



Estimation of musculotendon kinematics in large musculoskeletal models using multidimensional B-splines

Massimo Sartori^{a,b,*}, Monica Reggiani^c, Antonie J. van den Bogert^d, David G. Lloyd^e

^a Department of Information Engineering, University of Padova, Via Gradenigo 6/B, Padova, IT 35131, Italy

^b Institute of Biomedical Engineering, National Research Council, Corso Stati Uniti 4, Padova, IT 35127, Italy

^c Department of Management and Engineering, University of Padova, Stradella S. Nicola 3, Vicenza, IT 36100, Italy

^d Orchard Kinetics LLC, Cleveland, OH, USA

^e Musculoskeletal Research Program, Griffith Health Institute, Griffith University, Gold Coast, QLD, 4222, Australia

ARTICLE INFO

Article history:

Accepted 31 October 2011

Keywords:

Musculoskeletal modeling
Musculotendon length
Muscle moment arm
Muscle force
Multidimensional spline interpolation

ABSTRACT

We present a robust and computationally inexpensive method to estimate the lengths and three-dimensional moment arms for a large number of musculotendon actuators of the human lower limb. Using a musculoskeletal model of the lower extremity, a set of values was established for the length of each musculotendon actuator for different lower limb generalized coordinates (joint angles). A multidimensional spline function was then used to fit these data. Muscle moment arms were obtained by differentiating the musculotendon length spline function with respect to the generalized coordinate of interest. This new method was then compared to a previously used polynomial regression method. Compared to the polynomial regression method, the multidimensional spline method produced lower errors for estimating musculotendon lengths and moment arms throughout the whole generalized coordinate workspace. The fitting accuracy was also less affected by the number of dependent degrees of freedom and by the amount of experimental data available. The spline method only required information on musculotendon lengths to estimate both musculotendon lengths and moment arms, thus relaxing data input requirements, whereas the polynomial regression requires different equations to be used for both musculotendon lengths and moment arms. Finally, we used the spline method in conjunction with an electromyography driven musculoskeletal model to estimate muscle forces under different contractile conditions, which showed that the method is suitable for the integration into large scale neuromusculoskeletal models.

© 2011 Elsevier Ltd. All rights reserved.

1. Introduction

Musculoskeletal modeling has applications ranging from the study of muscle contributions to movement (Lloyd and Besier, 2003; Winby et al., 2009) to the development of human-machine interfaces (Fleischer and Hommel, 2008). Musculoskeletal modeling and simulations require accurate estimates of musculotendon kinematics including musculotendon length (ℓ^{mt}) and three-dimensional moment arms (r) to accurately predict musculotendon forces (F^{mt}) and joint moments.

Musculotendon kinematics estimates can be produced by a software that models musculotendon paths wrapping around

points and/or surfaces (Delp et al., 1990, 2007). However, this is based on obstacle detection and may cause discontinuities in the predicted musculotendon kinematics (Gao et al., 2002; Garner and Pandey, 2000). Moreover, it is desirable that musculotendon kinematics equations are continuously differentiable to enable the computation of analytical Jacobians for the forward simulation of the musculoskeletal system (Ackermann and van den Bogert, 2010; Menegaldo et al., 2006).

Sets of differentiable polynomial regression equations (P^{eq}) have been proposed to estimate ℓ^{mt} and r from nominal values of both parameters corresponding to combinations of discrete joint angles or generalized coordinates (GC) (Menegaldo et al., 2004). This required gathering data sets on both ℓ^{mt} and r and manually identifying the best-performing equations that depended on the muscle, number of GCs, and the fitted musculotendon parameter. This necessitated computing and storing coefficients for all equations for ℓ^{mt} and r for each muscle before use.

Alternatively, continuously differentiable multidimensional cubic splines can be used, but are yet to be examined as a means to

* Corresponding author at: Department of Information Engineering, University of Padova, Intelligent Autonomous Systems Laboratory (Stanza L3), Via Gradenigo 6/B, Padova, IT 35131, Italy. Tel.: +39 049 827 78 33; fax: +39 049 827 78 26; Mobile: +39 349 615 64 10.

E-mail addresses: massimo.srt@gmail.com, m.sartori@unipd.it (M. Sartori).

URL: <http://www.massimosartori.net/> (M. Sartori).

Table 1

Generalized coordinates (GCs) and their angular ranges of motion (ROM), expressed in degrees, of the two general lower limb models. The LowerLimb4307 model had 43 musculotendon actuators (MTAs) and 7 GCs (Delp et al., 1990). This was used by Menegaldo et al. (Menegaldo et al., 2004) to study polynomial regression equations (P^{eq}) fitting accuracy and was used in this study as the comparison method. The LowerLimb1310 model had 13 MTAs crossing the knee joint and 10 GCs (Winby et al., 2009). It was used to study splines and P^{eq} fitting capabilities over a wider range of GC angles. The metatarsus-phalangeal angle was not used in this model because MTAs do not cross the metatarsus-phalangeal joint. Two different ROM arrangements were used for this model to examine how the fitting accuracy varied when the distance between nodes increased.

LowerLimb4307 model	Min.		Max.	
Hip internal–external rotation (HR)	–20		20	
Hip adduction–abduction (HA)	–50		15	
Hip flexion–extension (HF)	–10		95	
Knee flexion–extension (KF)	–10		95	
Subtalar flexion–extension (SF)	–20		20	
Ankle dorsi–plantar flexion (AF)	–30		30	
Metatarsus–phalangeal angle (MA)	–30		30	

LowerLimb1310 model	Reduced ROM		Extended ROM	
	Min.	Max.	Min.	Max.
Hip internal–external rotation (HR)	–20	20	–30	20
Hip adduction–abduction (HA)	–50	15	–50	30
Hip flexion–extension (HF)	0	95	–10	95
Knee internal–external rotation (KR)	–5	5	–10	10
Knee adduction–abduction (KA)	–3	3	–5	5
Knee flexion–extension (KF)	0	120	–10	120
Subtalar flexion–extension (SF)	–10	10	–20	20
Ankle adduction–abduction (AA)	–10	10	–20	20
Ankle dorsi–plantar flexion (AF)	–30	30	–40	30

estimate musculotendon kinematics. In this paper we show that a single spline function per muscle can be used to estimate ℓ^{mt} and r based on only nominal values of ℓ^{mt} . We then compare fitting accuracy and computation time to P^{eq} . The ℓ^{mt} nominal values were derived from musculoskeletal geometry models using OpenSim (Delp et al., 2007). Finally, we quantified how differences in the ℓ^{mt} estimates affect F^{mt} estimates using an electromyography (EMG) driven musculoskeletal model (Buchanan et al., 2004; Lloyd and Besier, 2003).

2. Methods

The performance of P^{eq} and spline methods were examined using two musculoskeletal models (Table 1): LowerLimb4307 (Delp et al., 1990) and LowerLimb1310 (Winby et al., 2009). The reader is referred to (Menegaldo et al., 2004) for P^{eq} theory, while a summary of the spline theory (Habermann and Kindermann, 2007) is now presented.

A cubic spline consists of a series of third-order polynomials spliced together to preserve continuity of the first and second-order derivatives. The points at which polynomials are spliced together, $a = x_0 < x_1 < \dots < x_n = b \in \mathbb{R}$ are called *nodes*, where $\Delta_n = [a = x_0, x_1, \dots, x_n = b]$ is the bounded interval of interest. If Δ_n and the interpolation data $y_i \in \mathbb{R}$, $i = 0, \dots, n$ are given, a spline interpolates y_i while satisfying $n+3$ conditions on the nodes: $s_1(x_i) = y_i$ for $i = 0, \dots, n$, and $\tilde{s}_1(a) = \alpha$, $\tilde{s}_1(b) = \beta$, where s_1 represents a one-dimensional spline function on Δ_n , \tilde{s}_1 is its second-order derivative, while α and β are real-valued terms (Habermann and Kindermann, 2007). A spline with $n+1$ nodes can be written as a linear combination of $n+3$ cubic polynomials u_1, u_2, \dots, u_{n+3} defining a *basis*:

$$s_1(x) = \sum_{i=1}^m c_i u_i(x) \quad (1)$$

with coefficients c_i , where $m = \min(l+3, n+3)$ and $l = (x-a)/h+1$. When the interpolation data vary as a function of d variables (d -dimensional case), Eq. (1) can be generalized (Habermann and Kindermann, 2007) thus requiring $N_c = \prod_{i=1}^d (n_i+3)$ coefficients whilst a spline value can be computed using $\prod_{i=1}^d (m_i - l_i + 1) \leq 3^d$ coefficients and basis functions.

Given two real-valued variables x and z on the bounded intervals $\Delta_{n_1} = [a_1, b_1]$ and $\Delta_{n_2} = [a_2, b_2]$, the spline partial derivative with respect to x can be computed using the spline coefficients $c_{i_1 i_2}$:

$$\frac{\partial s_2(x, z)}{\partial x} = \sum_{i_1=1}^{m_1} \sum_{i_2=1}^{m_2} c_{i_1 i_2} \dot{u}_{i_1}(x) v_{i_2}(z) \quad (2)$$

where $u_{i_1}(x)$ and $v_{i_2}(z)$ represent basis functions for the two variables, respectively, and $\dot{u}_{i_1}(x)$ is the first-order derivative.

The problem of finding ℓ^{mt} as a function of d GC-angles, (q_1, \dots, q_d) is formulated by determining the $(n_1+3) \dots (n_d+3)$ coefficients for the d -dimensional spline function $s_d(q_1, \dots, q_d)$, where ℓ^{mt} are the nominal data to be interpolated. Moment arms (r_{q_i}) with respect to a specific GC (q_i) are then found by the partial differentiation (Eq. (2)) of $s_d(q_1, \dots, q_d)$:

$$r_{q_i} = \frac{\partial \ell^{mt}(q_1, \dots, q_d)}{\partial q_i} = \frac{\partial s_d(q_1, \dots, q_d)}{\partial q_i} \quad (3)$$

The ℓ^{mt} values for q_i that extend beyond their intervals Δ_i , $\forall 1 \leq i \leq d$, are obtained by creating a tangent line at all end points using linear extrapolation with the slope given by the values of r_{q_i} at the end points. Moment arms are therefore kept constant for q_i values that extend beyond Δ_i .

2.1. Validation procedure

Validation comprised of four tests to assess fitting accuracy and one to examine computation time and memory storage requirements. GC ranges of motion (ROM) were sampled into equidistant nodes and input into OpenSim¹ to compute ℓ^{mt} and r nominal values for P^{eq} coefficient calculation. Only ℓ^{mt} values were used for splines coefficients. The fitting accuracy was assessed both on the interpolation nodes (*on-nodes*) and on points located midway between nodes (*off-nodes*).

The four fitting accuracy tests examined the ability of the spline and P^{eq} methods in reproducing OpenSim-generated ℓ^{mt} , r , and F^{mt} calculated using OpenSim-generated ℓ^{mt} (Section 2.2). The Mean Fitting Error (MFE) and its standard deviation (σ) were used to measure the expected error, and its variation, for P^{eq} and spline estimations of OpenSim-generated values:

$$MFE = \frac{1}{N} \sum_{i=1}^N |\hat{X}_i - X_i|, \quad (4)$$

where \hat{X} is the OpenSim-generated ℓ^{mt} or r , or F^{mt} calculated using OpenSim-generated ℓ^{mt} ; the X corresponds to the value estimated using the P^{eq} or spline methods, and N the number of fitting function points. The percent MFE (%) was estimated with respect to the corresponding expected OpenSim-generated value ($E[\hat{X}]$):

$$\%E = \frac{MFE}{E[\hat{X}]} \times 100 \quad (5)$$

The MFE with 95% confidence interval (CI) was estimated using the Central Limit Theorem, which enabled maximum expected errors (MEE) to be calculated:

$$MEE = MFE + 1.96\sigma, (95\%) \quad (6)$$

¹ In this study we used the OpenSim release 2.2.0. This is available for download from: <https://simtk.org/home/opensim>.

This assumed a random error with normal distribution that is generally not the case (Fig. 1a). A normality check was performed using a one-sample Kolmogorov–Smirnov test with a 5% significance level (Miller, 1956). If normality was rejected, then a more conservative MEE with 90% and 95% CIs was obtained using the Chebyshev's Theorem, which can be applied to any probability distribution with known mean and σ :

$$\text{MEE} = \text{MFE} + 3.16\sigma, (90\%) \quad (7)$$

$$\text{MEE} = \text{MFE} + 4.47\sigma, (95\%) \quad (8)$$

All metrics were applied to all fitting accuracy tests, except for % ϵ that was only used in the second test. These tests are now described in more detail.

2.2. Validation fitting accuracy tests

The first test of fitting accuracy examined splines and P^{eq} musculotendon kinematics using Menegaldo's et al. input data, musculoskeletal model (i.e. LowerLimb4307 model), analysis procedure, and resulting P^{eq} set (Menegaldo et al., 2004). GC ROMs were sampled with 20 points, although when ℓ^{mt} depended on four GCs, a 15-point sampling was used.

In the second test the semimembranosus was chosen to examine how fitting accuracy changed with a large number of GCs, and varying distance between nodes. The LowerLimb1310 model was used with a 10-point GC ROM sampling using two different ROM arrangements (Table 1). The number of GCs was varied from four to six. The 4-GC set included: hip internal–external rotation (HR), hip adduction–abduction (HA), hip flexion–extension (HF) and knee flexion–extension (KF). The 5-GC set added knee internal–external rotation (KR), and the 6-GC set included knee adduction–abduction (KA).

In the third and fourth tests, biarticular thigh muscles from the LowerLimb1310 model were chosen to assess the impact of the spline and P^{eq}

estimates of ℓ^{mt} on the calculation of F^{mt} . This employed four GCs with knee GCs reduced to KF only, and a 10-point extended ROM sampling (Table 1). F^{mt} was estimated as a function of EMG and ℓ^{mt} using an EMG-driven model (Lloyd and Besier, 2003), which we briefly summarize. Raw EMGs were first high-pass filtered (30 Hz), full wave rectified, and low-pass filtered (6 Hz) to create EMG linear envelopes ($e(t)$). The linear envelopes were normalized to maximum EMG data that were the peak $e(t)$ values from a set of motor tasks that included: three trials of running at 5 m/s, five repeated calf rises, and maximum exertion isometric knee flexion–extension trials twice repeated at each 100°, 70°, and 40° of knee flexion on a Biodex (NY, USA) dynamometer. The normalized linear envelopes were then processed using a model that represented the muscle's twitch response as a critically-damped second-order filter (Lloyd and Besier, 2003). These data series ($u(t)$) were subsequently adjusted to account for the non-linear EMG-to-force relationship (Lloyd and Besier, 2003). The resulting muscle activation ($a(t)$) was a dimensionless term varying between 0 and 1.

Each musculotendon actuator (MTA) had muscle fibers modeled as a Hill-type muscle model (Zajac, 1989) in series with a non-linear elastic tendon (Lloyd and Besier, 2003). Fibers were modeled using generic force–length $f(\ell^m)$, force–velocity $f(v^m)$, and parallel passive force–length curves $f_p(\ell^m)$. These were normalized to maximum isometric muscle force (F^{max}), optimal fiber length (ℓ_0^m), and maximum muscle contraction velocity (v^{max}) (Zajac, 1989). Tendons were modeled with a non-linear force–strain function $f(\epsilon)$ (Zajac, 1989), where $\epsilon = (\ell - \ell_s)/\ell_s$ is the tendon strain, ℓ the tendon length, and ℓ_s the tendon slack length, with $f(\epsilon)$ normalized to F^{max} (Zajac, 1989). Using biomechanical parameters from (Delp et al., 1990), F^{mt} was calculated as a function of $a(t)$, fiber length ℓ^m , and fiber contraction velocity v^m :

$$F^{mt} = F^t \cos(\varphi(t)) = [a(t)f(\ell^m)f(v^m) + f_p(\ell^m)]F^{max} \cos(\varphi(t)), \quad (9)$$

where F^t is the tendon force, and $\varphi(t)$ the pennation angle that changes with ℓ^m (Scott and Winter, 1991). ℓ^{mt} estimates were generated using OpenSim, P^{eq} , and

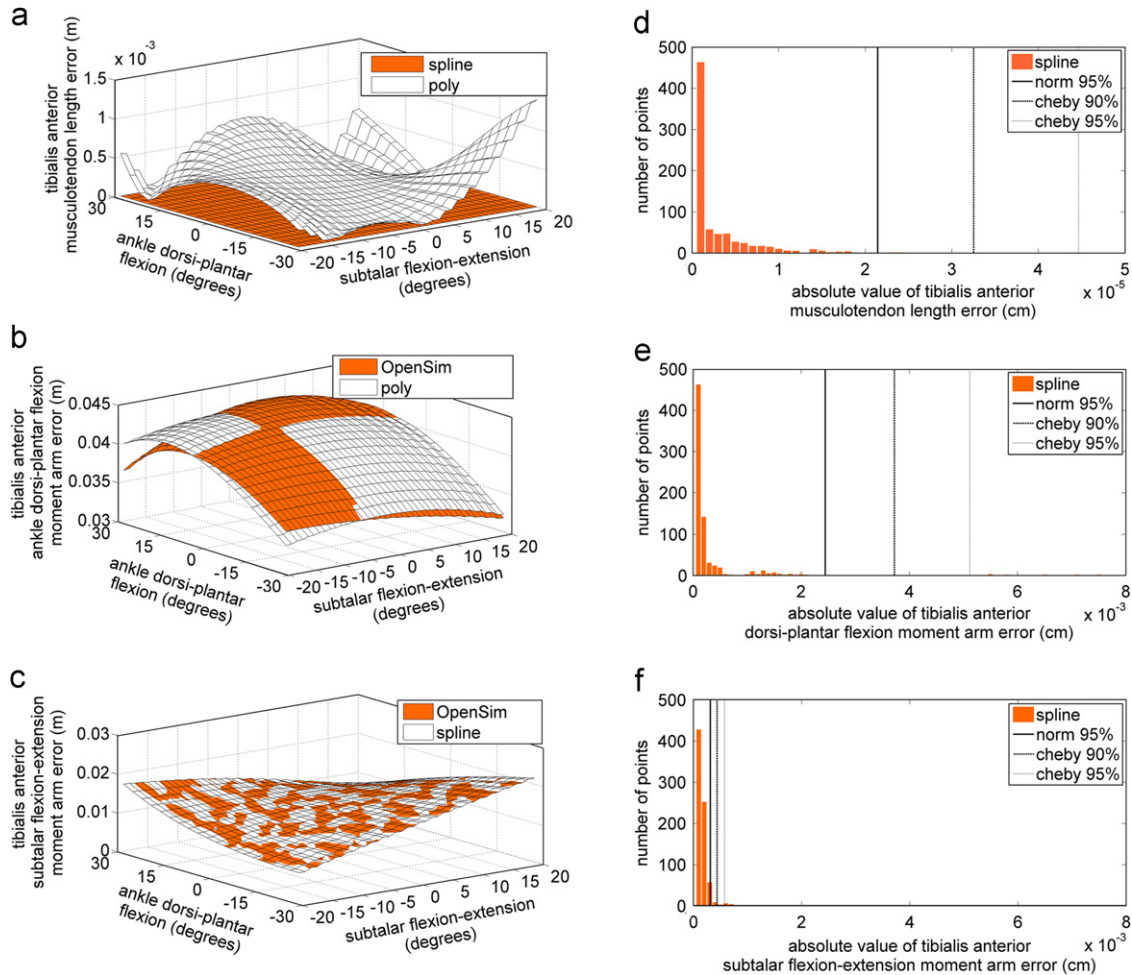


Fig. 1. First fitting accuracy test: all graphs report values for the tibialis anterior muscle using the LowerLimb4307 model (Delp et al., 1990). (a) Two surfaces represent the distribution of the difference between musculotendon length ℓ^{mt} computed using OpenSim, and ℓ^{mt} estimated by P^{eq} (poly) and spline function (spline), respectively. (b) Distribution of the ankle dorsi-plantar flexion moment arms (r_{AF}) computed using OpenSim and using P^{eq} , respectively. (c) Distribution of the subtalar flexion–extension moment arm (r_{SF}) computed using OpenSim and spline function, respectively. Histograms (d), (e), and (f) show the spline method fitting error distribution associated to ℓ^{mt} , r_{AF} , and r_{SF} , respectively. The maximum expected errors (MEEs) are calculated both using Chebyshev's Theorem (cheby) and the Central Limit Theorem (norm).

splines, which were used as the basis to calculate F^{mt} time series for each MTA (Lloyd and Besier, 2003). Briefly, this was performed using initial values of l^m from which tendon length was determined i.e. $l^t = l^{mt} - l^m \cos(\varphi(t))$. Then, tendon strain $\varepsilon = (l^t - l_s^t)/l_s^t$ was calculated from which F^t , and F^{mt} , were determined. By rearranging Eq. (9) the fiber velocity dependent muscle force F^v was determined (Buchanan et al., 2004):

$$F^v = \frac{(F^m - f_p(l^m))F^{max}}{a(t)f(l^m)F^{max}} \quad (10)$$

since activation $a(t)$ was known and both $f(l^m)$ and $f_p(l^m)$ were evaluated using the current l^m . Using F^v and inverting the fiber-velocity relationship $f(v^m)$, v^m could be determined, which was then numerically integrated to estimate the next l^m in time. And so the process continued until the end of the time series producing estimates for each MTA's time varying F^{mt} . In the special case for isometric contractile conditions, a simplified procedure (Buchanan et al., 2004) was used in which initial values for l^m and l^t were iteratively adjusted to ensure $F^t = F^{mt} \cos(\varphi(t))$ and $l^{mt} = l^t + l^m \cos(\varphi(t))$.

We compared F^{mt} computed using splines and P^{eq} -generated values for l^{mt} to our gold standard F^{mt} derived using the OpenSim-generated l^{mt} . Estimated F^{mt} values were multiplied by the respective muscle KF moment arms and summed to determine the total KF moment.

In the third test the EMG-driven model was used to examine the impact of l^{mt} estimates on F^{mt} estimates under maximum isometric contractile conditions (i.e. $a(t) = 1$ and $v^m(t) = 0$, $\forall t$) not involving real subject data. *On-nodes* and *off-nodes* l^{mt} values were calculated using OpenSim, P^{eq} , and splines, where each l^{mt} value was associated to a hypothetical instant of time t .

The fourth test studied how l^{mt} estimates affected F^{mt} evaluations in low-effort (walking at 1.4 ± 0.3 m/s) and higher-effort (jogging at 2.6 ± 0.4 m/s) dynamic contractile conditions. A healthy male subject (mass: 81.7 kg, height: 175.7 cm, age: 30 years) gave his informed, written consent prior to data collection. The subject's static anatomical poses and gait patterns were captured using a seven-camera Vicon MX Motion Analysis system (Oxford Metrics, UK) sampling at 100 Hz. During the gait trials, ground reaction forces (GRF) and EMG data were acquired at 2000 Hz synchronously with motion data. EMGs were collected from the muscles crossing the knee defined in the LowerLimb1310 model. Marker trajectories and GRF were low-pass filtered (6 Hz) as suggested by (McLean et al., 2005). OpenSim was used to create a dynamic motion simulation per gait trial. The LowerLimb1310 model was scaled (Delp et al., 2007) to match the subject's anthropometry based on experimentally measured marker positions from static poses. OpenSim created virtual representations of the experimental markers and an inverse kinematics algorithm solved for GC-angles that minimized the difference between experimental and virtual markers (Delp et al., 2007). GC-moments needed to generate the subject's movement were computed using the OpenSim inverse dynamics algorithm (Kuo, 1998). The inverse kinematics GC-angles from the walking and jogging trials were within the bounds of the extended range of motion of the LowerLimb1310 model (Table 1), from which the l^{mt} values were computed for each muscle using OpenSim, P^{eq} , and splines, respectively.

Some EMG-driven model parameters vary non-linearly with subject's anthropometry and the physiological conditions during testing and could not be obtained experimentally or from literature. These parameters were calibrated to the individual (Lloyd and Besier, 2003; Winby et al., 2009) and included two global activation filtering coefficients that were varied between -1 and 1 to realize a stable positive solution to the critically-damped recursive filter (Lloyd and Besier, 2003). One global activation shape parameter was also varied between -5 and 0 to account for the non-linearity of the EMG-to-force relationship (Lloyd and Besier, 2003). Finally, muscle-specific l_s^t values were varied so that $l_s^t = \text{initial value} \pm 5\%$, with initial values taken from literature (Delp et al., 1990; Lloyd and Besier, 2003). The parameters were adjusted using a simulated annealing algorithm that minimized² the least-squared differences between the KF moments estimated using the EMG-driven model and inverse dynamics (Winby et al., 2009). Three walking and two jogging trials were used for parameter adjustment. Using the adjusted parameters the EMG-driven model was then used to estimate F^{mt} 's (Eq. (9)) in other gait trials not used in the adjustment process. Forces estimated during the stance phase were time-normalized and two ensemble average curves computed from walking and jogging trials, respectively, and used to estimate the instantaneous force errors using Eq. (4) (with $N = 1$, $\forall 0\% \text{ stance} \leq i \leq 100\% \text{ stance}$).

Finally, we examined how computation time varied with the number of GC's. To this end, we computed the average time needed to repeatedly estimate one sample of l^{mt} and r 10,000 times with GCs varying from one to six, respectively. A 2.26 GHz 3 GB-RAM PC was used. Both spline and P^{eq} evaluation routines were implemented in C++ for speed comparison. Furthermore, the memory required to store the coefficients and nominal data for all muscles during each test for both P^{eq} and spline methods was assessed. For each test and method, two text files were created where coefficients and nominal data were stored, respectively, and then the files sizes were reported in megabytes.

3. Results

The tibialis anterior muscle was chosen to demonstrate the spline performances in the first test. This muscle's kinematics depended on ankle dorsi-plantar flexion (AF) and subtalar flexion-extension (SF). The variation of l^{mt} , r , and errors are easy to visualize in 3D (Fig. 1). The fitting errors between the OpenSim and spline l^{mt} values were uniform throughout the GC workspace reaching a 0.3×10^{-6} cm maximum while P^{eq} errors reached higher peaks (0.14 cm) at the boundaries³ (Fig. 1a). The P^{eq} fitting error for the AF moment arms reached peak values (0.59 cm) at the boundaries (Fig. 1b). There was good agreement between the OpenSim-generated surface associated to SF moment arms and those generated by the spline function with a 8×10^{-4} cm maximum error (Fig. 1c). The distribution of the spline-generated errors is shown in Fig. 1d–f. Errors were in the order of 10^{-5} cm for the l^{mt} estimation and in the order of 10^{-3} cm for the estimation of AF and SF moment arms. The distribution of the P^{eq} -generated error was previously published by Menegaldo et al. (Menegaldo et al., 2004) and is not reported in this study. However, for all muscles, P^{eq} MFEs were always three orders of magnitude larger than spline MFEs (Table 2).

From the second test, P^{eq} -generated l^{mt} fitting errors (Table 3) evaluated *off-nodes* (10^{-3} m) were smaller than those evaluated *on-nodes* (between 10^{-3} m and 10^{-2} m). *On-nodes* spline errors were negligible (10^{-10} m), due to the spline interpolation conditions, and *off-nodes* spline errors varied between 10^{-6} m and 10^{-4} m (Table 3). Finally, spline-generated % ε for the HA moment arms (Table 4) were smaller (between 0.04% and 0.26%) than that of P^{eq} -generated % ε (between 8.38% and 12%).

From the third test, the muscle forces estimated using OpenSim-generated l^{mt} are shown in Table 5. P^{eq} -generated force errors fell into classes with errors in the hundreds of newtons (N) (Fig. 2a). However, spline-generated force errors were always less than 30 N (Fig. 2b) and at all times fell in the first and smallest P^{eq} error class (Fig. 2a).

From the fourth test, the estimated forces generated by semimembranosus in walking and jogging are reported in Table 6. P^{eq} -generated force errors were higher during high-activation jogging tasks than during low-activation walking tasks as opposed to the spline case (Fig. 2c). The l^{mt} fitting errors significantly varied between spline and P^{eq} and directly impacted on the force prediction. During walking, P^{eq} -generated l^{mt} fitting errors reached peak values (0.5 ± 0.07 cm) from 50% to 70% of stance corresponding to peak force errors (4.5 ± 0.5 N) in the same range (Fig. 2c). During jogging, P^{eq} -generated l^{mt} fitting errors reached peak values (0.48 ± 0.02 cm) from 65% to 80% of stance corresponding to an increase in force error up to 20 N in a similar range (Fig. 2c). Spline-generated l^{mt} fitting errors were always less than 0.006 cm with corresponding force errors smaller than 1 N during walking and jogging (Fig. 2c).

In all tests, the Kolmogorov–Smirnov test rejected the normal distribution hypothesis for splines and P^{eq} fitting errors for all muscles justifying the use of the Chebyshev's Theorem for CI computation.

Spline computation time ranged from 10^{-5} ms (one-dimensional case) to 10^{-2} ms (six-dimensional case) as opposed to 10^{-5} ms with the P^{eq} . Even though the number of spline coefficients varies exponentially with the number of GCs and nodes, compared to the P^{eq} , the total memory required was smaller in the spline case since splines only needed nominal l^{mt} data to estimate both l^{mt} and r (Table 7).

³ The polynomial regression equations used were from (Menegaldo et al., 2004).

² OpenSim-generated l^{mt} were used.

Table 2

First fitting accuracy test: the mean fitting errors (MFE) and standard deviations (σ), expressed in meters, for both polynomial regression equation (P^{eq}) and splines. Values are associated to the estimation of musculotendon length (ℓ^{mt}) and moment arms (r) and were obtained by averaging MFEs and σ associated to all muscles in the LowerLimb4307 model (Delp et al., 1990). Moment arm estimates from the splines were obtained by differentiating the musculotendon length (ℓ^{mt}) spline function (Eq. (3)). A set of P^{eq} were individually fitted to ℓ^{mt} and r for each muscle.

MFE $\pm \sigma$	ℓ^{mt}	r_{HR}	r_{HA}	r_{HF}	r_{KF}	r_{SF}	r_{AF}	r_{MA}
P^{eq}	7.2E-04 \pm 6.4E-04	2.1E-03 \pm 1.6E-03	2.8E-03 \pm 2.2E-03	2.9E-03 \pm 2.3E-03	4.7E-04 \pm 5.1E-04	9.4E-04 \pm 9.1E-04	4.4E-04 \pm 7.2E-04	1.6E-04 \pm 9.3E-05
Spline	1.1E-07 \pm 3.2E-07	1.8E-06 \pm 3.3E-06	3.4E-06 \pm 1.2E-05	3.8E-06 \pm 1.4E-05	1.2E-05 \pm 4.1E-05	6.7E-06 \pm 1.9E-05	2.0E-06 \pm 3.8E-06	9.3E-07 \pm 9.9E-07

Table 3

Second fitting accuracy test: the semimembranosus's musculotendon length (ℓ^{mt}) fitting error (meters) using the LowerLimb1310 model (Winby et al., 2009). In this test the effect of the number ($n=4, 5$, and 6) of generalized coordinates (GC) and the distance between interpolation nodes (reduced/extended GC range of motion (ROM)) were evaluated. P_n^{eq} represents the n -dimensional polynomial regression equation that was a function of n GCs, while s_n represents the n -dimensional spline function. Both polynomial regression and spline methods were used to estimate ℓ^{mt} using the reduced and extended ROMs for the GCs listed in Table 1. With these two ROM arrangements, ℓ^{mt} was estimated both on the interpolation nodes (*on-nodes*) and at points placed midway between two consecutive interpolation nodes (*off-nodes*). Error measures used to quantify the fitting accuracy included: Mean Fitting Error (MFE), standard deviation (σ) and 95% confidence interval using the Chebyshev's distribution (MEE).

On or off nodes	Reduced ROM			Extended ROM		
	MFE	σ	MEE	MFE	σ	MEE
<i>On-nodes</i>						
P_4^{eq}	2.15E-03	1.74E-03	9.92E-03	2.35E-03	2.22E-03	1.23E-02
s_4	2.39E-10	1.69E-10	9.97E-10	2.25E-10	1.60E-10	9.42E-10
<i>Off-nodes</i>						
P_4^{eq}	1.86E-03	1.42E-03	8.19E-03	1.91E-03	1.71E-03	9.54E-03
s_4	6.93E-06	7.18E-06	3.90E-05	1.62E-05	2.11E-05	1.11E-04
<i>On-nodes</i>						
P_5^{eq}	2.16E-03	1.68E-03	9.69E-03	2.45E-03	2.26E-03	1.26E-02
s_5	2.46E-10	1.75E-10	1.03E-09	2.31E-10	1.66E-10	9.74E-10
<i>Off-nodes</i>						
P_5^{eq}	1.87E-03	1.34E-03	7.85E-03	2.02E-03	1.72E-03	9.71E-03
s_5	6.92E-06	7.17E-06	3.90E-05	1.62E-05	2.11E-05	1.11E-04
<i>On-nodes</i>						
P_6^{eq}	2.17E-03	1.68E-03	7.49E-03	2.53E-03	2.28E-03	1.27E-02
s_6	2.56E-10	1.83E-10	1.07E-09	2.51E-10	1.82E-10	1.06E-09
<i>Off-nodes</i>						
P_6^{eq}	1.86E-03	1.33E-03	7.81E-03	2.08E-03	1.74E-03	9.86E-03
s_6	1.35E-05	1.88E-05	9.74E-05	2.66E-05	3.52E-05	1.84E-04

Table 4

Second fitting accuracy test: the percent mean fitting error (%) obtained using the LowerLimb1310 model (Winby et al., 2009). Values of % ϵ are from the estimation of the semimembranosus hip adduction–abduction moment arms (r_{HA}). In this test the effect of the number ($n=4, 5$, and 6) of generalized coordinates (GC) and the distance between interpolation nodes (reduced/extended GC range of motion (ROM)) were evaluated. Values of % ϵ are calculated using P_n^{eq} that represents the n -dimensional polynomial regression equation that was a function of n GCs and was used to directly estimate r_{HA} . Furthermore, values of % ϵ are also calculated using $\partial s_n^{mt} / \partial q_{HA}$ that is the first-order partial derivative of the n -dimensional spline function that was used to directly estimate ℓ^{mt} (i.e. s_n^{mt}) with respect to hip adduction–abduction GC (q_{HA}). Values of % ϵ were calculated using both an extended and a reduced ROM as in Table 1. With these two arrangements, % ϵ was calculated both on the interpolation nodes (*on-nodes*) and on a set of nodes placed midway between the interpolation nodes (*off-nodes*).

	Reduced ROM		Extended ROM	
	<i>On-nodes</i>	<i>Off-nodes</i>	<i>On-nodes</i>	<i>Off-nodes</i>
P_4^{eq}	10.35	8.39	12.01	9.94
$\frac{\partial s_4^{mt}}{\partial q_{HA}}$	0.18	0.05	0.26	0.06
P_5^{eq}	10.34	8.38	12.00	9.93
$\frac{\partial s_5^{mt}}{\partial q_{HA}}$	0.18	0.04	0.26	0.06
P_6^{eq}	10.34	8.38	12.00	9.95
$\frac{\partial s_6^{mt}}{\partial q_{HA}}$	0.18	0.05	0.26	0.07

4. Discussion

Multidimensional cubic B-splines were used to estimate lower-limb musculotendon kinematics and were compared to a previous method that employed polynomial regression (Menegaldo et al.,

2004). Splines provided fitting errors that were uniform throughout the GC workspace and at least three orders of magnitude smaller than P^{eq} errors, the latter increasing at the workspace boundaries. P^{eq} -generated force errors were (i) larger than spline-generated errors during maximum isometric contractile conditions, and

Table 5

Third fitting accuracy test: the mean, standard deviation (σ), maximum (Max.), and minimum (Min.) musculotendon force (expressed in newtons) during maximum isometric contractile conditions. These were obtained using OpenSim-generated musculotendon lengths ℓ^{mt} , for the biarticular thigh muscles: semimembranosus (SM), semitendinosus (ST), biceps femoris long head (BFLH), sartorius (SAR), rectus femoris (RF), and gracilis (GRA). The ℓ^{mt} nominal values for the force estimation were obtained from the 10^4 data points corresponding to all configurations of hip adduction–abduction, hip internal–external rotation, hip flexion–extension, and knee flexion–extension (Section 2.2).

	SM	ST	BFLH	SAR	RF	GRA
Mean	1090.6	306.8	623.9	102.3	895.1	105.9
σ	886.2	56.7	351.6	8.5	603.4	17.5
Max.	5368.9	545.8	2084.4	168.85	3924.3	177.4
Min.	0	95.7	0	78.5	0.1	49.6

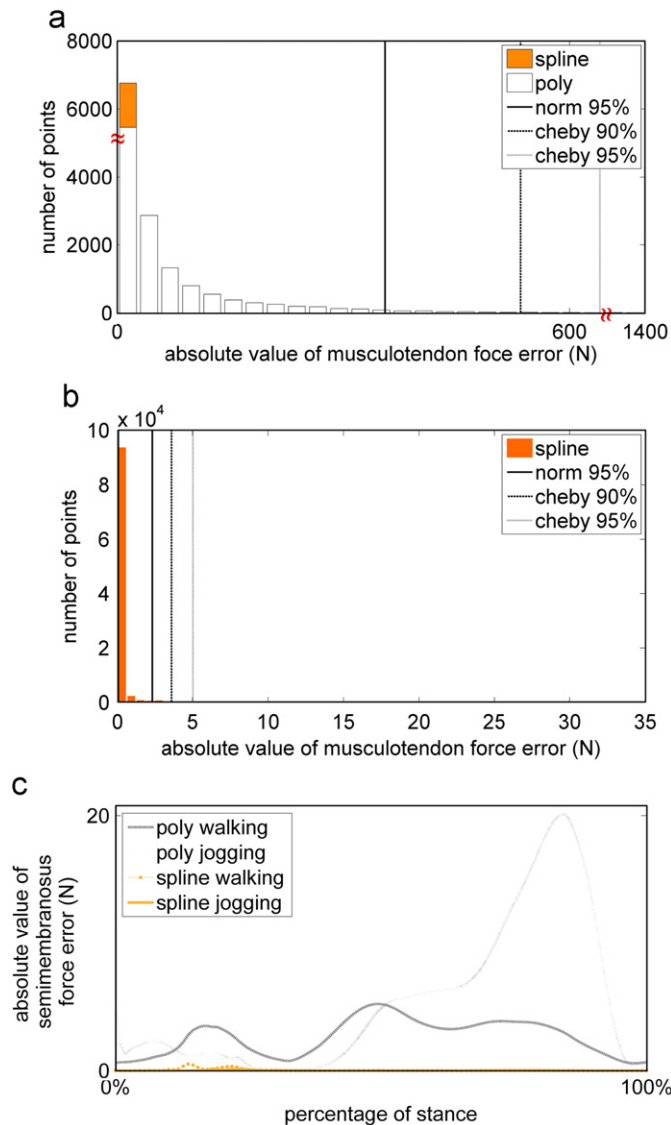


Fig. 2. Histograms (a) and (b) report the third test's error distribution associated to the prediction of the maximum isometric musculotendon force F^{mt} . The maximum expected error (MEE) calculated both using Chebyshev's Theorem (cheby) and the Central Limit Theorem (norm) are also reported. Histogram (a) shows the error distribution for both spline and polynomial regression equations P^{eq} (poly). MEEs are reported for P^{eq} only because spline's MEEs are contained in the first (smallest) error class. Histogram (b) shows the error distribution for the spline method only as well as associated MEEs. Graph (c) reports the absolute value of the instantaneous semimembranosus force prediction errors averaged over all walking and jogging trials during the fourth test. Histograms (a) and (b), and graph (c) combine values for all biarticular thigh muscles in the LowerLimb1310 model (Table 5) (Winby et al., 2009).

Table 6

Fourth fitting accuracy test: the semimembranosus's activation and musculotendon force, obtained using OpenSim-generated ℓ^{mt} , for the four discrete events in gait: beginning of stance, first activation/force peaks, second activation/force peaks, end of stance phase. Values are averaged over the 15 walking (Walk) and 5 jogging (Jog) trials (Section 2.2).

Walk				
Stance %	0	25	40	100
Activation	0.02 ± 0.01	0.49 ± 0.05	0.39 ± 0.08	0.03 ± 0.01
Force (N)	23.5 ± 12.4	369 ± 24.7	277 ± 51	5.14 ± 11.4
Jog				
Stance %	0	30	48	100
Activation	0.24 ± 0.03	0.65 ± 0.03	0.49 ± 0.04	0.28 ± 0.02
Force (N)	132.5 ± 17.6	441 ± 13	277.9 ± 11.3	95.38 ± 11.6

(ii) dependent on the level of muscle activation during dynamic tasks. This suggests splines may be more reliable for integration into forward dynamics musculoskeletal models, in which analytical Jacobians can be easily computed by differentiating the ℓ^{mt} spline function (Eq. (3)).

The time needed to evaluate one point on P^{eq} was always in the order of 10^{-5} ms whereas spline computation time varied exponentially with the number of GCs. However, this was always less than 0.01 ms (6-GC case) thus providing adequate speed for applications requiring real-time computation of ℓ^{mt} and r with the subsequent real-time estimation of F^{mt} and joint moments. Examples of these emerging applications are biofeedback (Murai et al., 2010) or EMG-controlled powered orthoses (Fleischer and Hommel, 2008), where the real-time deadline is the muscle electromechanical delay of about 20 ms (Nordez et al., 2009).

In general, spline fitting accuracy increases with the number of nodes with no over-fitting risk. We showed, in tests 1 and 2, that sampling GC-angles with 10, 15, or 20 equidistant nodes allowed for robust fitting of individual MTA kinematics. The number of nodes has to be chosen considering the number of spline coefficients and the associated memory storage requirements, which vary exponentially with the number of GCs (i.e. d) and nodes (Table 7). However, during our tests, the total memory to store coefficients and nominal values for splines was smaller than that needed for P^{eq} . One spline per muscle is required to estimate both ℓ^{mt} and r while $(d+1)$ P^{eq} are needed to individually fit ℓ^{mt} and r nominal values (Menegaldo et al., 2004) (Table 7). This is an issue when nominal values are obtained from imaging techniques, with large storage requirements, and from which values for r are difficult to determine.

As just suggested, one may use imaging methods to create splines from more realistic nominal values of ℓ^{mt} . These could be determined using the centroids of three-dimensional muscle geometry created from MRI data (Blemker et al., 2007) or by three-dimensional geometric modeling such as host-mesh fitting

Table 7

Number of coefficients per muscle, memory requirements in megabytes (MB) to store coefficients (Coefficient storage), and nominal values for coefficient calculation (Nominal data storage) both for polynomial regression equations (P^{eq}) and splines. The total memory to store both coefficients and nominal values is also reported (Total storage). For P^{eq} , these parameters vary with the number of generalized coordinates (GCs) and muscles. For splines they also vary with the number of nodes (Section 2). Because a different set of polynomial regression equations are generated depending on the number of GCs, the reported number of P^{eq} coefficients per muscle is relative to all coefficients from all polynomial regression equations within each set (Menegaldo et al., 2004). Data for tests 3 and 4 are the same as those for test 2 with 4 GCs.

	Test 1				Test 2			
GCs	1	2	3	4	4	5	6	
Muscles	5	6	27	6	9	9	9	
Nodes per muscle	20	20 ²	20 ³	15 ⁴	10 ⁴	10 ⁵	10 ⁶	
Polinomial Eq.								
Coefficients per muscle	18	60	116	195	195	300	434	
Coefficient storage (MB)	0.004	0.004	0.047	0.023	0.039	0.059	0.086	
Nominal data storage(MB)	0.008	0.085	9.385	17.32	3.36	200.6	482.8	
Total storage (MB)	0.012	0.089	9.432	17.343	3.399	200.7	482.9	
Spline								
Coefficients per muscle	23	529	12,167	104,976	28,516	371,293	4,826,809	
Coefficient storage (MB)	0.004	0.033	0.766	6.607	2.696	35.055	455.717	
Nominal data storage (MB)	0.002	0.027	2.227	3.079	0.6	29.9	61.8	
Total storage (MB)	0.006	0.06	2.993	9.686	3.296	64.955	517.52	

methods (Oberhofer et al., 2010). Future developments may relax the limitation of splines needing equally spaced nodes, using a generic spline basis as explained in (Habermann and Kindermann, 2007).

Finally, our method is coded in C++ with a MATLAB interface⁴ and is designed as a functional block that returns ℓ^{mt} and r as a function of GC-angles facilitating the integration into large and complex musculoskeletal models.

Conflict of interest statement

There were not conflicts of interest between these funding bodies and procedures and results in this study.

Acknowledgements

The authors would like to thank Dr Elena Ceseracciu for her contribution to the implementation of the C++/MATLAB interface.

Funding sources included the US National Institute of Health (R01EB009351-01A2), the Australian National Health and Medical Research Council, the Western Australian Medical and Health Research Infrastructure Council, and the CARIPARO foundation in Italy under Grant “Progetto Dottorati di Ricerca 2008–2010”.

Appendix A. Supplementary materials

Supplementary data associated with this article can be found in the online version at doi:10.1016/j.jbiomech.2011.10.040 and from <http://code.google.com/p/mcbs/>.

References

- Ackermann M., van den Bogert A.J., 2010. Predictive simulation of gait in rehabilitation. In: Proceedings of the 32nd Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Buenos Aires, Argentina.
- Blemker, S.S., Asakawa, D.S., Gold, G.E., Delp, S.L., 2007. Image-based musculoskeletal modeling: applications, advances, and future opportunities. *Journal of Magnetic Resonance Imaging* 25, 441–451.

- Buchanan, T.S., Lloyd, D.G., Manal, K., Besier, T.F., 2004. Neuromusculoskeletal modeling: estimation of muscle forces and joint moments and movements from measurements of neural command. *Journal of Applied Biomechanics* 20, 367–395.
- Delp, S.L., Anderson, F.C., Arnold, A.S., Loan, P., Habib, A., John, C.T., Guendelman, E., Thelen, D.G., 2007. OpenSim: open-source software to create and analyze dynamic simulations of movement. *IEEE Transactions on Biomedical Engineering* 54, 1940–1950.
- Delp, S.L., Loan, J.P., Hoy, M.G., Zajac, F.E., Topp, E.L., Rosen, J.M., 1990. An interactive graphics-based model of the lower extremity to study orthopaedic surgical procedures. *IEEE Transactions on Biomedical Engineering* 37, 757–767.
- Fleischer, C., Hommel, G., 2008. A human-exoskeleton interface utilizing electromyography. *IEEE Transactions on Robotics* 24, 827–882.
- Gao, F., Damsgaard, M., Rasmussen, J., Christensen, S.T., 2002. Computational method for muscle-path representation in musculoskeletal models. *Biological Cybernetics* 87, 199–210.
- Garner, B.A., Pandey, M.G., 2000. The obstacle-set method for representing muscle paths in musculoskeletal models. *Computer Methods in Biomechanics and Biomedical Engineering* 3, 1–30.
- Habermann, C., Kindermann, F., 2007. Multidimensional spline interpolation: theory and applications. *Computational Economics* 30, 153–166.
- Kuo, A.D., 1998. A least-squares estimation approach to improving the precision of inverse dynamics computations. *Journal of Biomechanical Engineering-Transactions of the ASME* 120, 148–159.
- Lloyd, D.G., Besier, T.F., 2003. An EMG-driven musculoskeletal model to estimate muscle forces and knee joint moments in vivo. *Journal of Biomechanics* 36, 765–776.
- McLean, S.G., Huang, X., van den Bogert, A.J., 2005. Association between lower extremity posture at contact and peak knee valgus moment during side-stepping: implications for ACL injury. *Clinical Biomechanics* 20, 863–870.
- Menegaldo, L.L., de Toledo Fleury, A., Weber, H.L., 2004. Moment arms and musculotendon lengths estimation for a three-dimensional lower-limb model. *Journal of Biomechanics* 37, 1447–1453.
- Menegaldo, L.L., de Toledo Fleury, A., Weber, H.L., 2006. A ‘cheap’ optimal control approach to estimate muscle forces in musculoskeletal systems. *Journal of Biomechanics* 39, 1787–1795.
- Miller, L.H., 1956. Table of percentage points of Kolmogorov statistics. *Journal of the American Statistical Association* 51, 111–121.
- Murai, A., Kurosaki, K., Yamane, K., Nakamura, Y., 2010. Musculoskeletal-see-through mirror: computational modeling and algorithm for whole-body muscle activity visualization in real time. *Progress in Biophysics and Molecular Biology* 103, 310–317.
- Nordez, A., Gallot, T., Catheline, S., Guevel, A., Cornu, C., Hug, F., 2009. Electromechanical delay revisited using very high frame rate ultrasound. *Journal of Applied Physiology* 106, 1970–1975.
- Oberhofer, K., Stott, N.S., Mithraratne, K., Anderson, I.A., 2010. Subject-specific modelling of lower limb muscles in children with cerebral palsy. *Clinical Biomechanics* 25, 88–94.
- Scott, S.H., Winter, D.A., 1991. A comparison of three muscle pennation assumptions and their effect on isometric and isotonic force. *Journal of Biomechanics* 24, 163–167.
- Winby, C.R., Lloyd, D.G., Besier, T.F., Kirk, T.B., 2009. Muscle and external load contribution to knee joint contact loads during normal gait. *Journal of Biomechanics* 42, 2294–2300.
- Zajac, F.E., 1989. Muscle and tendon: properties, models, scaling, and application to biomechanics and motor control. *Critical Reviews in Biomedical Engineering* 17, 359–411.

⁴ Spline and polynomial regression code, C++/MATLAB interface, experimental data, and documentation is available as Supplementary Web Material from the Journal of Biomechanics web site.