith and jth segments' coordinate systems, respectively. The constraints used as subject-to constraints are as follows. These are also used in the gait model solved using standard kinematic analysis.

The spherical joint constraints can be written as:

$$r_i + A(p_i)s'_i - (r_j + A(p_j)s'_j) = 0. \tag{17}$$

For the revolute joints, when the rotation axis is around the z-axis in both coordinate systems, the equations can be written as:

$$\begin{aligned} r_i + A(p_i)s'_i - (r_j + A(p_j)s'_j) &= 0, \\ a_{3_i}^T a_{1_j} &= 0, \\ a_{3_i}^T a_{2_j} &= 0. \end{aligned} \tag{18}$$

For the universal joints, where the rotation axes are the z-axis and y-axis in the jth coordinate system, the equations

Table 4. Definition of local joint coordinates.

Segment	Name	Coordinates (*m*)
Pelvis	Left hip	(0.0328, −0.1094, −0.0875)
Pelvis	Right hip	(0.0328, −0.1094, 0.0875)
Left thigh	Left hip	(0, 0.1909, 0.03682)
Left thigh	Left knee	(−0.0242, −0.2500, 0.02713)
Left shank	Left knee	(0, 0.1918, 0)
Left shank	Left ankle	(0, −0.2511, 0)
Left foot	Left ankle	(0.0648, 0.0648, 0)
Right thigh	Right hip	(0, 0.1909, −0.03682)
Right thigh	Right knee	(−0.0242, −0.2500, −0.02713)
Right shank	Right knee	(0, 0.1918, 0)
Right shank	Right ankle	(0, −0.2511, 0)
Right foot	Right ankle	(0.0648, 0.0648, 0)

can be written as:

$$\begin{aligned} r_i + A(p_i)s_i' - (r_j + A(p_j)s_j') = 0, \\ a_{3_i}^T a_{2_j} = 0. \end{aligned} \tag{19}$$

The constraint on the Euler parameters:

$$p_i^T p_i - 1 = 0. \tag{20}$$

The constraint equations, Ψ, used in the objective function are all based on the difference between the segment-fixed marker positions in the model and the measured marker trajectories in global coordinates. Let $s'_{i_{m_j}}$ denote the local coordinates of the *j*th marker on the *i*th segment and let $z_{i_{m_j}}$ denote the corresponding measurement of the marker:

$$r_i + A(p_i)s'_{i_{m_j}} - z_{i_{m_j}} = 0. \tag{21}$$

In the model using standard kinematic analysis, the errors between segment-fixed markers and the measured marker trajectories are calculated in local coordinates:

$$A(p_i)^{\mathrm{T}}(r_i - z_{i_{m_j}}) + s'_{i_{m_j}} = 0. \tag{22}$$

Since not all the measured differences can be fulfilled in standard kinematic analysis, only the directions specified in Table 3 are included. It should be noted that this choice of driver directions is not a random choice, but indeed an attempt to do as well as possible with the limited number of driver equations. It has taken many manual trials and errors to find a solution as good as the one in Figure 3(b). Indeed, the differences between the measured points and the points in model could have been calculated in global coordinates, but local coordinates, as in Equation (22), made it much easier to select a constant usable driver set.

The specific definition of the local joint and local marker coordinates are given in Table 4 and Table 5. As we saw earlier, the derivative of the marker trajectories with respect to time is required to perform velocity and acceleration analysis. These are obtained by interpolating the sampled points with a 4th order B-spline and the derivatives found as the first and second derivative of the B-spline evaluated in the sampled points.

The computed trajectories of the segment-fixed markers and the measured marker trajectories are shown together with the gait models in Figure 3. Additionally, a close-up of the trajectories of the markers on pelvis can be seen in Figure 4. As is already visually clear from these pictures, the two analysis results are not the same, and particularly the results for the pelvis segment appear to be substantially different. Moreover, it appears that the optimisation-based approach is following the measured trajectories closer than the standard kinematic analysis approach.

A comparison between the calculated linear and angular accelerations for all the segments for the two methods is shown in Figure 5. The graphs are calculated by taking the magnitude (i.e. two-norm) of the vector differences between the results for the two methods. As seen in this figure, particularly around the start of the motion and around 0.9 s, the two methods produce significantly different results with a maximum difference of 7.8 $\mathrm{m\,s^{-2}}$ (left thigh) and 60.1 $\mathrm{rad\,s^{-2}}$ (the left foot). An interesting observation is that the differences in the estimations are largest during the beginning of both swing phases of the gait cycle where the accelerations are largest.

Table 5. Definition of local marker coordinates.

Segment	Name	Coordinates (*m*)
Pelvis	SACRUM	(−0.045, 0.0, 0.0)
Pelvis	ASISR	(0.1245, −0.0991, 0.1307)
Pelvis	ASISL	(0.1430, −0.1055, −0.1144)
Left thigh	KNEEL	(−0.0110, −0.2499, −0.0140)
Left shank	MALLEOLUSL	(−0.0036, −0.2578, −0.0556)
Left foot	METATARSALL	(0.0126, −0.0502, −0.0580)
Left foot	HEELL	(0.0248, 0.1292, 0.0)
Right thigh	KNEER	(−0.0189, −0.2591, 0.0166)
Right shank	MALLEOLUSR	(0.0020, −0.2615, 0.0462)
Right foot	METATARSALR	(0.01265, −0.0502, 0.0527)
Right foot	HEELR	(0.0248, 0.1292, 0.0)

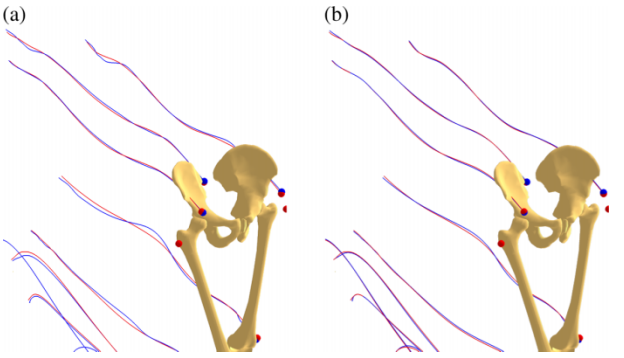

Figure 4. Close-up of the gait models' trajectories. In both models, the blue trajectories illustrate the motion of the points in the model and the red trajectories the measured trajectories. (a) Shows the result using standard kinematic analysis where some of the measured marker trajectories are neglected. (b) Shows a close-up on the results with the optimisation-based approach.

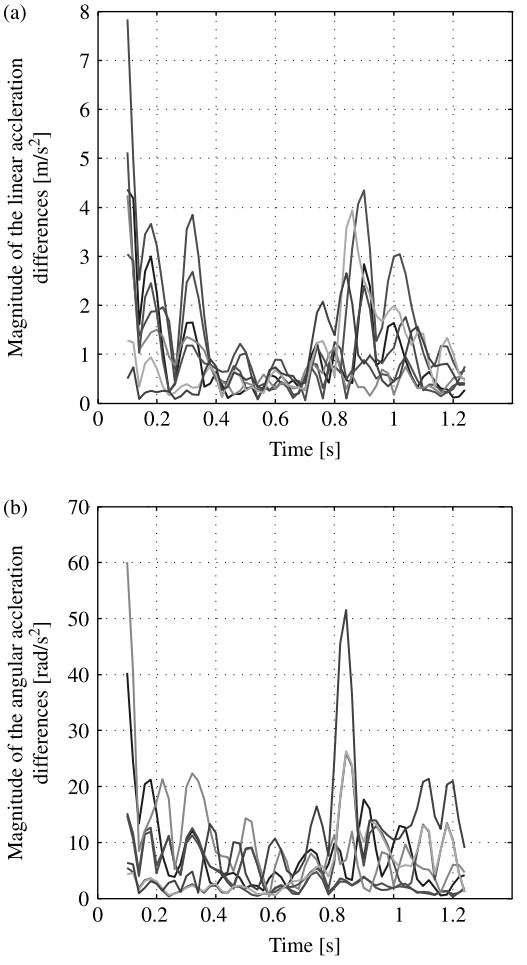

Figure 5. Magnitude of the differences between the estimated linear and angular accelerations of the segments using the optimisation-based approach and standard kinematic analysis. The figure to the left shows the result for the linear accelerations and the figure to the right, for the angular accelerations. Each curve in the figure represents the results for a single segment.

In order to investigate these differences, we focus on the results for the pelvis segment and a direct comparison between the two analysis results can be seen in Figures 6–8, which show the results for the translational coordinate, the linear and angular velocities, and the linear and angular accelerations, respectively. In these figures, particularly the large differences in the calculated velocities and accelerations should be noticed. As seen in the figures, differentiation of the data amplifies the differences between the two analysis results. Considering how large differences there are in the calculated velocities and accelerations, one might ask which of the two results are best to use as input to inverse dynamics. One observation is that the calculated velocities and accelerations are indeed smaller using the optimisation-based approach, but which of the two results are more correct cannot be answered from these calculations.

To give a more concrete measure of how well the methods are at not only estimating the measured trajectories but also the derivatives of these, the magnitude (i.e. two-norm) of the differences between the positions, velocities and accelerations of all the measured trajectories and the corresponding points in the model have been calculated. The results using the optimisation-based approach can be seen in Figure 9 and the results using standard kinematic analysis in Figure 10. As seen in the figures, the optimisation-based approach appears to estimate the accelerations of the marker trajectories best with a peak error of $2.8\,\mathrm{m\,s^{-2}}$, while the standard kinematic analysis method has a peak error of $8.7\,\mathrm{m\,s^{-2}}$. The root-mean-square of the estimated acceleration errors for all marker coordinates and all samples are $0.36\,\mathrm{m\,s^{-2}}$ for the optimisation-based approach and $0.86\,\mathrm{m\,s^{-2}}$ for the standard kinematic analysis approach.

6. Discussion and conclusion

In this paper, we presented a general and natural extension of the standard kinematic analysis for determinate systems to the over-determinate case. This was accomplished by re-writing the over-determinate kinematical equations into a constrained optimisation problem. The given formulation was made such that theoretically any desired choice of system coordinates and differentiable objective function can be used.

By applying the chain rule to the KKT optimality conditions, we showed how it is possible to also perform exact velocity and acceleration analysis by solving linear sets of equations.

We then presented how the various required terms appearing in the optimisation problem, velocity, and acceleration analysis can be calculated from knowledge of the constraint equations and their partial derivatives. This we only showed in the case of a least-square objective function with constant diagonal weight matrix. However,

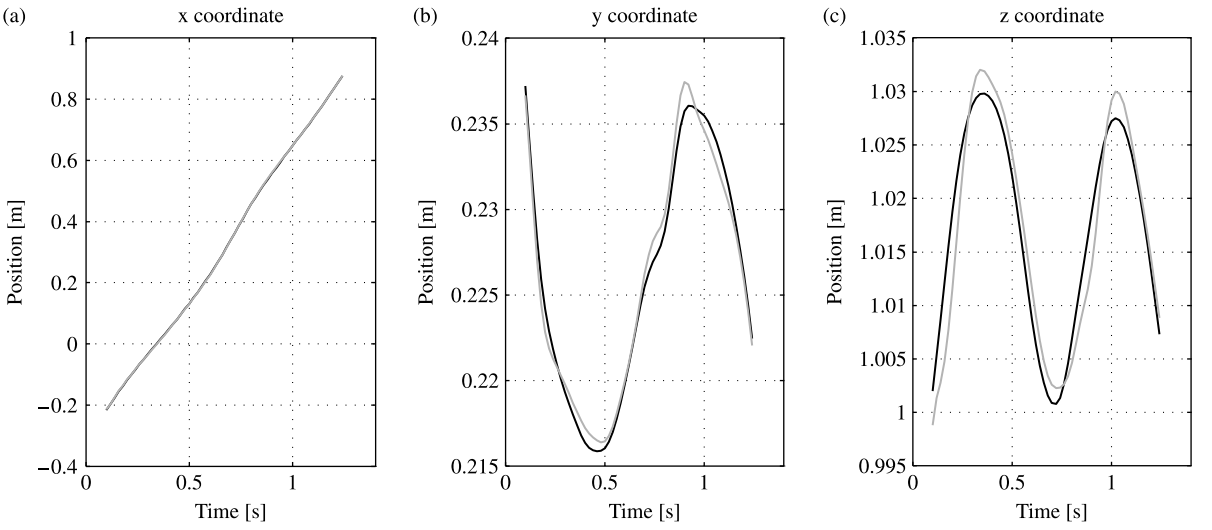

Figure 6. Position analysis results for the pelvis segment. The black lines show the results from solution of Equation (9) with a diagonally weighted least-square objective function. The grey lines show the results when standard kinematic analysis is performed.

the derivation of the terms for another objective function is not difficult.

As briefly mentioned earlier, the optimisation problem given in Equation (9) will generally be non-linear and non-convex due to the kinematical equations that it is composed of. This means that there may exist multiple local minima in the problem. In case local minimisation is performed, a good starting guess is therefore required in order to find the desired solution. The other option is global minimisation, which unfortunately is very computationally demanding. With that said, it should be noted that no problems with local minima were encountered with the gait model as presented here and local minimisation appears to be adequately robust in practice.

In order to perform velocity and acceleration analysis, a number of partial derivatives are required for both the sets of equations used in the objective function, the objective function itself, and the subject-to constraints. These derivatives are listed in Table 1. When all kinematical equations are analytically known, Equations (9), (12) and (13) will ensure that the calculated best-fit positions, velocities and accelerations are consistent. However, obtaining these partial derivatives is not a trivial matter. In the example shown in this paper, these derivatives were obtained using a homemade automatic differentiation package. For details about automatic differentiation see Bischof et al. (1995).

Although it is possible to hard-code models using the equations directly, this is a difficult task. Instead, a general-purpose modelling system should be designed and implemented such that users do not need to worry about getting all the functions and derivatives right. Precisely for this reason, the examples used in this paper were derived using a full Cartesian formulation. The key reason for using this formulation is the generality and ease of use in general-purpose software, especially when closed-loop mechanical systems are considered. Although it is theoretically always possible to find a set of generalised coordinates, in practice it might be difficult to find a set of generalised coordinates that is valid over the whole time-period of interest for a complicated close-loop mechanical system. Therefore, it might be necessary to change coordinates during the simulation. This makes generalised coordinates difficult to use in general-purpose software. The use of a full Cartesian formulation eliminates this concern, as the coordinates are always valid. The drawback of a full Cartesian formulation is that there will be more coordinates and equations than strictly needed.

Another issue is whether it is better to approximate the velocities and accelerations of the marker trajectories and use these derivatives in the velocity and acceleration analysis in Equations (12) and (13), as opposed to only solving the position analysis optimisation problem in Equation (9) and subsequently approximate the velocities and accelerations of the system coordinates instead. When the model is formulated using generalised coordinates and the optimisation problem is consequently unconstrained, the differences in the calculated velocities and accelerations are probably going to be minor. On the other hand, when a non-minimal set of coordinates is used, it is a completely different story. In this case, it is important that the subject-to constraints, Φ, are satisfied not only on position level, but also on velocity and acceleration level. If the optimisation problem is only solved on position level, and the velocities and accelerations are only

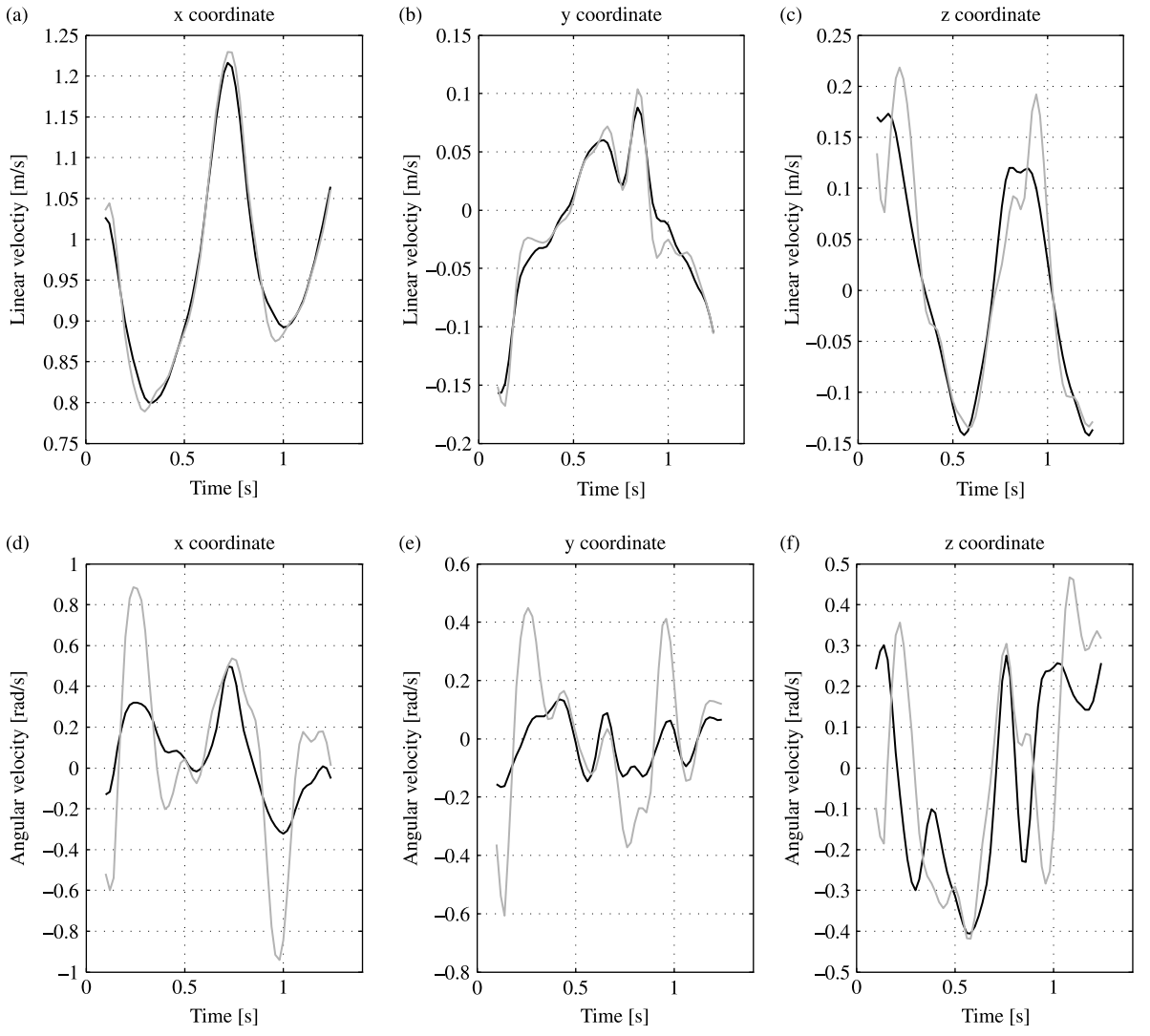

Figure 7. Velocity analysis results for the pelvis segment. The black lines show the results from solution of Equation (9) with a diagonally weighted least-square objective function. The grey lines show the results when standard kinematic analysis is performed.

approximated, the found velocities and accelerations of the system coordinates will end up not satisfying the subject-to constraints on velocity and acceleration level. This poses a serious issue when kinetic analysis is performed since it is assumed in the equations of motion that these equations are satisfied for positions, velocities, and accelerations. If marker-based motion capture systems also provided accurate measurements of velocities and accelerations of the marker trajectories, using e.g. accelerometers, these measurements could be used directly in the velocity and acceleration equations and would improve the accuracy of the estimates.

A number of existing methods described in the literature are special cases of the optimisation problem given in Equation (9) and therefore Equations (12) and (13) extend these methods with the capability of performing velocity and acceleration analysis. It was already mentioned that the methods described by Lu and O'Connor (1999) and Ausejo et al. (2006) are special cases. However, a number of methods that reconstruct the motion of each model segment without considering joint constraints can also be considered special cases, e.g. Veldpaus et al. (1988), Söderkvist and Wedin (1993) and Cheze et al. (1995).

Although the method we presented in this paper was specially designed to obtain accurate and consistent positions, velocities and accelerations required for inverse dynamic analysis, forward dynamic simulations that rely on tracking both the measured kinematical data as well measured reaction forces can also benefit from this method such as Thelen and Andersen (2006), Fregly et al. (2007) and Seth and Pandy (2007). These methods rely on having

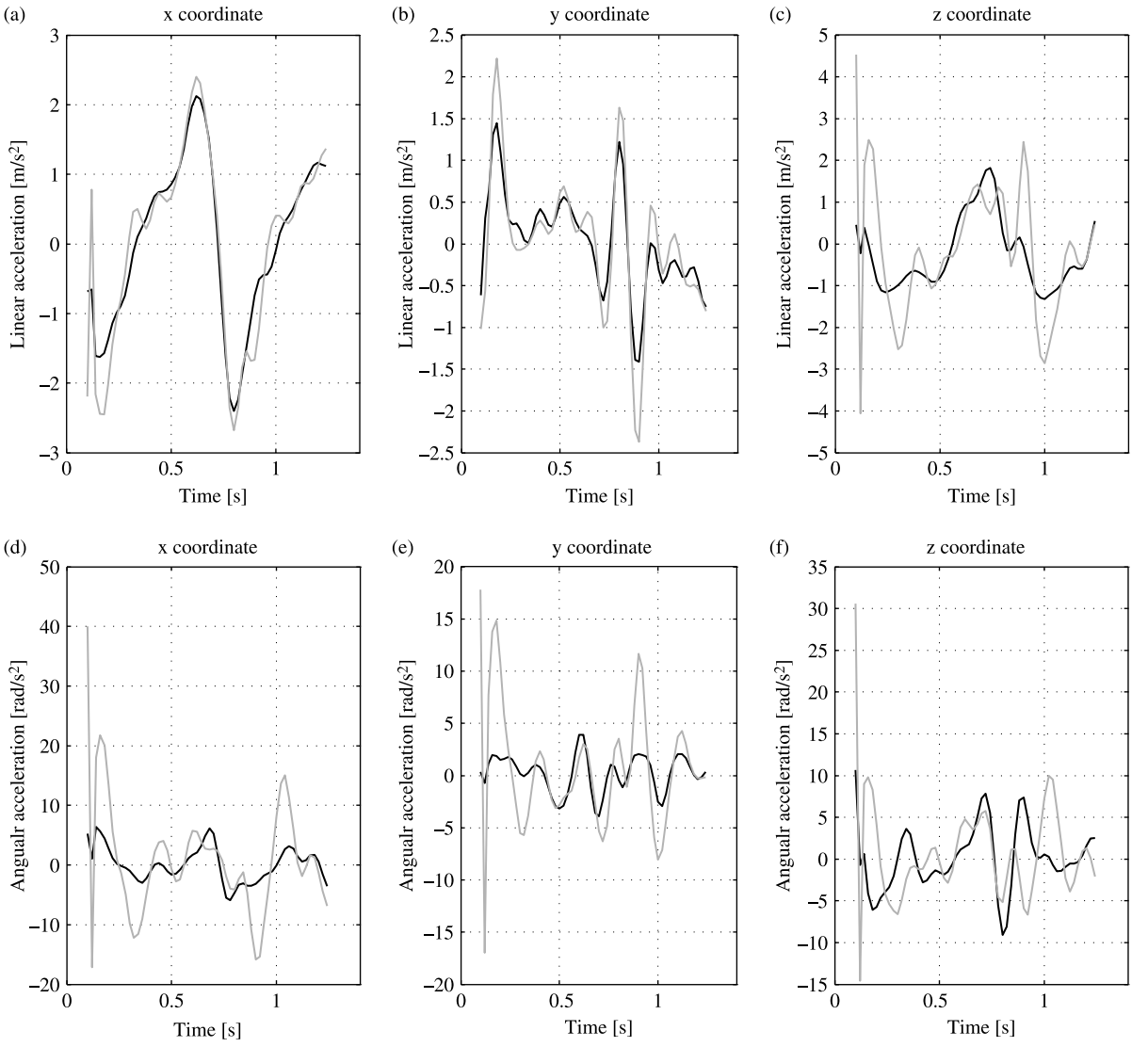

Figure 8. Acceleration analysis results for the pelvis segment. The black lines show the results from solution of Equation (9) with a diagonally weighted least-square objective function. The grey lines show the results when standard kinematic analysis is performed.

computed positions, velocities and accelerations of the system coordinates available for the forward dynamics tracking algorithm and these can be obtained using Equations (9), (12) and (13).

As we saw in the 3D gait example, significant differences were obtained between excluding some of the measured marker trajectories and applying standard kinematic analysis compared to including them all and minimising the least-square errors. This indicates that the assumption of a rigid-body motion is not completely fulfilled (e.g. due to STA), or that the chosen model does not capture all the motion patterns of the bones, completely. However, since the method we presented allows the system equations to be described by any sufficiently differentiable nonlinear functions of the coordinates and time, more complicated models can be handled. For instance for gait analysis, this could be more complicated knee or ankle models, or even models of STA. Another cause of the differences is the precise placement of the markers on the model and the local joint coordinates compared to the test subject. For the example shown here, these were approximated to the best of our ability by solving another optimisation problem, which is a generalisation of the optimisation problem in Equation (9) extended to allow one to find constant model parameters optimally over the whole sampling period in addition to the time-dependent system coordinates. The method, recently developed by the authors, is presented in Andersen et al. (2007) together with some preliminary results. A similar idea can also be found in (Reinbolt et al. 2005), which uses a two-level optimisation

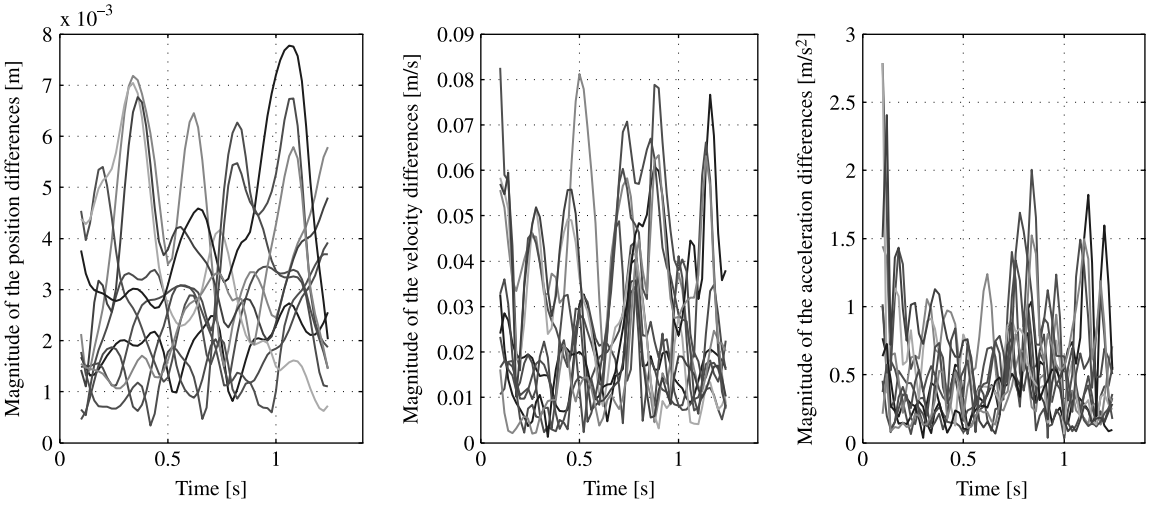

Figure 9. Magnitude of the differences between the marker positions, velocities, and accelerations from the measurement and estimated derivatives by b-splines, and the segment-fixed markers in the gait model solved using the optimisation-based approach. Each curve represents the result for a single marker.

strategy in order to partition the large-scale optimisation problem, but is limited to unconstrained optimisation.

Even though we demonstrated that the optimisation-based approach does provide a better solution than a standard kinematic approach, it remains unknown whether the results are good enough to be used as input to inverse dynamics to calculate joint reaction forces and muscle forces. As the results showed, the optimisation-based approach produced a peak acceleration error of 2.8 ms^{-2} compared to the estimated accelerations of the marker trajectories (i.e. almost 1/3 g). In order to answer this question, measurements of the true bone motions are required. This can be accomplished in an invasive experiment involving bone-mounted markers or by advanced radiographic methods, both of which require ethical approval.

Since the main source of measurement error in marker-based motion capture is STA, and it has been shown that STA is highly correlated with the motion performed (Benoit et al. 2006; Stagni et al. 2005; Tayler et al. 2005), the choice of a least-square objective function is not justifiable because it assumes that the measurement errors are randomly distributed around the true solution. Derivation of methods to take STA into account

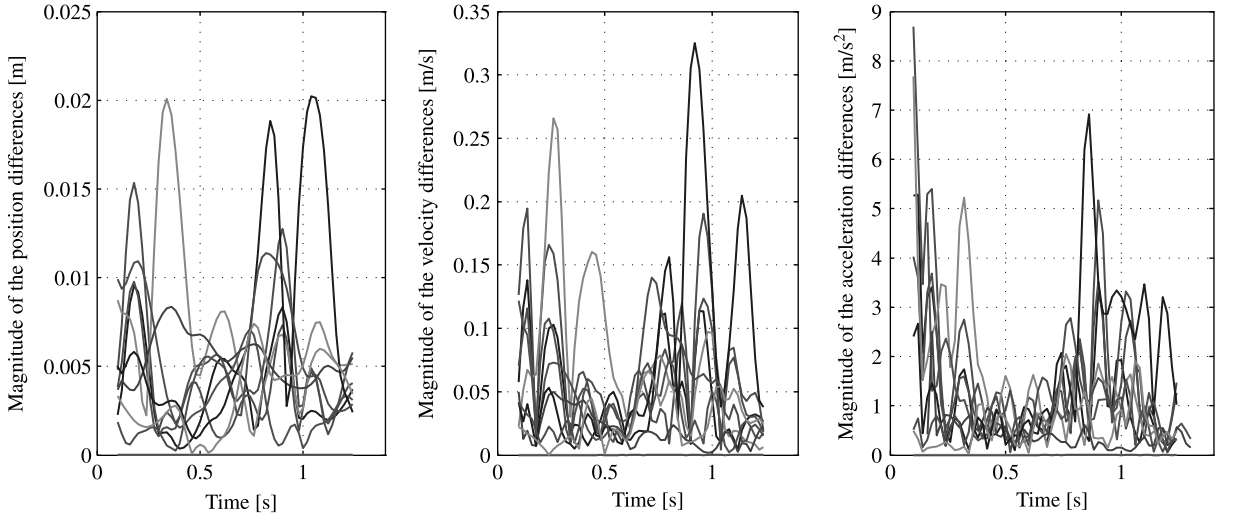

Figure 10. Magnitude of the differences between the marker positions, velocities, and accelerations from the measurement and estimated derivatives by b-splines, and the segment-fixed markers in the gait model solved using standard kinematic analysis. Each curve represents the result for a single marker.

in computer simulation models is an important future research task. Notice that the presented method allows this.

References

Andersen MS, Damsgaard M, Rasmussen J. 2007. Scaling and local marker coordinate determination for musculoskeletal systems. Proceedings of Digital Human Modeling for Design and Engineering Conference. Seattle, USA.

Anderson FC, Pandy MG. 1999. A dynamic optimization solution for vertical jumping in three dimensions. Comput Methods Biomech Biomed Engin. 2(3):1–30.

Anderson FC, Pandy MG. 2001. Dynamic optimization of human walking. J Biomech Eng. 123(5):381–390.

Ausejo S, Suescun A, Celigueta J, Wang X. 2006. Robust human motion reconstruction in the presence of missing markers and the absence of markers for some body segments. Proceedings of Digital Human Modeling for Design and Engineering Conference. Lyon, France.

Benoit DL, Ramsay DK, Lamontage M, Xu L, Wretenberg P, Renström P. 2006. Effect of skin movement artifact on knee kinematics during gait and cutting motions measured *in vivo*. Gait Posture. 24(2):152–164.

Bischof C, Carle A, Khademi P, Pusch G. 1995. Automatic differentiation: obtaining fast and reliable derivatives fast. In: Proceedings of Control Problems in Industry. Boston, USA.

Boyd S, Vandenberghe L. 2004. Convex Optimization. Cambridge, UK: Cambridge University Press.

Cappozzo A, Catani F, Leardini A, Benedetti MG, Croce UD. 1996. Position and orientation in space of bones during movement: experimental artifacts. Clin Biomech. 11(2): 90–100.

Cappozzo A, Cappello A, Croce UD, Pensalfini F. 1997. Surface-marker cluster design criteria for 3D bone movement reconstruction. IEEE Trans Biomed Eng. 44(12): 1165–1174.

Cerveri P, Pedotti A, Ferrigno G. 2003. Robust recovery of human movement from video using kalman filters and virtual humans. Hum Mov Sci. 22:377–404.

Cerveri P, Pedotti A, Ferrigno G. 2005. Kinematic models to reduce the effects of skin artifacts on marker-based human motion estimation. J Biomech. 38:2228–2236.

Cheze L, Fregly BJ, Dimnet J. 1995. A solidification procedure to facilitate kinematic analysis based on video system data. J Biomech. 28(7):879–884.

Crowninshield RD. 1978. Use of optimization techniques to predict muscle forces. J Biomech Eng. 100:88–92.

Crowninshield RD, Brand RA. 1981. A physiologically based criterion of muscle force prediction in locomotion. J Biomech. 14:793–801.

Dul J, Johnson GE, Shiavi R, Townsend MA. 1984a. Muscular Synergism – I. On criteria for load sharing between synergistic muscles. J Biomech. 17:663–673.

Dul J, Johnson GE, Shiavi R, Townsend MA. 1984b. Muscular Synergism – II. A minimum fatigue criterion for load sharing between synergistic muscles. J Biomech. 17: 675–684.

Erdemir A, McLean S, Herzog W, van den Bogert AJ. 2007. Model-based estimation of muscle forces exerted during movements. Clin Biomech. 22(2):131–154.

Fregly BJ, Reinbolt JA, Rooney KL, Mitchell KH, Chmielewski TL. 2007. Design of patient-specific gait modifications for knee osteoarthritis rehabilitation. IEEE Trans Biomed Eng. 54(9): 1687–1695.

Garcia de Jalon J, Unda J, Avello A. 1986. Natural coordinates for the computer analysis of multibody systems. Comput Methods Appl Mech Eng. 56:309–327.

Jung S, Wohn K. 1997. Tracking and motion estimation of the articulated object: a hierarchical kalman filter approach. Real-Time Imaging. 3:415–432.

Lu TW, O'Connor JJ. 1999. Bone position estimation from skin marker co-ordinates using global optimization with joint constraints. J Biomech. 32:129–134.

Manal K, McClay I, Stanhope S, Richards J, Galinat B. 2000. Comparison of surface mounted markers and attachment methods in estimating tibial rotations during walking: an *in vivo* study. Gait Posture. 11:38–45.

Nikravesh PE. 1988. Computer-aided analysis of mechanical systems. Upper Saddle River, NJ: Prentice-Hall International, Inc.

Rasmussen J, Damsgaard M, Voigt M. 2001. Muscle recruitment by the min/max criterion – a comparative numerical study. J Biomech. 34(3):409–415.

Reinbolt JA, Schutte JF, Fregly BJ, Koh BIL, Haftka RT, George AD, Mitchell KH. 2005. Determination of patient-specific multi-joint kinematic models though two-level optimization. J Biomech. 38:621–626.

Seth A, Pandy MG. 2007. A neuromusculoskeletal tracking method for estimating individual muscle forces in human movement. J Biomech. 40(2):356–366.

Stagni R, Fantozzi S, Cappello A, Leardini A. 2005. Quantification of soft tissue artefact in motion analysis by combining 3D fluoroscopy and stereophotogrammetry: a study of two subjects. Clin Biomech. 20:320–329.

Söderkvist I, Wedin P. 1993. Determining the movement of the skeleton using well-configured markers. J Biomech. 26: 1473–1477.

Taylor WR, Ehrig RM, Duda GN, Schell H, Seebeck P, Heller MO. 2005. On the influence of soft tissue coverage in the determination of bone kinematics using skin markers. J Orthop Res. 23:726–734.

Thelen DG, Andersen FC. 2006. Using computed muscle control to generate forward dynamic simulations of human walking from experimental data. J Biomech. 39(6):1107–1115.

Vaughan CL, Davis BL, O'Connor JC. 1992. Dynamics of human gait. 2nd ed.. Cape Town, South Africa: Kiboho Publishers.

Veldpaus FE, Woltring HJ, Dortmans LJMG. 1988. A least-square algorithm for the equiform transformation from spatial marker co-ordinates. J Biomech. 21(1):45–54.

Wang X, Chevalot N, Monnier G, Ausejo S, Suescun A, Celigueta J. 2005. Validation of a model-based motion reconstruction method developed in the REALMAN project. In: Proceedings of Digital Human Modeling for Design and Engineering Conference. Iowa City, USA.

Zakotnik J, Matheson T, Dürr V. 2004. A posture optimization algorithm for model-based motion capture of movement sequences. J Neurosci Methods. 135:43–54.