

**Fast Operation of Anatomical and Stiff Tendon Neuromuscular Models in EMG-driven Modeling**

**Author**

Sartori, M, Lloyd, DG, Reggiani, M, Pagello, E

**Published**

2010

**Conference Title**

Proceedings - IEEE International Conference on Robotics and Automation

**DOI**

[10.1109/ROBOT.2010.5509932](https://doi.org/10.1109/ROBOT.2010.5509932)

**Rights statement**

© 2010 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.

**Downloaded from**

<http://hdl.handle.net/10072/40078>

**Griffith Research Online**

<https://research-repository.griffith.edu.au>

# Fast Operation of Anatomical and Stiff Tendon Neuromuscular Models in EMG-driven Modeling

Massimo Sartori, David G. Lloyd, Monica Reggiani, and Enrico Pagello

**Abstract**—The inclusion of robotic systems in physiotherapy allows developing new solutions for the rehabilitation and support of disabled people. Our research addresses the core problem for the advancement of such applications: the availability of a human machine interface offering intuitive control of robotic devices. In this paper we present an EMG-driven model of the human lower limb based on that previously developed by Lloyd et al [1]. We then introduce a set of enhancements that allow reducing time and memory requirements and provide real-time performances for the control of a lower limb powered orthosis.

## I. INTRODUCTION

The role of technology in the health sector has been growing increasingly in the past years as well as the capability and the power of the systems involved. People suffering from a debilitating disease can now benefit from the recent progresses in healthcare technologies but many issues are still open. Current motor rehabilitation therapies still cannot give back the patients their lost motor capabilities within a short timeframe or at an affordable cost. The possibility of integrating robot-assisted treatments in the physiotherapy could help decrease the length of the recovery process as well as the cost involved. Researchers are currently developing models of the human body that can simulate the motion of the human limbs and help design better assistive robotic devices for people with disabilities. Indeed, the availability of accurate and comprehensive human limbs models is required for the development of effective human-robot interfaces (HRI) and control systems for rehabilitation robotic devices such as powered orthoses. Biomechanists have developed complex models of the human limbs that combine together kinematic and kinetic data with neural signals to study human motor control strategies [1], [2], [3], [4]. These models implements several complex functions into separate *computing blocks*. Due to their complexity, these blocks are not integrated into a single framework and do not execute simultaneously. Rather, they execute separately at different times. Only at a second stage, results from different computing blocks are combined together and analyzed. This considerably slows down the data investigation process and bounds researchers to perform offline analysis only. While this does not represent a problem when using biomechanical

models for pure analysis purposes, it clearly does when applied in the field of robotics. The high complexity of each computing block and their asynchronous operations would prevent the biomechanical model real-time execution. Only recently, robotics researchers have developed models that are suitable for real-time applications [5], [6], [7], [8], [9]. In [5], the authors present a model of the human upper limb that is able to predict elbow joint torque as a function of the joint kinematics and muscle neural activation level. The controlled device is an arm exoskeleton in which the user straps into and performs exercises imparted by the machine. Another effective research is presented in [6]. Muscles electromyographic (EMG) signals are used as input for a simplified EMG-driven model of the human lower extremity that is able to derive the knee flexion-extension (FE) torque produced by the user. The estimated torque is used to control a powered orthosis that can provide extra support to the operator when needed. These works utilize simplified models of the human limbs that allow achieving real-time performances. However, it is not clear the accuracy of these models and how the applied simplifications affect their accuracy with respect to complete original models developed by biomechanists. In general, the degree of simplification often makes them unsuitable for being integrated into the control system of assistive devices that are to support a wide range of movements. The availability of accurate and comprehensive human limbs models, combining high reliability and real-time operation, is therefore needed for the development of effective human-robot interfaces (HRI) and control systems for rehabilitation robotic devices. In this paper we discuss the developmental work that has been done towards the design of an EMG-driven model of the human lower extremity that combines together comprehensiveness, accuracy, and simultaneous real-time operation of all its computing blocks. Our model processes EMG signals to estimate muscle force and knee joint FE torque. The model is initially derived from the well-established one developed by Lloyd et al. [1], [2], [10] which, however, requires large memory space and high computation time. Our goal is twofold. First, we want to reduce the model computation time and memory requirements while preserving high estimation accuracy. Second, we want to combine together the individual model's computing blocks to execute simultaneously in a common framework. To achieve the first goal we have investigated the impact of modeling the tendon as an infinitely stiff body. This avoided the integration of the musculotendon force differential equations, greatly reducing computation time. We directly compared the behavior of the modified model to that of the original

M. Sartori and E. Pagello are with the Department of Information Engineering, University of Padova and with the Institute of Biomedical Engineering - National Research Council, Padova, Italy. [massimo.sartori@dei.unipd.it](mailto:massimo.sartori@dei.unipd.it),

M. Reggiani is with the Department of Technique and Management, University of Padova, Vicenza, Italy.

D. G. Lloyd is with the School of Sport Science Exercise and Health, University of Western Australia, Perth, Australia.

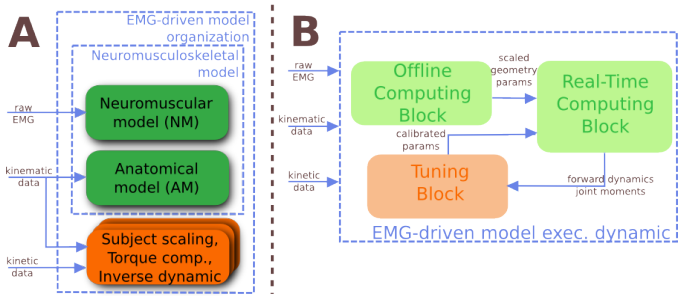


Fig. 1. (A) High-level representation of the EMG-driven model computing blocks. (B) Classification of the computing blocks relative to the way they execute. Green boxes represent the computing blocks that have been enhanced. Orange boxes refer to those that have not been modified.

one to quantitatively evaluate the effects of this change. We also designed a novel approach for computing and storing musculotendon estimates that greatly reduced the required memory space. To achieve the second goal we designed a new anatomical model that allowed integrating all EMG-driven model's computing blocks into a common framework and executing them in real-time. In the previous version the anatomical model was executed offline through external programs breaking the EMG-driven model's execution into different steps. The paper is organized as follows. Section II briefly overviews Lloyd's EMG-driven model as a baseline for our research. In section III we discuss the changes we applied to the model to achieve our goals. Section IV presents the experimental procedure adopted in this research while in V and VI, results about accuracy and real-time performances are analyzed. Section VII suggests how our enhanced EMG-driven model can be used to control a lower limb powered orthosis whereas Section VIII concludes the paper and sketches future research directions.

## II. EMG-DRIVEN MODEL STRUCTURE

Our EMG-driven model includes several computing blocks (Fig. 1-A) that can be grouped in *offline*, *real-time* and *tuning* computing blocks, depending on the way they execute (Fig 1-B). The original model by Lloyd et al. has a greater number of computing blocks working offline (Fig.2-A) [1], [2], [10], [11] and is briefly reviewed in the following of this section.

### A. Offline Computing Blocks

1) *Subject Scaling Model*: This block scales the mass and the dimension of muscles and bones in the model to those of the subject being studied. The scaling procedure is based on data obtained from static poses. In this process some muscle-specific parameters are also tuned.

2) *Anatomical Model*: The *anatomical model* (AM) describes the organization of the muscular and skeletal systems and defines the way muscles and bones interact towards the generation of joint motions. The anatomical model is fed with joint angular positions and generates instantaneous estimation of musculotendon lengths  $l^{mt}$ , velocities  $v^{mt}$ , and moment arms  $r$ . These estimates are the input for the

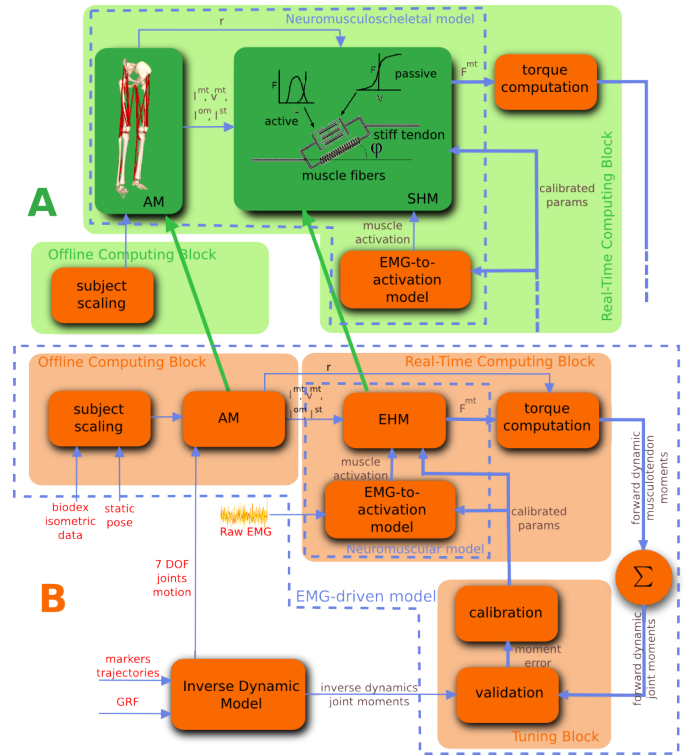


Fig. 2. (A) View of the modified EMG-driven model in which the entire neuromusculoskeletal model can now be executed in real-time. (B) Schematic view of the blocks of the original version of the model and the way they are executed. The neuromusculoskeletal model is not entirely executed in real-time as the anatomical model has to be run offline in a separate step.

subsequent blocks (Fig. 2). The AM is implemented using the SIMM Biomechanics Software Suite (Musculographics Inc., US) based on the results presented in [12], [11]. The anatomical model consists of line segment representations of the following 13 musculotendon units (MTUs) spanning the knee joint: semimembranosus (semimem), semitendinosus (semiten), biceps femoris long head (bifemlh), biceps femoris short head (bifemsh), sartorius (sar), tensor fascia latae (tfl), gracilis (gra), vastus lateralis (vaslat), vastus medialis (vasmed), vastus intermedius (vasint), rectus femoris (rf), medial gastrocnemius (medgas), and lateral gastrocnemius (latgas). The lengths of the modeled MTUs are linearly scaled to the actual subject's size according to the scaling factors computed by the subject scaling model. The AM also defines 7 degrees of freedom (DOF). The hip has 3 DOFs: flexion-extension ( $\theta_{h,1} \in [-20; 95]$ ), abduction-adduction ( $\theta_{h,2} \in [-50; 30]$ ), and internal-external rotation ( $\theta_{h,3} \in [-30; 20]$ ). The knee is modeled with only 1 DOF: flexion-extension ( $\theta_{k,1} \in [-10; 120]$ ). The final 3 DOFs are at the ankle joint: plantar-dorsi flexion ( $\theta_{a,1} \in [-80; 33]$ ), abduction-adduction ( $\theta_{a,2} \in [-20; 20]$ ), and internal-external rotation ( $\theta_{a,3} \in [-20; 20]$ ). The DOF are expressed in degrees and it is important to account for all of them as  $l^{mt}$ ,  $v^{mt}$ , and  $r$  highly vary with respect to the angular joint position.

## B. Real-Time Computing Blocks

Real-time computing blocks include the *neuromuscular* and the *torque computation* models. As shown in Fig. 2-A the neuromuscular model is composed of the *EMG-to-activation* and the *Hill-type muscle (EHM)* models.

1) *EMG-to-activation Model*: Raw EMG signals are input to this model and are converted into muscle activation. This is a percentage value that expresses how much a specific muscle is activated toward the generation of force. The transformation from EMG to activation accounts for the non-linear relationship between EMG and muscle force [2].

2) *Hill-type Muscle Model*: It calculates the tension developed by each musculotendon unit in the model [2]. The model assumes that each muscle consists of an active force generating component in series with a passive elastic one. The *force generating component* models the muscle fiber behavior. It is widely accepted that the muscle fiber force depends on three factors:  $f_A(\tilde{l}^m)$ ,  $f_P(\tilde{l}^m)$ , and  $f_V(v^m)$ . The first one,  $f_A(\tilde{l}^m)$ , is the *active force-length function* that expresses the ability of muscle fibers to produce tension at different lengths. The second factor,  $f_P(\tilde{l}^m)$ , is the *passive force-length function* that represents the passive element of the fiber producing resistive force when stretched. It relates the instantaneous amount of fiber length ( $l^m$ ) to the force produced in the muscle. The final factor,  $f_V(v^m)$ , is the *force-velocity function* that expresses the influence of the fiber contraction velocity ( $v^m$ ) on the muscle fiber force production. Fiber length ( $l^m$ ) and velocity ( $v^m$ ) are derived from the musculotendon length ( $l^{mt}$ ) and velocity ( $v^{mt}$ ) output of the AM block. The *elastic component* of the musculotendon unit refers to the tendon. This is modeled to act like a rubber band. As the muscle fiber force grows, the tendon gets stretched. When the tendon resting length is exceeded, the tendon produces a resistive force that is non-linearly related to the amount of tendon strain. Muscle fiber force and elastic tendon force are then combined together with muscle activation level computed by the EMG-to-activation model to estimate the total force produced by the musculotendon unit. Section III provides additional details about this component.

3) *Torque Computation*: The net knee joint flexion-extension moment is finally computed as the sum of the product of each muscle's force by its flexion-extension moment arm ( $r$ ).

## C. Tuning Components

Methodological difficulties in measuring individual muscle force prevent any direct validation of the NM model on human. Therefore, an indirect validation process was designed to both validate the model's prediction ability and to support its calibration.

1) *Validation Model*: To validate the EMG-driven model, the estimated knee joint flexion-extension (FE) moment, that is the output of the torque computation model, is compared to the net FE moment experimentally measured through an inverse dynamic model. Indeed, when the model is able to accurately estimate the muscle forces, the computed knee FE

moment should be equal to the measured one in the whole range of joint values [1], [2].

2) *Calibration Model*: The aim of the calibration is to define the values of a set of EMG-driven model parameters w.r.t. the subject's characteristics for a proper estimation of the FE knee moment. Starting from a set of calibration trials, a simulated annealing process alters the selected parameters to minimize the sum of squared differences between the net knee FE moment approximated by the model and that experimentally measured [1].

## III. REAL-TIME NEUROMUSCULOSKELETAL MODEL

To achieve real-time execution of the EMG-driven model we designed more efficient algorithms that allowed integrating all computing blocks into a common framework and decreasing the required memory capacity.

### A. Real-Time Neuromuscular Model

To reduce the computation time required to estimate the tension developed by each musculotendon unit, we replaced the elastic tendon in the Hill-type muscle model with a stiff one (Fig. 2). The force produced by the musculotendon unit only depends now on the action of the muscle fibers. We introduced this simplification under the hypothesis it does not affect the model prediction ability. Indeed, according to [4] the tendon is rather stiff. The strain, i.e. the percentage of length variation w.r.t. the resting length, is only the 3.3% of the tendon length when the muscle generates the maximum isometric force. In the following of this section we will refer to this model as to the Stiff tendon Hill-type Muscle (SHM) model and to the previous one as to the Elastic tendon Hill-type Muscle (EHM) model (Sec. II-B.2).

The force produced by the muscle fibers alone, is expressed in the SHM model in the same way it is expressed in the EHM. This can be written as follows:

$$F^m = (f_A(\tilde{l}^m) \cdot f_V(v^m) \cdot a(u) + f_P(\tilde{l}^m) + d^m \cdot \tilde{v}^m) \cdot F_0^m \cdot \delta \quad (1)$$

where  $d^m$  represents a passive parallel damping element that was added to the force-velocity relationship to account for the muscles damping characteristics as suggested in [3]. The term  $a(u)$  is the muscle activation as a function of the post processed EMG signal  $u$ . The term  $F_0^m$  expresses the force generated by the muscle when fibers are at their optimal length, while the term  $\delta$  is the muscle strength coefficients. The force produced by the musculotendon unit,  $F^{mt}$  can then be derived by projecting the fiber force  $F^m$  onto the tendon line of action as follows:  $F^{mt} = F^m \cdot \cos(\varphi)$ , where the term  $\varphi$  is the pennation angle. This is the angle at which fibers are oriented with respect to the tendon (Fig. 2-B). The computation of  $F^m$  and  $F^{mt}$  can then be done when the instant values of  $l^m$  and  $v^m$  are known at all times. The method used to calculate these values is the main difference between the SHM and the EHM model. The SHM model is designed so that the tendon does not produce any resistive force to stretch. This means the projection of  $F^m$  onto the tendon line of action does not affect its length in any way. The tendon length ( $l^t$ ) is therefore constant regardless of

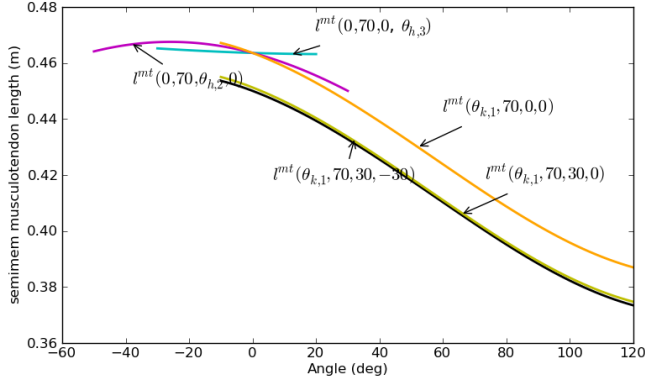


Fig. 3.  $\theta_{h,2}$  and  $\theta_{h,3}$  range from  $0^\circ$  to  $30^\circ$  and  $-30^\circ$  respectively, while  $\theta_{h,1} = 70^\circ$ . The curves expressed by  $l^{mt}(\theta_{k,1}, 70, 0, 0)$ ,  $l^{mt}(\theta_{k,1}, 70, 30, 0)$  and  $l^{mt}(\theta_{k,1}, 70, 30, -30)$  differ just by a constant value  $\forall \theta_{k,1}$ .  $\forall \theta_{h,2}, \theta_{h,3}$  the curves  $l^{mt}(0, 70, \theta_{h,2}, 0)$  and  $l^{mt}(0, 70, 0, \theta_{h,3})$  define the amount of shift applied to  $l^{mt}(\theta_{k,1}, 70, 0, 0)$  according to (3) and (4).

the force projected onto the tendon and is always equal to the resting tendon length  $l_s^t$ . The assumption that  $l^t = l_s^t$ , provides a closed-form expression that allows for a very fast computation of the muscle fiber length  $l^m$ :

$$l^m = \sqrt{(l_0^m \cdot \sin(\varphi_0))^2 + (l^{mt} - l^t)^2} \quad (2)$$

where  $\varphi_0$  is the pennation angle at optimal fiber length. The fiber contraction velocity  $v^m$  can then be found by differentiation. The calculations involved in the EHM model are more complex. Since both  $l^m$  and  $l^t$  are unknown, (2) cannot be solved. When the tendon is defined as an elastic body, the musculotendon unit dynamics is well described by an ordinary differential equation (ODE) that has to be solved at every time step to estimate both  $l^m$  and  $l^t$ . This involves the execution of several computation steps including: the estimation of the current musculotendon velocity, the apportion of it to the muscle fibers and tendon (based on their relative stiffness), and the integration of the derived fiber and tendon contraction velocities. When several samples of  $l^m$  need to be estimated, the numerical integration of the ODE considerably increases the computation time involved. As shown in Sec. VI, SHM has better time performances with respect to the EHM.

### B. Real-Time Anatomical Model

The AM is a complex system as it requires 7 DOFs and 13 MTUs to correctly estimate the musculotendon lengths ( $l^{mt}$ ), velocities ( $v^{mt}$ ), and moment arms ( $r$ ). When so many DOFs and muscles are involved, a direct implementation of the musculoskeletal organization and interaction between bones and muscles would result in complex algorithms that would be costly in terms of time. To overcome this difficulty, we adopted the following strategy. Using SIMM, it is possible to precompute subject-specific values of  $l^{mt}$  at discrete joint configuration steps and store them in a multidimensional table for each muscle. A multidimensional cubic spline

can then be used to interpolate the values within discrete intervals. At run time, the tables are indexed based on the current joint positions. The dimension of the tables and interpolating splines depend on the number of the muscle's DOFs. Uniarticular muscles crossing the knee have one DOF only. Therefore a simple 1-dimensional table (i.e. an array) can store the required data. Biarticular thigh and shank muscles cross the knee as well as the hip or the ankle respectively. Such muscles have therefore 4 DOF. In this specific case a 4-dimensional table is required per muscle as well as a 4-dimensional cubic spline to interpolate the data. If the range of motion of the limb about each DOF is sampled with intervals of 2 degrees, the length of a MTU crossing the hip and the knee joint could assume up to  $n = 4.080.648$  different values. Furthermore, in order to evaluate the spline function several times on the interpolation data, one has to calculate  $4 \cdot n$  coefficients and store them permanently [13]. With 13 muscles, one would therefore need to store  $13 \cdot n$  data in tables and compute  $13 \cdot 4 \cdot n$  spline coefficients which may be costly in terms of time and space. An analogous analysis can be made about muscles crossing both knee and ankle. In order to relax these space and time constraints we separately considered the variation of three DOFs with respect to the forth. In other words, we created three different 2-dimensional tables per muscle instead of a single 4-dimensional table. The first table stores the data relative to the variation of  $(\theta_{x,1}, \theta_{x,4})$ , the second stores the data relative to  $(\theta_{x,2}, \theta_{x,4})$  and the third stores the data relative to  $(\theta_{x,3}, \theta_{x,4})$ , where  $\theta_{x,i}$  is the  $i^{th}$  DOF about the  $x$  joint. In the case of biarticular thigh muscles for instance, the size of the data that have to be stored in all three tables per muscles, is  $m = 7.714$  and the total number of spline coefficient is  $4m$  which is nearly 530 times smaller than the data stored in the 4-D table. The reason why three 2-D tables per muscle are enough to properly estimate musculotendon length and moment arms without loss of accuracy w.r.t. the complete 4-D table, is here illustrated. Let us fix two arbitrary DOF to constant values, e.g.  $\theta_{h,1} = h_1$  and  $\theta_{h,2} = h_2$ . Let us then compute the values assumed by  $l^{mt}$  first when  $\theta_{h,3} = \theta_a$  and then  $\theta_{h,3} = \theta_b$ , with  $\theta_a \neq \theta_b$  and  $\forall \theta_{h,1}$ . We then obtain two functions  $l_a^{mt}(\theta_{k,1})$  and  $l_b^{mt}(\theta_{k,1})$  for which it holds that  $l_a^{mt}(\theta_{k,1}) - l_b^{mt}(\theta_{k,1}) \approx K, \forall \theta_{k,1}$ . The term  $K$  represents a constant value. In other words,  $l^{mt}(\theta_{k,1})$  is linearly shifted when  $\theta_{h,3}$  changes from  $\theta_a$  to  $\theta_b$  and  $\theta_{h,1}$  and  $\theta_{h,2}$  are fixed. The same transformations apply to any choice of constant DOFs (Fig. 3). The following elementary relationship therefore holds:

$$l^{mt}(k_1, h_1, h_2, 0) = l^{mt}(k_1, h_1, 0, 0) + s(h_2). \quad (3)$$

where  $s(h_2) = l^{mt}(k_1, h_1, h_2, 0) - l^{mt}(k_1, h_1, 0, 0)$ . Furthermore, from our previous discussion, the following relation also holds:  $l^{mt}(\theta_{k,1}, h_1, a, 0) - l^{mt}(\theta_{k,1}, h_1, b, 0) \approx K, \forall \theta_{k,1}$ . This implies that  $s(h_2)$  in (3) can be replaced by:

$$s(h_2) = l^{mt}(0, h_1, h_2, 0) - l^{mt}(0, h_1, 0, 0). \quad (4)$$

An analogous transformation can be applied to compute  $l^{mt}(k_1, h_1, h_2, h_3)$  where  $l^{mt}(k_1, h_1, h_2, 0)$  is shifted by:

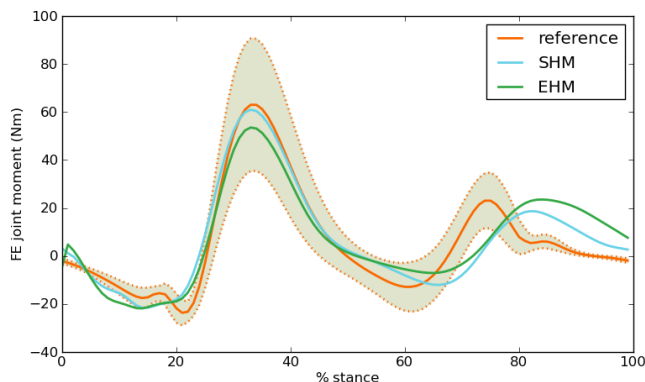


Fig. 4. The ensemble average of the reference FE knee moment is compared to the that estimated by the SHM and to that estimated by the EHM.

$s(h3) = l^{mt}(0, h1, h3, 0) - l^{mt}(0, h1, 0, 0)$ . The terms  $l^{mt}(k1, h1, 0, 0)$ ,  $l^{mt}(0, h1, h2, 0)$  and  $l^{mt}(0, h1, h3, 0)$  needed to compute  $l^{mt}(k1, h1, h2, h3)$  can be easily stored in a 2-dimensional tables including the values of MTU length relative to the combined variation of  $(\theta_{k,1}, \theta_{h,1})$ ,  $(\theta_{h,1}, \theta_{h,2})$  and  $(\theta_{h,1}, \theta_{h,3})$  respectively. The transformations previously described can be applied to the muscles crossing the knee and the ankle therefore the actual formalization is omitted. The moment arms along all DOF don't need to be stored in tables but can be directly derived by differentiating the interpolated musculotendon length with respect to each DOF angular position thus saving further memory space.

#### IV. EXPERIMENTAL PROCEDURE

Experiments were performed at the Gait Laboratory of the School of Sport Science Exercise and Health of the University of Western Australia. Tests involved capturing *anatomical static poses* of 6 male subjects (mass =  $81.73 \pm 16.4Kg$ , height =  $175.75 \pm 6.39cm$ , age =  $30.67 \pm 7.84$ years). A 7 camera motion capture system (Vicon, Oxford, UK) was used.

*EMG signals* were collected from the selected muscles using double-differential surface electrodes. A telemetered system (Noraxon, Scottsdale, USA) was used to transfer the EMG signals to a 16 channel amplifier (Delsys, Boston, USA) with sampling frequency at  $2kHz$ . EMG were collected during isometric and gait trials. Isometric trials were performed using a dynamometer (Biodex, New York, USA). Gait trials included the acquisition of *ground reaction forces* and *joints kinematics*, as well. Different sets of data were then created. The first was used to calibrate the EMG-driven model (Sec. II-C) and it included isometric and gait trials. The second data set comprised gait trials used for model validation purposes only (Sec. V). For each subject a minimum of 10 trials were recorded. The dataset comprised a total of 75 trials.

#### A. Neuromusculoskeletal Model

The model's prediction ability is evaluated through a direct comparison of the estimated joint torque with the reference data measured with the inverse dynamic model. Additionally, to quantify the impact on the model's behavior when the tendon is assumed infinitely stiff, the SHM's predicted torque was compared to that predicted by the EHM. All trials from all subjects were time normalized using a natural cubic spline. This allowed comparing *in-between* trials. A statistical analysis on the normalized dataset resulting from the entire population was then applied. A brief description of the procedure follows. The SHM was used to predict FE knee moments during *stance*. The stance is defined as the period of time between the *heel-strike* and *toe-off* events and it represents the interval during which the foot is in contact with the ground floor. The average of the FE knee torque estimated by the SHM over all subjects' trials was first computed. This curve expresses how good the model is at estimating the overall FE knee torque. Similarly, the average of the FE knee torque for all subject's trials measured by the inverse dynamic system was calculated. Fig. 4 shows that the average of the estimated FE knee torque very closely replicates the behavior of the average of the reference knee torque. The joint moment's peaks for both curves occur at the same time during the stance phase. A small discrepancy is noticeable at the edges of the graph. This is due to filtering issues. For each time step, filter algorithms need to process some data over a predefined time window. At the edges of the data time series the window is not full and therefore approximation errors are introduced. This, however, is not an SHM issue. The Pearson correlation coefficient  $R$  is 0.932. This means that the model generates estimates that are highly correlated to reference values over all subjects' trials. The root mean squared error  $RMSE$  is  $7.898Nm$ . This means the model introduces an error that, on average, is less than the 10% of the range of variation observed in the reference values (i.e.  $86.673Nm$ ). The maximum absolute error ( $e_{max}$ ) is equal to  $21.494Nm$ . It occurs at 73% of the stand phase and is located in the edge region where the observed data is least significant. The minimum absolute error ( $e_{min}$ ) is equal to  $0.114Nm$  and occurs at 47%. It is located in the peak region. This proves that the model is capable of providing high performances within the most significant part of the observed data. Fig. 4 also shows the predictive ability of the EHM that is found to be slightly lower on average than that provided by the SHM. Statistical coefficients assume poorer values with respect to the SHM ones:  $R = 0.893$ ,  $RMSE = 9.714Nm$ . Minimum and maximum absolute errors provide, however, slightly better results:  $e_{max} = 19.031Nm$  and  $e_{min} = 0.012Nm$ . The analyzed data shows the SHM improved the estimation ability of the whole EMG-driven model. The lower EHM performances can be related to the fact that the model does not always correctly estimates the amount of stiffness in the musculotendon unit. This leads to incorrect estimations of

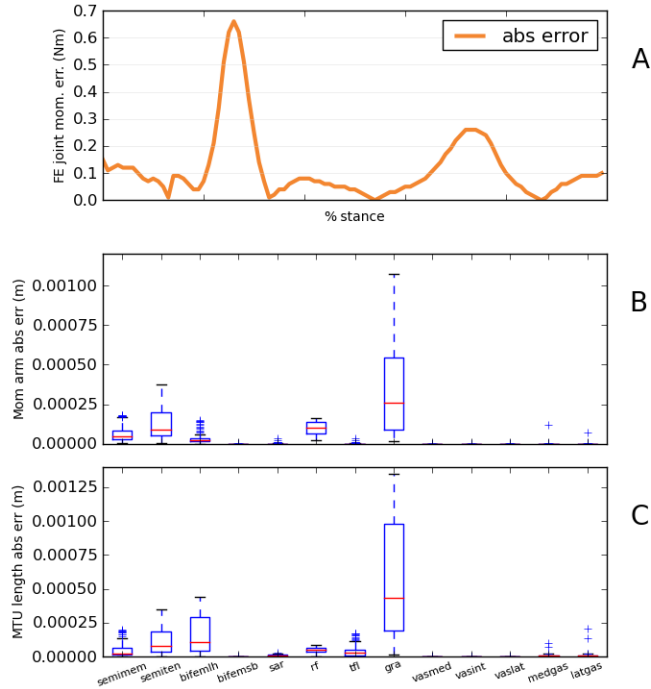


Fig. 5. (A) Instantaneous absolute error between the FE knee moments obtained by using MTU estimates computed with SIMM and with the RTAM. Range of variation of the instantaneous absolute errors between MTU lengths (B) and moment arms (C) computed by SIMM and RTAM.

the muscle fiber length that results in inaccurate muscle force and joint torque predictions. On the contrary, the SHM does not rely on the tendon stiffness to compute the muscle fiber length. Furthermore, although the tendon length is constant in the SHM, its actual variation in the human MTU is small anyway. The SHM is therefore more robust during gait as it is shown in Fig. 4. Further studies have to be carried out to assess the prediction ability of both SHM and EHM on a wider range of movements including squats in which the EHM may provide better results.

### B. Real-Time Anatomical Model

As described in III-B, the real-time anatomical model (RTAM) uses a set of subject-specific tables to compute MTU estimates. The RTAM accuracy fully depends on how *descriptive* the numbers in the 2-D tables are. Since the tables are generated by using SIMM, the same level of descriptiveness is assured for each subject and therefore the accuracy analysis can be made on one individual only. For the selected subject, the reported experimental results were relative to 20 gait trials. To assess the accuracy of the real-time anatomical model (RTAM), estimates of MTU moment arms and lengths computed by the RTAM were compared to those computed with SIMM (Fig. 5-B and C). Furthermore, the FE knee joint torque computed using MTU estimates produced by SIMM was compared to the FE knee joint torque computed using MTU estimates produced by the

RTAM (Fig. 5-A). The data were first time normalized to allow an in-between trial analysis. Then, the instantaneous absolute error between values computed by SIMM and by the RTAM were calculated. For each muscle, the absolute errors were then averaged over all trials. Fig. 5-A shows how the estimation of FE knee joint torque changes when using RTAM rather than SIMM. The joint torque estimated using RTAM differs from that estimated using SIMM of  $0.119Nm$ , on average. The maximum and the minimum discrepancy between the two curves are  $0.657Nm$  and  $0.001Nm$ , respectively. Therefore RTAM does not have negative impact on the EMG-driven model's capability of producing FE knee torque estimates. Fig. 5-B and C show how the estimation of musculotendon moment arm and length change when using RTAM rather than SIMM. Uniarticular muscles have no errors at all since the anatomical model tables directly store all MTU values computed by SIMM for the whole range of motion. Biarticular muscles, on the other hand, have higher errors due to the approximations introduced by (3) and (4). However, these errors are extremely small as their range of variation is within a millimeter for all muscles. The RTAM therefore generates estimations that are quantitatively identical to the ones generated by SIMM.

## VI. PERFORMANCE ANALYSIS

To verify whether the enhanced EMG-driven model can be used to control powered orthoses with low computation resources, we tested the model performances on an embedded system (Digital Logic, Switzerland) equipped with a  $1.6Ghz$  Atom CPU with  $512MB$  RAM.

### A. Stiff Tendon Hill-type Muscle Model

Both the SHM and EHM models were run on each trial for each subject. Trial execution time was averaged yielding to an average execution time per subject. Data are summarized in table I. A significant improvement in computation time is achieved when using the SHM. While EHM average execution time is longer than the trial duration the SHM takes just a little fraction of the trial time. This leaves enough time to execute the other blocks of the whole neuromusculoskeletal model at runtime.

### B. Real-Time Anatomical Model

The same subject in V-B was used to assess the execution time of the RTAM. The average time required to estimate MTU lengths and moment arms over all trials is  $0.1185s$  while the average trial duration is  $1.3324s$ . The RTAM, only needs to access the 2-D tables, read the MTU estimates and perform the linear shift described in (3) and (4). All tables and interpolating spline coefficients are precomputed offline. The memory space required to store all tables and spline coefficients is  $1.44MB$ . This allows using the RTAM even on systems with very limited hardware resources.

### C. EMG-driven Model Real-Time Analysis

The average time needed by the EMG-driven model to execute each computing block follows: 1) Input data reading:

TABLE I  
COMPARISON PERFORMANCES BETWEEN SHM AND LHM

Subject	1	2	3	4	5	6
SHM(s)	0.002	0.001	0.002	0.001	0.002	0.002
LHM(s)	1.232	0.897	1.268	0.604	1.068	1.306
Duration(s)	1.015	0.803	1.161	0.403	1.224	1.332

0.17697s, 2) RTAM: 0.1185s 3) EMG-to-activation model: 0.0028s, 4) SHM and torque computation model: 0.00199s. The total EMG-driven model's execution time was therefore 0.3003s while the actual average trial duration was 1.3324. The total number of torque samples computed within the trial was 67. To assure real-time execution, the EMG-driven model has to compute the torque samples within a *reasonable* amount of time, which varies depending on the application. The control of a torque-driven powered orthosis improves its performance as the torque sample computation rate increases. To provide a comfortable control interface to the user, the orthosis needs to actuate itself by the time the operator's muscles contract. To achieve this, the EMG-driven model has to compute each torque sample within the muscle *electromechanical delay* (EMD). This is the interval between the moment the neural system generates EMG signals in the muscles and the moment when the muscles contract mechanically and generate moments around the joints. In lower limb muscles, EMD is about 15ms [5]. From the data above reported, our EMG-driven model took 4.58ms to compute each torque sample. This makes the model suitable to cope with the real-time control of powered orthoses. We also assessed the time needed to calibrate the model (Sec. II-C) and compared it with that needed by the original model. While SHM was calibrated in 158.84s it took 4 hours and 48 minutes for EHM calibration.

## VII. CONTROL OF A POWERED ORTHOSIS

As shown in [5], [6], [8], [9], an EMG-driven model can be effectively used to control a lower limb powered orthosis to enhance disabled people's mobility. An orthosis control system can use our EMG-driven model to estimate the force the user's muscles can produce and the resulting *user's torque* produced at the knee joint. When muscles forces are not *strong* enough to allow a proper joint movement, they can be amplified to produce the *target torque* at the knee. This is the torque the subject should produce to properly execute the desired movement. The powered orthosis controller can then be fed with the difference between the target torque and the *current torque* produced by the user wearing the powered orthosis. This difference defines the amount of extra torque the orthosis actuation system should add to obtain the target motion. The muscle amplification factor that leads to the target torque can be initially defined by a medical doctor who had previously analyzed the patient. However, automated procedures for the calibration of this parameter can be implemented.

## VIII. CONCLUSIONS AND FUTURE WORK

The objective of this research was to demonstrate to which extent accuracy had to be sacrificed in EMG-driven modeling to cope with real-time requirements. We found out that assuming the tendon infinitely stiff does not affect the ability of the model to predict muscle force and joint moment during gait but significantly reduce processing time. We then presented a new method for interpolating 4-dimensional musculotendon estimates by using 2-dimensional cubic splines. This allowed significantly relaxing the memory requirements and processing the musculotendon estimates at runtime. Our experimental results proved our enhanced EMG-driven model can generate control signal at an adequate rate for the control of a lower limb orthosis and can be used on embedded systems with limited power capabilities. More importantly, the use of a well-established biomechanical model as a baseline for our research [1], [2], [10], will allow designing more effective and safety *man-in-the-loop* control strategies for robot-assisted applications. Future research will focus on the integration and testing of our enhanced EMG-driven model on an lower limb powered orthosis control unit.

## REFERENCES

- [1] D. G. Lloyd and T. F. Besier, "An EMG-driven musculoskeletal model to estimate muscle forces and knee joint moments in vivo," *J. of Biomech.*, vol. 36, no. 6, pp. 765–776, 2003.
- [2] T. Buchanan, D. G. Lloyd, K. Manal, and T. Besier, "Neuromusculoskeletal modeling: Estimation of muscle forces and joint moments and movements from measurements from neural command," *Journal of Applied Biomechanics*, vol. 20, pp. 367–395, 2004.
- [3] L. Schutte, "Using musculoskeletal models to explore strategies for improving performance in electrical stimulation-induced leg cycle ergometry." Ph.D. dissertation, Stanford University, 1992.
- [4] F. E. Zajac, "Muscle and tendon: properties, models, scaling, and application to biomechanics and motor control," *Crit. Rev. Biomed. Eng.*, vol. 17, no. 4, pp. 359–411, 1989.
- [5] E. Cavallaro, J. Rosen, J. Perry, and S. Burns, "Real-time myoprocessors for a neural controlled powered exoskeleton arm," *Biomedical Engineering, IEEE Trans. on*, vol. 53, no. 11, pp. 2387–2396, 2006.
- [6] C. Fleischer and G. Hommel, "A human-exoskeleton interface utilizing electromyography," *Robotics, IEEE Trans. on*, vol. 24, no. 4, pp. 872–882, 2008.
- [7] A. Dollar and H. Herr, "Lower extremity exoskeletons and active orthoses: Challenges and state-of-the-art," *Biomedical Engineering, IEEE Transactions on*, vol. 24, pp. 144–158, 2008.
- [8] D. P. Ferris, K. E. Gordon, G. S. Sawicki, and A. Peethambaran, "An improved powered ankle-foot orthosis using proportional myoelectric control," *Gait Posture*, vol. 23, no. 4, pp. 425–428, 2006.
- [9] H. Kawamoto and Y. Sankai, "Comfortable power assist control method for walking aid by hal-3," Oct. 2002.
- [10] C. Winby, D. Lloyd, and T. Kirk, "Evaluation of different analytical methods for subject-specific scaling of musculotendon parameters," *J. of Biomech.*, vol. 41, no. 8, pp. 1682–1688, 2008.
- [11] D. G. Lloyd and T. S. Buchanan, "A model of load sharing between muscles and soft tissues at the human knee during static tasks," *J. of Biomechanical Engineering*, vol. 118, pp. 367–376, 1996.
- [12] S. Delp, J. Loan, M. Hoy, F. Zajac, E. Topp, and J. Rosen, "An interactive graphics-based model of the lower extremity to study orthopaedic surgical procedures," *Biomedical Engineering, IEEE Transactions on*, vol. 37, pp. 757–767, 1990.
- [13] C. Habermann and F. Kindermann, "Multidimensional spline interpolation: theory and applications," *Comput. Econ., Journal of*, no. 30, pp. 153–169, 2007.